

Créer des utilisateurs dans l'Active Directory via powershell à partir d'un fichier CSV

Comme nous l'avons vu précédemment, nous pouvons créer un utilisateur via powershell avec la commande ci-dessous.

Cette commande est valable pour créer un utilisateur unique, nous allons voir comment automatiser ce processus afin de créer des utilisateurs en masse.

```
40 New-ADUser -Name "Testy Test" `
41     -DisplayName "Test Testy" `
42     -GivenName "Testy" `
43     -Surname "Test" `
44     -SamAccountName "testy" `
45     -UserPrincipalName "testy@gsb.lan" `
46     -EmailAddress "testy@gsb.fr" `
47     -Path "OU=Test,OU=GSB,DC=gsb,DC=lan" `
48     -AccountPassword(ConvertTo-SecureString "Passw0rd" -AsPlainText -Force) `
49     -Enabled $true
50
```

Tout d'abord il faut un fichier CSV avec des utilisateurs.

Ce fichier se compose de 3 colonnes:

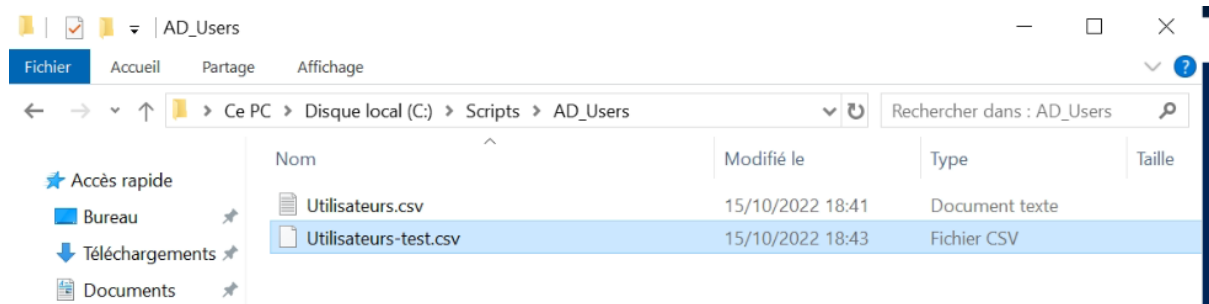
- Nom
- Prénom
- Service
-

A	B	C
Prenom	Nom	Service
Jacques	Pert	Service1
Jordan	Lecky	Service2
Lucky	Luke	Service3
Mathieu	Dupuy	Service4

Nous pouvons maintenant voir ce fichier via powershell avec la commande suivante.

```
$CSVData = Import-CSV -Path $CSVFile -Delimiter ";" -Encoding UTF8
```

J'ai placé ce fichier dans C:\Scripts\AD_Users



Nous allons maintenant créer une boucle et des fonctions.

Par exemple \$UtilisateurPrenom est composé de \$Utilisateur et de la colonne prénom dans le fichier CSV.

```
$CSVFile = "C:\Scripts\AD_Users\Utilisateurs-test.csv"
$CSVData = Import-CSV -Path $CSVFile -Delimiter ";" -Encoding UTF8

Foreach($Utilisateur in $CSVData){

    $UtilisateurPrenom = $Utilisateur.Prenom
    $UtilisateurNom = $Utilisateur.Nom
    $UtilisateurLogin = ($UtilisateurPrenom).Substring(0,1).ToLower() + "." + $UtilisateurNom.ToLower()
    $UtilisateurEmail = "$UtilisateurLogin@gsb.com".ToLower()
    $UtilisateurService = $Utilisateur.Service
```

La boucle quant à elle sert à vérifier que l'utilisateur n'est pas déjà présent dans l'AD. Si l'utilisateur est déjà présent, le script renvoie la valeur: L'identifiant \$UtilisateurLogin existe déjà dans l'AD.

A noter qu'avec les paramètres de la commande, \$UtilisateurLogin correspond à la première lettre du prénom ainsi que du nom tout attaché et en minuscule grâce à l'argument .ToLower().

```
# Vérifier la présence de l'utilisateur dans l'AD
if (Get-ADUser -Filter {SamAccountName -eq $UtilisateurLogin})
{
    Write-Warning "L'identifiant $UtilisateurLogin existe déjà dans l'AD"
}
```

Si le script ne trouve pas d'utilisateur dans l'AD avec ces informations, il lance la commande vue dans le tutoriel précédent.

Afin d'automatiser le processus, les informations sont remplacées par les fonctions que nous avons mises en place précédemment.

