



Introduzione a Windows

Lezione 4 PowerShell

Ruggero Donida Labati

Laboratorio di Sistemi Operativi

Università degli Studi di Milano
Dipartimento di Informatica
A.A. 2024/2025

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

1

Panoramica della lezione

- Verrà presentata brevemente la PowerShell
- Verranno presentati concetti programmazione in PowerShell
- Verranno svolti esercizi pratici

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

2

Sommario

1. La PowerShell
2. Interfacce
3. Come scrivere in PowerShell
4. Script in PowerShell
5. Credits e letture consigliate
6. Esercizi

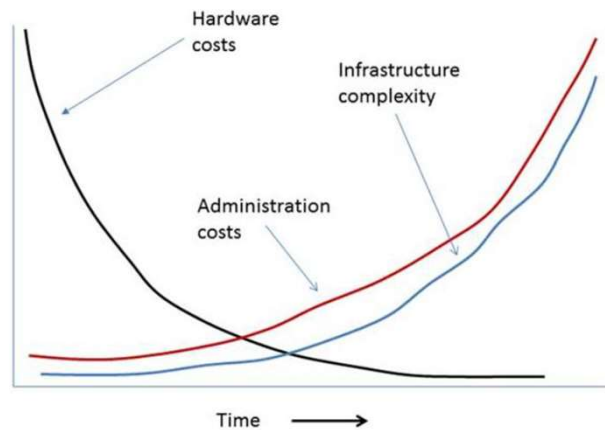


1. La PoweShell



Windows 10

Amministrazione dei sistemi operativi



R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

5

Cos'è la PowerDhell

- Terminale per gestire l'infrastruttura di Windows utilizzando script e comandi per automatizzare processi e workflow
 - Task di amministrazione
 - Configurazioni del sistema
 - Operazioni ripetute periodicamente
- Linguaggio di scripting object oriented in grado di gestire numerose componenti del sistema
 - Accesso ai componenti .NET, WMI e COM
- Gestione di sistema locale e remota
- Sviluppata con framework .NET

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

6

Sistemi compatibili

- Windows 8.1 x86 and x64
- Windows Management Framework (WMF)
 - Windows 7 SPI (or above) x86 and x64
 - Windows Embedded Standard 7
 - Windows Server 2008 R2 SPI (or above) x64
 - Windows Server 2012
 - Versioni di Windows successive
- Installabile su diversi SO (anche Linux)

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

7

Versioni

Versione	Incluso con Windows	Data di rilascio
1.0	XP / Server 2008	01/11/2006
2.0	7 / Server 2008 R2	01/11/2009
3.0	8 / Server 2012	01/08/2012
4.0	8.1 / Server 2012 R2	01/11/2013
5.0	10 / Server 2016 Tech Preview	16/12/2015
5.1	10 Anniversary edition / Server 2016	27/01/2017

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

8

Storia della gestione dei componenti Windows

	Before	Now
GUI	MMC	GUIs built on top of Powershell
Interactive Shell	CMD	PowerShell
Scripting	BAT in CMD	PowerShell
COM	WMI(VBScript and Jscript)	PowerShell

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

9

2. Interfacce



R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

10

Interfacce alla PowerShell

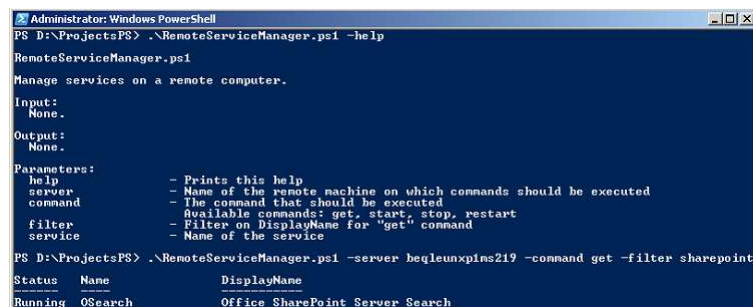
- PowerShell
 - Console
 - ISE: Integrated Scripting Environment
- Editor di terze parti per Powershell
 - PrimalScript
 - PowerGUI
 - PowerShell +
 - PowerSE

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

11

Interfaccia PowerShell

1. Fare clic su Start e successivamente cliccare sul pulsante Tutte le app in basso a destra
 2. Nel gruppo di Sistema Windows, fare clic su Windows PowerShell
- Oppure sulla riga del Prompt dei comandi, digitare PowerShell



