Progetto S11 – L5

Usare Windows PowerShell

Executive Summary

L’esercizio ha approfondito l’**utilizzo di Windows PowerShell per l’amministrazione e l’analisi del sistema**. Sono stati confrontati **CMD e PowerShell, analizzati cmdlet e connessioni di rete con netstat**, e svolte operazioni amministrative da console.

Introduzione

L’attività ha lo scopo di **acquisire familiarità con PowerShell**, **comprendere il funzionamento dei cmdlet e utilizzare la console per analisi di rete e gestione del sistema** in ottica sicurezza.

Obiettivi:

- Parte 1: Accedere alla console PowerShell
- Parte 2: Esplorare comandi Prompt dei Comandi e PowerShell
- Parte 3: Esplorare i cmdlet
- Parte 4: Esplorare netstat usando PowerShell
- Parte 5: Svuotare il cestino usando PowerShell

Parte 1 – Accedere alla console PowerShell

Istruzioni Parte 1

Accedere alla console PowerShell.

a. Faccio clic su **Start**, cerco e seleziono **PowerShell**.
b. Faccio clic su **Start**, cerco e seleziono **Prompt dei comandi (Command Prompt)**.

Parte 2 – Esplorare i comandi del Prompt dei Comandi e di PowerShell

a. Inserisco **dir** al prompt in entrambe le finestre (Prompt dei Comandi e PowerShell).

Domanda:

Quali sono gli output del comando dir?

Risposta:

In entrambi gli ambienti ottengo l'elenco dei file e delle cartelle presenti nella directory corrente.

Nel Prompt dei Comandi visualizzo un **output testuale classico, con intestazione, elenco file e riepilogo finale con numero totale di file e byte**.

In PowerShell visualizzo un output più strutturato, organizzato in colonne come **Mode, LastWriteTime, Length e Name**, perché PowerShell lavora con oggetti.

---

b. Provo altri comandi che ho usato nel Prompt dei Comandi, come **ping, cd e ipconfig**, inserendoli anche in PowerShell.

Domanda:

Quali sono i risultati?

Risposta:

- Con **ping 8.8.8.8** ottengo risposte ICMP con tempo di latenza (TTL) oppure un messaggio di timeout se l'host non è raggiungibile.
- Con **cd "C:\Program Files"** cambio la directory corrente.
- Con **ipconfig** visualizzo la configurazione di rete del sistema (indirizzo IP, subnet mask, gateway e DNS).

Verifico che i comandi tradizionali del Prompt dei Comandi funzionano anche in PowerShell, poiché molti sono eseguibili esterni richiamabili dalla console.

Parte 3 – Esplorare i cmdlet

a. So che i comandi PowerShell, chiamati cmdlet, sono costruiti nella forma verbo-nome.
Per identificare il comando PowerShell equivalente a **dir**, inserisco al prompt:
Get-Alias dir

Domanda:

Qual è il comando PowerShell per dir?

Risposta:

Osservo che il comando PowerShell equivalente a **dir** è **Get-ChildItem**.
Verifico quindi che **dir è un alias del cmdlet Get-ChildItem**.

