# PowerShell - Guide des Commandes et Concepts Utiles

PowerShell est un interpréteur de commandes puissant combinant une interface en ligne de commande (CLI) et un langage de script. Il permet l’automatisation et l’administration du système sous Windows, avec compatibilité étendue avec d'autres outils système.

## Définitions et Syntaxes de base

Une cmdlet PowerShell suit le format Verbe-Nom (ex: Get-Service). Les commentaires s’écrivent avec # ou dans des blocs <# ... #>.

# Ceci est un commentaire

<#
Ceci est un commentaire
sur plusieurs lignes
#>

## Premiers pas et aide intégrée

$PSVersionTable

Get-Alias

Get-Command

Get-Command \*user\*

Get-Command get\*

Update-Help

Get-Help

Get-Help nomdelacommande -examples

## Utilisation du Pipe

Get-ChildItem | Format-List

Get-ChildItem | Where-Object { $\_.Extension -eq ".txt" }

Get-Process | Where-Object { $\_.Name -eq "notepad" }

Get-Service | Where-Object { $\_.Name -eq "wuauserv" } | Stop-Service

## Commandes Utiles

Set-ComputerName -ComputerName "newservername"

New-LocalUser -Name "UserPrincipalName" -Description "description" -NoPassword

Set-ExecutionPolicy RemoteSigned

## Windows Subsystem for Linux (WSL)

wsl --install

wsl --list --online

wsl --install -d Ubuntu

## Connexion à distance

Enter-PSSession -ComputerName 192.168.0.1 -Credential administrateur@maboite.home

$MASESSION = New-PSSession -ComputerName monpc.home

Get-PSSession

Remove-PSSession -Session $MASESSION

## WinRM & Confiance hôtes

Get-Service WinRM

Enable-PSRemoting

winrm get winrm/config

Set-Item WSMan:\localhost\Client\TrustedHosts -Value "192.168.29.10"

## Gestion des disques simples

Get-Disk

Initialize-Disk -Number 2

Clear-Disk -Number 2

New-Partition –DiskNumber 2 -AssignDriveLetter –UseMaximumSize

Format-Volume -DriveLetter E -FileSystem NTFS -Confirm:$false

Get-Volumes

Get-Disk | Where-Object IsSystem -eq $False

Get-Disk | Where-Object IsOffline –Eq $True | Initialize-Disk

## Créer un volume miroir RAID1

Get-Disk

Get-PhysicalDisk

New-StoragePool -FriendlyName "RAID1\_pool" -StorageSubsystemFriendlyName "Windows Storage\*" -PhysicalDisks (Get-PhysicalDisk -CanPool $True | Where-Object {$\_.UniqueId -eq "Disk1UniqueID" -or $\_.UniqueId -eq "Disk2UniqueID"})

New-Volume -StoragePoolFriendlyName "RAID1" -FriendlyName "RAID1\_volume" -FileSystem CSVFS\_NTFS -Size 512GB -ResiliencySettingName Mirror

Get-Volume

## Gestion des fichiers et répertoires

Get-ChildItem

Set-Location

New-Item -Path D: -Name NOM -ItemType Directory

New-Item -Path PATH -Name NOM -ItemType File

Set-Content -Path "fichier.txt" -Value "Bonjour"

Add-Content -Path "fichier.txt" -Value "Bonjour"

Get-Content "C:\...log.txt"

Copy-Item "source" -Destination "dest"

Copy-Item -Path "C:\Logfiles\\*" -Destination "C:\Presentation" -Recurse

$MASession = New-PSSession -ComputerName "Server01" -Credential "upn@domaine.fr"

Copy-Item "D:\test.log" -Destination "C:\Folder\_Copy\" -ToSession $MASession

Remove-PSSession -Session $MASession

## Configuration Réseau

Get-NetAdapter

New-NetIPAddress -IPAddress 192.168.0.1 -PrefixLength 23 -InterfaceIndex 8

Set-NetIPAddress -IPAddress 192.168.0.1 -PrefixLength 23 -InterfaceIndex 8

Set-NetIPAddress -InterfaceIndex 8 -DHCP Enabled

Set-DnsClientServerAddress -InterfaceIndex 12 -ServerAddresses ("10.0.0.1","10.0.0.2")

netsh advfirewall set allprofiles state off

netsh advfirewall set allprofiles state on

Get-NetConnectionProfile

Set-NetConnectionProfile -InterfaceIndex 8 -NetworkCategory Private

Get-NetTCPConnection

Get-NetTCPConnection -State Listen

Get-NetTCPConnection -localport 53

Get-NetTCPConnection -State Listen | Select-Object LocalAddress,LocalPort,@{'Name' = 'ProcessName';'Expression'={(Get-Process -Id $\_.OwningProcess).Name}}

Get-NetTCPConnection -AppliedSetting Internet