Progetto S11/L5

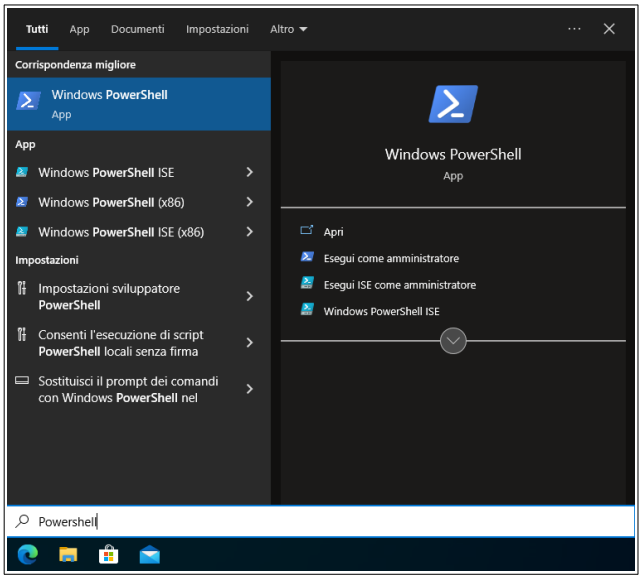
Utilizzo di PowerShell

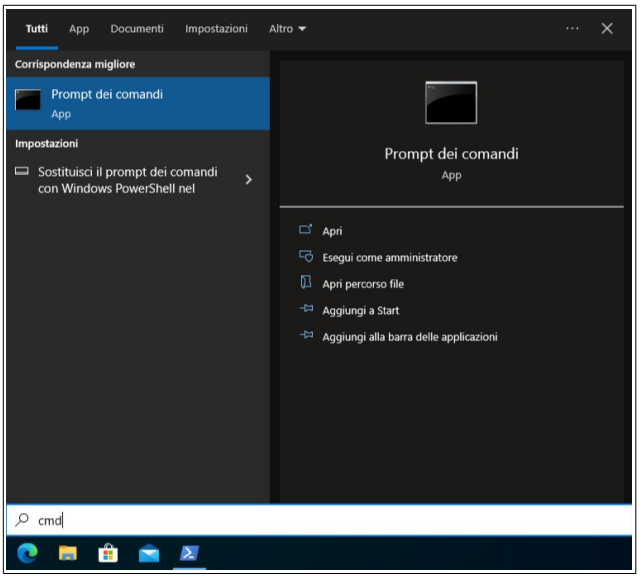
Introduzione

L’obiettivo di questo esercizio è stato esplorare le funzionalità di Windows PowerShell.

Parte 1: Accedere alla console PowerShell.

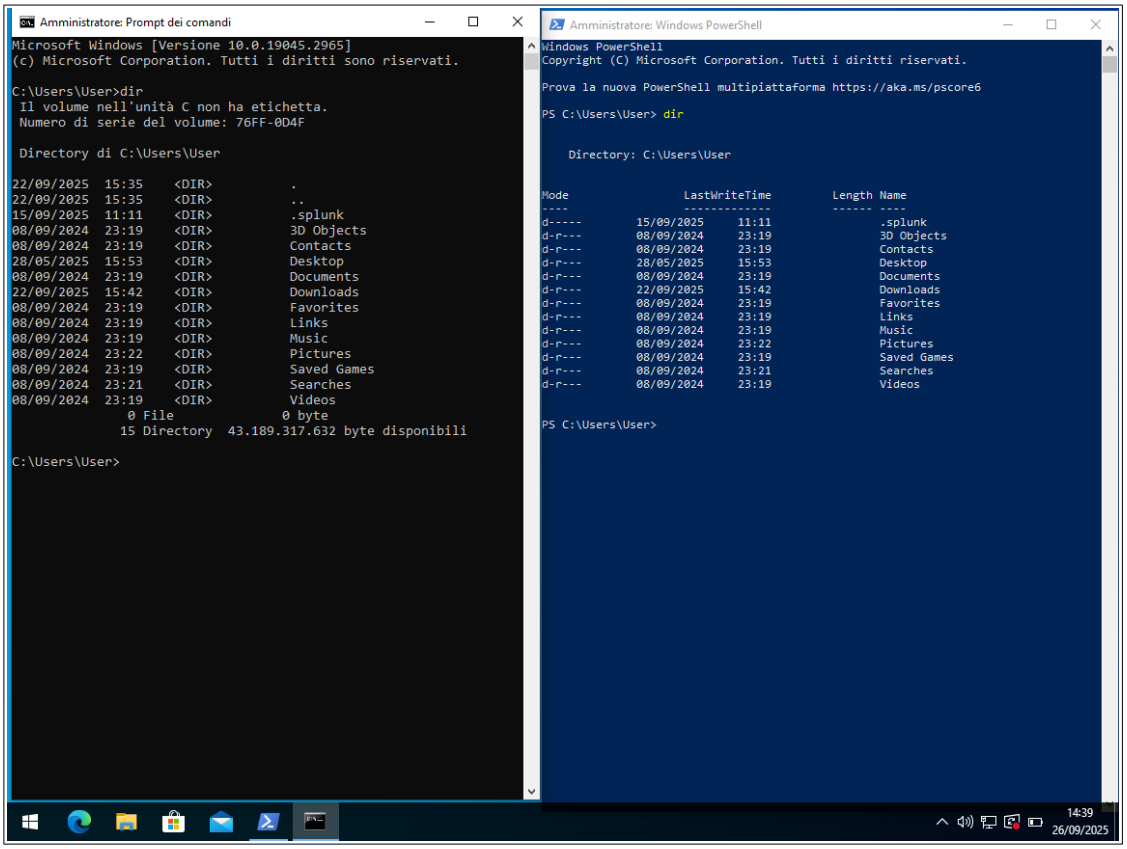
Il primo passo è stato accedere alla PowerShell e al Prompt dei Comandi per fare un confronto tra le due console.





