

Nom de l'établissement :

CFA ROBERT SCHUMAN

Réalisé par:

AOURDOU BEN KHAYI BRAHIM

Date :

02/04/2025

Encadrant :

M.ROTH

Intro :

Le script PowerShell permet d'automatiser l'importation d'utilisateurs à partir d'un fichier CSV. Cela simplifie et accélère la gestion des utilisateurs en lisant les données du fichier et en créant automatiquement les comptes dans un système, comme Active Directory. Cette méthode permet de gagner du temps et d'éviter les erreurs lors de l'ajout de nombreux utilisateurs.

Création du fichier CSV

--> Création du fichier CSV et y rentrer les données concernant les utilisateurs :

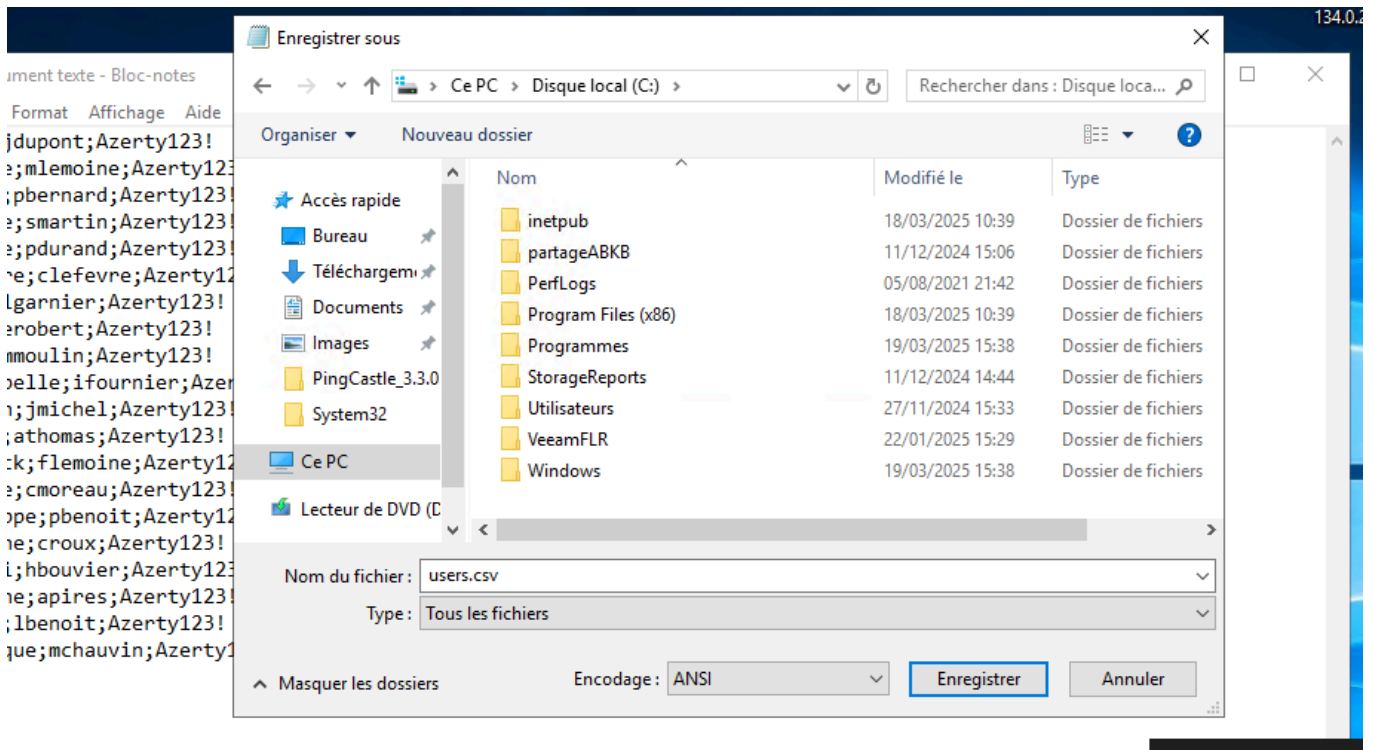
--> Ouvrir un nouveau doc texte et y rentrer les utilisateurs

```

Fichier Edition Format Affichage Aide
Firstname;Lastname;Username;Password
Dupont;Jean;jdupont;Azerty123!
Lemoine;Marie;mlemoine;Azerty123!
Bernard;Paul;pbernard;Azerty123!
Martin;Sophie;smartin;Azerty123!
Durand;Pierre;pdurand;Azerty123!
Lefevre;Claire;clefevre;Azerty123!
Garnier;Luc;lgarnier;Azerty123!
Robert;Emma;erobert;Azerty123!
Moulin;Marc;mmoulin;Azerty123!
Fournier;Isabelle;ifournier;Azerty123!
Michel;Julien;jmichel;Azerty123!
Thomas;Alice;athomas;Azerty123!
Lemoine;Franck;flemoine;Azerty123!
Moreau;Carole;cmoreau;Azerty123!
Benoit;Philippe;pbenoit;Azerty123!
Roux;Catherine;croux;Azerty123!
Bouvier;Henri;hbouvier;Azerty123!
Pires;Amandine;apires;Azerty123!
Benoit;Louis;lbenoit;Azerty123!
Chauvin;Monique;mchauvin;Azerty123!

