CyberOps

Laboratorio - Utilizzo di Windows PowerShell

In questo laboratorio, esploreremo alcune delle funzioni di PowerShell.

https://itexamanswers.net/3-3-11-lab-using-windows-powershell-answers.html

Laboratorio - Utilizzo di Wireshark per Esaminare il Traffico HTTP e HTTPS

In questo laboratorio, completa i seguenti obiettivi:

- Catturare e visualizzare il traffico HTTP
- Catturare e visualizzare il traffico HTTPS

https://itexamanswers.net/10-6-7-lab-using-wireshark-to-examine-http-and-https-traffic-answers.html

Bonus 1

Laboratorio - Esplorazione di Nmap

La scansione delle porte è solitamente parte di un attacco di ricognizione. Esistono diversi metodi di scansione delle porte che possono essere utilizzati.

https://itexamanswers.net/9-3-8-lab-exploring-nmap-answers.html

Bonus 2

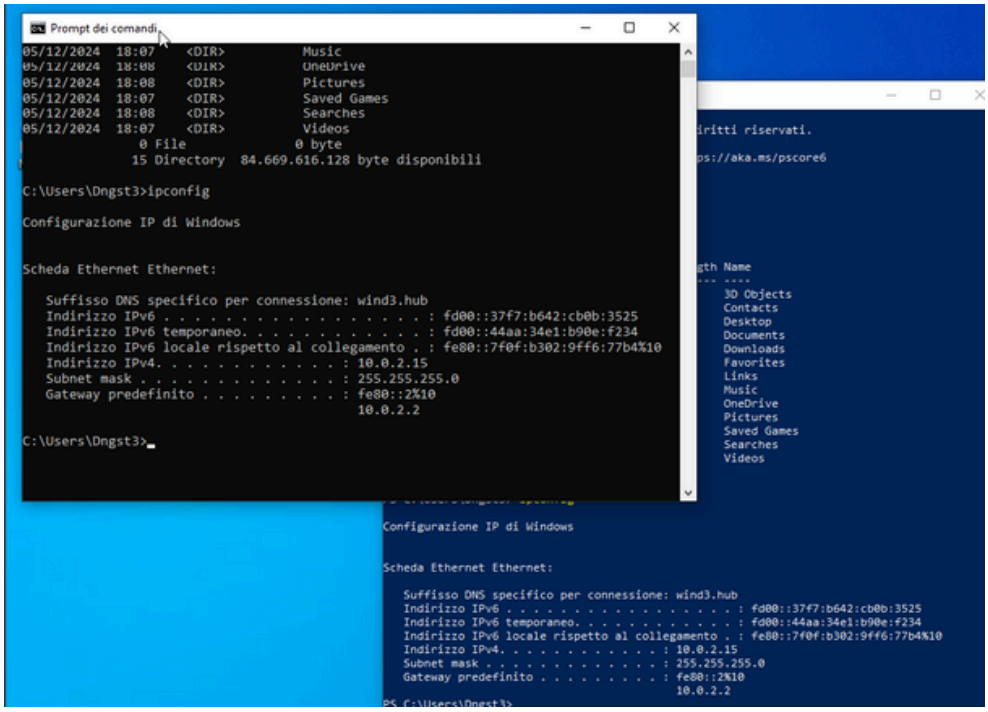
Attacco a un Database MySQL

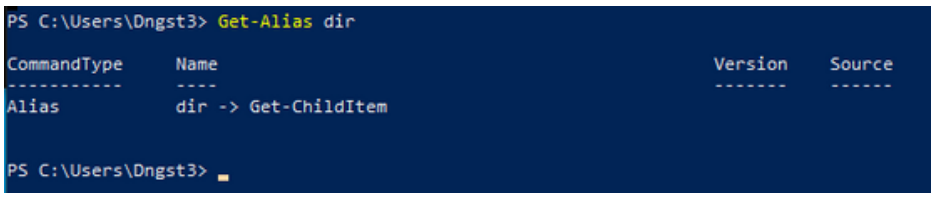
In questo laboratorio, completa il seguente obiettivo:

Visualizzare un file PCAP relativo a un attacco precedente contro un database SQL.