---

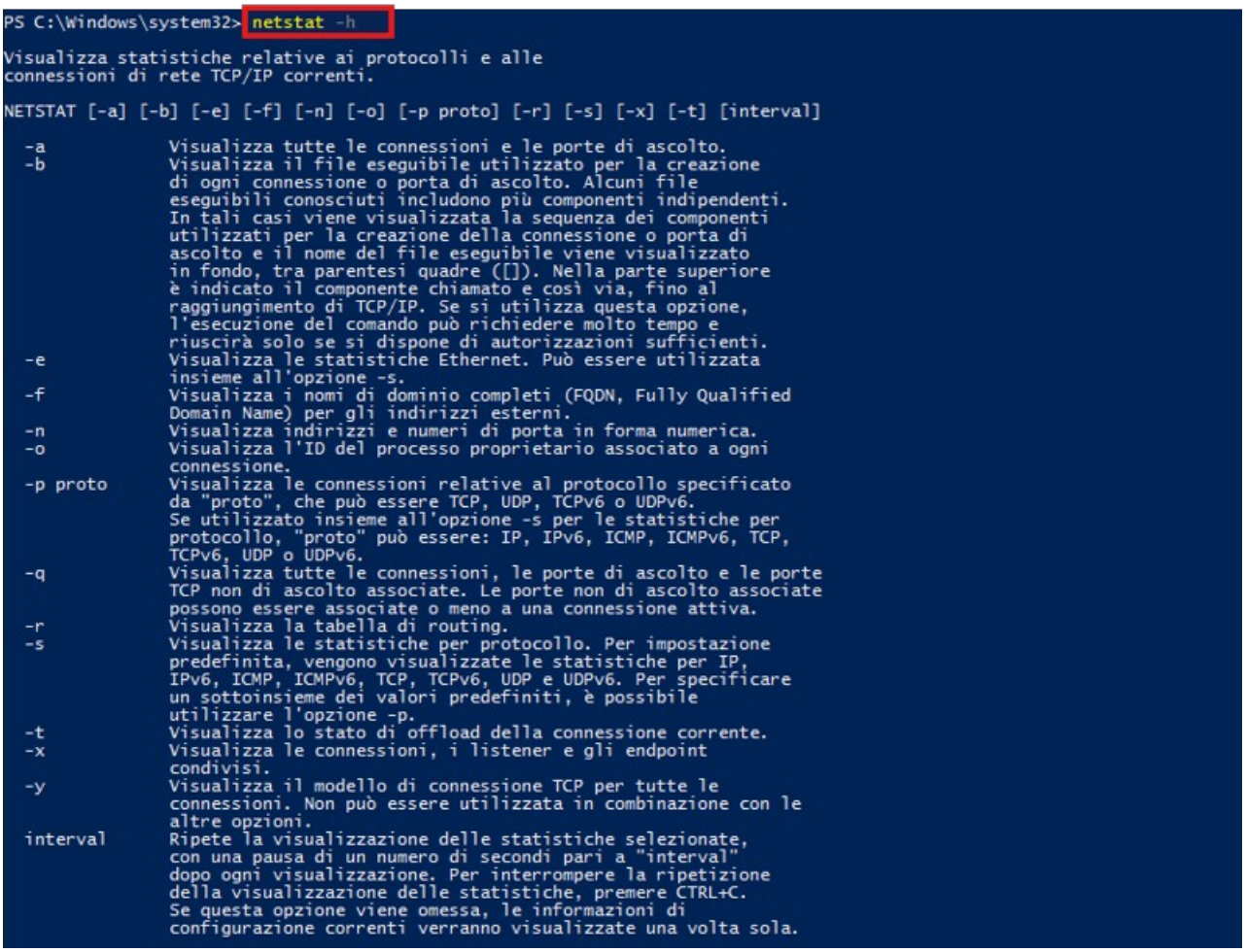
b. Per approfondire, eseguo una ricerca su internet digitando **Microsoft PowerShell cmdlets** per consultare la documentazione ufficiale.

c. Quando ho terminato, chiudo la finestra del Prompt dei Comandi.