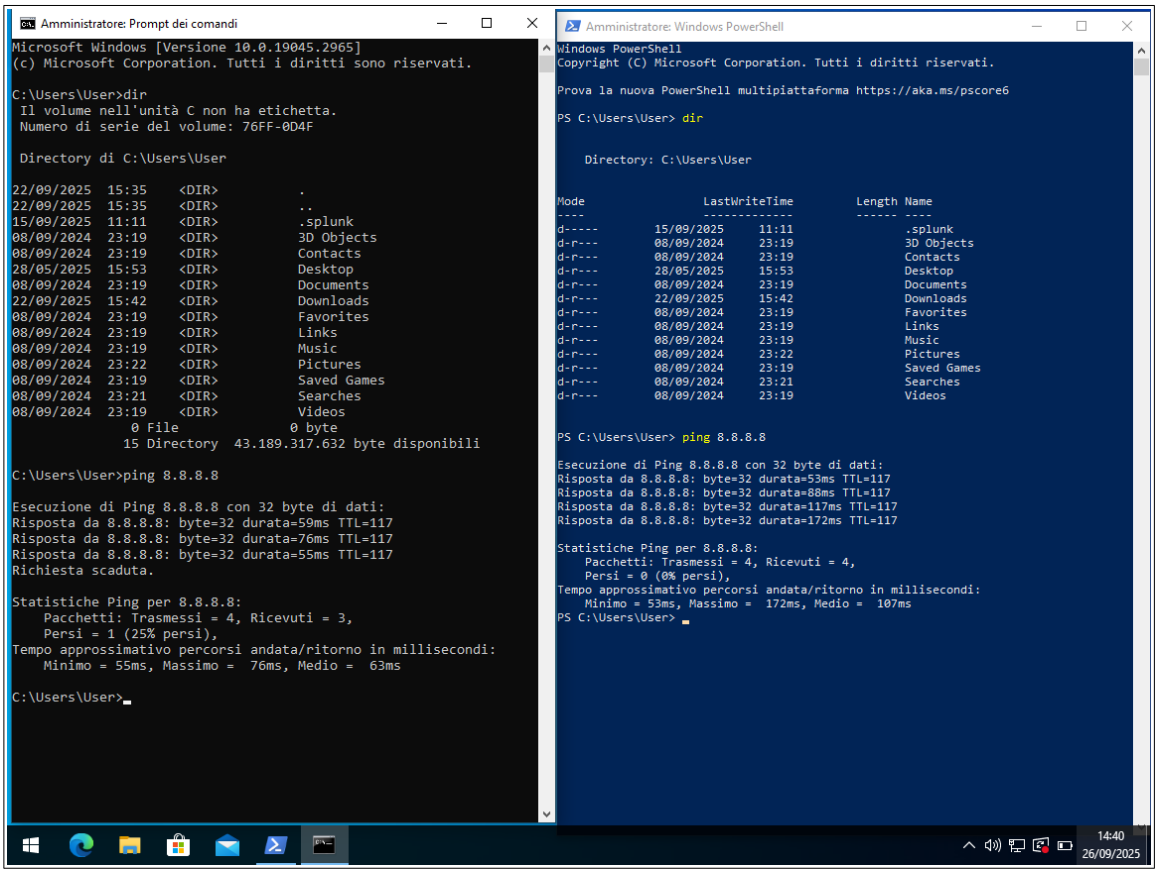
Parte 2: Esplorare i comandi del Prompt dei Comandi e di PowerShell.

Dopodiché ho digitato lo stesso comando su entrambe le console ovvero “dir”.



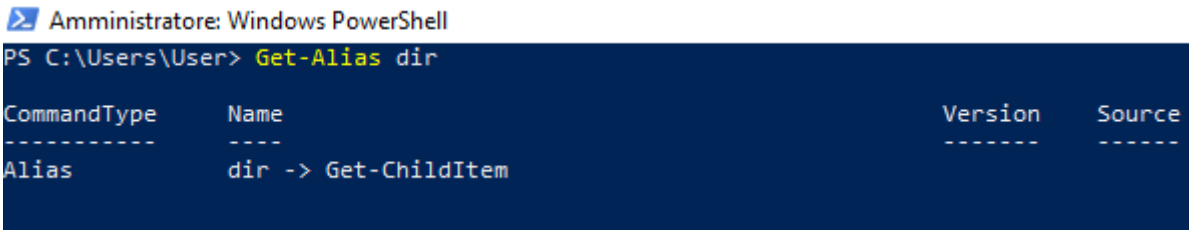
Quali sono gli output del comando dir?
Elenca file e sottocartelle nella directory corrente con attributi: data/ora, dimensione e nome.

Come passo successivo ho provato ad eseguire un “ping”su entrambe le console.



Quali sono i risultati?
Entrambe le console hanno dato lo stesso risultato.