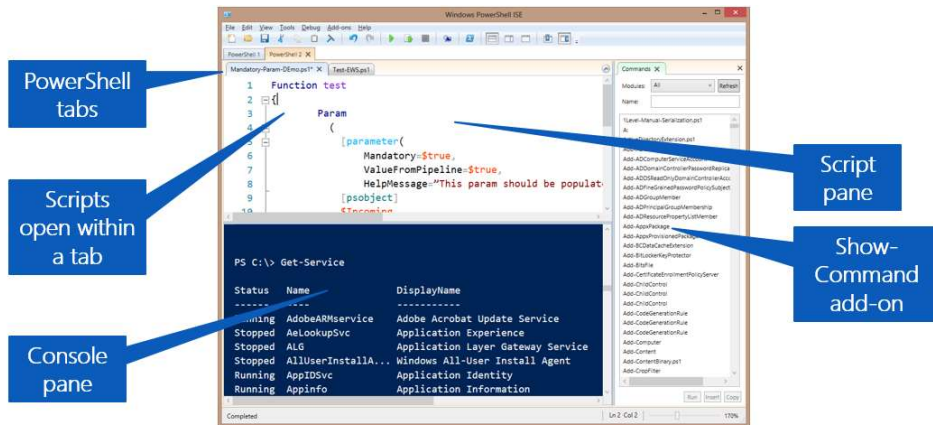
```
Administrator: Windows PowerShell
PS D:\ProjectsPS> .\RemoteServiceManager.ps1 -help
RemoteServiceManager.ps1
Manage services on a remote computer.
Input:
None.
Output:
None.
Parameters:
  help          - Prints this help
  server        - Name of the remote machine on which commands should be executed
  command       - The command that should be executed
                 Available commands: get, start, stop, restart
  filter        - Filter on DisplayName for "get" command
  service       - Name of the service

PS D:\ProjectsPS> .\RemoteServiceManager.ps1 -server beqleunixpms219 -command get -filter sharepoint
Status Name DisplayName
-----
Running OSearch Office SharePoint Server Search
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

12

Interfaccia ISE



R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

13

3. Come scrivere in PowerShell



R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

14

La shell

- La PowerShell non è case-sensitive, ovvero non distingue caratteri maiuscoli e minuscoli
- I tasti freccia facilitano la scrittura delle linee di comando
 - Con le frecce su e giù si possono richiamare comandi inseriti in precedenza
 - Con le frecce destra e sinistra ci si muove all'interno della riga per eventuali modifiche
 - Il tasto Tab consente di completare in modo automatico i comandi

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

15

Cmdlet

- Nella costruzione delle linee di comando in PowerShell si utilizza il concetto di cmdlet, esso indica un piccolo programma che svolge una specifica funzione e che può essere attivato da riga di comando
- Il cmdlet può essere classificato in
 - Get : per ottenere dati
 - Set : per impostare o modificare dati
 - Format : per formattare i dati
 - Out : per indirizzare l'output di un'elaborazione

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

16

La cronologia dei comandi

- La Power Shell conserva la cronologia dei comandi utilizzati
 - Questa funzione viene chiamata history e viene invocata con il comando

```
get-history
```

Il comando più utile

- La guida in linea per uno specifico cmdlet, viene visualizzato con il comando Get-Help

```
nomecmdlet -?
```

Hello everybody, I'm Merlin.
I'll guide you through the
application windows.



Commenti

- Commento di una singola riga con #

```
# This is a single-line comment  
Get-ChildItem
```

- Commento di più righe con <# ... #>

```
<# This is a  
multi-line  
comment #>  
Get-ChildItem
```

3. Script in PowerShell



La programmazione nella shell

- Per shell script si intendono righe di comandi della shell che diventano comandi eseguibili
- Gli script PowerShell sono caratterizzati dall'estensione **.ps1**
- Per scrivere una shell script si usa un editor qualsiasi di Windows

```
# Hello.ps1
# scrittura di messaggio
write-host "*****"
write-host " Hello, World " -foregroundcolor Red -backgroundcolor Black
write-host "*****"
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

21

Le variabili

- Le variabili servono per modificare stringhe di carattere o numeri
- I cmdlet che gestiscono le variabili sono
 - **Get-Variable** per visualizzare le variabili già definite;
 - **Set-Variable** per impostare una variabile assegnando nome e valore;
 - **New-Variable** per creare una nuova variabile senza assegnare alcun valore;
 - **Clear-Variable** per eliminare il valore di una variabile mantenendone il nome;
 - **Remove-Variable** per eliminare la variabile.