Dans cette commande l'OU où doit être placé l'utilisateur est \$UtilisateurService et cette fonction renseigne ce qui se trouve dans utilisateur.

```
else
{
    New-ADUser -Name "$UtilisateurPrenom $UtilisateurNom" `
        -DisplayName "$UtilisateurNom $UtilisateurPrenom" `
        -GivenName $UtilisateurPrenom `
        -Surname $UtilisateurNom `
        -SamAccountName $UtilisateurLogin `
        -UserPrincipalName "$UtilisateurLogin@gsb.lan" `
        -EmailAddress $UtilisateurEmail `
        -Path "OU=$UtilisateurService,OU=GSB,DC=gsb,DC=lan" `
        -AccountPassword (ConvertTo-SecureString "Password" -AsPlainText -Force) `
        -Enabled $true

    Write-Output "Création de l'utilisateur : $UtilisateurLogin ($UtilisateurNom $UtilisateurPrenom)"
}
```

Si un utilisateur à été ajouté suite à cette commande, le terminal nous renverra :
Création de l'utilisateur : \$UtilisateurLogin (\$UtilisateurNom \$UtilisateurPrenom)
Voici la commande complète.

```
$CSVFile = "C:\Scripts\AD_Users\Utilisateurs.csv"
$CSVData = Import-CSV -Path $CSVFile -Delimiter ";" -Encoding UTF8

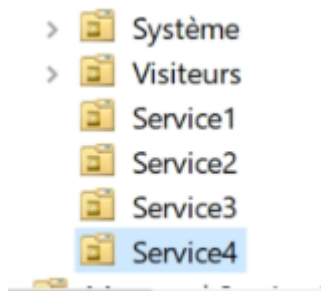
Foreach($Utilisateur in $CSVData){

    $UtilisateurPrenom = $Utilisateur.Prenom
    $UtilisateurNom = $Utilisateur.Nom
    $UtilisateurLogin = ($UtilisateurPrenom).Substring(0,1).ToLower() + "." +
$UtilisateurNom.ToLower()
    $UtilisateurEmail = "$UtilisateurLogin@gsb.com".ToLower()
    $UtilisateurService = $Utilisateur.Service

    # Vérifier la présence de l'utilisateur dans l'AD
    if (Get-ADUser -Filter {SamAccountName -eq $UtilisateurLogin})
    {
        Write-Warning "L'identifiant $UtilisateurLogin existe déjà dans l'AD"
    }
    else
    {
        New-ADUser -Name "$UtilisateurPrenom $UtilisateurNom" `
            -DisplayName "$UtilisateurNom $UtilisateurPrenom" `
            -GivenName $UtilisateurPrenom `
            -Surname $UtilisateurNom `
            -SamAccountName $UtilisateurLogin `
            -UserPrincipalName "$UtilisateurLogin@gsb.lan" `
            -EmailAddress $UtilisateurEmail `
            -Path "OU=$UtilisateurService,OU=GSB,DC=gsb,DC=lan" `
            -AccountPassword(ConvertTo-SecureString "Passw0rd" -AsPlainText
-Force) `
            -Enabled $true

        Write-Output "Création de l'utilisateur : $UtilisateurLogin ($UtilisateurNom
$UtilisateurPrenom)"
    }
}
```

Pour que les utilisateurs soient créés, il ne faut pas oublier de créer les OU qui correspondent à la colonne service.



Nous pouvons maintenant lancer le script.

