Laboratorio CyberOps

Autore: Manfredi Lo Piparo
Data: 20 febbraio 2026

Introduzione

Il presente laboratorio si propone di esplorare strumenti e tecniche fondamentali per la sicurezza informatica attraverso esercitazioni pratiche mirate. Gli obiettivi principali includono l'utilizzo di Windows PowerShell per l'automazione e l'analisi di sistema, lo studio di Indicatori di Compromissione (IoC) tramite piattaforme di analisi dinamica, l'esplorazione di Nmap per la scansione di rete, e l'analisi di un attacco SQL injection mediante traffico di rete catturato.

Lo scopo di questa relazione è documentare i risultati e le analisi degli esercizi pratici, dimostrando la comprensione delle tecniche e degli strumenti esaminati, competenze essenziali per un professionista del settore della cybersecurity.

L'ambiente di laboratorio comprende una macchina Windows con PowerShell per le esercitazioni su automazione di sistema, la piattaforma Any.Run per l'analisi dinamica di malware, e una macchina virtuale CyberOps Workstation basata su Linux per le attività di network scanning e analisi del traffico di rete.

Esercizio 1: Utilizzare Windows PowerShell

Windows PowerShell rappresenta uno strumento strategico per l'automazione delle attività e la gestione dei sistemi operativi Windows. A differenza del tradizionale Prompt dei Comandi, PowerShell offre una shell a riga di comando più potente e un linguaggio di scripting integrato, basato su oggetti. La sua padronanza è essenziale per un analista di sicurezza, poiché consente di eseguire compiti complessi di diagnostica, configurazione e gestione in modo efficiente e automatizzato.

Obiettivi

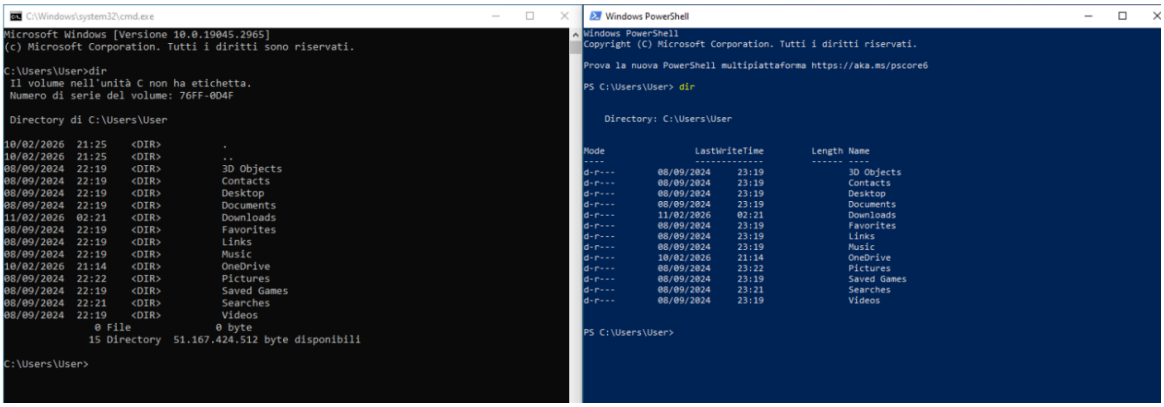
1. Accedere alla console PowerShell
2. Esplorare e confrontare i comandi del Prompt dei Comandi e di PowerShell
3. Esplorare i cmdlet
4. Esplorare il comando netstat usando PowerShell
5. Svuotare il cestino usando PowerShell

Per questo test è stata utilizzata una macchina Windows con PowerShell installata e connessione a internet attiva.

Parte 1: Accesso e Comandi di Base

L'analisi comparativa del comando `dir` eseguito nel Prompt dei Comandi e in Windows PowerShell rivela una differenza sostanziale nella presentazione dell'output. PowerShell formatta le informazioni in una tabella strutturata, con colonne chiare come Mode,

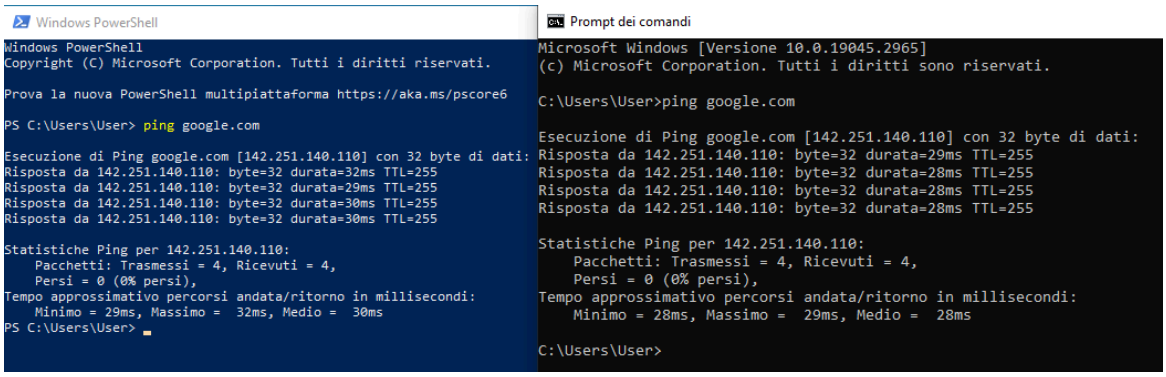
LastWriteTime, Length e Name, offrendo una leggibilità e un'utilità superiori rispetto all'output più semplice del Prompt dei Comandi.



Quali sono gli output del comando `dir`?

In entrambi i terminali viene mostrato l'elenco dei file e delle sottocartelle presenti nella directory corrente. La differenza principale risiede nel fatto che l'output di Windows PowerShell è formattato, offrendo la possibilità di estrapolare e manipolare maggiori informazioni, mentre l'output del cmd restituisce testo grezzo con una formattazione basilare.

Successivamente, è stato testato il comando di verifica della connettività:



Quali sono i risultati?

I risultati sono positivi, confermando che l'host è correttamente connesso a internet. L'output risulta sostanzialmente identico in entrambe le console. Questo dimostra la piena retrocompatibilità di PowerShell, che è in grado di eseguire i comandi tradizionali del Prompt dei Comandi senza alcuna modifica, garantendo una transizione fluida per gli amministratori di sistema.

Parte 2: Esplorazione dei Cmdlet

I comandi PowerShell, chiamati **cmdlet**, seguono una convenzione di denominazione **Verbo-Nome** (es. Get-ChildItem, Set-Location) che rende intuitiva la comprensione della loro funzione.

Un cmdlet è un comando specifico di PowerShell, strutturato secondo questa convenzione, che opera su oggetti anziché su stringhe di testo, offrendo maggiore flessibilità.

Per identificare il cmdlet equivalente al comando dir, ho eseguito

Qual è il comando PowerShell per dir?

Il comando nativo PowerShell per `dir` è `Get-ChildItem`. `dir` è semplicemente un alias, utile per mostrare il contenuto della directory corrente.