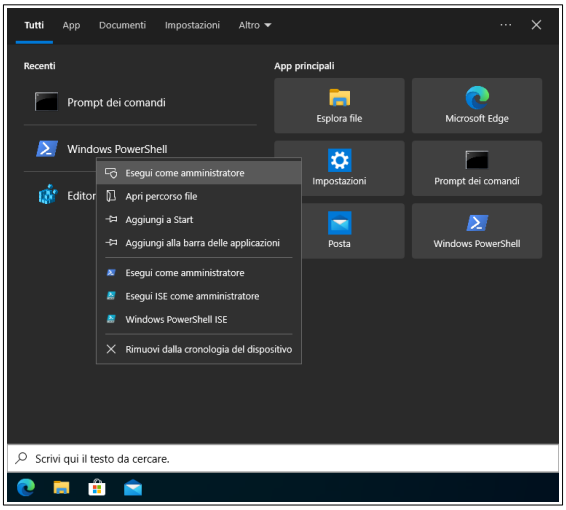
Parte 3: Esplorare i cmdlet.
Per esplorare i comandi PowerShell, chiamati cmdlet, è stato inserito il comando “Get-Alias” per elencare le sottodirectory e i file di una directory.



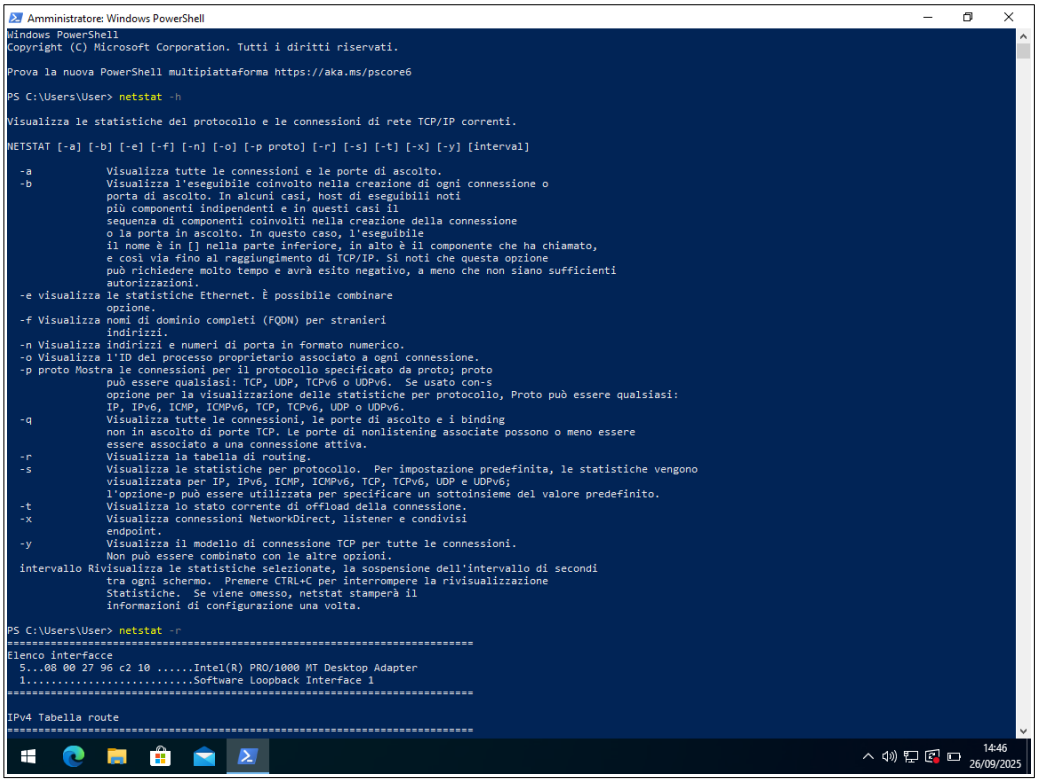
Qual è il comando PowerShell per dir?
Get-ChildItem

I cmdlet di PowerShell sono comandi nativi con sintassi verbo–nome (ad esempio Get-Process, Set-ExecutionPolicy) che operano su oggetti .NET invece che su testo, consentendo una pipeline ricca e affidabile. La documentazione ufficiale Microsoft spiega struttura, parametri, esempi e note d'uso; si possono scoprire cmdlet disponibili con Get-Command, approfondire la guida integrata con Get-Help e aprire la pagina online con Get-Help -Online, aggiornando i contenuti locali tramite Update-Help. I moduli estendono l'ambiente introducendo nuovi cmdlet specifici per ruoli e prodotti, individuabili con Get-Module e installabili da repository come PSGallery. Rispetto ai comandi tradizionali del Prompt, i cmdlet favoriscono automazione e gestione remota in modo coerente e ripetibile, risultando particolarmente utili per attività di amministrazione e sicurezza.