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

22

Gestione dei file e delle directory (1/3)

- **new-item**: crea un nuovo file o directory
 - Esempio: creazione di un file di testo *testo1*

```
new-item testo1 -itemtype file
```

- **remove-item**: cancella un file / directory , se la directory non è vuota richiede la conferma a meno che non ci sia il parametro
 - Opzione **-recurse**: che cancella la directory senza chiedere conferma

```
remove-item prove -recurse
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

23

Gestione dei file e delle directory (2/3)

- **set-location**: seleziona una nuova directory come directory corrente

```
set-location c:\users
```

- **get-childitem**: visualizza il contenuto di un'unità o di una directory

```
get-childitem c:\windows
```

- **get-content**: visualizza il contenuto di un file indicato col nome

```
get-content testo1
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

24

Gestione dei file e delle directory (3/3)

- **copy-item**: copia un file o una directory

```
copy-item -path c:\esercizi\prova1 -destination c:\consegnati\prova2
```

- **rename-item** (Alias REN): cambia il nome di un file o una directory specificata

```
rename-item prova1 esercizio1
```

- **move-item**: sposta un file o una directory all'interno dell'albero della directory

```
move-item c:\esercizi\prova1 c:\consegnati\prova2
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

25

L'output di dati e messaggi

- Per la visualizzazione del valore delle variabili viene utilizzato il comando write-host
 - Visualizza il valore della variabile a

```
write-host $a
```

- Il Write-Host consente anche la visualizzazione di stringhe di messaggi
 - Esempio

```
write-host Windows
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

26

Sequenze di escape

- Combinazioni del carattere ` (accento grave o apice rovesciato) con altri caratteri dette sequenze di escape assumono significati speciali con il comando write-host

` `	Virgolette semplici
` "`	Virgolette doppie
` 0	Null
` a	Avviso
` b	Backspace
` f	Avanzamento modulo
` n	Nuova riga
` r	Ritorno a capo
` t	Tabulazione orizzontale
` v	Tabulazione verticale

- Per inviare messaggi di avvertimento o di errore è opportuno utilizzare il cmdlet o il comando write-warning

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

27

Politiche di esecuzione

- Le politiche di esecuzione consentono di definire le modalità in PowerShell carica i file di configurazione ed esegue gli script
 - **Restricted**: nessuno script può essere eseguito e i comandi PowerShell possono essere usati solo da linea di comando
 - **AllSigned**: indica la politica più sicura, perché possono essere eseguiti solo gli script aventi firma digitale, con l'utilizzo di un certificato emesso da un autore attendibile
 - **RemoteSigned**: consente di eseguire gli script locali anche se privi di firma digitale
 - **Unresctricted**: non ci sono limitazioni e tutti gli script possono essere eseguiti

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

28

Come eseguire i nostri script

- Non vogliamo cambiare le impostazioni di sicurezza del nostro sistema operativo solo per fare esperienza con PowerShell
- Utilizzo di **Bypass** come politica di esecuzione
 - Da Powershell o sessione ISE

```
Set-ExecutionPolicy Bypass Process
```

- Da cmd

```
C:\>powershell -ExecutionPolicy Unrestricted
```

Eseguire uno script

- Richiamando lo script in PowerShell
- All'esterno di PowerShell
 - Scrivendo nella casella Esegui
 - Scrivendo il comando nel cmd

```
powershell.exe -noexit c:\Scripts\Hello
```

Operatori aritmetici

-op	inverte il segno dell'operando
op1 + op2	somma i due operandi
op1 - op2	sottrae dal primo il secondo operando
op1 * op2	moltiplica i due operandi
op1 / op2	divide il primo operando per il secondo
op1 % op2	restituisce il modulo, cioè il resto della divisione tra il primo e il secondo operando
op1 += op2	equivale a op1 = op1 + op2
op1 -= op2	equivale a op1 = op1 - op2
op1 /= op2	equivale a op1 = op1 / op2
op1 *= op2	equivale a op1 = op1 * op2
op1 %= op2	equivale a op1 = op1 % op2.

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

31

Operatori di confronto

-eq	uguale a
-ne	diverso da
-gt	maggiore di
-ge	maggiore o uguale a
-lt	minore di
-le	minore o uguale a
-like	confronto con caratteri jolly
-notlike	confronto negativo con caratteri jolly
-match	confronto tra espressioni regolari
-notmatch	confronto negativo tra espressioni regolari.