Parte 3: Analisi di Rete con netstat

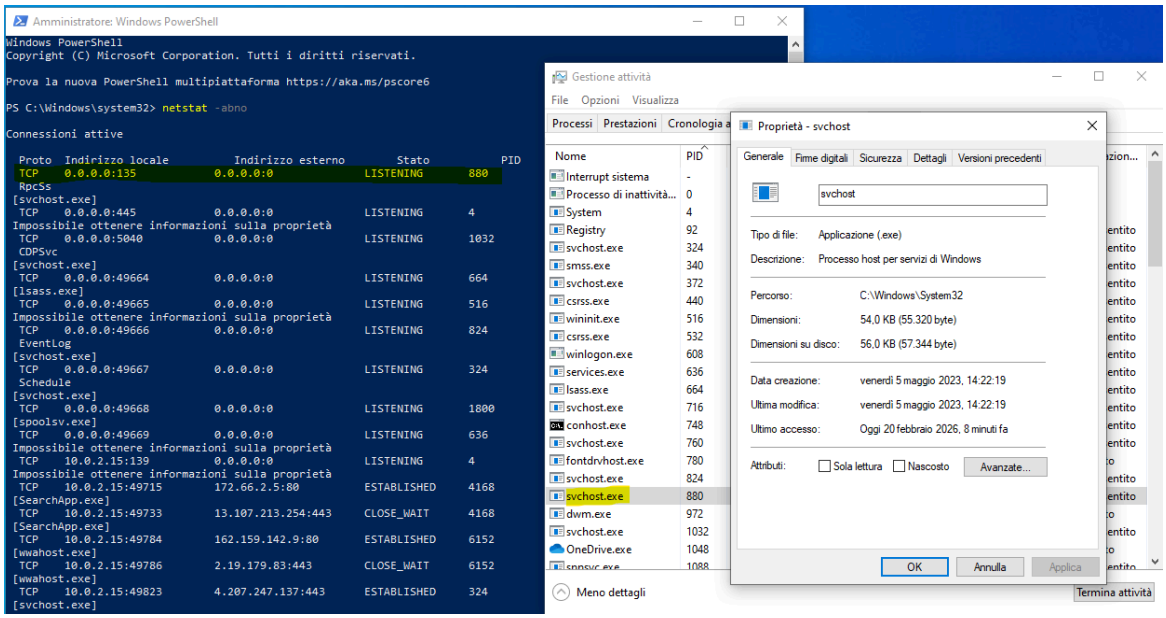
Eseguendo `netstat -h` è possibile visualizzare l'help e le opzioni disponibili per il comando. Successivamente, tramite il comando `netstat -r`, è stata visualizzata la tabella di routing dell'host.

```
C:\Users\User>netstat -r
===========================================================================
Elenco interfacce
  5...08 00 27 96 c2 10 ......Intel(R) PRO/1000 MT Desktop Adapter
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete             Mask          Gateway     Interfaccia Metrica
          0.0.0.0          0.0.0.0         10.0.2.2        10.0.2.15     25
         10.0.2.0    255.255.255.0         On-link         10.0.2.15    281
        10.0.2.15  255.255.255.255         On-link         10.0.2.15    281
       10.0.2.255  255.255.255.255         On-link         10.0.2.15    281
        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
        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         10.0.2.15    281
  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         10.0.2.15    281
===========================================================================
Route permanenti:
  Nessuna

IPv6 Tabella route
===========================================================================
Route attive:
 Interf Metrica Rete Destinazione      Gateway
  5    281 ::/0                     fe80::2
  1    331 ::1/128                  On-link
  5    281 fd17:625c:f037:2::/64    On-link
  5    281 fd17:625c:f037:2:7899:5949:534:2173/128
                                    On-link
  5    281 fd17:625c:f037:2:fcb2:c30b:a64e:97a5/128
                                    On-link
  5    281 fe80::/64                On-link
  5    281 fe80::7de5:ce64:b266:fed3/128
                                    On-link
  1    331 ff00::/8                 On-link
  5    281 ff00::/8                 On-link
===========================================================================
Route permanenti:
  Nessuna
```

Qual è il gateway IPv4?

Dall'analisi dell'output del comando `netstat -r`, che visualizza la tabella di routing del sistema, è possibile identificare il gateway predefinito. La rotta corrispondente all'indirizzo di rete 10.0.2.2 indica il gateway IPv4. Questa è la rotta predefinita, ovvero la "catch-all" utilizzata per tutto il traffico destinato a una rete non esplicitamente elencata nella tabella.

Avviando PowerShell con privilegi di amministratore e aprendo il Task Manager ordinato per PID, è stata selezionata una connessione specifica visualizzata tramite il comando `netstat -abno` per analizzare le proprietà.



Quali informazioni puoi ottenere dalla scheda Dettagli e dalla finestra di dialogo Proprietà per il PID selezionato?

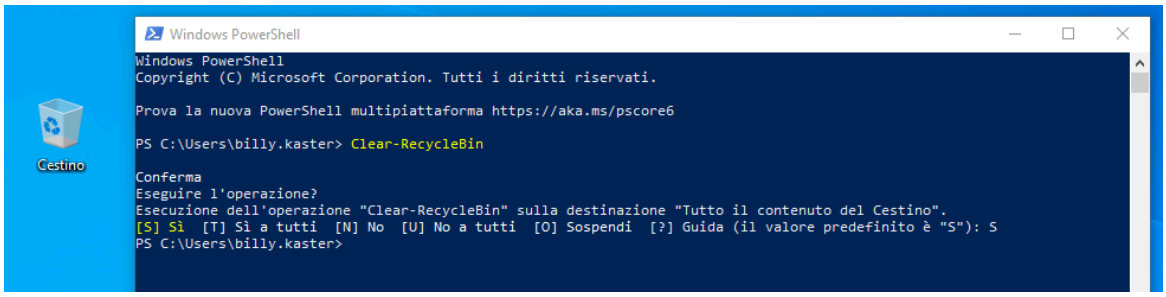
Per il processo `svchost.exe` con Process ID (PID) specifico, è stato possibile ottenere una serie di informazioni dettagliate combinando i dati di Gestione Attività e della finestra Proprietà del file eseguibile:

- **Da Gestione Attività (scheda Dettagli):** Nome processo, PID, Stato, Nome utente
- **Dalla finestra Proprietà (scheda Generale):** Tipo di file (Applicazione .exe), Descrizione, Percorso completo, Dimensioni, Date di creazione/modifica/accesso
- **Dalla finestra Proprietà (scheda Dettagli):** Versione del file, Nome del prodotto, Copyright, Firme digitali
- **Dalla finestra Proprietà (scheda Sicurezza):** Permessi e privilegi di accesso

È possibile ottenere numerose informazioni sul processo, tra cui il tipo di file, il percorso assoluto, le dimensioni, le date (creazione, modifica, ultimo accesso), i permessi di sicurezza, le firme digitali e le versioni precedenti.

Parte 4: Svuotare il Cestino

In questa fase, il cestino di Windows viene svuotato direttamente dalla riga di comando. Tramite il cmdlet `Clear-RecycleBin`, vengono eliminati tutti gli elementi presenti nel cestino.



Cosa è successo ai file nel Cestino?

Gli elementi presenti nel cestino sono stati eliminati in modo definitivo dal sistema. Un primo tentativo di eseguire il comando potrebbe fallire con un errore di tipo InvalidOperation ("Segmento già sbloccato"), che potrebbe indicare un problema temporaneo o uno stato inconsistente del sistema - tipicamente quando il cestino è già vuoto. Dopo averlo popolato con alcuni file, un secondo tentativo del comando ha cancellato con successo i file all'interno del cestino svuotandolo completamente.

Domanda di Riflessione: PowerShell per l'Analista di Sicurezza

PowerShell è stato sviluppato per l'automazione delle attività e la gestione della configurazione, e le sue capacità si estendono ampiamente al campo della sicurezza informatica. Un analista di sicurezza può sfruttare PowerShell per semplificare e automatizzare numerose attività di monitoraggio, analisi e risposta.

Alcuni cmdlet essenziali per il lavoro di un analista SOC o di sicurezza includono:

- **Get-Process e Stop-Process:** Utili per elencare tutti i processi in esecuzione su un sistema. Fondamentali per la threat hunting, poiché i malware spesso si nascondono o si iniettano nei processi in esecuzione. Consentono di identificare attività sospette o non autorizzate e di terminare i processi malevoli in tempo reale.
- **Get-NetTCPConnection:** Permette di verificare tutte le connessioni TCP correnti, le porte aperte e il loro stato, sostituendo in modo più efficiente `netstat`. Cruciale per identificare connessioni anomale o non autorizzate verso indirizzi IP sospetti.
- **Get-FileHash:** Fondamentale per calcolare l'hash di un file (es. SHA256) e confrontarlo con database di threat intelligence come VirusTotal per identificare file malevoli.
- **Get-EventLog o Get-WinEvent:** Per analizzare i log di eventi di Windows. Questi cmdlet permettono di filtrare e correlare eventi di sicurezza, accessi e errori di sistema, facilitando l'individuazione di indicatori di compromissione (IoC).
- **Invoke-WebRequest:** Per interagire con servizi web e API. Un analista può usarlo per testare la sicurezza di endpoint web, scaricare file da analizzare (es. firme di malware) o automatizzare le query verso piattaforme di threat intelligence.