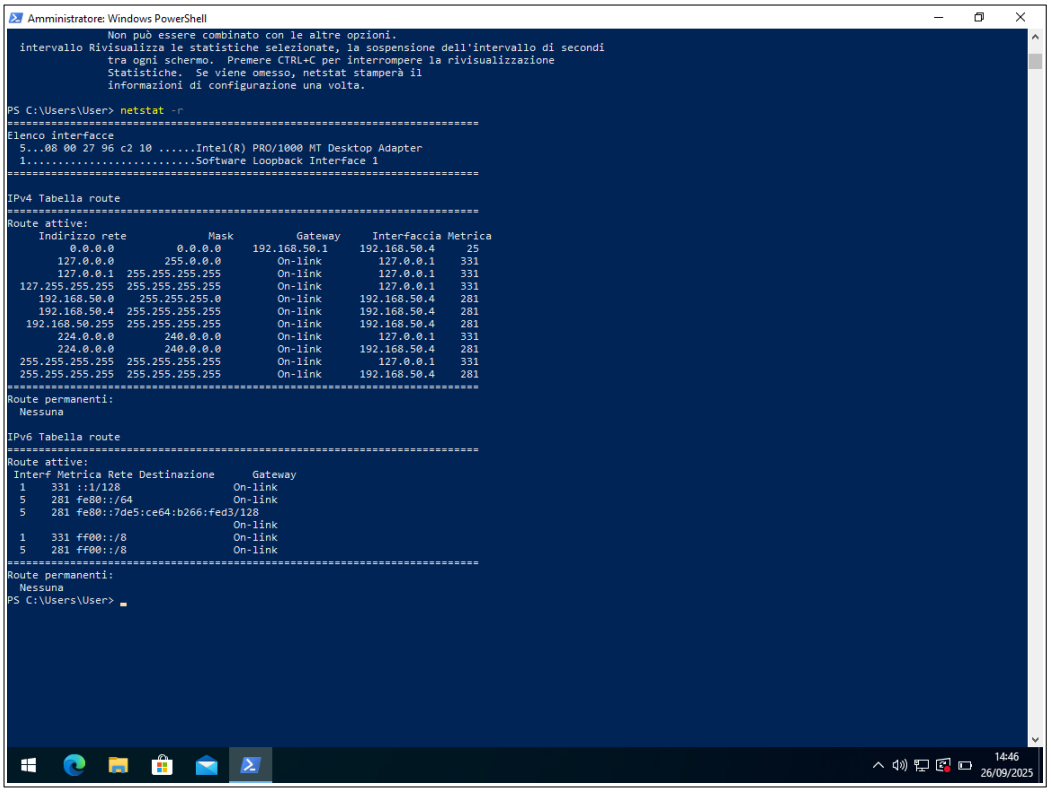
Parte 4: Esplorare il comando netstat usando PowerShell.
Per utilizzare alcuni comandi "netstat" (tipo "netstat -abno") è necessario avviare PowerShell come amministratore.



1) netstat -h (elenco dei parametri disponibili e la descrizione delle opzioni)

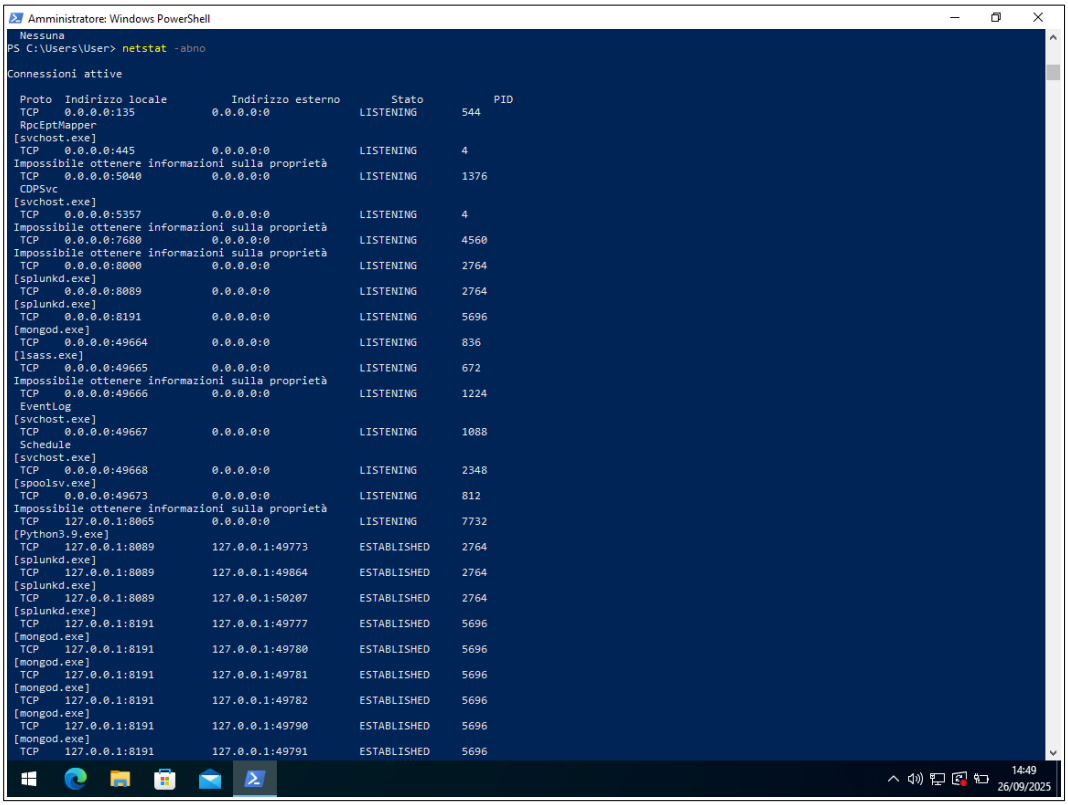


2) netstat -r (tabella di routing del sistema)