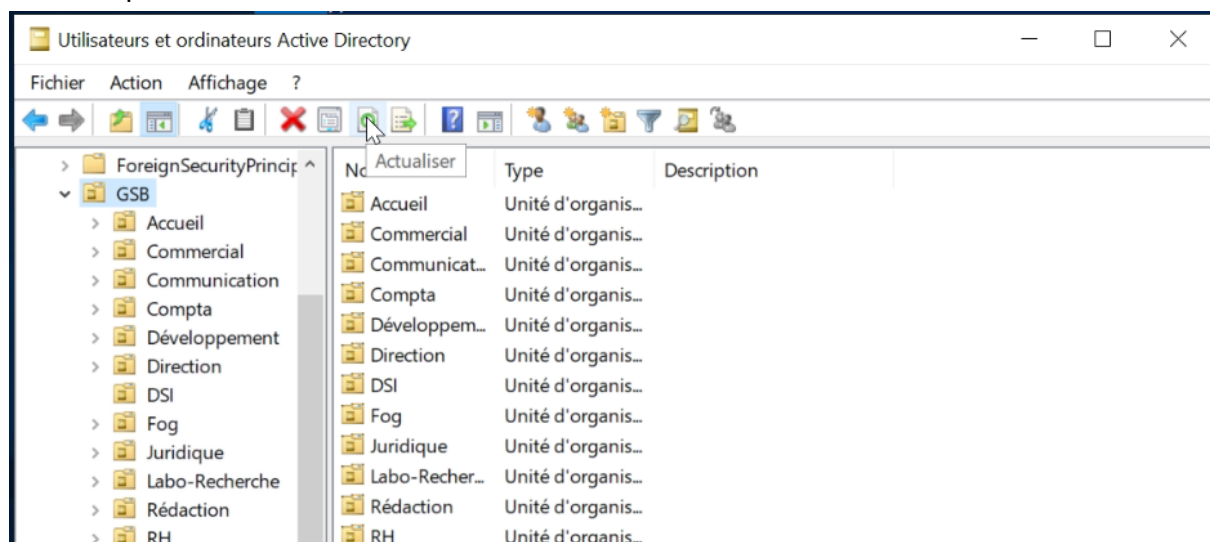
```
Administrateur : Windows PowerShell
Windows PowerShell
Copyright (c) Microsoft Corporation. Tous droits réservés.

PS C:\Users\Administrateur.GSB> $CSVFile = "C:\Scripts\AD_Users\Utilisateurs-test.csv"
>> $CSVData = Import-CSV -Path $CSVFile -Delimiter ";" -Encoding UTF8
>>
>> Foreach($Utilisateur in $CSVData){
>>     $UtilisateurPrenom = $Utilisateur.Prenom
>>     $UtilisateurNom = $Utilisateur.Nom
>>     $UtilisateurLogin = ($UtilisateurPrenom).Substring(0,1).ToLower() + "." + $UtilisateurNom.ToLower()
>>     $UtilisateurEmail = "$UtilisateurLogin@gsb.com".ToLower()
>>     $UtilisateurService = $Utilisateur.Service
>>
>>     # Vérifier la présence de l'utilisateur dans l'AD
>>     if (Get-ADUser -Filter {SamAccountName -eq $UtilisateurLogin})
>>     {
>>         Write-Warning "L'identifiant $UtilisateurLogin existe déjà dans l'AD"
>>     }
>>     else
>>     {
>>         New-ADUser -Name "$UtilisateurPrenom $UtilisateurNom" `
>>             -DisplayName "$UtilisateurNom $UtilisateurPrenom" `
>>             -GivenName $UtilisateurPrenom `
>>             -Surname $UtilisateurNom `
>>             -SamAccountName $UtilisateurLogin `
>>             -UserPrincipalName "$UtilisateurLogin@gsb.lan" `
>>             -EmailAddress $UtilisateurEmail `
>>             -Path "OU-$UtilisateurService,OU=GSB,DC=gsb,DC=lan" `
>>             -AccountPassword(ConvertTo-SecureString "Password" -AsPlainText -Force) `
>>             -Enabled $true
>>
>>         Write-Output "Création de l'utilisateur : $UtilisateurLogin ($UtilisateurNom $UtilisateurPrenom)"
>>     }
>> }
```