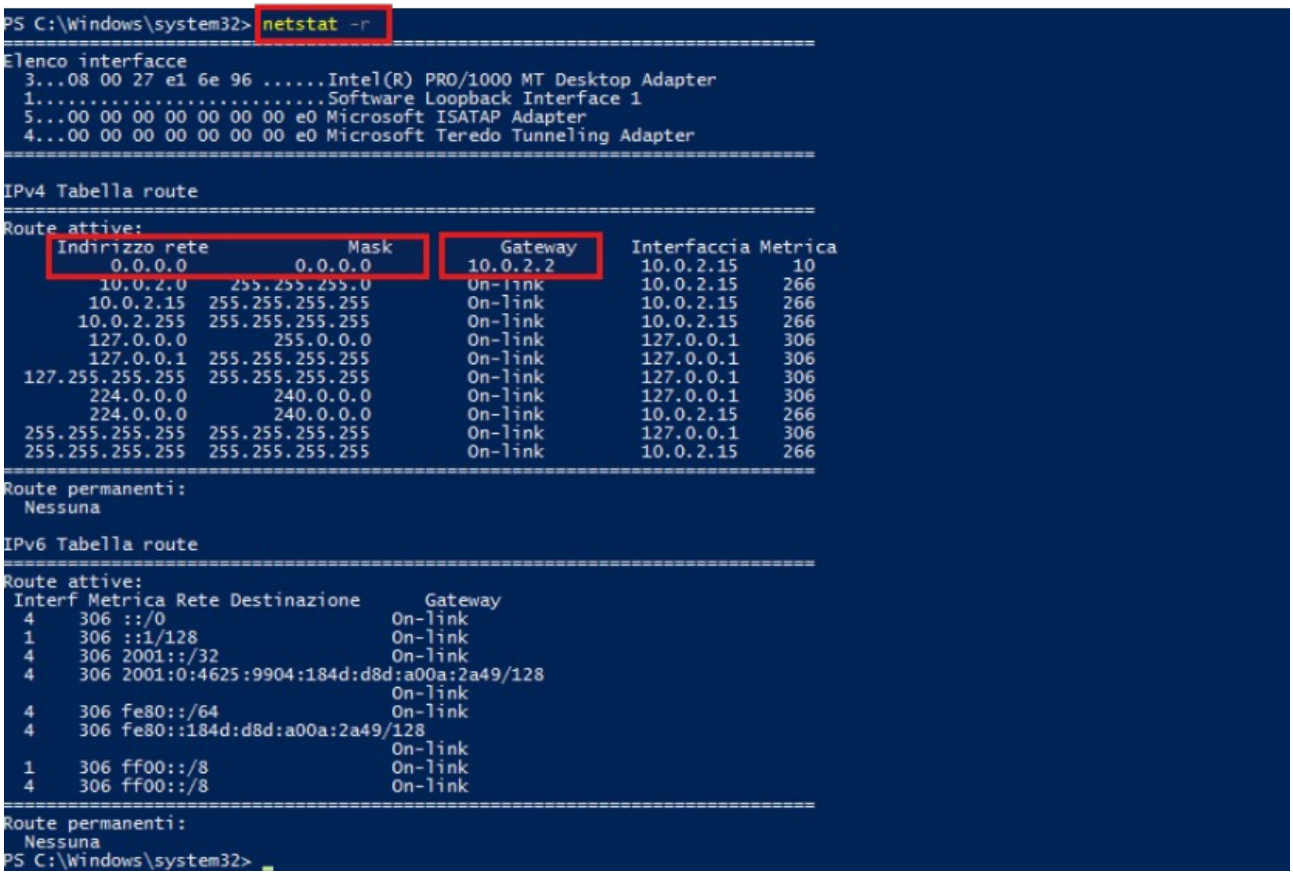
Parte 4 – Esplorare il comando netstat usando PowerShell

a. Al prompt di PowerShell inserisco:
netstat -h

per visualizzare tutte le opzioni disponibili del comando netstat.



b. Per visualizzare la tabella di routing con le rotte attive inserisco:
netstat -r



Domanda:

Qual è il gateway IPV4?

Risposta:

Osservo l’output del comando e individuo la riga della default route:
0.0.0.0 0.0.0.0

Nella colonna **Gateway** leggo l’indirizzo del gateway IPv4 predefinito.