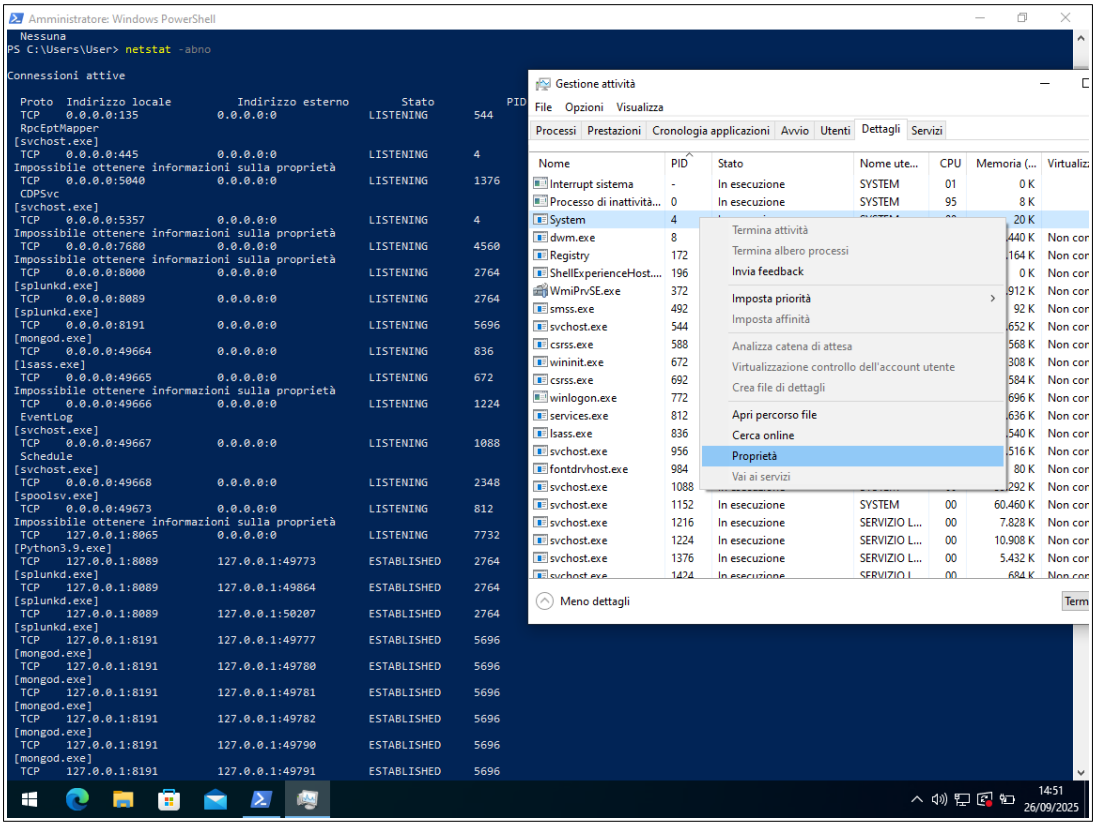


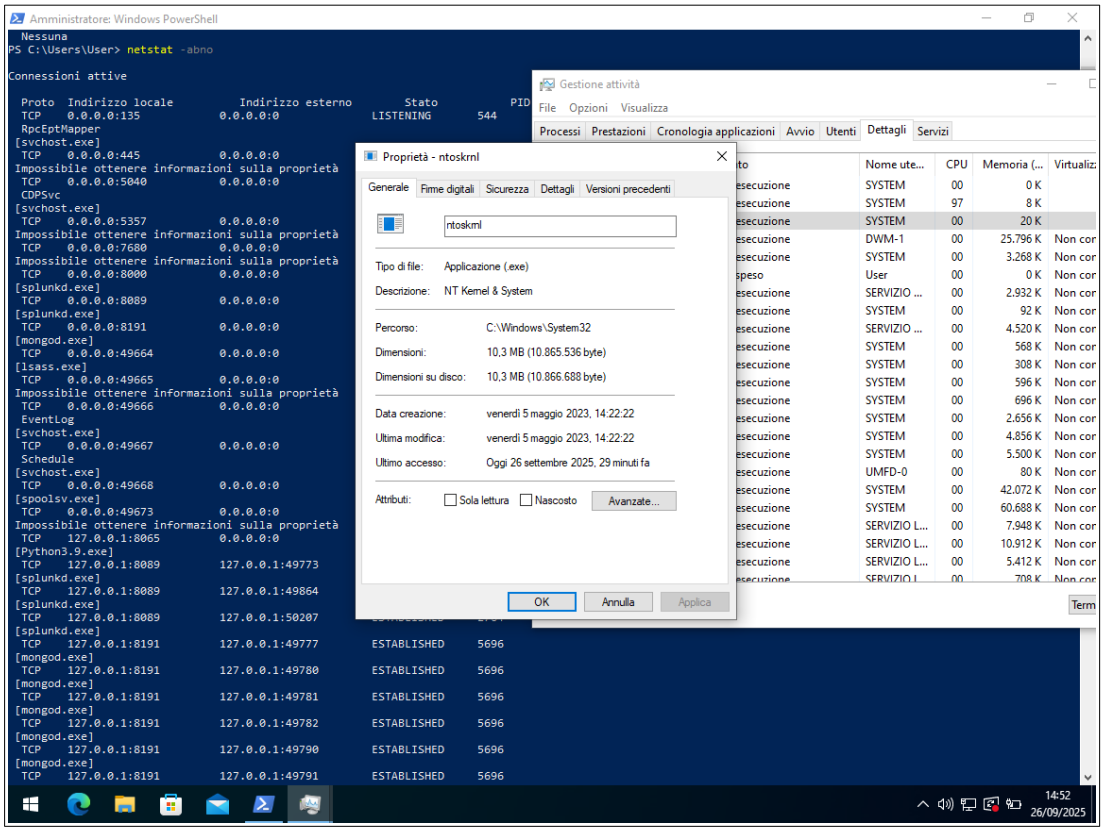
Qual è il gateway IPv4?
192.168.50.1

3) netstat -abno (connessioni di rete attive con informazioni molto dettagliate)