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

32

Operatori booleani

- L'operatore di negazione si indica con **-not**, produce falso se l'espressione è vera e viceversa
- L'operatore di congiunzione si indica con **-and**, produce vero se entrambe le espressioni sono vere
- L'operazione di disgiunzione si indica con **-or**, produce vero solo se un'espressione è vera

Operatori condizionali (1/4)

- Sintassi del costrutto **if**

```
if (condizione) {...}  
elseif (condizione) {...}  
else {...}
```

- Esempio di **if**

```
# Controllo.ps1  
# esistenza del file  
$a = test-path "C:\Scripts\Arch1"  
if ($a -eq $true)  
    {write-host "il file esiste"}
```

Operatori condizionali (2/4)

- Sintassi del costrutto **switch**

```
switch ('Condition')
{
    'Condition'
    {
        'First Action'
    }
    'Condition'
    {
        'Second Action'
        break
    }
    'Condition'
    {
        'Third Action'
    }
}
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

35

Operatori condizionali (3/4)

- Sintassi del costrutto **for**

```
for (valoreiniziale; condizione; operazioni)
{comandi}
```

- Esempio di **for**

```
# Numeri.ps1
# elenco dei numeri da 1 a 50
for($i=1; $i -le 50; $i++)
{write-host $i}
```

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

36

Operatori condizionali (4/4)

- Sintassi del costrutto **while**

```
while (condizione)
{comandi}
```

- Sintassi del costrutto **do while**

```
do
{comandi}
while (condizione)
```

```
do
{comandi}
until (condizione)
```

Operatori booleani

- L'operatore di negazione si indica con **-not**, produce falso se l'espressione è vera e viceversa
- L'operatore di congiunzione si indica con **-and**, produce vero se entrambe le espressioni sono vere
- L'operazione di disgiunzione si indica con **-or**, produce vero solo se un'espressione è vera

Gli alias (1/2)

- Gli alias consentono di assegnare un nome più semplice da ricordare o da scrivere
 - L'elenco completo si ottiene digitando il comando

```
get-alias
```

- Caso particolare

- Write-Host
Scrive solo su standard output

```
Write-Host "Hello World"
```

- Write-Output
Scrive su standard output e consente il reindirizzamento

```
Write-Output "Hello World"
```

Gli alias (2/2)

- Esempio

```
C:\Windows> dir  
C:\Windows> ls  
C:\Windows> Get-ChildItem
```

- Definire alias

```
Set-Alias -Name ping -Value Test-NetConnection
```

La pipeline

- Consente di manipolare l'output da un cmdlet per eseguire un'altra azione
- Esempio
 - Utilizzare Select-Object per mostrare solo la proprietà Name di un file mostrato da Get-ChildItem

```
Get-ChildItem | Select-Object Name
```

5. Credits e letture consigliate



Credits

- Approfondimento PowerShell
<https://riptutorial.com/Download/powershell-it.pdf>
- Francesco Parisi, PowerShell di Windows
<https://slideplayer.it/slide/17163449/>
- Awase Khirni Syed, PowerShell
<https://www.slideshare.net/awasekhirni/powershell-training-material>

Lecture consigliate

- Microsoft, "Documentazione di PowerShell,"
<https://docs.microsoft.com/it-it/powershell/>
- E. Wilson, "Windows PowerShell. La guida completa al linguaggio di script per l'amministrazione dei sistemi Windows. Con CD-ROM," Mondadori Informatica.
- Approfondimento PowerShell
<https://riptutorial.com/Download/powershell-it.pdf>
- Awase Khirni Syed, PowerShell
<https://www.slideshare.net/awasekhirni/powershell-training-material>

6. Esercizi



R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

45

6. Esercizi

- Scrivere uno script che somma i primi 10 numeri naturali
- Eseguire lo script scritto in precedenza in PowerShell e in cmd
- Scrivere uno script che accetta come argomento una delle seguenti lettere: D, U, L, C e restituisce rispettivamente: la data di oggi, il nome dell'utente collegato, la lista dei file della directory corrente e il nome della directory corrente

R. DONIDA LABATI – INTRODUZIONE A WINDOWS – POWERSHELL

46

In sintesi

1. La PowerShell
2. Interfacce
3. Come scrivere in PowerShell
4. Script in PowerShell
5. Credits e letture consigliate
6. Esercizi