Nel mio caso, il gateway IPv4 è:

10.0.2.2

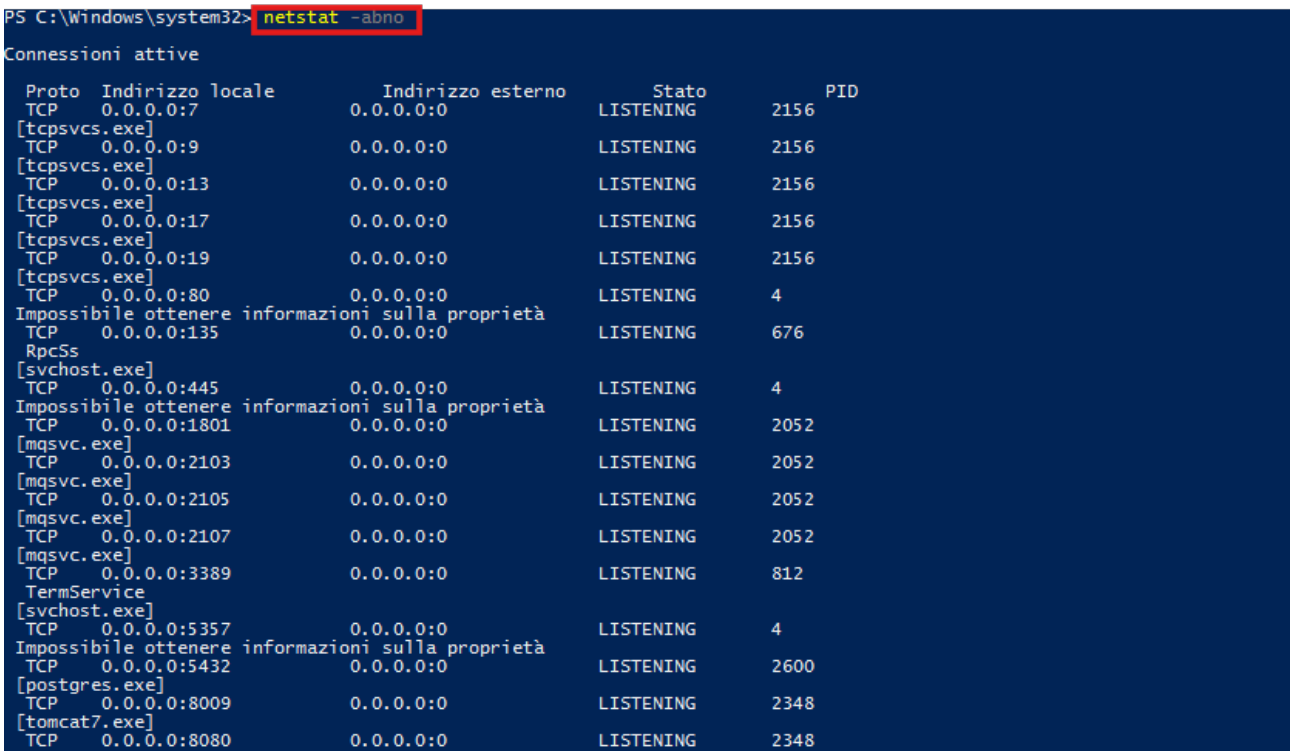
NB: L’indirizzo 10.0.2.2 indica che il sistema è probabilmente configurato in modalità NAT all’interno di una macchina virtuale.

---

c. Apro una seconda PowerShell con privilegi elevati.
Faccio clic su **Start**, cerco **PowerShell**, faccio clic con il pulsante destro su **Windows PowerShell** e seleziono **Esegui come amministratore**. Confermo con **Sì**.

---

d. Nella PowerShell amministratore inserisco:
netstat -abno



Visualizzo:

- **Protocollo** (TCP)
- **Indirizzo locale** (IP: porta del mio sistema)
- **Indirizzo esterno** (IP remoto o 0.0.0.0 se in ascolto)
- **Stato della connessione** (LISTENING nel mio caso)
- **PID del processo**
- **Nome dell’eseguibile associato alla porta**

---

e. Apro **Gestione Attività (Task Manager)** e navigo nella scheda **Dettagli (Details)**.
Faccio clic sull’intestazione **PID** per ordinare i processi in base al numero PID.

