Utilizzo di Windows PowerShell

L’obiettivo principale di questo esercizio è acquisire familiarità con alcune delle funzionalità base di PowerShell, esplorando comandi comuni, cmdlet e strumenti di diagnostica di rete.
Le attività si suddividono in:

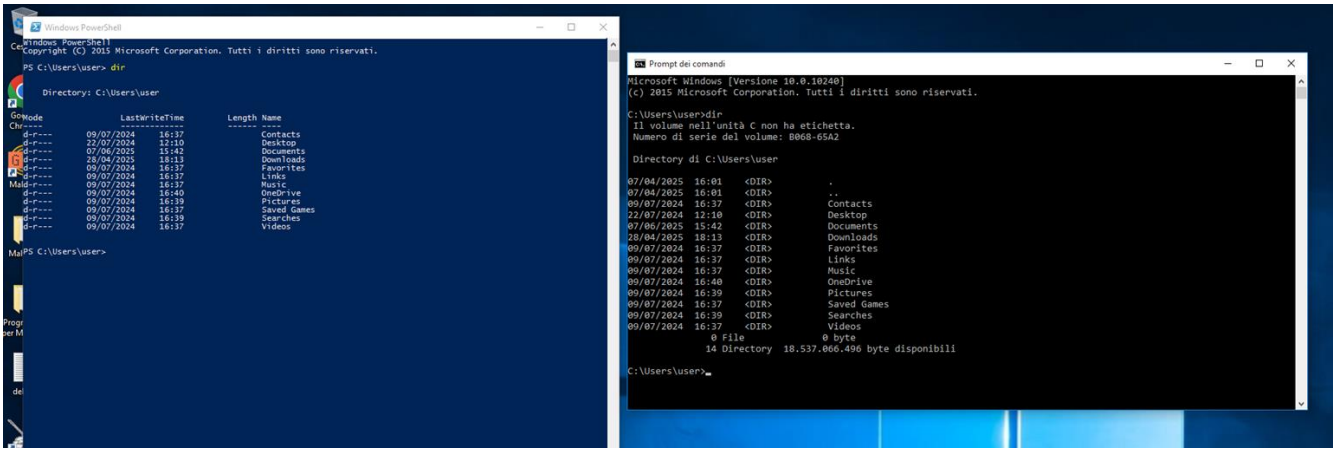
- Accesso alla console PowerShell
- Confronto tra comandi del Prompt dei Comandi e PowerShell
- Analisi dei cmdlet
- Utilizzo del comando `netstat` per la rete
- Pulizia del Cestino tramite comando PowerShell

Parte 1: Avvio di PowerShell

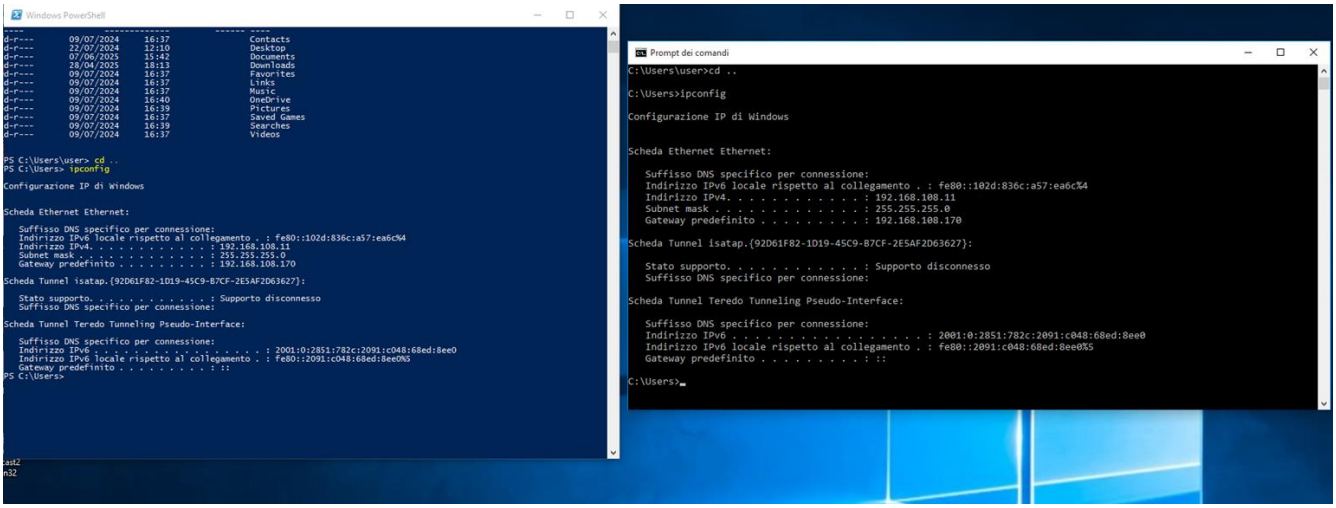
La prima fase ha previsto l’apertura di PowerShell dal menu Start. In parallelo è stato avviato anche il Prompt dei Comandi, per permettere un confronto diretto tra i due ambienti. Sebbene visivamente simili, PowerShell si distingue per il supporto a comandi più evoluti e una struttura più orientata alla programmazione e automazione.