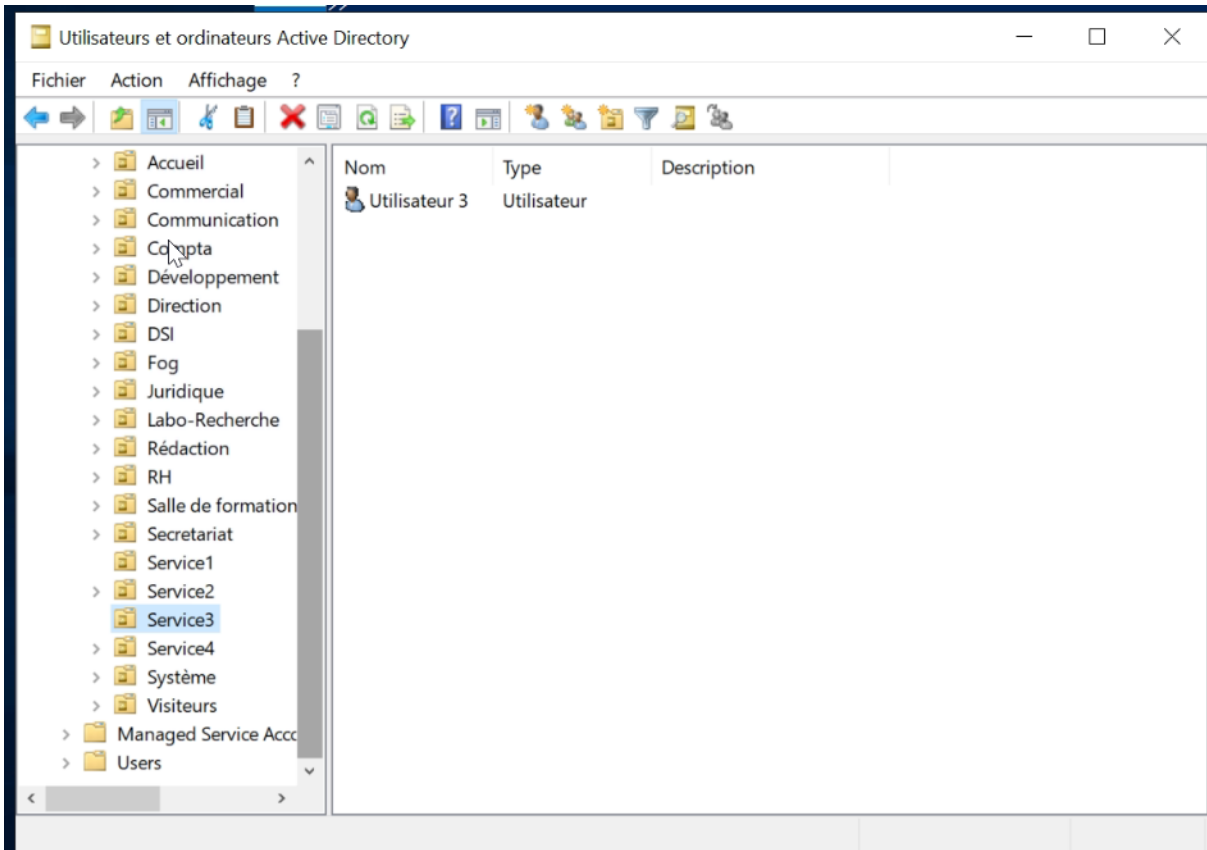
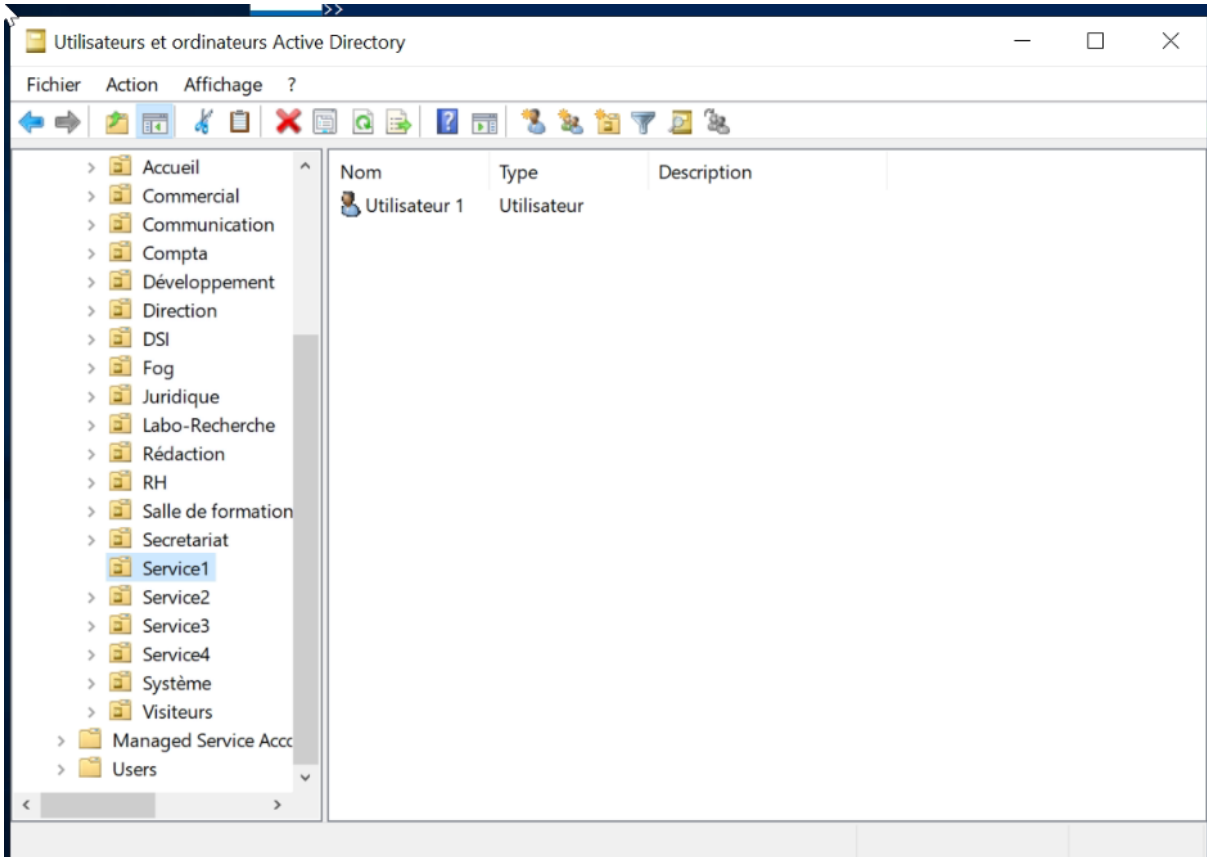
Nous pouvons constater que les utilisateurs ont bien été créés.

```
Administrateur : Windows PowerShell
PS C:\Users\Administrateur.GSB> $CSVFile = "C:\Scripts\AD_Users\Utilisateurs-test.csv"
>> $CSVData = Import-CSV -Path $CSVFile -Delimiter ";" -Encoding UTF8
>> Foreach($Utilisateur in $CSVData){
>>     $UtilisateurPrenom = $Utilisateur.Prenom
>>     $UtilisateurNom = $Utilisateur.Nom
>>     $UtilisateurLogin = ($UtilisateurPrenom).Substring(0,1).ToLower() + "." + $UtilisateurNom.ToLower()
>>     $UtilisateurEmail = "$UtilisateurLogin@gsb.com".ToLower()
>>     $UtilisateurService = $Utilisateur.Service
>>
>>     # Vérifier la présence de l'utilisateur dans l'AD
>>     if (Get-ADUser -Filter {SamAccountName -eq $UtilisateurLogin})
>>     {
>>         Write-Warning "L'identifiant $UtilisateurLogin existe déjà dans l'AD"
>>     }
>>     else
>>     {
>>         New-ADUser -Name "$UtilisateurPrenom $UtilisateurNom" `
>>             -DisplayName "$UtilisateurNom $UtilisateurPrenom" `
>>             -GivenName $UtilisateurPrenom `
>>             -Surname $UtilisateurNom `
>>             -SamAccountName $UtilisateurLogin `
>>             -UserPrincipalName "$UtilisateurLogin@gsb.lan" `
>>             -EmailAddress $UtilisateurEmail `
>>             -Path "OU=$UtilisateurService,OU=GSB,DC=gsb,DC=lan" `
>>             -AccountPassword(ConvertTo-SecureString "Passw0rd" -AsPlainText -Force) `
>>             -Enabled $true
>>