Esercizio 2: Studio di Indicatori di Compromissione (IoC)

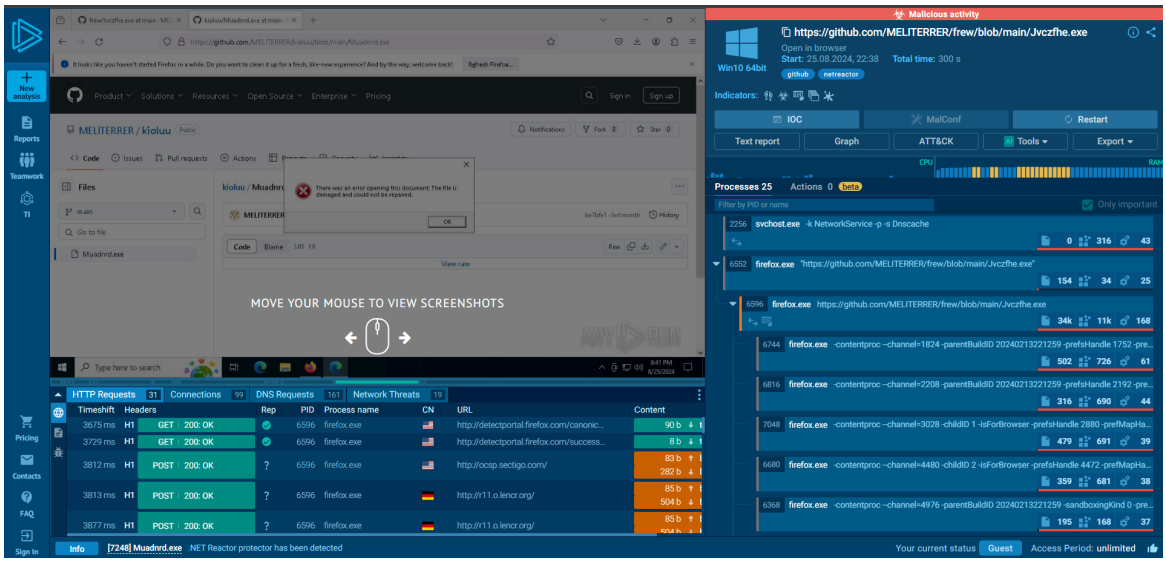
Descrizione del Task

L'esercizio richiede l'analisi di un campione di malware tramite la piattaforma **Any.Run**, un servizio di analisi dinamica interattiva che esegue file sospetti in ambienti sandbox virtualizzati.

Questo esercizio documenta l'analisi dinamica di un'esecuzione che coinvolge il download e l'attivazione di file binari eseguibili, in un ambiente che presenta una potenziale attività malevola, evidenziata dall'interazione con un tool di sandbox e analisi comportamentale.

Link analisi: https://app.any.run/tasks/9a15871843fe-45ce-85b366203dbc2281

L'obiettivo è identificare le minacce presenti, comprenderne il comportamento e documentare gli Indicatori di Compromissione per supportare attività di threat hunting e incident response.

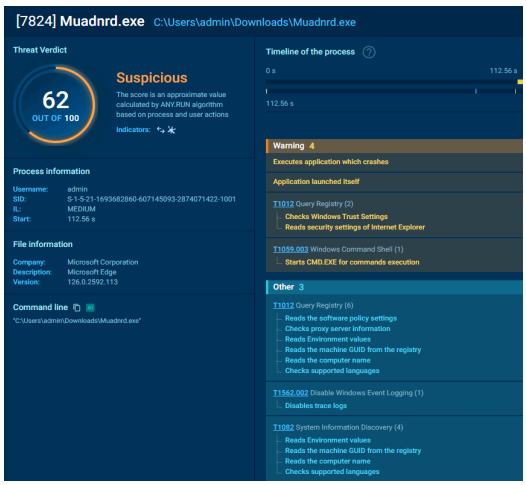


Informazioni Generali

- **Nome file 1**: Jvczfhe.exe
- **Nome file 2**: Muadnrd.exe
- **Tipo file**: PE32 executable (Windows)

Tipologia di Minaccia

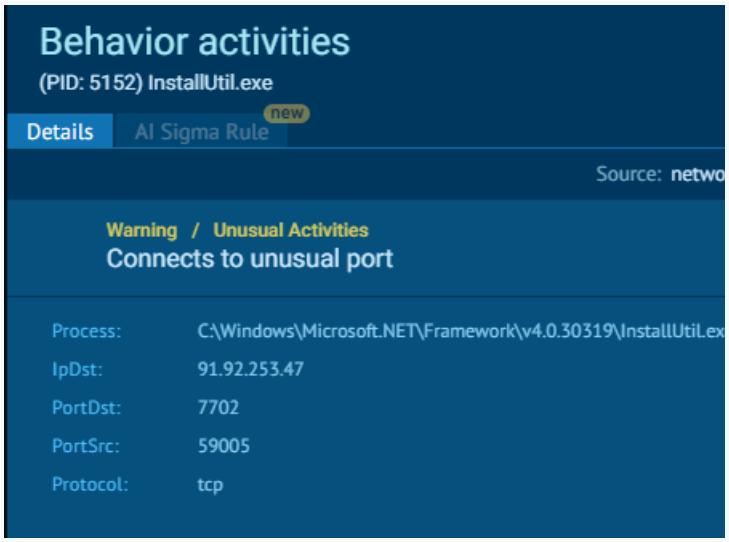
Il sistema di analisi ha classificato l'attività come **Malicious** con punteggi di rischio elevati (51/100 per il primo file e 62/100 per il secondo). Classificazione del malware: Information Stealer / Trojan Dropper con comportamenti di evasione delle sandbox.



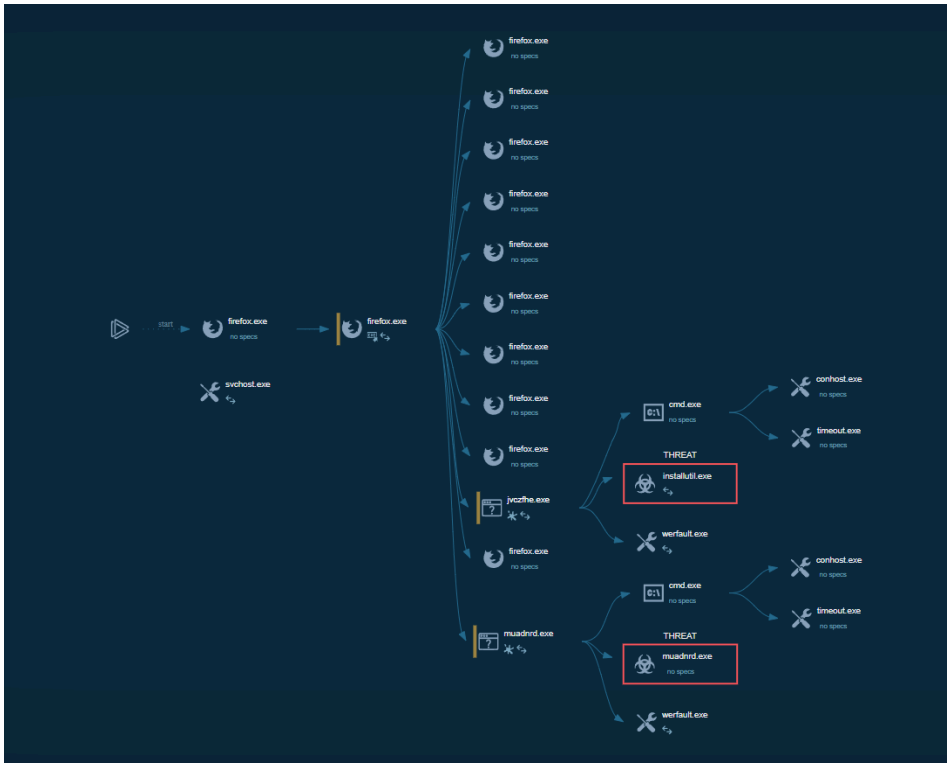
Indicatori e Attività Sospette

Il sistema ha identificato i seguenti comportamenti chiave nonostante i crash apparenti riportati all'utente:

- **Esecuzione di File Droppato**: Il processo `firefox.exe` ha innescato l'esecuzione di file eseguibili scaricati che, una volta avviati, hanno mostrato comportamenti malevoli
- **Tentativo di Evasione**: Entrambi i processi, `Jvczfhe.exe` (PID 7492) e `Muadrid.exe` (PID 7824), hanno avviato `CMD.EXE` per eseguire comandi, in particolare utilizzando `TIMEOUT.EXE` per ritardare l'esecuzione. Questa è una tattica comune per eludere le sandbox automatiche con limiti di tempo brevi
- **Crash e Segnalazione Errori**: Entrambi i binari hanno eseguito un'applicazione che è poi andata in crash. I crash sono stati gestiti dai processi di segnalazione errori di Windows (`WerFault.exe`)
- **Connessione Sospetta**: È stata registrata una connessione a una porta non usuale (porta 7702) dal processo `InstallUtil.exe`



- **Messaggi di Errore Ingannevoli**: Gli eseguibili mostravano messaggi di errore ("There was an error opening this document. The file is damaged and could not be opened") per ingannare l'utente, mentre in realtà eseguivano payload malevoli in background



