

Fiche 08 : Amélioration de la gestion d'Active Directory avec de nouvelles commandes

1. Constat et autre méthode de gestion d'Active Directory

Dans PowerShell, le mode de fonctionnement d'ADSI n'est pas toujours des plus pratiques. Pour combler cette lacune, une société (Quest Software) propose gratuitement de nouveaux applets de commandes conçus spécialement pour faciliter la gestion d'un annuaire Active Directory. C'est la même société qui propose l'éditeur PowerGUI (powergui.com). Ces applets sont téléchargeables depuis quest.com/powershell (PowerShell Commands for Active Directory).

Après avoir installé ces nouveaux applets de commande avec l'exécutable fourni par la société, il est nécessaire, pour rendre ces commandes opérationnelles dans la console PowerShell, de lancer l'instruction suivante :

```
Add-PSSnapin Quest.ActiveRoles.ADManagement
```

Pour obtenir le même résultat avec l'éditeur PowerGUI :
Menu Fichier, Bibliothèques PowerShell..., sélectionner la case à cocher Quest.ActiveRolesADManagement, Bt OK.

Maintenant, la touche tabulation a le même effet sur ces nouvelles commandes, qui possèdent toutes la lettre 'Q' comme (Quest) dans leur nom : Get-QADuser, Set-QADuser, New-QADuser, Enable-QADuser, etc...

2. Création de comptes utilisateurs à partir d'un fichier texte au f

Exemple, mise en place du script de la fiche N°07 (pb pour l'instant avec le mot de passe et password

() même instruction sur plusieurs lignes

```
1 Import-Csv "c:\temp\users.txt" | ForEach-Object {
2     New-QADUser -organizationalUnit 'labo.ig/test' -Name ($_.prenom+" "+"_.nom) `
3     -SamAccountName $_.nom -FirstName $_.prenom -LastName $_.nom -Description $_.description `
4     -DisplayName ($_.nom+" "+"_.prenom) -UserPassword 'toto' `
5     -UserPrincipalName $_.nom -PhoneNumber $_.tel } |
6     enable-QADUser | Set-QADUser -UserMustChangePassword $True
```

Idem script fiche 07

New-QADUser possède un paramètre pour chaque attribut

Remarques : - Le résultat de la commande New-QADUser est transmis par pipeline à la commande enable-QADUser qui active donc le compte de l'utilisateur correspondant (l'objet reçu) et transmet au pipeline suivant un objet (utilisateur) pour modifier une propriété (changer le mot de passe à la première connexion).
- Cette commande est sur une seule ligne, pas toujours pratique, mais très puissante.

3. Idem mais avec un test d'existence du compte utilisateur.

```
1 Import-Csv "c:\temp\users.txt" | ForEach-Object {
2     if (!(Get-QADUser -Name ($_.prenom+" "+"_.nom))){
3         New-QADUser -organizationalUnit 'labo.ig/test' -Name ($_.prenom+" "+"_.nom) `
4         -SamAccountName $_.nom -FirstName $_.prenom -LastName $_.nom -Description $_.description `
5         -DisplayName ($_.nom+" "+"_.prenom) -UserPassword 'toto' `
6         -UserPrincipalName $_.nom -PhoneNumber $_.tel |
7         enable-QADUser | Set-QADUser -UserMustChangePassword $True
8     }
9     else{
10        Write-Host ("le compte "+$_.prenom+" "+"_.nom+" existe déjà")
11    }
12 }
```

Remarque : La nouvelle commande Get-QADUser permet de réaliser plus simplement le test que dans la version de la fiche N°7.

4. Encore plus spectaculaire !!

Un autre organisme, PowerShell Community Extension propose un complément (en cours d'évolution) qui permet de diriger simplement l'interpréteur de commandes vers le nom de domaine pour gérer l'annuaire comme un simple lecteur de disque. Ce complément 'Open Source' est disponible sur codeplex.com/powershellcx.

Si vous n'avez pas coché la mise à jour du profil lors de l'installation, il est nécessaire de lancer l'instruction suivante :

```
Add-PSSnapin PSCX
```

Exemple, résultat des commandes CD et DIR sur le domaine labo et l'unité d'organisation test :

```
cd labo:
labo.ig\
cd test
labo.ig\test
dir
```

	LastWriteTime	Type	Name
	-----	----	----
	05/11/2008	18:40 user	Daniel Regnier
	05/11/2008	18:40 user	Michael Tutin
	05/11/2008	18:40 user	Pierre Loisel