Report sull’Esercitazione S11L5: Analisi e Attacco con Strumenti di Cybersecurity

1. Accesso a PowerShell

Per avviare PowerShell ho seguito questi passaggi:

1. **Apertura di PowerShell**:
    - Ho cliccato su **Start**, digitato `PowerShell` nella barra di ricerca e selezionato **Windows PowerShell**.
    - Ho avviato la console con privilegi amministrativi cliccando con il tasto destro e scegliendo **Esegui come amministratore**.
2. **Apertura del Prompt dei comandi**:
    - Ho ripetuto la procedura cercando `cmd` e avviando il Prompt dei comandi.

---

2. Confronto tra il Prompt dei comandi e PowerShell

Per confrontare le funzionalità dei due strumenti, ho eseguito alcuni comandi base:

2.1. Utilizzo del comando `dir`

Nel **Prompt dei comandi**, ho digitato:

```
dir
```

- Questo ha mostrato un elenco di file e cartelle della directory corrente.

In **PowerShell**, ho eseguito lo stesso comando:
`dir`

- Il risultato è stato simile, ma il formato dell'output conteneva più dettagli sulle proprietà dei file.

Per verificare la differenza interna, ho usato il comando:

```
Get-Alias dir
```

Output:

```
Alias           dir -> Get-ChildItem
```

Questo mi ha confermato che `dir` in PowerShell è un alias per il cmdlet `Get-ChildItem`.

2.2. Utilizzo di altri comandi di base

Ho eseguito alcuni comandi comuni in entrambi gli ambienti:

- **`ping google.com`** → Test della connettività di rete.
- **`ipconfig`** → Visualizzazione della configurazione IP del computer.
- **`cd`** → Navigazione tra le cartelle.

Tutti questi comandi hanno dato risultati simili in entrambi gli ambienti.

---

3. Esplorazione dei Cmdlet in PowerShell

Dopo aver verificato che `dir` è un alias di `Get-ChildItem`, ho provato altri cmdlet nativi di PowerShell:

Visualizzazione degli alias disponibili:
`Get-Alias`

1. Questo comando ha mostrato l'elenco completo degli alias disponibili.

Elenco dei comandi di PowerShell:
`Get-Command`

2. Ho ottenuto una lista di tutti i comandi e cmdlet disponibili nel sistema.

Ottenere informazioni su un cmdlet:

```
Get-Help Get-ChildItem
```

3. Questo ha mostrato una descrizione dettagliata del cmdlet `Get-ChildItem`, con esempi di utilizzo.

---

4. Uso di `netstat` per monitorare la rete

Ho poi testato `netstat`, un comando utile per visualizzare connessioni attive e statistiche di rete.

Visualizzazione delle connessioni attive:
`netstat -a`

1. Questo ha mostrato un elenco delle connessioni di rete attive e delle porte in ascolto.

Visualizzazione dettagliata con informazioni sui processi:

```
netstat -abno
```

2. Questo comando ha fornito informazioni dettagliate sui processi associati a ciascuna connessione.

Verifica della tabella di routing:
`netstat -r`

3. Ho ottenuto una panoramica delle rotte di rete configurate sul sistema.

---

5. Pulizia del Cestino con PowerShell

Infine, ho testato l'utilizzo di PowerShell per eseguire operazioni amministrative, come lo svuotamento del Cestino.

Visualizzazione del contenuto del Cestino:

```
Get-ChildItem C:\$Recycle.Bin -Recurse
```

