

- TP 4 -
Les scripts sous Windows :
PowerShell
par Édouard Lumet

Sommaire

- 1. Découverte de PowerShell.....3
- 2. Scripts PowerShell élémentaires.....4
- 3. Scripts PowerShell plus complexes.....5

1. Découverte de PowerShell

- Get-Help Get-Process nous fournit une aide à propos de la commande Get-Process qui permet elle de récupérer la liste des processus. L'ajout de more via un pipe (ou tube : |), permet d'afficher le résultat comme sous Linux avec less ou more.
- Analyse de commandes

\$UneVar="Cours PowerShell"	Affectation d'une chaîne de caractères à une variable \$UneVar
\$UneVar	Affichage du contenu de la variable \$UneVar
\$UneVar.GetType()	Affichage du type de la variable \$UneVar
\$UneVar.GetType().FullName	Affichage du type complet de la variable \$UneVar
\$lesProcessus=Get-Process	Affectation de la liste des processus courants à la variable \$lesProcessus
\$lesProcessus[0]	Affichage de la première ligne de la variable \$lesProcessus
\$lesProcessus[0].ProcessName	Affichage du nom seulement du processus de la première ligne de \$lesProcessus

- Analyse de commandes

dir C:\temp*	Affichage du contenu du répertoire 'temp' sur le lecteur C:
Set-Location HKLM:	Changement du répertoire courant pour HKLM:
dir HARDWARE	Permet de lister (dans HKLM:) les périphériques matériels
Get-Help about_Core_Commands	Affiche l'aide du cmdlet about_Core_Commands

- Analyse de commandes utilisant les pipe

Get-Process	Affiche les processus
Get-Process where {\$_.handlecount -gt 400}	Affiche les processus dont le numéro est supérieur à 400
Get-Process where {\$_.handlecount -gt 400} Format-List	Même chose que précédemment, sous forme de liste

- Analyse de commandes utilisant les objets WMI

Get-WmiObject win32_logicaldisk	Affiche les périphériques de stockage de l'ordinateur
Get-WmiObject win32_logicaldisk Format-Table deviceid, size, freespace	Même chose que précédemment, sous forme d'un tableau avec l'ID, la taille et l'espace libre du disque

- Analyse de commandes avec redirection du résultat

Get-Process where {\$_.handlecount -gt 1500}	Affiche les processus dont le numéro est supérieur à 1500
Get-Process where {\$_.handlecount -gt 1500} _Export-Clixml test.xml	Même chose que précédemment, le résultat étant redirigé dans un fichier au format XML
\$fich=Get-Content test.xml	Affectation du contenu du fichier test.xml à la variable \$fich
\$fich	Affichage du contenu de la variable \$fich (= contenu de test.xml)

2. Scripts PowerShell élémentaires

- Script affichant la liste des services du PC, en vert si le service est actif ou en rouge s'il est arrêté :

```
$service=Get-Service|Select-Object name, status
foreach ($s in $service)
{
    if ($s.status -eq "running")
    {
        Write-Host $s.name -ForegroundColor Green
    }
    else
    {
        Write-Host $s.name -ForegroundColor Red
    }
}
```

- Script qui ferme les fenêtres du bloc-notes ouvertes :

```
$ps=Get-Process|Select-Object ProcessName, Id
foreach ($i in $ps)
{
    if ($i.ProcessName -eq "notepad")
    {
        Stop-Process $i.Id
    }
}
```

3. Scripts PowerShell plus complexes