https://itexamanswers.net/17-2-6-lab-attacking-a-mysql-database-answers.html

Screen

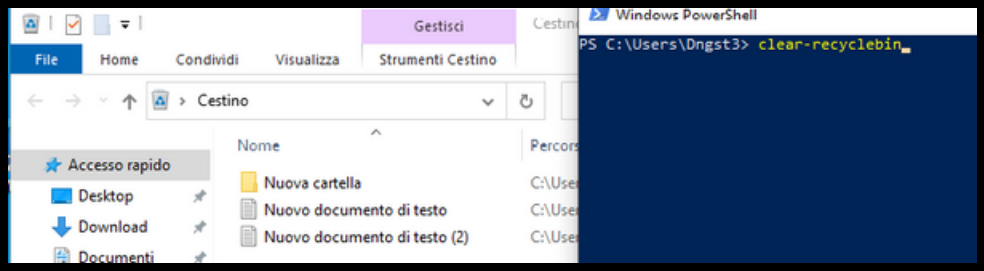




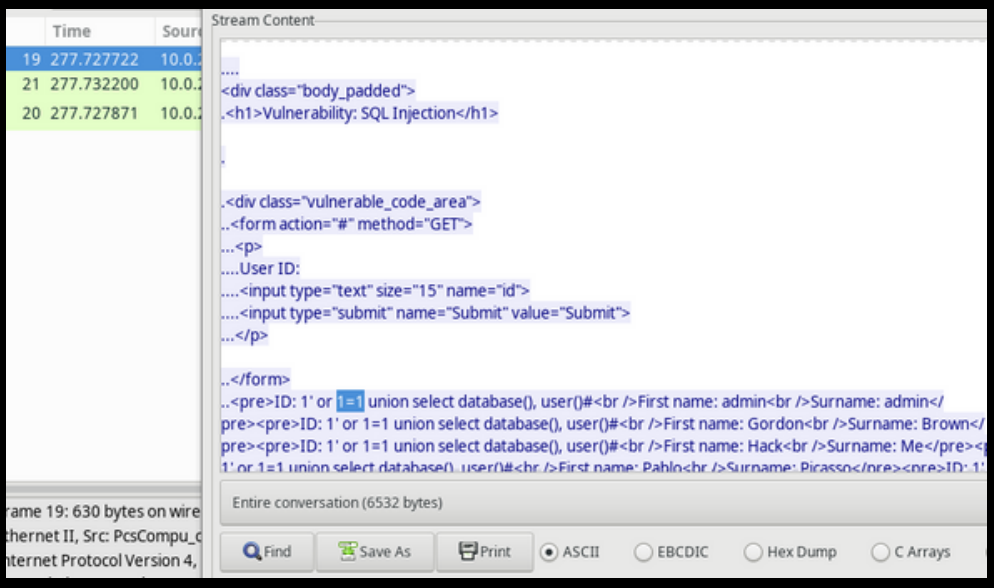
```
PS C:\Users\Dngst3> netstat

Connessioni attive

  Proto  Indirizzo locale       Indirizzo esterno       Stato
  TCP    10.0.2.15:53832        192.229.221.95:http     CLOSE_WAIT
  TCP    10.0.2.15:53862        4.231.66.184:https      ESTABLISHED
  TCP    10.0.2.15:53904        ppp-82-209:https        FIN_WAIT_2
  TCP    10.0.2.15:53907        a104-104-52-99:https    FIN_WAIT_2
  TCP    10.0.2.15:54004        98.64.238.3:https       ESTABLISHED
  TCP    10.0.2.15:54005        98.64.238.3:https       ESTABLISHED
  TCP    10.0.2.15:54006        192.229.221.95:http     CLOSE_WAIT
  TCP    10.0.2.15:54007        20.191.45.158:https     ESTABLISHED
  TCP    10.0.2.15:54008        20.191.45.158:https     ESTABLISHED
  TCP    10.0.2.15:54009        20.191.45.158:https     ESTABLISHED
  TCP    10.0.2.15:54011        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54012        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54013        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54014        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54015        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54016        ppp-188-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54079        13.107.21.239:https     TIME_WAIT
  TCP    10.0.2.15:54081        217.20.58.99:http       ESTABLISHED
  TCP    10.0.2.15:54091        ppp-115-209:https       CLOSE_WAIT
  TCP    10.0.2.15:54093        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54094        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54095        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54096        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54097        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54098        a104-104-52-91:https    CLOSE_WAIT
  TCP    10.0.2.15:54124        204.79.197.239:https    TIME_WAIT
  TCP    10.0.2.15:54128        13.107.246.60:https     FIN_WAIT_2
  TCP    10.0.2.15:54130        40.127.240.158:https    TIME_WAIT
  TCP    10.0.2.15:54131        40.127.240.158:https    TIME_WAIT
  TCP    10.0.2.15:54138        51.124.78.146:https     TIME_WAIT
  TCP    10.0.2.15:54139        51.124.78.146:https     TIME_WAIT
  TCP    10.0.2.15:54145        204.79.197.239:https    ESTABLISHED
  TCP    10.0.2.15:54146        204.79.197.239:https    ESTABLISHED
PS C:\Users\Dngst3>
```