(Tabella riassuntiva)

Conclusioni e Valutazione del Rischio

L'analisi dinamica evidenzia le seguenti criticità:

1. **Fonte Sospetta**: I file vengono scaricati da repository GitHub con nomi che non sembrano appartenere a progetti noti o ufficiali (MELITERRER/frew, MELITERRER/kioluu). Questa è una tecnica comune per la distribuzione di malware
2. **Publisher Sconosciuto**: Il sistema operativo Windows ha mostrato avvisi di sicurezza per applicazioni con Publisher Sconosciuto
3. **Comportamento Malevolo Tipico**: Entrambi i binari hanno tentato di eseguire comandi di sistema (cmd.exe e timeout.exe). L'uso di timeout.exe è un classico segnale di evasione, mirando a ritardare l'esecuzione per eludere i sistemi di analisi automatica
4. **Rischio di Infezione**: Il primo file (Jxfzbc.exe) ha tentato di eseguire un installer (InstallUtil.exe), indicando il probabile obiettivo di installare componenti malevoli (malware, downloader, o trojan)
5. **Punteggi di Rischio Elevati**: Any.Run ha assegnato punteggi di 51/100 e 62/100, classificando i file come Suspicious/Malicious

In sintesi, l'utente ha scaricato e tentato di eseguire due file ad alto rischio (molto probabilmente malware) da una fonte non attendibile. L'esecuzione apparentemente fallita

con messaggi di errore era in realtà una tecnica di inganno, mentre i payload venivano eseguiti in background mostrando comportamenti tipici di malware (evasione, persistenza, comunicazione di rete sospetta).

IOC trovati :

- **Esecuzione ritardata** (timeout 21).
- **Process Injection** (InstallUtil.exe).
- **Scoperta:** Lettura chiavi di registro e lingua di sistema per profilazione (Fingerprinting).

Indicatori di Rete :

- **IPv4 Destination**: 91.92.253.47 (Porta TCP: 7702)

Indicatori di Sistema :

- **Nomi file sospetti:** Jvczfhe.exe, Muadnrd.exe
- **Percorsi file:** C:\Users\admin\Downloads\Muadnrd.

Tecniche di Evasione e Persistenza:

- **Anti-analisi**: Uso di timeout.exe per ritardare l'esecuzione e superare i limiti temporali delle sandbox automatiche
- **Messaggi di errore ingannevoli**: Mostrare messaggi di file corrotto mentre il payload viene eseguito in background
- **Offuscamento**: Esecuzione tramite processi legittimi (cmd.exe, InstallUtil.exe)
- **Persistence**: Tentativo di installazione tramite InstallUtil.exe

Esercizio 3: Information Gathering tramite Nmap

Nmap (Network Mapper) è uno strumento open-source indispensabile per la scansione di rete e l'audit di sicurezza. La sua funzione principale rientra nella fase di ricognizione o reconnaissance, permettendo di scoprire host attivi su una rete, identificare le porte aperte, determinare i servizi in esecuzione su tali porte e, in alcuni casi, rilevare il sistema operativo e potenziali vulnerabilità.

In questo esercizio viene utilizzato Nmap per effettuare la fase di ricognizione sulla superficie di attacco di un target. Questa fase è vitale per mappare le porte aperte e i servizi esposti. Verrà utilizzata una macchina virtuale Cyber Ops Workstation.

Obiettivi

1. Esplorazione del tool Nmap
2. Scansione delle porte aperte (Locale, Rete locale, Remoto)

Parte 1: Esplorazione di Nmap

Aperto il terminale nella VM CyberOps Workstation, digitando il comando `man nmap` si accede al manuale ufficiale del tool.

Cos'è Nmap? Per cosa viene usato Nmap?

Nmap è uno dei più diffusi e potenti strumenti open source per l'esplorazione e l'auditing della sicurezza di una rete. Tramite l'invio di pacchetti IP grezzi, Nmap permette di ricevere informazioni cruciali sul sistema operativo, sull'eventuale presenza di filtri o firewall in uso e, in particolare, sui servizi in esecuzione sulle varie porte di rete.

Nmap viene utilizzato principalmente per:

- **Network Discovery (Esplorazione della Rete):** Per identificare quali host sono attivi e disponibili sulla rete
- **Port Scanning (Scansione delle Porte):** Per determinare quali porte sono aperte e quali servizi (come Web server, SSH, FTP, ecc.) sono in ascolto su ciascun host
- **Service and Version Detection:** Per identificare non solo che un servizio è in esecuzione, ma anche la sua applicazione specifica e il numero di versione
- **OS Detection (Rilevamento del Sistema Operativo):** Per dedurre il sistema operativo e il tipo di hardware dell'host di destinazione
- **Security Auditing (Audit di Sicurezza):** Utilizzando il Nmap Scripting Engine (NSE), può automatizzare la rilevazione di vulnerabilità

Il risultato di queste scansioni è un report dettagliato che classifica lo stato di ogni porta analizzata: risulta **aperta** se un'applicazione è in ascolto e accetta connessioni, **chiusa** se nessun servizio la sta utilizzando pur essendo accessibile, oppure **filtrata** quando un ostacolo di rete blocca i pacchetti, impedendo a Nmap di capirne il reale stato.

Tramite il flag `--help` o consultando gli esempi, è possibile consultare gli esempi pratici forniti dalla documentazione.

Qual è il comando nmap usato nell'esempio? Cosa fanno le opzioni -A e -T4?

Il comando usato nell'esempio è:

nmap -v -A -T4 scanme.nmap.org

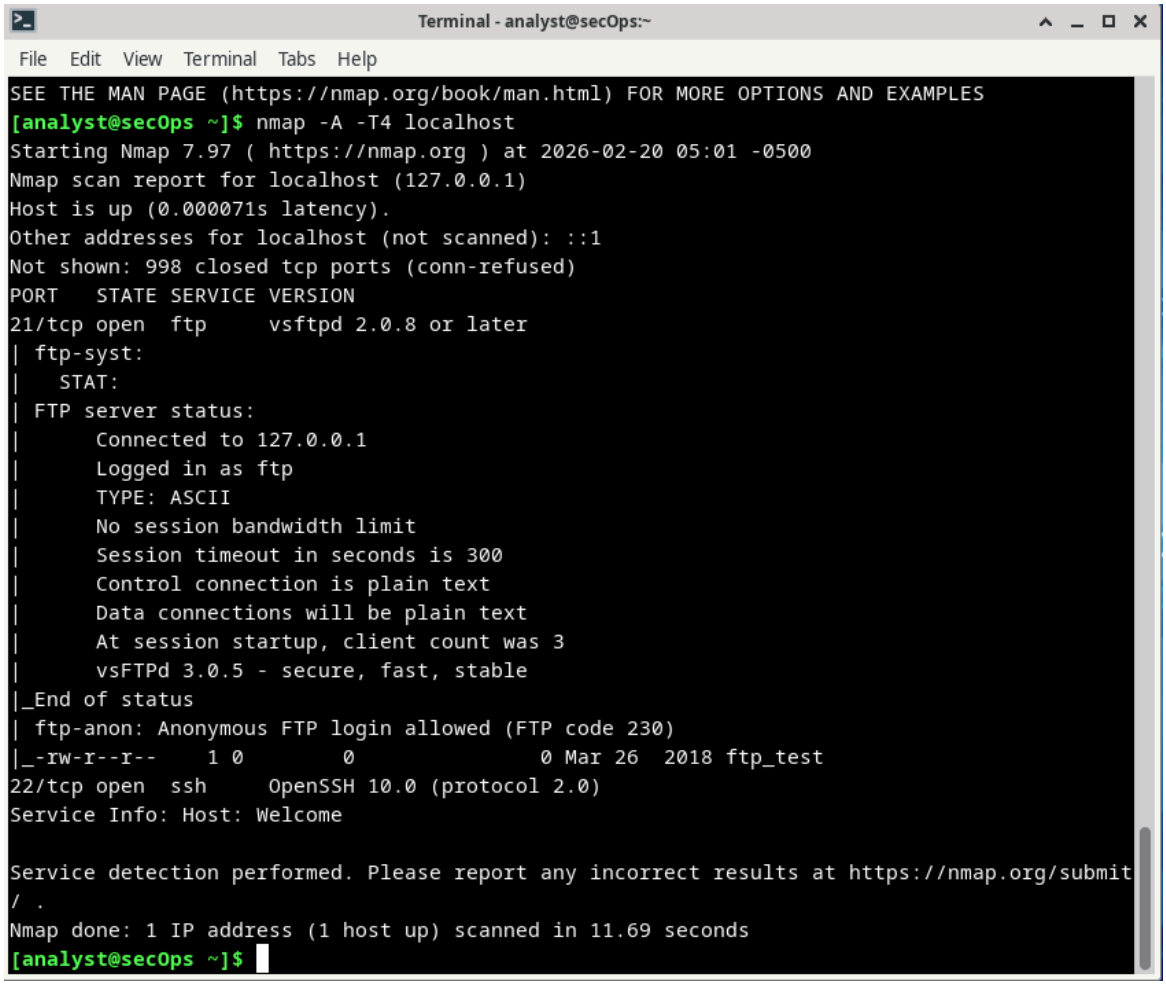
- **-A (Aggressive Scan):** Abilita un set avanzato di scansioni che include il rilevamento del sistema operativo (OS detection), la rilevazione della versione dei servizi (Version detection), l'esecuzione degli script di default di Nmap e il traceroute. Fornisce la massima quantità di informazioni sull'obiettivo in un singolo comando, ma è più rumorosa e facile da rilevare per i sistemi di sicurezza.
- **-T4 (Timing Template):** Regola la velocità della scansione. Il livello 4 (Aggressive) accelera i tempi di esecuzione, presumendo di trovarsi su una rete veloce e affidabile. I valori vanno da T0 (Paranoid - più lento e furtivo) a T5 (Insane - più veloce e rumoroso).