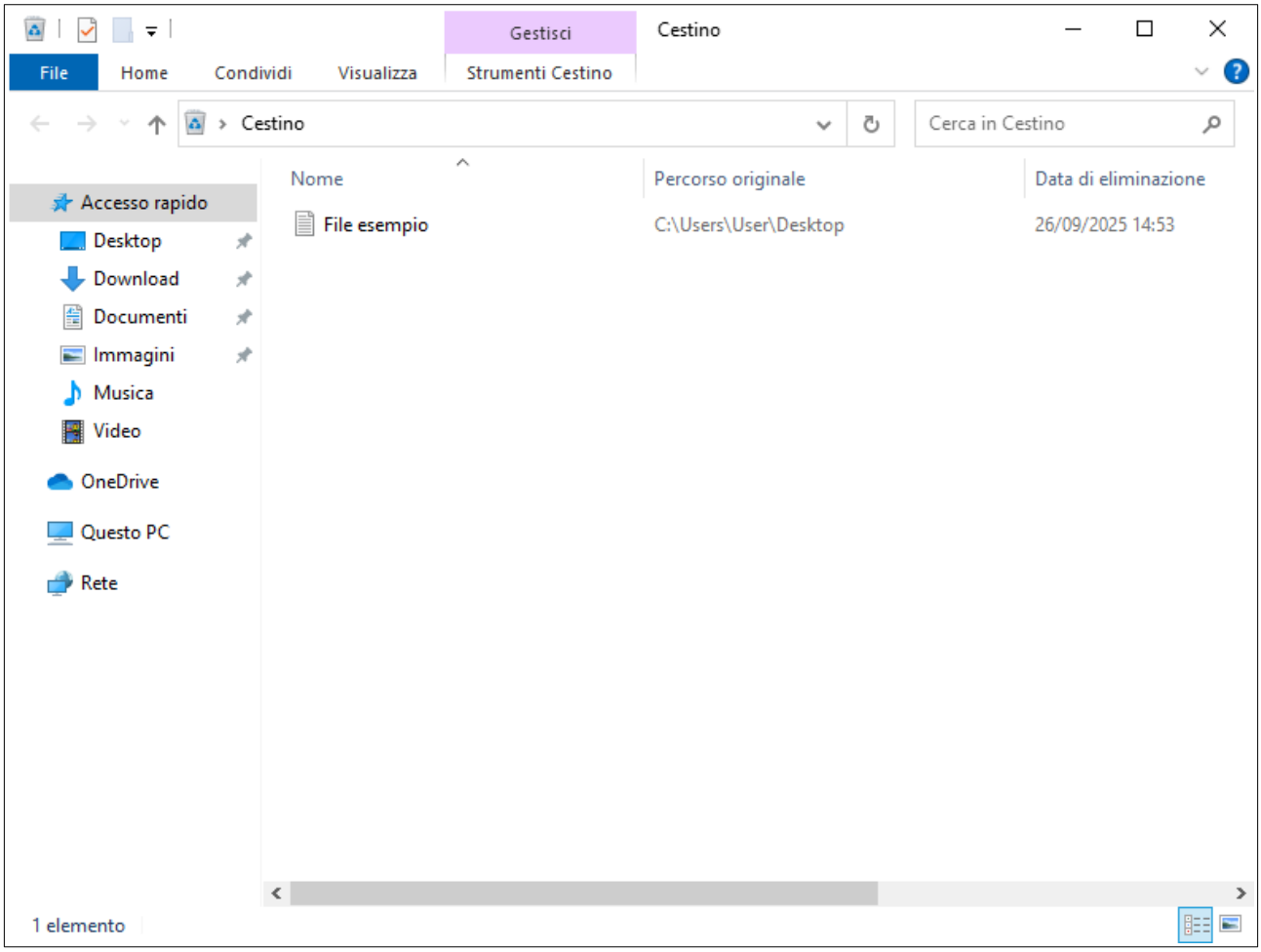
Nell’output di netstat -abno sono presenti dei PID ovvero dei numeri univoci che il sistema operativo assegna a ogni processo in esecuzione. Le proprietà dei PID possono essere analizzati dal Task Manager.