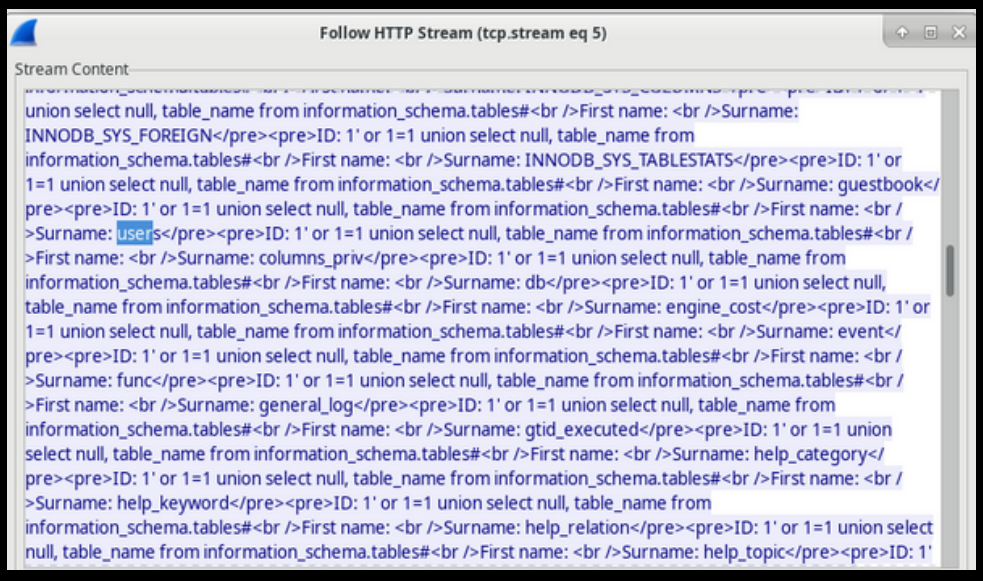
```
PS C:\Users\Dngst3> netstat -r
===========================================================================
Elenco interfacce
 10...08 00 27 86 34 36 ......Intel(R) PRO/1000 MT Desktop Adapter
  1...........................Software Loopback Interface 1
===========================================================================

IPv4 Tabella route
===========================================================================
Route attive:
     Indirizzo rete            Mask          Gateway     Interfaccia Metrica
          0.0.0.0          0.0.0.0    192.168.1.1    192.168.1.183     25
        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     192.168.1.183    281
    192.168.1.183  255.255.255.255         On-link     192.168.1.183    281
    192.168.1.255  255.255.255.255         On-link     192.168.1.183    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     192.168.1.183    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     192.168.1.183    281
===========================================================================
Route permanenti:
  Nessuna

IPv6 Tabella route
===========================================================================
Route attive:
 Interf Metrica Rete Destinazione      Gateway
  1    331 ::1/128                  On-link
 10    281 fe80::/64                On-link
 10    281 fe80::7f0f:b302:9ff6:77b4/128
                                    On-link
  1    331 ff00::/8                 On-link
 10    281 ff00::/8                 On-link
===========================================================================
Route permanenti:
  Nessuna
PS C:\Users\Dngst3>
```

```
PS C:\Windows\system32> netstat -abno

Connessioni attive

  Proto  Indirizzo locale          Indirizzo esterno        Stato           PID
  TCP    0.0.0.0:135             0.0.0.0:0              LISTENING       964
  RpcSs
 [svchost.exe]
  TCP    0.0.0.0:445             0.0.0.0:0              LISTENING       4
 Impossibile ottenere informazioni sulla proprietà
```




```
PS C:\Users\Dngst3> clear-recyclebin

Conferma
Eseguire l'operazione?
Esecuzione dell'operazione "Clear-RecycleBin" sulla destinazione "Tutto il cont
del Cestino".
[S] Sì  [T] Sì a tutti  [N] No  [U] No a tutti  [O] Sospendi  [?] Guida
(il valore predefinito è "S"):s
PS C:\Users\Dngst3>
```





Utilizzo di Windows PowerShell

Accesso alla Console di PowerShell

L'accesso alla console di PowerShell è stato effettuato tramite il menu Start, dimostrando la semplicità di utilizzo anche per utenti meno esperti. In parallelo, è stato utilizzato il prompt dei comandi per confrontare funzionalità e output.