Parte 2: Scansione delle Porte Aperte

Scansione del Localhost

Inizialmente è stata effettuata una scansione sul localhost tramite il comando:

nmap -A -T4 localhost



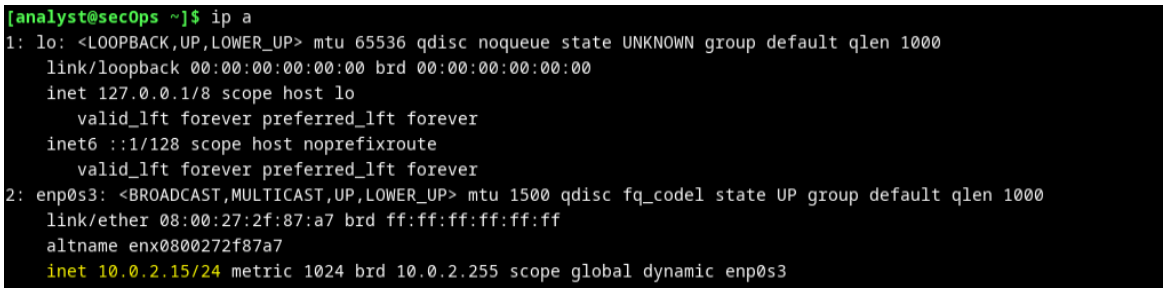
Quali porte e servizi sono aperti sul localhost?

Porta/Protocollo	Stato	Servizio
21/tcp	Aperta	FTP (vsftpd 2.0.8 or later)
22/tcp	Aperta	SSH (OpenSSH 10.0)

Scansione della Rete Locale

Tramite i comandi di rete, sono stati ricavati l'IP e la subnet mask della macchina:

- **IP:** 10.0.2.15
- **Subnet Mask:** 255.255.255.0 (/24)



A quale rete appartiene la VM e quanti host sono attivi?

La VM risiede nella rete 10.0.2.0/24. Avviando la scansione della rete con la VM impostata in modalità NAT, Nmap individua un solo host attivo: la VM stessa. Questo accade perché il NAT isola la macchina in una rete privata virtuale.

Scansione di un Server Remoto

Lo step successivo prevede la scansione del server `scanme.nmap.org`, un host fornito appositamente per scopi didattici. Il comando eseguito è:

nmap -A -T4 scanme.nmap.org

Risultati della scansione:

Qual è lo scopo di questo sito?

Come indicato dal titolo HTTP rilevato nell'output della scansione ("Go ahead and ScanMe!"), questo sito è un bersaglio messo a disposizione legalmente dal team di Nmap per permettere a studenti e professionisti di esercitarsi con le tecniche di scansione in modo sicuro ed etico.

Quali porte e servizi sono aperti?

Porta/Protocollo	Servizio
22/tcp	SSH
53/tcp	DNS
80/tcp	HTTP (servizio web)

Quali porte e servizi sono filtrati?

L'output indica la presenza di circa 997 porte TCP in stato **filtered** (no-response).

Qual è l'indirizzo IP risolto del server?

L'indirizzo IP del server è **45.33.32.156**.

Qual è il sistema operativo target?

Il Service Info ha identificato il sistema operativo come **Linux** (CPE: cpe:/o:linux:linux_kernel).

Domanda di Riflessione: Uso Etico e Malevolo di Nmap

Nmap è uno strumento potente per l'esplorazione e la gestione della rete. Come può Nmap aiutare con la sicurezza della rete? Come può Nmap essere usato da un attore malevolo come strumento nefasto?

Uso per la Sicurezza di Rete (Etico):

Dal lato difensivo, Nmap è essenziale per l'auditing continuo: permette di scoprire servizi esposti accidentalmente, porte non necessarie e software non aggiornato, aiutando a ridurre la superficie d'attacco. Per gli amministratori di rete e i professionisti della sicurezza, Nmap è uno strumento di audit fondamentale utilizzato per mappare la topologia di una rete aziendale, identificare porte aperte che potrebbero essere non necessarie e rappresentare una superficie di attacco, verificare che le regole dei firewall siano implementate correttamente e inventariare i servizi in esecuzione sui server.

1. **Asset discovery e inventory**: Mantenere un inventario aggiornato di tutti i dispositivi connessi alla rete. Viene utilizzato per mappare la topologia di una rete aziendale
2. **Vulnerability assessment**: Identificare servizi obsoleti, configurazioni insicure, porte non necessarie esposte. Identificare porte aperte che potrebbero essere non necessarie e rappresentare una superficie di attacco
3. **Compliance verification**: Verificare che le policy di sicurezza siano correttamente implementate (es. nessuna porta Telnet aperta). Verificare che le regole dei firewall siano implementate correttamente
4. **Change detection**: Monitorare cambiamenti nella rete che potrebbero indicare dispositivi non autorizzati o compromissioni
5. **Penetration testing**: Fase di reconnaissance durante security assessment autorizzati. Queste attività, note come ethical hacking o penetration testing, sono cruciali per rafforzare le difese di un'organizzazione prima che un malintenzionato possa sfruttarne le debolezze
6. **Firewall testing**: Verificare l'efficacia delle regole firewall. Inventariare i servizi in esecuzione sui server
7. **Service inventory**: Identificare versioni specifiche dei servizi per patch management

Uso Malevolo:

Dal lato offensivo, lo strumento è principe della fase di Reconnaissance. Viene utilizzato per mappare l'infrastruttura bersaglio e identificare i vettori d'ingresso vulnerabili prima di lanciare un exploit. Nelle mani di un aggressore, Nmap diventa uno strumento di ricognizione per pianificare un attacco. Un attore malevolo lo utilizza per raccogliere informazioni su un bersaglio senza autorizzazione, identificando host attivi, sistemi operativi vulnerabili tramite fingerprinting e servizi con vulnerabilità note. Questa fase di scansione è spesso uno dei primi e più rumorosi indicatori di un potenziale attacco imminente.

1. **Reconnaissance**: Primo stadio della kill chain - mappare la rete target e identificare potenziali punti di ingresso. Un attore malevolo lo utilizza per raccogliere informazioni su un bersaglio senza autorizzazione, identificando host attivi

2. **Service enumeration**: Identificare servizi vulnerabili da sfruttare. Servizi con vulnerabilità note da exploitare
3. **OS fingerprinting**: Adattare gli exploit al sistema operativo target. Sistemi operativi vulnerabili tramite fingerprinting
4. **Firewall evasion**: Utilizzare tecniche di scansione stealth per evitare detection
5. **Vulnerability identification**: Tramite NSE scripts, identificare vulnerabilità note da exploitare
6. **Attack planning**: Le informazioni raccolte con Nmap sono il primo passo per selezionare un vettore d'attacco e lanciare exploit mirati. Questa fase di "scansione" è spesso uno dei primi e più rumorosi indicatori di un potenziale attacco imminente

Esercizio 3: Analisi di un Attacco SQL Injection (SQLi)

Gli attacchi di tipo SQL Injection (SQLi) rappresentano una delle vulnerabilità più critiche e diffuse nelle applicazioni web. Questo tipo di attacco sfrutta una gestione non sicura degli input utente per manipolare le query SQL inviate a un database. Un malintenzionato può così aggirare i meccanismi di autenticazione e accedere, modificare o cancellare dati sensibili, fino a compromettere l'intero server.