---

f. Seleziono uno dei PID visualizzati nell’output di **`netstat -abno`** (ad esempio PID 756).

g. Individuo lo stesso PID in Gestione Attività.
Faccio clic con il pulsante destro sul processo selezionato e apro la finestra **Proprietà (Properties)**.

Domanda:

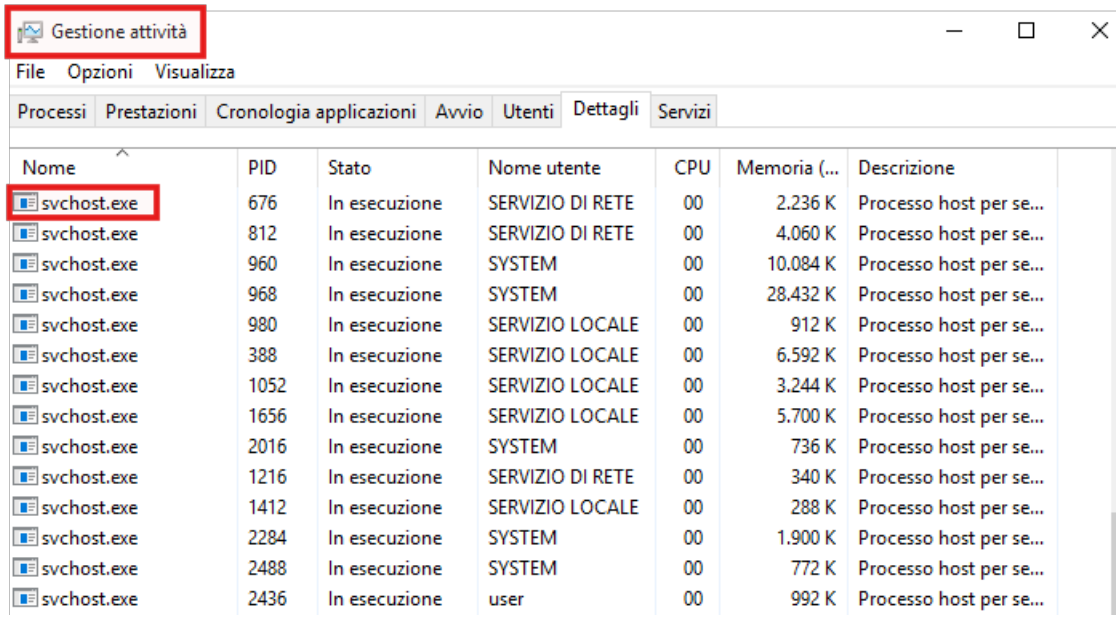
Quali informazioni puoi ottenere dalla scheda Dettagli e dalla finestra di dialogo Proprietà per il PID selezionato?

Risposta:

Dalla scheda **Dettagli** e dalla finestra **Proprietà** posso ottenere:

- Il nome del processo (svchost.exe)
- Il percorso completo dell'eseguibile
- Il produttore del software
- La versione del file
- La descrizione del processo
- L'utente che esegue il processo
- L'utilizzo di CPU e memoria
- Eventuale riga di comando
- La correlazione tra PID e porta/connessione individuata con `netstat -abno`

Queste informazioni mi permettono di capire quale applicazione sta utilizzando una determinata porta di rete. Questa attività dimostra la capacità di correlare porte di rete e processi in esecuzione, competenza fondamentale per l'analisi e il monitoraggio della sicurezza di sistema.