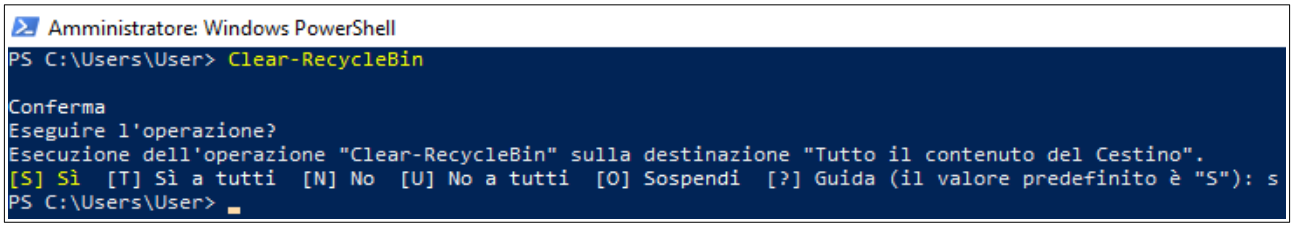




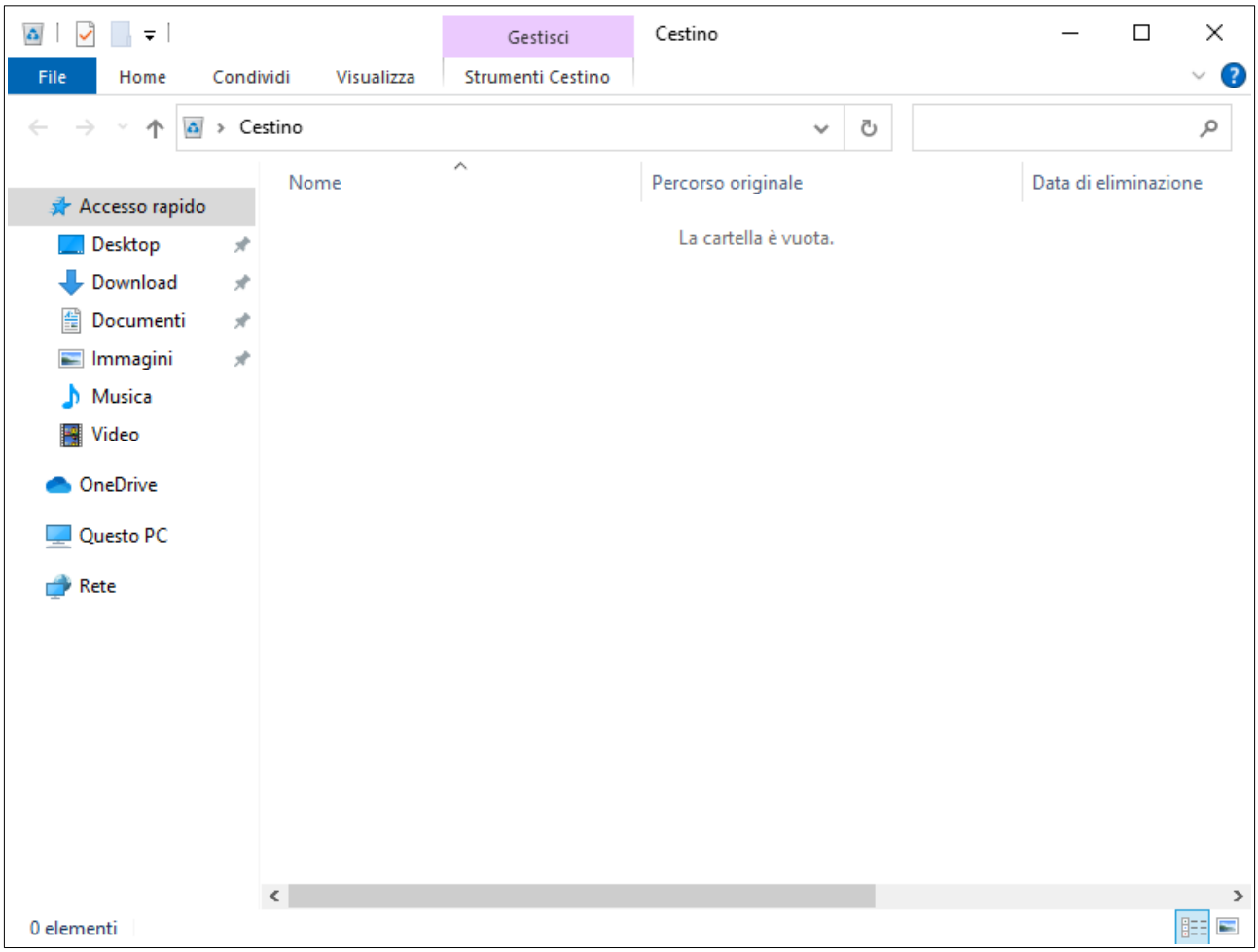
Quali informazioni puoi ottenere dalla scheda Dettagli e dalla finestra di dialogo Proprietà per il PID selezionato?
Nome dell'eseguibile, percorso file, dimensione, versione, firma digitale (se presente), utente che esegue il processo, data di creazione, uso CPU/memoria corrente.

Parte 5: Svuotare il cestino usando PowerShell.
Da PowerShell si possono eseguire anche comandi di gestione del computer. Come esempio è stato usato il comando "Clear-RecycleBin" per svuotare il cestino.





Cosa è successo ai file nel Cestino?
Il file di testo all'interno del cestino è stato eliminato definitivamente.



Conclusione
PowerShell rappresenta uno strumento molto potente per semplificare e automatizzare le attività quotidiane legate alla sicurezza e all'amministrazione di un sistema. Grazie ai cmdlet è possibile interrogare in tempo reale processi, servizi, log di sistema e connessioni di rete senza dover ricorrere a strumenti grafici o a utility esterne, con il vantaggio di poter integrare tutto in script riutilizzabili. Per un analista di sicurezza, comandi come Get-Process e Get-Service permettono di monitorare in modo immediato lo stato delle applicazioni, Get-EventLog consente di analizzare gli eventi registrati dal sistema, mentre Get-NetTCPConnection e Test-NetConnection forniscono informazioni utili sulle connessioni di rete e la raggiungibilità di host remoti. Anche Get-ChildItem -Recurse risulta utile per esplorare file e directory alla ricerca di contenuti sospetti. L'uso combinato di questi strumenti consente di aumentare il livello di controllo, velocizzare indagini e verifiche e ridurre i margini di errore umano.