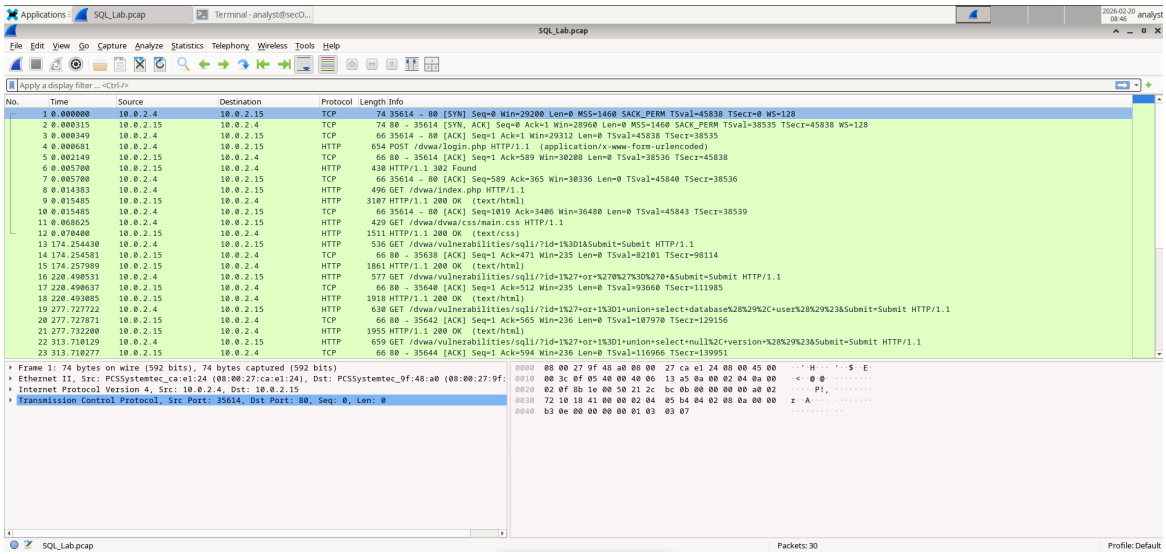
In questo esercizio viene utilizzato Wireshark per ispezionare il file PCAP di una cattura di rete contenente un attacco SQL Injection su un database MySQL.

Obiettivi

1. Caricare il file PCAP in Wireshark
2. Tracciare, visualizzare e analizzare le query iniettate per comprendere l'esfiltrazione dei dati

Parte 1: Analisi del Traffico

Il file analizzato rappresenta un attacco SQLi della durata di circa 8 minuti.



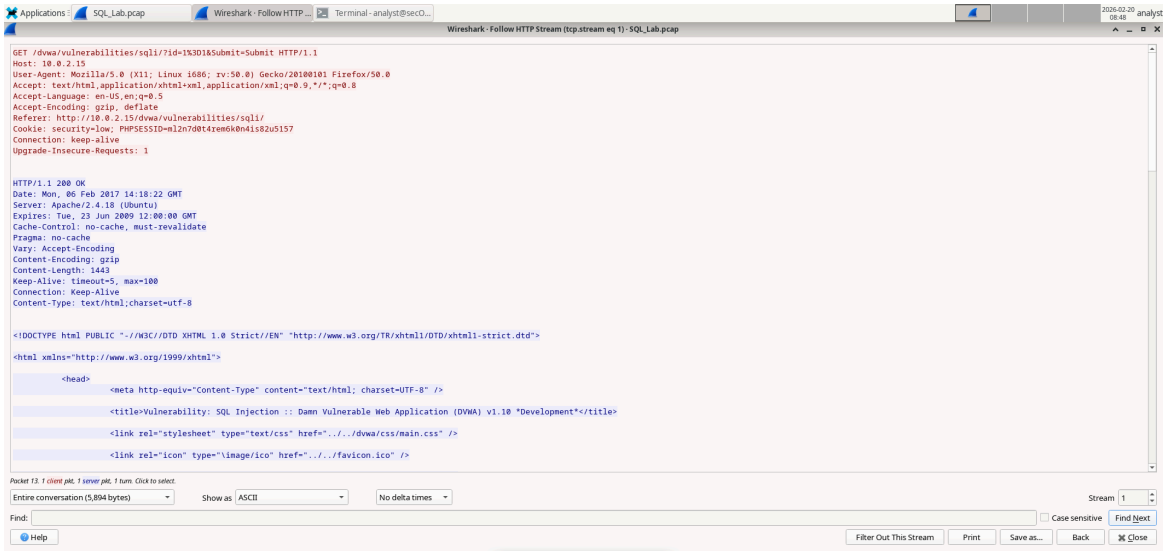
Quali sono i due indirizzi IP coinvolti in questo attacco in base alle informazioni visualizzate?

Dall'analisi del file di cattura del traffico, è possibile identificare chiaramente i due attori coinvolti nella comunicazione:

- **Sorgente (Attaccante):** 10.0.2.4
- **Destinazione (Server Target):** 10.0.2.15

Parte 2: Fasi dell'Attacco e Raccolta di Informazioni

Selezionando il pacchetto numero 13 (una richiesta HTTP GET) e seguendo lo stream TCP, è possibile ispezionare il payload dell'attacco in chiaro.

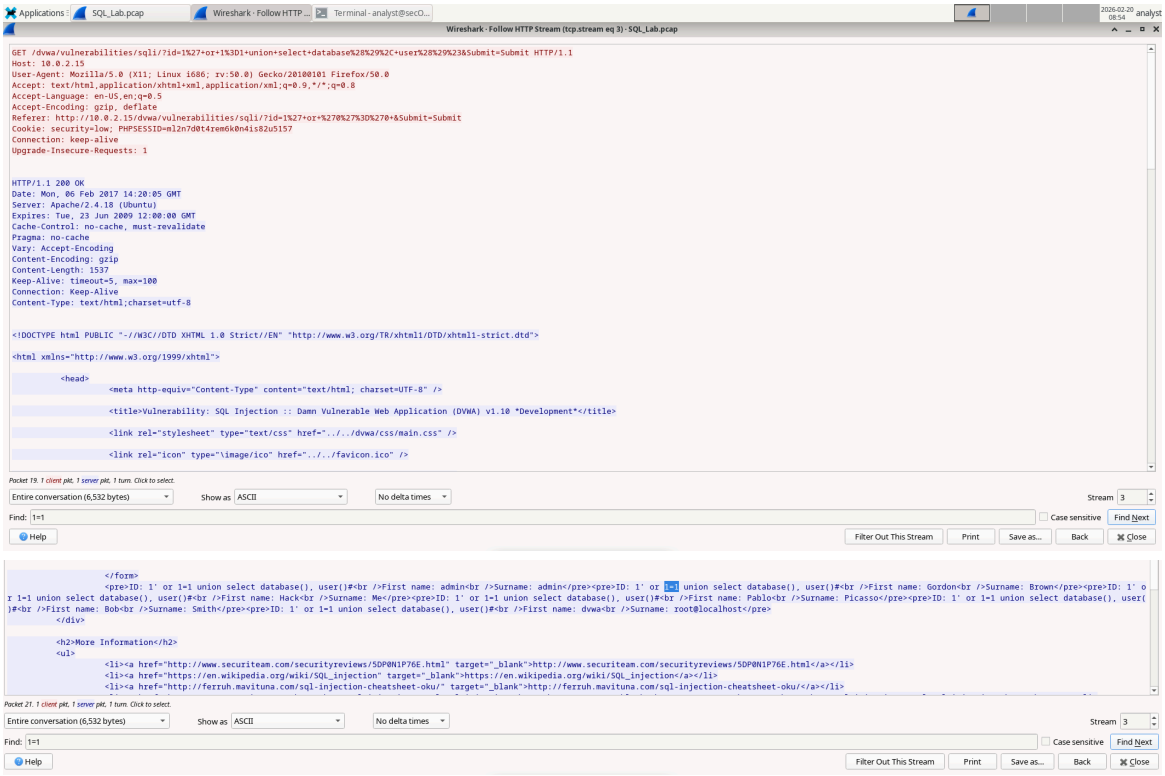


Fase 1: Verifica della Vulnerabilità

L'attaccante inietta la stringa `1=1`. Questa tecnica viene utilizzata nella fase iniziale (ricognizione della vulnerabilità) per forzare una condizione sempre vera. Se il sito restituisce un risultato imprevisto senza errori, conferma all'attaccante che l'input non è sanitizzato e che il form è vulnerabile alle SQL Injection.