1. Ho potuto vedere tutti i file eliminati ma ancora presenti nel Cestino.

Svuotamento del Cestino:

```
Clear-RecycleBin -Force
```

2. Questo ha eliminato definitivamente tutti i file nel Cestino senza richiedere conferma.

Analisi del Traffico HTTP e HTTPS con Wireshark

Parte 1: Cattura e Analisi del Traffico HTTP

1. **Avvio della Macchina Virtuale**:
   - Ho avviato la VM CyberOps Workstation e ho effettuato l'accesso con le credenziali fornite.
2. **Avvio di `tcpdump` per Catturare il Traffico HTTP**:
   - Ho aperto un terminale e identificato l'interfaccia di rete attiva utilizzando il comando appropriato.
   - Ho avviato `tcpdump` sull'interfaccia identificata, specificando la cattura completa dei pacchetti e salvando l'output in un file denominato `httpdump.pcap`.
3. **Generazione di Traffico HTTP**:
   - Ho aperto un browser web all'interno della VM e navigato al sito web `http://www.altoromutual.com/login.jsp`, che utilizza il protocollo HTTP non crittografato.
   - Nella pagina di login, ho inserito le credenziali "Admin" sia per il nome utente che per la password, quindi ho cliccato su "Login".
   - Dopo aver completato queste operazioni, ho chiuso il browser.
4. **Interruzione della Cattura e Analisi dei Dati**:
   - Sono tornato al terminale e ho interrotto `tcpdump` utilizzando la combinazione di tasti appropriata.
   - Ho aperto il file `httpdump.pcap` con Wireshark per analizzare il traffico catturato.
   - Filtrando per il protocollo HTTP, ho individuato le richieste e le risposte HTTP associate al login effettuato. Ho notato che le credenziali inserite erano visibili in chiaro all'interno dei pacchetti, confermando la mancanza di crittografia nel protocollo HTTP.

Parte 2: Cattura e Analisi del Traffico HTTPS

1. **Avvio di una Nuova Cattura con `tcpdump`**:
   - Ho aperto un nuovo terminale e avviato `tcpdump`, salvando l'output in un file denominato `httpsdump.pcap`.
2. **Generazione di Traffico HTTPS**:
   - Ho aperto il browser web e navigato al sito `https://www.google.com`, che utilizza il protocollo HTTPS crittografato.
   - Ho interagito con la pagina per generare traffico, ad esempio effettuando una ricerca.

3. **Interruzione della Cattura e Analisi dei Dati**:
    - Ho interrotto `tcpdump` e aperto il file `httpsdump.pcap` con Wireshark.
    - Filtrando per il protocollo TLS (Transport Layer Security), ho osservato che, sebbene fosse possibile identificare la comunicazione tra il client e il server, il contenuto dei messaggi risultava crittografato, impedendo la visualizzazione dei dati effettivi scambiati.

Esplorazione di Nmap

Parte 1: Esplorazione di Nmap

1. **Avvio della Macchina Virtuale**:
    - Ho avviato la VM CyberOps Workstation e ho effettuato l'accesso con le credenziali fornite.
2. **Apertura del Terminale**:
    - Ho aperto una finestra del terminale per interagire con il sistema.
3. **Consultazione del Manuale di Nmap**:
    - Ho digitato `man nmap` per accedere alle pagine manuali di Nmap.
    - All'interno del manuale, ho utilizzato le frecce per navigare e la barra spaziatrice per avanzare di una pagina.
    - Per cercare termini specifici, ho utilizzato la funzione di ricerca digitando `/example` e premendo Invio, trovando così esempi di utilizzo di Nmap.
4. **Analisi degli Esempi**:
    - Nel primo esempio, il comando mostrato era `nmap -A -T4 scanme.nmap.org`.
    - Ho approfondito il significato delle opzioni:
        - `-A`: Abilita il rilevamento del sistema operativo, la rilevazione delle versioni, la scansione degli script e il traceroute.
        - `-T4`: Imposta la velocità della scansione su un livello più veloce, adatto per connessioni a banda larga o Ethernet.

Parte 2: Scansione per Porte Aperte

1. **Scansione del Localhost**:
    - Ho eseguito una scansione sul mio host locale per identificare le porte aperte e i servizi in esecuzione.
    - L'output ha mostrato diverse porte aperte, indicando i servizi attivi sul sistema.
2. **Scansione della Rete Locale**:

    - Ho eseguito una scansione sulla rete locale per identificare gli host attivi e le loro porte aperte.
    - La scansione ha rilevato diversi dispositivi sulla rete, ciascuno con un elenco di porte aperte e servizi associati.
3. **Scansione di un Server Remoto**:
    - Ho eseguito una scansione sul server remoto `scanme.nmap.org` per identificare le porte aperte e i servizi offerti.
    - L'output ha mostrato diverse porte aperte, indicando i servizi disponibili su quel server.