Esplorazione dei Comandi dir, ping e ipconfig

Dall'esecuzione del comando dirin entrambe le console, l'output ha fornito un elenco dettagliato di sottodirectory e file. In PowerShell, l'aggiunta degli attributi/modalità ha mostrato una maggiore ricchezza di informazioni rispetto al prompt dei comandi. Anche altri comandi come ping, cde ipconfighanno restituito output simili, evidenziando la compatibilità tra le due console.

Utilizzo dei cmdlet di PowerShell

Con il comando **Get-Alias dir**, è stato possibile identificare che il cmdlet equivalente in PowerShell per dirè Get-ChildItem. Successivamente, la ricerca di ulteriori cmdlet su fonti ufficiali ha permesso di approfondire le possibilità offerte da PowerShell, ampliando la conoscenza di comandi specifici e delle loro applicazioni pratiche.

Analisi del Comando "netstat" tramite PowerShell

Il comando netstatè stato utilizzato per analizzare le connessioni di rete attive e la tabella di routing. L'esecuzione di netstat -rha permesso di visualizzare i percorsi attivi ei relativi gateway, identificando, ad esempio, il gateway IPv4 come 192.168.1.1.
L'uso del comando netstat -abno, seguito dall'analisi dei PID associati tramite il Task Manager, ha consentito di ottenere dettagli sui processi, come l'identificazione del PID 756 associato al processo svchost.exe, che utilizza il servizio NETWORK SERVICE con un consumo di memoria di 4132K .

Eliminazione del Cestino tramite PowerShell

L'utilizzo del comando Clear-Recyclebin ha permesso di eliminare permanentemente i file presenti nel Cestino. L'azione, confermata tramite un prompt, ha dimostrato come PowerShell possa semplificare le operazioni che richiederebbero più passaggi tramite interfaccia grafica.

Utilizzo di Wireshark per esaminare il traffico HTTP e HTTPS

Introduzione ai protocolli HTTP e HTTPS

HTTP è un protocollo che consente la trasmissione di dati tra il browser e il server, ma senza alcuna protezione. Le informazioni inviate, come credenziali o dati sensibili, sono visibili a chiunque intercetti il traffico. HTTPS, invece, utilizza la crittografia TLS/SSL per proteggere questi dati, ma non garantisce automaticamente la legittimità del sito web, poiché anche siti malevoli possono implementare HTTPS.

Analisi del traffico HTTP

Per iniziare, ho avviato la macchina virtuale CyberOps Workstation e utilizzata tcpdumpper catturare il traffico HTTP generato collegandomi al sito http://www.altoromutual .com/login.jsp. Il file .pcap generato è stato poi analizzato con Wireshark.
L'analisi ha mostrato chiaramente i dati scambiati: attraverso i filtri per HTTP, è stato possibile identificare un messaggio POST contenente le credenziali d'accesso (nome utente e password) trasmesse in chiaro. Questo dimostra come HTTP non protegge le informazioni sensibili, esponendole a potenziali intercettazioni.

Analisi del traffico HTTPS

Successivamente, ho ripetuto il processo utilizzando un sito HTTPS, come www.netacad.com . Anche in questo caso, ho catturato il traffico con tcpdumpe analizzato il file risultante in Wireshark.
A differenza del traffico HTTP, i dati scambiati tramite HTTPS risultano crittografati. Wireshark ha mostrato l'implementazione di TLS/SSL, che protegge le informazioni durante la trasmissione. Tuttavia, i dettagli del messaggio, inclusi eventuali credenziali o dati sensibili, non erano visibili poiché crittografati.

Esplorazione di Nmap

In questa parte, è stata utilizzata la pagina del manuale di Nmap per apprendere le sue funzionalità principali. Nmap è uno strumento di esplorazione della rete che consente di rilevare l'attività dell'host, scansionare le porte e determinare i servizi in esecuzione. La ricerca della parola "esempio" all'interno della pagina del manuale ha permesso di esplorare diverse opzioni di comando, tra cui l'interruttore -Ache abilita il rilevamento del sistema operativo, della versione, la scansione degli script e il traceroute, e l' interruttore -T4, che velocizza la scansione senza superare i 10 ms di ritardo per le porte TCP, rendendolo adatto per una rete a banda larga.

Scansione delle Porte Aperte

Scansione del Localhost:

Eseguendo la scansione del localhost con il comando nmap -A -T4 localhost, sono stati rilevati diversi servizi attivi, tra cui FTP sulla porta 21 con il servizio vsftpd e SSH sulla porta 22 con OpenSSH. Il comando ha identificato anche la possibilità di accesso FTP anonimo sulla macchina locale.