Fase 2: Estrazione Informazioni sul Database

Ispezionando il pacchetto 19, si rileva una query di tipo UNION basata sull'estrazione di informazioni di sistema. L'attaccante ha eseguito una query per estrarre il nome del database e l'utente in uso:



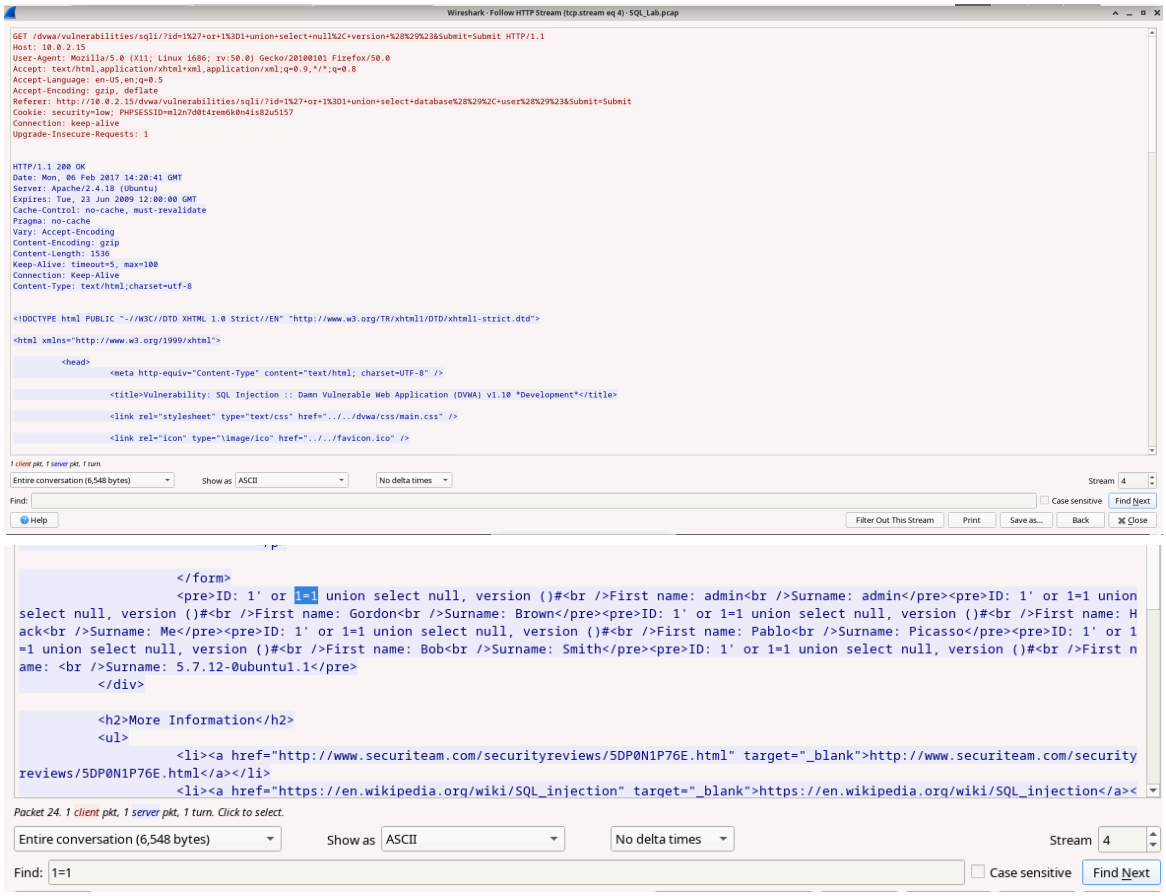
1' UNION SELECT database(), user() --

- **Nome Database:** dvwa

- **Utente Database:** root@localhost

Fase 3: Estrazione Versione del Sistema

Seguendo lo stream HTTP del pacchetto 22, l'attaccante esegue una query per determinare la versione del DBMS



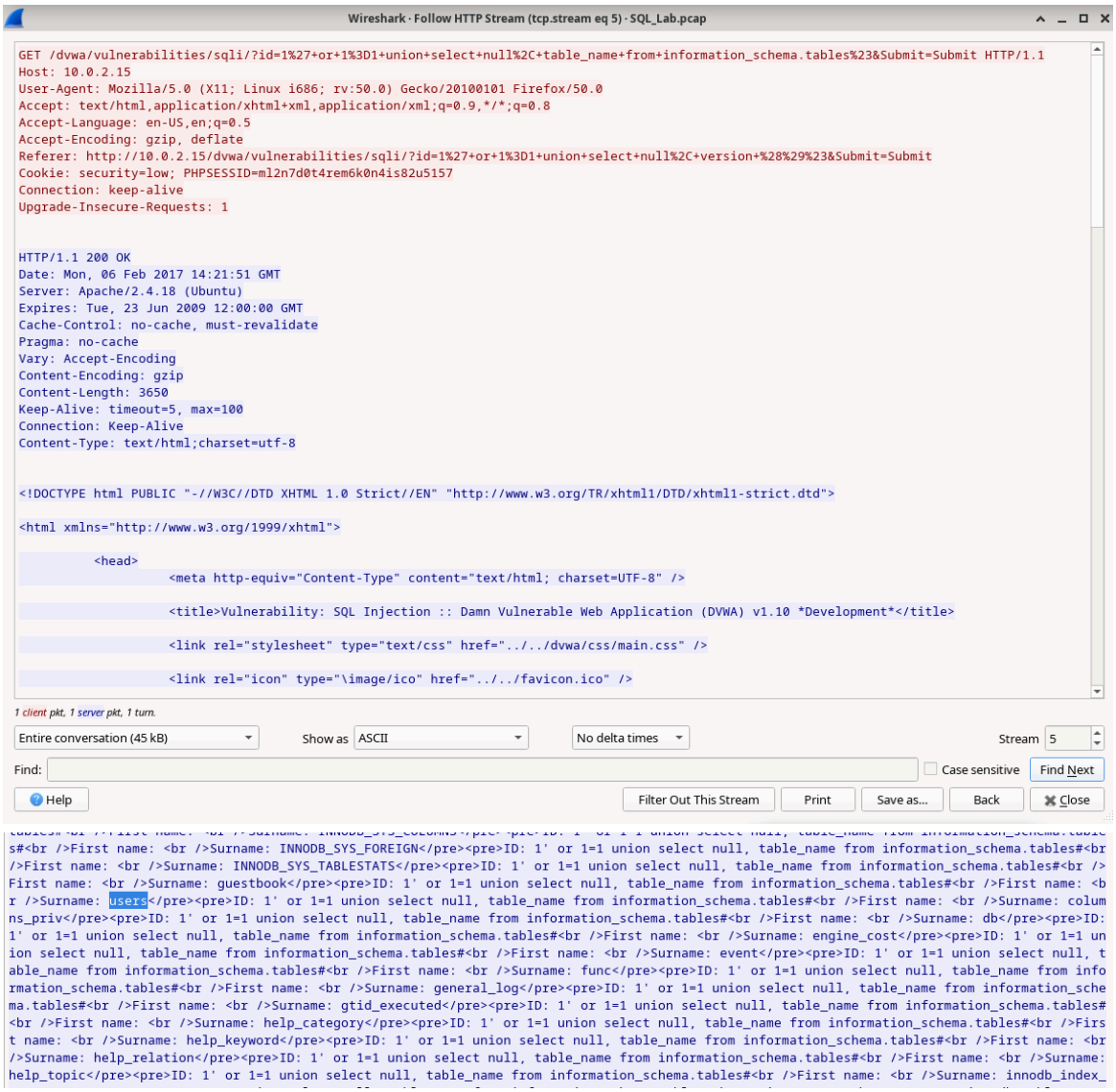
1' UNION SELECT NULL, version() --

Qual è la versione del database?

La versione estratta è **5.7.12-0ubuntu1.1**.

Fase 4: Enumerazione delle Tabelle

Procedendo con l'analisi (pacchetto 25), l'attaccante passa all'enumerazione dello schema del database interrogando `information_schema.tables`. L'output di questa query restituisce l'elenco delle tabelle, permettendo all'attaccante di individuare quelle contenenti dati sensibili come la tabella `users`.