>>         Write-Output "Création de l'utilisateur : $UtilisateurLogin ($UtilisateurNom $UtilisateurPrenom)"
>>     }
>> }
Création de l'utilisateur : u.1 (1 Utilisateur)
Création de l'utilisateur : u.2 (2 Utilisateur)
Création de l'utilisateur : u.3 (3 Utilisateur)
Création de l'utilisateur : u.4 (4 Utilisateur)
PS C:\Users\Administrateur.GSB>
```

Il ne faut pas oublier d'actualiser la liste des utilisateurs.




Nous pouvons constater que les utilisateurs sont bien présents dans l'AD.



Les utilisateurs ont bien les caractéristiques voulues.

Propriétés de : Utilisateur 2 ? X

Environnement Sessions Contrôle à distance Profil des services Bureau à distance COM+
Général Adresse Compte Profil Téléphones Organisation Membre de Appel entrant

 Utilisateur 2

Prénom : Initiales :

Nom :

Nom complet :

Description :

Bureau :

Numéro de téléphone :

Adresse de messagerie :

Page Web :

Propriétés de : Utilisateur 2 ? X

Environnement Sessions Contrôle à distance Profil des services Bureau à distance COM+
Général Adresse **Compte** Profil Téléphones Organisation Membre de Appel entrant

Nom d'ouverture de session de l'utilisateur :
 @gsb.lan

Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000) :

☐ Déverrouiller le compte

Options de compte :

- ☐ L'utilisateur devra changer le mot de passe
- ☐ L'utilisateur ne peut pas changer de mot de passe
- ☐ Le mot de passe n'expire jamais
- ☐ Enregistrer le mot de passe en utilisant un chiffrement réversible

Date d'expiration du compte

☒ Jamais

☐ Fin de :