Studio Ioc

Introduzione
Obiettivo dell'esercizio: analizzare il report AnyRun relativo un campione potenzialmente malevolo e produrre una sintesi tecnica che descriva il comportamento osservato, le tattiche/tecniche utilizzate e gli indicatori di compromissione (IoC).

Profilo del campione (Jvczfhe.exe)
Verdetto di AnyRun: Malicious Activity
Hash segnalato da AnyRun:
MD5: 00B5E91B42712471CDFBDB37B715670C
SHA1: D9550361E5205DB1D2DF9D02CC7E30503B8EC3A2
SHA256: 0307EE805DF8B94733598D5C3D62B28678EAEADBF1CA3689FA678A3780DD3DF0

Sintesi delle tattiche e tecniche osservate
Esecuzione
- Drop e uso di file .exe secondari (Muadnrd.exe), esecuzioni tramite processi legittimi (InstallUtil.exe) e iterazioni con WerFault.exe (Windows Error Reporting).
- Abuso di LOLBins: InstallUtil.exe e processi legittimi per eseguire o mascherare codice malevolo.

Evasione e anti-analisi
- Uso di timeout.exe per introdurre ritardi (tecnica anti-sandbox).
- Induzione di crash e gestione tramite WerFault, probabilmente per deviare log/artefatti o alterare il flusso di esecuzione.

Fingerprinting e raccolta dati
- Lettura di informazioni di sistema: nome macchina, Machine GUID, variabili d'ambiente, lingue supportate.
- Lettura chiavi relative a Microsoft Office e impostazioni Internet Explorer / WinINET.

Anti-forensics / Anti-logging
- Modifiche al registro per disattivare il RAS tracing relative ai nomi dei processi coinvolti:
  - HKLM\SOFTWARE\WOW6432Node\Microsoft\Tracing\Jvczfhe_*
  - HKLM\SOFTWARE\WOW6432Node\Microsoft\Tracing\Muadnrd_*
    Valori impostati su EnableFileTracing=0, EnableAutoFileTracing=0, EnableConsoleTracing=0, ecc per ridurre i log e maggiore difficoltà nelle attività di analisi forense.

Manipolazione delle impostazioni di rete (WinINET/IE)
- Scritture in HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap
  - ProxyBypass=1, IntranetName=1, UNCAsIntranet=1, AutoDetect=0.
    Effetto: bypass proxy e considerazione di risorse UNC come Intranet (zone più permissive), agevolando comunicazioni esterne e aggirando controlli di rete.

Artefatti di crash e identità
- WerFault genera dump (*.dmp) e report WER per Jvczfhe.exe e Muadnrd.exe.
- Sono presenti scritture sensibili in HKCU\SOFTWARE\Microsoft\IdentityCRL\Immersive\production\… (DeviceTicket, AppInit_DLLs) possibili artefatti collegati a identità o token utente.

Persistenza
Nel blocco analizzato non sono state rilevate scritture classiche per persistenza. Possibile spiegazione:
- persistenza implementata in fasi successive non osservate.
- tentativi falliti a causa di crash.
- modalità fileless o persistente tramite meccanismi non tipici. Va verificato con analisi Process tree / Autoruns / EDR.

Valutazione funzionale
Il comportamento è coerente con un trojan loader/dropper offuscato (.NET) che:

- esegue fingerprinting e raccolta ambiente.
- disabilità/trama log per nascondere attività.
- Modifica impostazioni di rete per facilitare comunicazioni.
- scarica/avvia payload secondari (Muadnrd.exe).
- sfrutta strumenti legittimi per esecuzione e mascheramento.

Conclusione: si classifica come un Trojan Loader/Dropper (con possibili funzionalità di infostealer) piuttosto che ransomware o worm.

Indicatori di Compromissione (IoC) consolidati
Hash (dal dump)

- Jvczfhe.exe (Downloads): SHA256 E6A7AAFF54EB6D06ACFC6F1DFA21A85B767DBF7FF3E9BFDF2DDBDECED86AA9B2
- Muadndr.exe (Downloads): SHA256 B662E7213F4985684439E655BD92EA4B9A1566E76712BB86E1238113A35B90A0

Percorsi e artefatti

- C:\Users\admin\Downloads\Jvczfhe.exe (+ :Zone.Identifier)
- C:\Users\admin\Downloads\Muadnrd.exe (+ :ZoneIdentifier)
- Crash dumps e report:
  - C:\Users\admin\AppData\Local\CrashDumps*.dmp
  - C:\ProgramData\Microsoft\Windows\WER\ReportArchive\...\Report.wer
  - C:\ProgramData\Microsoft\Windows\WER\Temp*.dmp, *.xml

Chiavi di registro modificate

- Anti-logging RAS:
  - HKLM\SOFTWARE\WOW6432Node\Microsoft\Tracing\Jvczfhe_RASAPI32| RASMANCS
  - HKLM\SOFTWARE\WOW6432Node\Microsoft\Tracing\Muadnrd_RASAPI32| RASMANCS
    (DeviceTicket / DeviceId / ApplicationFlags)

Conclusione
L’analisi mostra un campione offuscato (.NET Reactor) che si comporta come trojan loader/dropper, con chiare misure di evasione e anti-forensics, e con capacità di preparare la rete e il sistema per successive comunicazioni o download di payload. Sebbene non siano emerse evidenze di cifratura tipiche di ransomware, il comportamento è pericoloso: raccomandata immediata azione di containment, raccolta prove e bonifica seguendo le raccomandazioni fornite.