```

--> Lors de l'enregistrement, attention à bien enregistrer le fichier en .csv



--> Vérifier l'extension du fichier dans l'emplacement celui-ci a été enregistré.

Création du script PowerShell

--> Ouvrir PowerShell ISE

--> Création de deux variables:

--> Une permettant de récupérer le fichier csv

--> Une montrant ou les utilisateurs devront être enregistrés dans l'AD (notation dn)

```
1 # Variables
2 $csvPath = « C:\\users.csv »
3 $ouPath = « OU=utilisateurs,Ou=ABKB,DC=brahimABKB,DC=local »
4
```

--> Importer le module Active Directory

```
# Importer le module Active Directory
Import-Module ActiveDirectory
```

--> commande permettant au script PowerShell de lire le fichier CSV

```
Import-Module ActiveDirectory

# Lire le fichier CSV et ajouter les utilisateurs
$users = Import-Csv -Path $csvPath
```

--> Création d'une boucle qui va pour chaque utilisateur :

--> Convertir le MDP en une chaîne sécurisée

--> Propriétés de l'utilisateur qui vont être définies

```
foreach ($user in $users) {
    $password = ConvertTo-SecureString $user.password -Force
    $userProps = @{
        'Name' = $user.FirstName + ~ ~ + $user.LastName
        'GivenName' = $user.FirstName
        'Surname' = $user.LastName
        'SamAccountName' = $user.UserName
        'UserPrincipalName' = $user.UserName + « brahimABKB.local »
        'DisplayName' = $user.FirstName + ~ ~ + $user.LastName
        'Path' = $ouPath
        'AccountPassword' = $password
        'Enabled' = $false
        'ChangePasswordAtLogon' = $true
    }
}
```

--> ajouter la commande new-ADUser qui va créer l'utilisateur dans l'AD en utilisant les propriétés créées précédemment.

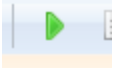
```
New-ADUser @userProps
```

--> Ajouter la commande Write-Host permettant d'afficher un message quand le script a été exécuté.

```
Write-Host « Importation des utilisateurs terminée. »
```

Exécuter le script

--> Appuyer sur la touche verte en forme de triangle



--> Script final

```
# Variables
$csvPath = 'C:\\users.csv'
$ouPath = 'OU=utilisateurs,Ou=ABKB,DC=brahimABKB,DC=local'

# Importer le module Active Directory
Import-Module ActiveDirectory

# Lire le fichier CSV et ajouter les utilisateurs
$users = Import-Csv -Path $csvPath

foreach ($user in $users) {
    $password = ConvertTo-SecureString $user.password -Force
    $userProps = @{
        'Name'           = $user.FirstName + ' ' + $user.LastName
        'GivenName'      = $user.FirstName
        'Surname'        = $user.LastName
        'SamAccountName' = $user.UserName
        'UserPrincipalName' = $user.UserName + 'brahimABKB.local'
        'DisplayName'    = $user.FirstName + ' ' + $user.LastName
        'Path'           = $ouPath
        'AccountPassword' = $password
        'Enabled'        = $false
        'ChangePasswordAtLogon' = $true
    }

    New-ADUser @userProps
}

Write-Host « Importation des utilisateurs terminée. »
```

--> Le script a bien été exécuté

```
Write-Host "Importation des utilisateurs terminée"
Importation des utilisateurs terminée
PS C:\Users\Administrateur>
```

--> Ouvrir l'AD et vérifier que les utilisateurs ont bien été créés

Utilisateurs et ordinateurs Active Directory			
Requêtes enregistrées			
brahimABKB.local			
ABKB			
groupes			
ordinateurs			
RDS			
utilisateurs			
Builtin			
Computers			
Domain Controllers			
ForeignSecurityPrincipal			
Managed Service Accounts			
ServerBrahim			
Users			
Nom	Type	Description	
ThomasAlice	Utilisateur		
RouxCatherine	Utilisateur		
RobertEmma	Utilisateur		
PiresAmandine	Utilisateur		
MoulinMarc	Utilisateur		
MoreauCarole	Utilisateur		
MichelJulien	Utilisateur		
MartinSophie	Utilisateur		
LemoineMarie	Utilisateur		
LemoineFranck	Utilisateur		
LefevreClaire	Utilisateur		
GarnierLuc	Utilisateur		
FournierIsabelle	Utilisateur		
DurandPierre	Utilisateur		
DupontJean	Utilisateur		
co-glpi co-glpi	Utilisateur		
ChauvinMonique	Utilisateur		
BouvierHenri	Utilisateur		
Binfo	Utilisateur		
BernardPaul	Utilisateur		
BenoitPhilippe	Utilisateur		

Conclusion:

En conclusion, l'utilisation d'un script PowerShell pour importer des utilisateurs à partir d'un fichier CSV est une méthode efficace et rapide pour gérer les comptes. Elle permet d'automatiser une tâche répétitive, de réduire les risques d'erreurs humaines et de gagner du temps, surtout lorsqu'il s'agit d'ajouter un grand nombre d'utilisateurs dans un système. C'est une solution pratique pour simplifier la gestion des utilisateurs et améliorer l'efficacité administrative.