Parte 2: Confronto tra comandi del Prompt e di PowerShell

Per iniziare, è stato utilizzato il comando dir in entrambi gli ambienti. Il risultato è stato simile: l’elenco dei file e delle cartelle della directory corrente. Tuttavia, PowerShell fornisce un output più chiaro e dettagliato.



Successivamente sono stati testati comandi di rete come cd .. e ipconfig. Entrambi funzionano anche in PowerShell, dimostrando una compatibilità con i comandi classici del Prompt, pur offrendo maggiore versatilità.

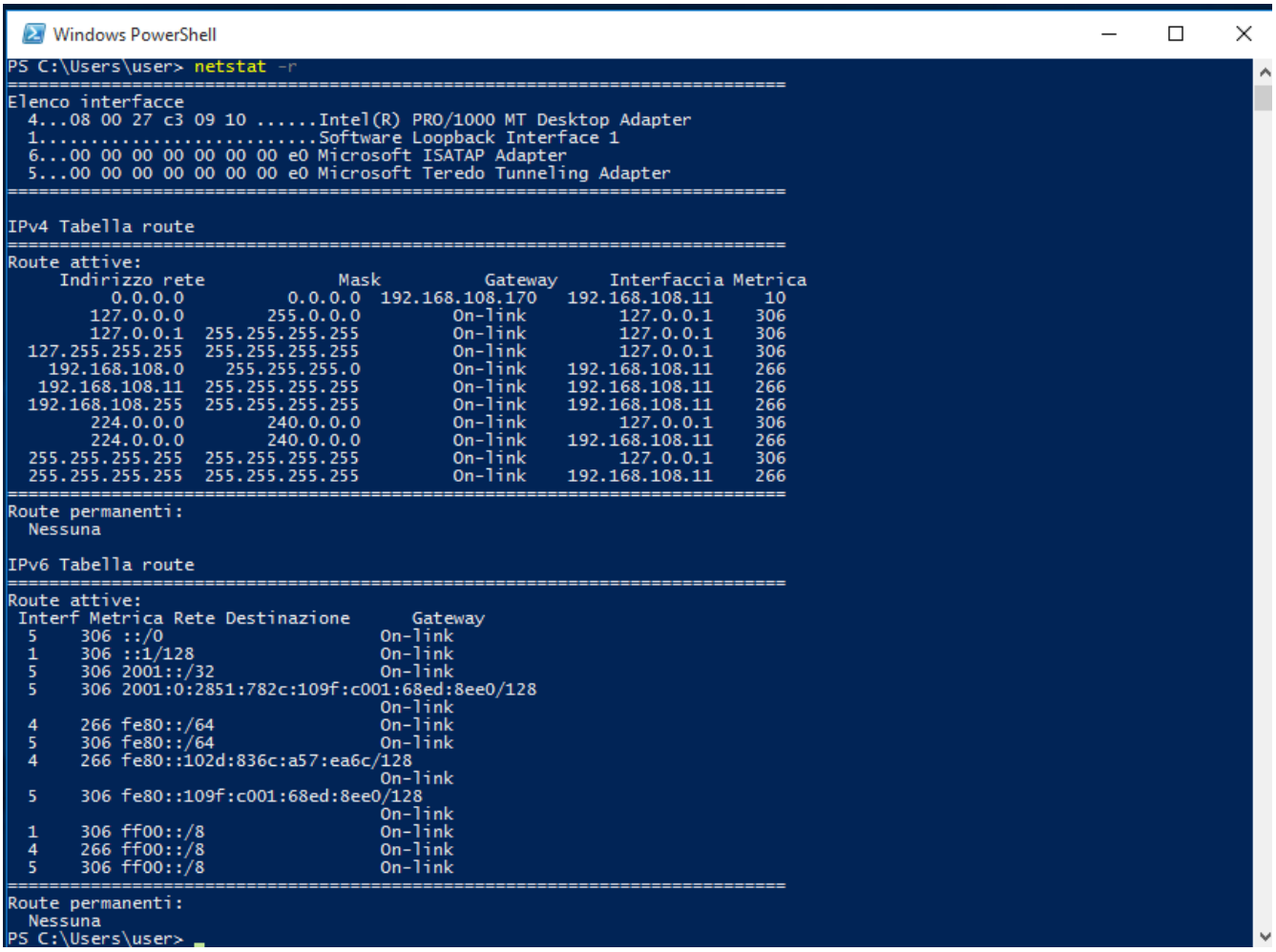


Parte 3: Scoperta dei cmdlet

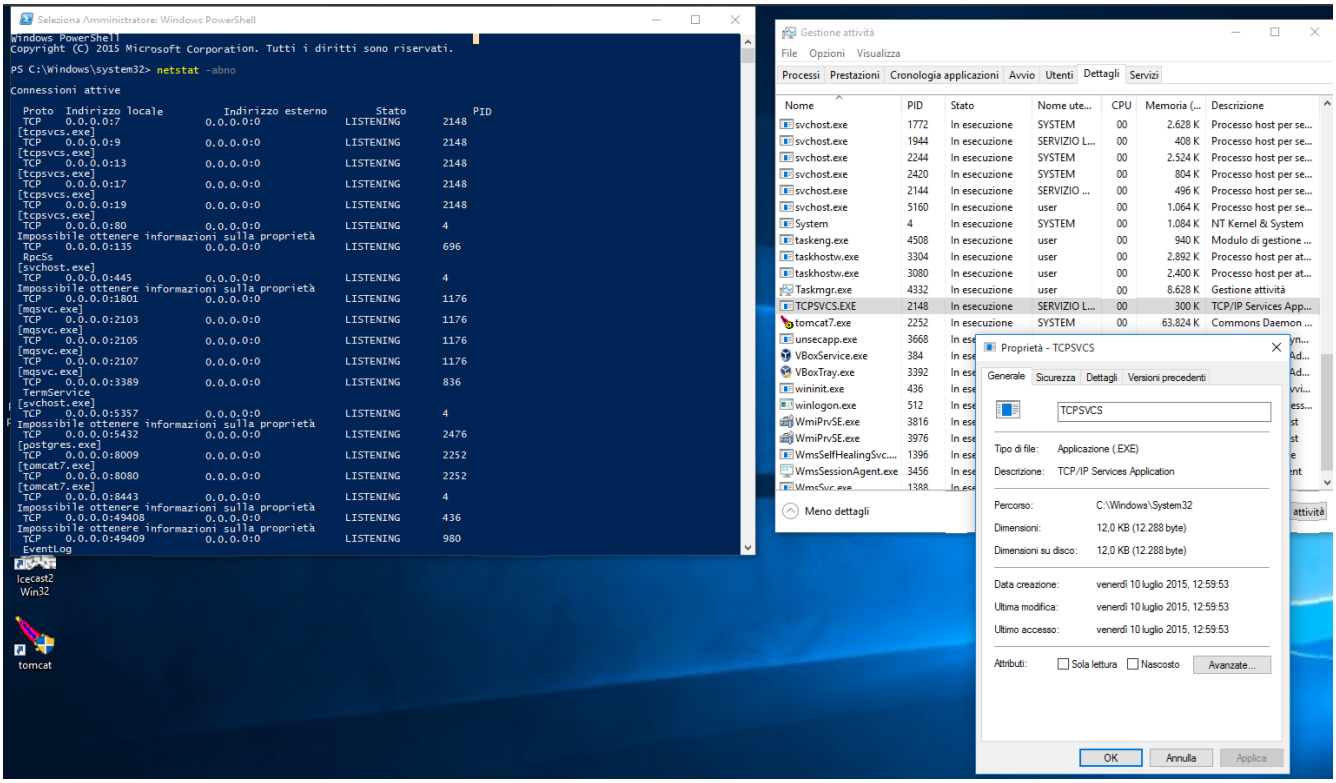
PowerShell introduce i **cmdlet**, comandi strutturati nella forma Verbo-Nome (es. Get-Process, Set-Date), pensati per essere autoesplicativi.
Ad esempio, il comando dir in PowerShell è in realtà un alias di Get-ChildItem. Questo approccio favorisce una maggiore coerenza e leggibilità, soprattutto quando si creano script più complessi per l'automazione.