Scansione della Rete Locale:

Dopo aver determinato l'indirizzo IP della macchina (10.0.2.15) utilizzando il comando ip address, è stata eseguita la scansione della rete locale (10.0.2.0/24) con il comando nmap -A -T4 10.0.2.0/24. Durante la scansione, sono stati individuati diversi host attivi, tra cui il proprio sistema e altri dispositivi della rete. I risultati hanno mostrato che diverse porte erano aperte, inclusi FTP (porta 21), SSH (porta 22) e Telnet (porta 23), indicando una varietà di servizi in esecuzione sui dispositivi della rete locale.

Scansione di un Server Remoto (**scanme.nmap.org**)**:**

La scansione del server remoto scanme.nmap.orgha rivelato diverse porte aperte, tra cui la porta 22 per SSH, la porta 80 per HTTP (Apache HTTPD), e la porta 9929 per Nping echo. Queste informazioni sono state utilizzate per comprendere i servizi attivi su un server remoto e come Nmap rileva i servizi in esecuzione.

Attacco a un Database MySQL

In questo laboratorio, abbiamo esplorato un attacco di iniezione SQL contro un database MySQL utilizzando Wireshark per analizzare un file di cattura (PCAP) che documenta il traffico di rete relativo a un conto attacco. Lo scenario ha messo in evidenza come un attaccante possa sfruttare una minaccia di iniezione SQL per compromettere la sicurezza di un'applicazione web basata sul database.

Il laboratorio è stato suddiviso in diverse fasi, ognuna delle quali ha evidenziato aspetti chiave dell'attacco:

- Caricamento del file PCAP in Wireshark Dopo aver avviato Wireshark, abbiamo caricato il file SQL_Lab.pcap, che conteneva il traffico di rete di un attacco SQL. Questo file mostra la comunicazione tra due indirizzi IP coinvolti nell'attacco, ovvero 10.0.2.4 e 10.0.2.15. Il traffico catturato ha una durata complessiva di circa 8 minuti, durante i quali si sviluppa l'iniezione SQL.

- Inizio dell'attacco SQL Injection Abbiamo seguito il flusso di dati HTTP, identificando il punto in cui l'attaccante ha tentato di iniettare una query SQL (1=1) nel campo di ricerca UserID. Questa query è stata progettata per testare se l'applicazione fosse vulnerabile alle iniezioni SQL. Poiché l'applicazione ha risposto con un record del database invece di un messaggio di errore, l'attaccante ha confermato la presenza di una debolezza.

- Continuazione dell'attacco Nella fase successiva, l'attaccante ha ampliato l'iniezione SQL utilizzando una query più complessa per recuperare informazioni dal database. La query 1' or 1=1 union select database(), user()#ha permesso di ottenere dettagli sensibili, come il nome del database (dvwa) e l'utente del database (root@localhost).

- Raccolta di informazioni sul sistema L'attaccante ha continuato a sfruttare la vulnerabilità per raccogliere informazioni specifiche sul sistema, come la versione di MySQL utilizzata. La query 1' or 1=1 union select null, version()ha restituito la versione del database, che era MySQL 5.7.12-0.

- Esplorazione delle tabelle del database L'attaccante ha tentato di ottenere un elenco delle tabelle nel database utilizzando la query 1' or 1=1 union select null, table_name from information_schema.tables#. Successivamente, ha affinato la ricerca per ottenere informazioni specifiche sulle colonne della tabella degli utenti, cercando dettagli sensibili come i nomi utente e le password.

- Estrazione degli hash delle password Alla fine, l'attaccante ha cercato di ottenere gli hash delle password degli utenti con la query 1' or 1=1 union select user, password from users#. Uno degli utenti identificativi è risultato avere l'hash della password 8d3533d75ae2c3966d7e0d4fcc69216b, che successivamente è stato decifrato come la password "Carlo".

Riflessioni sul rischio degli attacchi di iniezione SQL:

L'iniezione SQL rappresenta una debolezza critica nelle applicazioni web che interagiscono con database. Se non gestita correttamente, può consentire agli attaccanti di ottenere accesso non autorizzato ai dati sensibili, modificare i contenuti del database, e compromettere l'integrità e la riservatezza del sistema.
Questo tipo di attacco è particolarmente pericoloso perché non richiede l'accesso diretto al sistema, ma può essere effettuato attraverso l'interfaccia web vulnerabile.

Metodi per prevenire gli attacchi di iniezione SQL:

- Filtraggio dell'input dell'utente: Valutare e sanificare tutti i dati immessi dagli utenti per prevenire l'inserimento di codice malizioso.
- Utilizzare query parametrizzate: Le query parametrizzate impediscono l'esecuzione di codice SQL arbitrario, separando i dati dagli script SQL.

Daniel_Gabriel_Costeanu