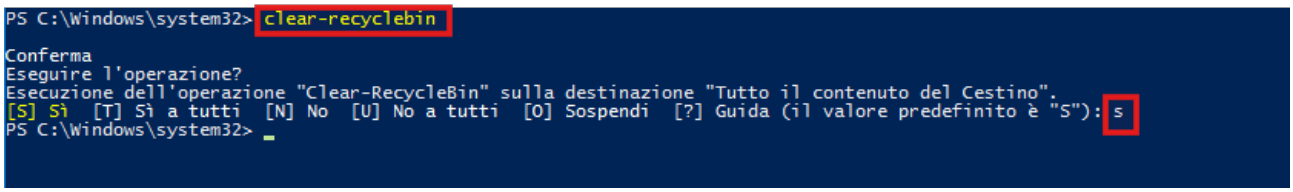


Nome	PID	Stato	Nome utente	CPU	Memoria (...	Descrizione
svchost.exe	676	In esecuzione	SERVIZIO DI RETE	00	2.236 K	Processo host per se...
svchost.exe	812	In esecuzione	SERVIZIO DI RETE	00	4.060 K	Processo host per se...
svchost.exe	960	In esecuzione	SYSTEM	00	10.084 K	Processo host per se...
svchost.exe	968	In esecuzione	SYSTEM	00	28.432 K	Processo host per se...
svchost.exe	980	In esecuzione	SERVIZIO LOCALE	00	912 K	Processo host per se...
svchost.exe	388	In esecuzione	SERVIZIO LOCALE	00	6.592 K	Processo host per se...
svchost.exe	1052	In esecuzione	SERVIZIO LOCALE	00	3.244 K	Processo host per se...
svchost.exe	1656	In esecuzione	SERVIZIO LOCALE	00	5.700 K	Processo host per se...
svchost.exe	2016	In esecuzione	SYSTEM	00	736 K	Processo host per se...
svchost.exe	1216	In esecuzione	SERVIZIO DI RETE	00	340 K	Processo host per se...
svchost.exe	1412	In esecuzione	SERVIZIO LOCALE	00	288 K	Processo host per se...
svchost.exe	2284	In esecuzione	SYSTEM	00	1.900 K	Processo host per se...
svchost.exe	2488	In esecuzione	SYSTEM	00	772 K	Processo host per se...
svchost.exe	2436	In esecuzione	user	00	992 K	Processo host per se...

Parte 5 – Svuotare il cestino usando PowerShell

a. Apro il **Cestino** e verifico che siano presenti elementi eliminabili permanentemente.
Se non ci sono file, creo un file di testo con Notepad e lo sposto nel Cestino.

b. Apro PowerShell e inserisco:
clear-recyclebin



Confermo l'operazione premendo **Y**.

Domanda:

Cosa è successo ai file nel Cestino?

Risposta:

Dopo aver eseguito il comando e confermato l'operazione, i file presenti nel Cestino vengono eliminati definitivamente dal sistema.

Domanda di Riflessione

Domanda:

PowerShell è stato sviluppato per l'automazione delle attività e la gestione della configurazione. Usando internet, ricerca comandi che potresti usare per semplificare i tuoi compiti come analista di sicurezza. Registra le tue scoperte.

Risposta:

Dalla mia ricerca ho individuato alcuni cmdlet utili per un analista di sicurezza:

Rete e connessioni

- Test-NetConnection
- Get-NetTCPConnection
- Get-NetIPAddress
- Get-NetRoute

Processi e servizi

- Get-Process
- Get-Service
- Get-ScheduledTask

Log e monitoraggio

- Get-WinEvent
- Get-EventLog

Analisi file e IOC

- Get-FileHash
- Get-ChildItem -Recurse
- Select-String

Microsoft Defender

- Get-MpComputerStatus
- Get-MpThreat
- Update-MpSignature

Reporting e audit

- Export-Csv
- ConvertTo-Json
- Start-Transcript

Questi comandi mi permettono di automatizzare attività di monitoraggio, analisi e verifica in ambito cybersecurity.

Conclusioni

L’esercitazione ha dimostrato come **PowerShell rappresenti uno strumento fondamentale per amministrazione e cybersecurity**.

La capacità di a**nalizzare processi, connessioni e automatizzare attività operative rende PowerShell una risorsa strategica per un analista di sicurezza**.
Questa attività dimostra la capacità di correlare porte di rete e processi in esecuzione.

L’obiettivo è stato raggiunto con successo.