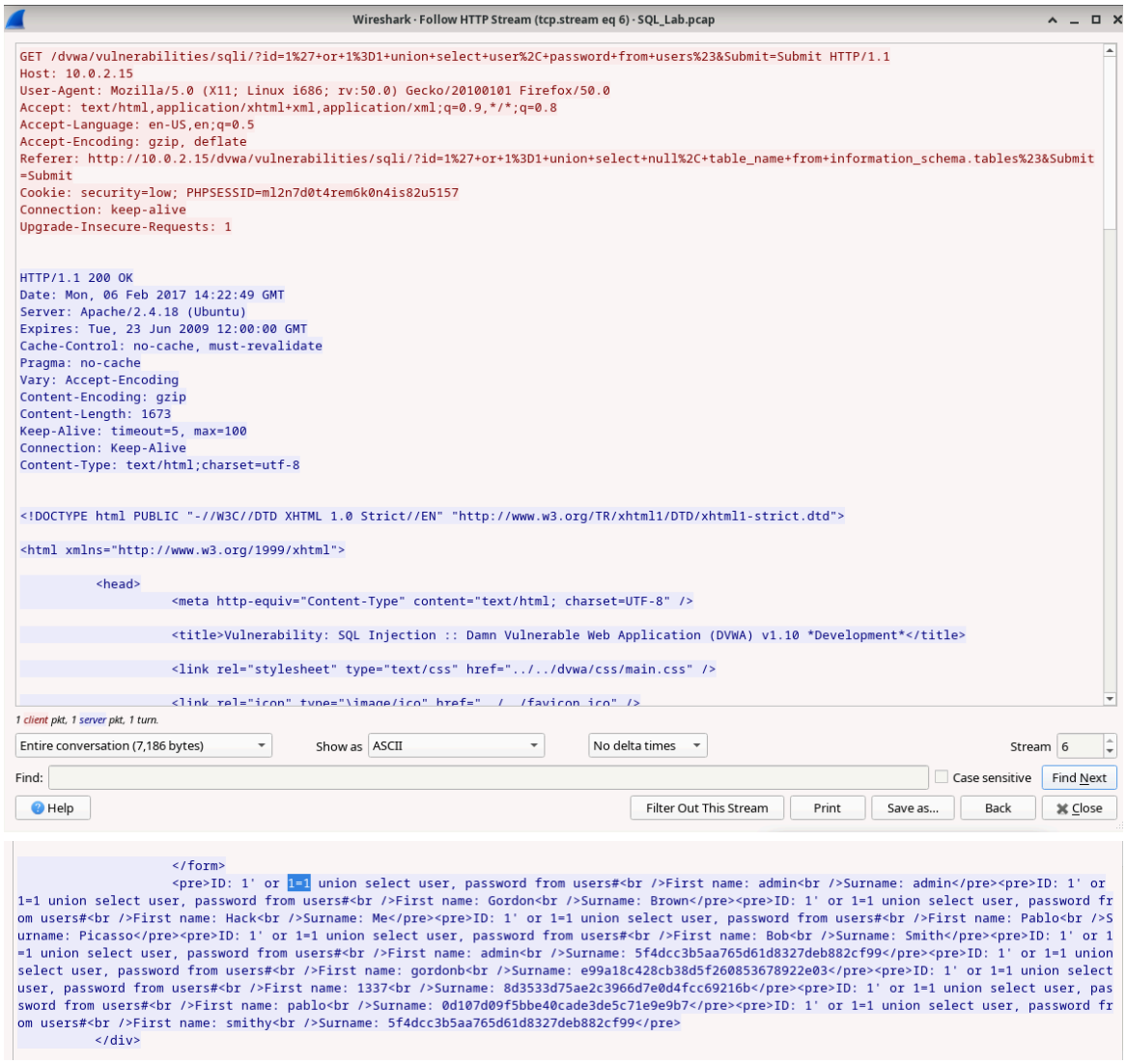


Cosa farebbe per l'aggressore il comando modificato con WHERE table_name='users'?

Dopo aver tentato di elencare tutte le tabelle (producendo un output enorme), l'uso di un comando modificato con la clausola `WHERE table_name='users'` permetterebbe all'attaccante di rendere l'attacco più mirato ed efficiente. Questa query filtrerebbe i risultati per ottenere solo i nomi delle colonne della tabella specifica `users`, che è il bersaglio più probabile per contenere credenziali di accesso.

Fase 5: Compromissione Finale - Estrazione delle Credenziali

Nella fase finale dell'attacco (pacchetto 28), la SQL Injection viene sfruttata per esfiltrare il contenuto della tabella utenti, estraendo username e hash delle password:



1' UNION SELECT user, password FROM users --

Quale utente possiede l'hash della password 8d3533d75ae2c3966d7e0d4fcc69216b?

Qual è la password in chiaro?

Dall'output della query, si evince che l'hash **8d3533d75ae2c3966d7e0d4fcc69216b** è associato all'utente **1337**. L'hash della password è stato successivamente analizzato con un servizio di cracking online. Il processo ha avuto successo, rivelando che la password in chiaro è **charley**.

Domande di Riflessione: Rischi e Prevenzione dell'SQL Injection

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

Sebbene il linguaggio SQL sia fondamentale per il funzionamento della maggior parte dei siti web e delle applicazioni moderne basate su database, il suo utilizzo introduce il rischio intrinseco di attacchi di tipo SQL Injection. Il pericolo non risiede in SQL stesso, ma nella gestione non sicura degli input degli utenti da parte delle applicazioni.

Oggi giorno quasi tutti i siti web si basano su database che comunicano tramite linguaggio SQL. Il rischio principale è che, se non protetti adeguatamente, questi dati sensibili possano essere estratti, alterati o addirittura cancellati. La gravità dell'attacco SQL Injection varia molto in base alle intenzioni dell'attaccante e al tipo di informazioni che mira a raccogliere o distruggere. La gravità può variare dall'esposizione di dati sensibili dei clienti (violazione della privacy) e dalla manipolazione dei dati, fino al bypass completo dei controlli di accesso e alla compromissione totale del server che ospita il database.

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

Esistono diverse tecniche efficaci per mitigare e prevenire gli attacchi SQLi. Due delle più importanti sono:

1. **Query Parametrizzate (Prepared Statements) e Stored Procedure:** Questo è considerato il metodo più robusto e efficace. Consiste nel separare nettamente l'istruzione SQL (il codice) dai dati forniti dall'utente (i parametri). In questo modo, il motore del database non interpreta mai l'input dell'utente come parte eseguibile del comando, annullando di fatto la possibilità di un'iniezione. Consiste nello scrivere il codice SQL in modo che il database tratti i dati inseriti dall'utente solo come semplici parametri testuali e mai come comandi eseguibili, neutralizzando così l'attacco alla radice.
2. **Sanitizzazione e Validazione dell'Input:** Consiste nel validare e pulire rigorosamente tutti i dati provenienti dall'esterno prima che vengano incorporati in una query. La principale azione preventiva è verificare rigorosamente i dati inseriti dall'utente (anche tramite espressioni regolari - regex) per bloccare o ripulire caratteri speciali e sequenze che compongono query malevole prima che vengano elaborate. Ciò include la convalida del tipo di dati atteso (es. un numero deve essere un numero), la limitazione della lunghezza e l'escape dei caratteri speciali come apici, doppi apici, punto e virgola che potrebbero essere usati per manipolare la sintassi SQL.

Conclusione

Questo laboratorio ha fornito un'esperienza pratica e approfondita su strumenti e tecniche fondamentali nel campo della cybersecurity. Attraverso gli esercizi, sono state acquisite competenze pratiche nell'uso di PowerShell per l'amministrazione e l'analisi di sistemi Windows, nell'impiego di Nmap per la ricognizione di rete e l'identificazione di potenziali superfici d'attacco, e nell'analisi forense di un attacco SQL Injection tramite Wireshark.

L'insieme di queste attività ha offerto una base solida per comprendere le metodologie operative di un professionista della sicurezza, coprendo le fasi di amministrazione di sistema, ricognizione e analisi post-incidente. La padronanza di questi strumenti è essenziale per difendere efficacemente le infrastrutture informatiche moderne e per operare come analista SOC o professionista della cybersecurity.

Le competenze acquisite includono:

- Utilizzo avanzato di PowerShell per l'automazione e il monitoraggio della sicurezza

- Tecniche di network reconnaissance tramite Nmap per la mappatura della superficie d'attacco
- Analisi forense del traffico di rete per identificare e comprendere gli attacchi
- Comprensione delle vulnerabilità comuni e delle migliori pratiche di prevenzione

Questi strumenti e tecniche rappresentano la base del toolkit di ogni professionista della sicurezza informatica e la loro padronanza continua è fondamentale per mantenere un livello adeguato di protezione delle infrastrutture digitali moderne.