Attacco a un Database MySQL

Parte 1: Apertura di Wireshark e Caricamento del File PCAP

1. **Avvio della Macchina Virtuale**:
    - Ho avviato la VM CyberOps Workstation e ho effettuato l'accesso con le credenziali fornite.
2. **Apertura di Wireshark**:
    - Ho cliccato su "Applicazioni" > "CyberOps" > "Wireshark" per avviare l'applicazione.
3. **Caricamento del File PCAP**:
    - All'interno di Wireshark, ho cliccato su "File" > "Open" e ho navigato fino alla directory `/home/analyst/lab.support.files/` per aprire il file `SQL_Lab.pcap`.
4. **Esame del Traffico Catturato**:
    - Il file PCAP conteneva il traffico di rete catturato durante un attacco di SQL Injection, con una durata complessiva di circa 8 minuti (441 secondi).
5. **Identificazione degli IP Coinvolti**:
    - Analizzando il traffico, ho identificato due indirizzi IP coinvolti nell'attacco: `10.0.2.4` (attaccante) e `10.0.2.15` (vittima).

Parte 2: Analisi dell'Attacco di SQL Injection

1. **Esame Iniziale dell'Attacco**:
    - Ho individuato una richiesta HTTP GET sospetta al pacchetto numero 13.
    - Cliccando con il tasto destro su questa riga, ho selezionato "Follow" > "HTTP Stream" per visualizzare l'intero flusso di dati.
2. **Identificazione del Payload Malevolo**:
    - All'interno del flusso HTTP, ho cercato la stringa `1=1` per individuare l'iniezione SQL.
    - L'attaccante aveva inserito la query `1=1` in un campo UserID sul server `10.0.2.15`.

- Invece di restituire un messaggio di errore, l'applicazione ha risposto con un record del database, indicando che l'iniezione SQL aveva avuto successo.

Parte 3: Continuazione dell'Attacco di SQL Injection

1. **Ulteriore Analisi del Traffico**:
   - Ho esaminato il pacchetto numero 19 seguendo lo stesso metodo descritto in precedenza.
2. **Raccolta di Informazioni Sensibili**:
   - L'attaccante ha eseguito la query: `1' or 1=1 union select database(), user()#`.
   - Questa query ha restituito il nome del database (`dvwa`) e l'utente del database (`root@localhost`), oltre a diversi account utente presenti nel sistema.

Parte 4: Informazioni di Sistema Ottenute dall'Attacco

1. **Determinazione della Versione del Database**:
   - Analizzando il pacchetto numero 22, ho osservato che l'attaccante ha eseguito la query: `1' or 1=1 union select null, version()#`.
   - Questa query ha restituito la versione del database: `MySQL 5.7.12-0`.

Parte 5: Informazioni sulle Tabelle del Database Ottenute dall'Attacco

1. **Elenco delle Tabelle del Database**:
   - Nel pacchetto numero 25, l'attaccante ha eseguito la query: `1'or 1=1 union select null, table_name from information_schema.tables#`.
   - Questa query ha restituito un elenco di tutte le tabelle presenti nel database, fornendo all'attaccante una panoramica completa della struttura del database.
2. **Accesso ai Dati Sensibili**:
   - Modificando la query in: `1' OR 1=1 UNION SELECT null, column_name FROM INFORMATION_SCHEMA.columns WHERE table_name='users'`, l'attaccante potrebbe ottenere i nomi delle colonne della tabella `users`, facilitando l'accesso a dati sensibili come nomi utente e password.

Parte 6: Conclusione dell'Attacco di SQL Injection

1. **Estrazione di Hash delle Password**:
   - Nel pacchetto numero 28, l'attaccante ha eseguito una query per ottenere gli hash delle password degli utenti, concludendo l'attacco.