Parte 4: Analisi di rete con netstat

Utilizzando il comando netstat -r, è stato possibile consultare la tabella di routing del sistema. Il dato più rilevante è stato l'individuazione del **gateway IPv4** (192.168.108.170), che rappresenta il punto di accesso verso la rete esterna.

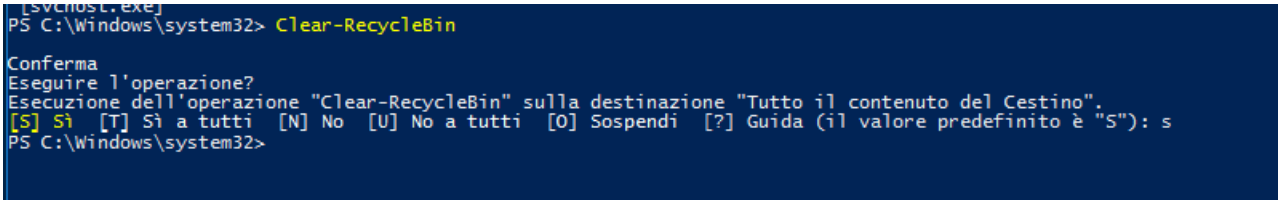


Con netstat -abno, è stata visualizzata la lista delle connessioni attive, assieme ai processi che le gestiscono. È stato analizzato in particolare il **PID 2148**, associato al processo tcpsvcs.exe. Esplorandolo tramite **Gestione Attività**, la finestra "Proprietà" ha mostrato informazioni chiave come il percorso eseguibile, il produttore Microsoft e la firma digitale. Questi dati sono fondamentali per riconoscere processi legittimi da quelli potenzialmente malevoli.



Parte 5: Svuotare il Cestino con PowerShell

Infine, è stato provato il comando Clear-RecycleBin per svuotare il Cestino. Dopo avervi spostato manualmente alcuni file, il comando è stato eseguito, eliminando definitivamente i contenuti senza dover ricorrere all'interfaccia grafica.



Riflessione: PowerShell per la Sicurezza Informatica

PowerShell è particolarmente utile nel contesto della cybersecurity grazie alla sua capacità di automatizzare compiti e raccogliere informazioni in modo efficiente. Alcuni comandi rilevanti per la sicurezza includono:

- Get-EventLog – per consultare i log di sistema, utile in fase di auditing o incident response.
- Get-Process – per monitorare i processi in esecuzione.
- Get-NetTCPConnection – per analizzare le connessioni di rete e identificare eventuali attività sospette.

- Get-LocalUser e Get-LocalGroup – per gestire utenti e gruppi locali e controllare accessi non autorizzati.

Conclusione

PowerShell si è dimostrato uno strumento avanzato per la gestione dei sistemi Windows, combinando la potenza di uno scripting engine con la semplicità di una console testuale.
Rispetto al Prompt dei Comandi, offre funzionalità decisamente più evolute, rendendolo indispensabile in contesti professionali dove automazione, controllo e sicurezza sono elementi cruciali.