

Projet : Gestion d'un Annuaire Active Directory via PowerShell

Auteur : Mamady Konate

Environnement : Windows Server

2022 Domaine : mamady.local

Date : 13 Novembre 2025



• Introduction et objectifs

Ce projet a pour objectif d'automatiser la gestion des comptes utilisateurs au sein d'un annuaire Active Directory via un script PowerShell complet et fiable. Il s'agit notamment de faciliter la création, la modification et la suppression des comptes, ainsi que la génération d'un rapport structuré permettant de visualiser l'état de l'annuaire.

Ce travail a été réalisé sur une machine Windows Server 2022, de poste d'administration PowerShell. Cette approche simplifie la configuration réseau tout en permettant de développer un environnement Active Directory pleinement fonctionnel.

• Environnement et configuration technique

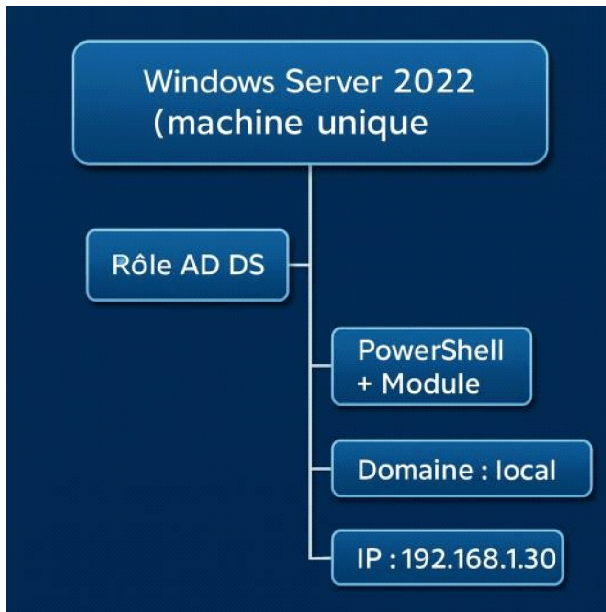
Le projet a été mené sur une machine virtuelle Windows Server 2022, configurée pour héberger :

- Le rôle Active Directory Domain Services (AD DS)
 - Le domaine mamady.local (les membres utilisés pour ce projet concernent mon AD principal directement).
- Le serveur DNS intégré ;
- L'environnement PowerShell nécessaire à l'exécution des scripts.

```
-----  
                               Bienvenue dans Windows Server 2022 Standard  
-----  
1) Domaine ou groupe de travail :      Domaine : Mamady.local  
2) Nom de l'ordinateur :                WIN-40J9LFQ5SUB
```

- **Architecture et topologie du système**

L'architecture du projet repose sur un modèle simple : une machine servant de contrôleur de domaine, d'hôte DNS, et de station d'administration PowerShell.



*À noter : j'ai une deuxième machine avec une interface graphique, sur le même réseau, j'ai donc pu vérifier si l'action menée fonctionnait correctement.

- **Déroulement du projet et procédures**

Le projet a suivi plusieurs étapes organisées :

- Installation du rôle AD DS et promotion en domaine mamady.local.
- Installation du module PowerShell « ActiveDirectory ».
- Création du répertoire de travail C:\Scripts\AD-Powershell.
- Création du script principal AD-UserMgmt.ps1.
- Exécution des tests et génération des rapports.

1/2.

```
+ FullyQualifiedErrorId : NamedParameterNotFound,Microsoft.Windows.ServerMa
PS C:\Users\Administrateur> Install-WindowsFeature -Name RSAT-AD-PowerShell

Success Restart Needed Exit Code      Feature Result
-----
True      No           NoChangeNeeded {}

PS C:\Users\Administrateur> Import-Module ActiveDirectory
PS C:\Users\Administrateur>
```

Ces commandes permettent d'activer les fonctionnalités PowerShell nécessaires pour administrer Active Directory depuis une machine Windows.

- **RSAT-AD-PowerShell** installe les outils d'administration à distance pour AD.
- **Import-Module ActiveDirectory** rend disponibles les cmdlets comme Get- ADUser, New-ADGroup, etc.

3.

```
PS C:\Users\Administrateur> New-Item -Path "C:\Scripts\AD-powershell" -ItemType Directory

Répertoire : C:\Scripts

Mode                LastWriteTime         Length Name
----                -
d-----          06/11/2025   08:19         AD-powershell

PS C:\Users\Administrateur>
```

Sert à créer un dossier nommé **AD-powershell** dans C:\Scripts. Ce dossier est destiné à organiser les scripts liés à l'administration d'Active Directory via PowerShell.

Description du script PowerShell

Le script AD-UserMgmt.ps1 automatise l'ensemble du processus de gestion des utilisateurs Active Directory.

Il repose sur un fichier CSV contenant les informations nécessaires à chaque opération : prénom, nom, SamAccountName,

mot de passe, groupe, action, etc.

Les fonctionnalités principales du script :

- Lecture du fichier CSV et validation des données ;
- Création, modification ou suppression des utilisateurs en fonction du champ « Action » ;
- Gestion automatique de l'appartenance aux groupes de sécurité ;
- Génération d'un rapport CSV comprenant : liste des utilisateurs, derniers logons, groupes associés.

```
PS C:\Users\Administrateur> @"
>> Action;SamAccountName;GivenName;Surname;DisplayName;UserPrincipalName;OU;Enabled;Password;ChangePasswordAtLogon;Groups;Create;mkonate;Mamady;Konate; Mamady Konate;mkonate@isitech.local;OU=UtilisateursDC=isitech,DC=local;True;1234;True;GRP-Informatique;GRP-Administration;Create;mmacron;Emmanuel;Macron;Emmanuel Macron;mmacron@isitech.local;OU=Utilisateurs,DC=isitech,DC=local;True;;False;GRP-Administration;Create;lhamilton;Lewis;Hamilton;Lewis Hamilton;lhamilton@isitech.local;OU=Utilisateurs,DC=isitech,DC=local;False;;False;
>> "@
Action;SamAccountName;GivenName;Surname;DisplayName;UserPrincipalName;OU;Enabled;Password;ChangePasswordAtLogon;Groups;Create;mkonate;Mamady;Konate; Mamady Konate;mkonate@isitech.local;OU=UtilisateursDC=isitech,DC=local;True;1234;True;GRP-Informatique;GRP-Administration;Create;mmacron;Emmanuel;Macron;Emmanuel Macron;mmacron@isitech.local;OU=Utilisateurs,DC=isitech,DC=local;True;;False;GRP-Administration;Create;lhamilton;Lewis;Hamilton;Lewis Hamilton;lhamilton@isitech.local;OU=Utilisateurs,DC=isitech,DC=local;False;;False;
PS C:\Users\Administrateur> _
```

Ainsi, ce script permet d'automatiser de manière efficace et fiable l'ensemble des opérations de gestion des utilisateurs au sein de l'annuaire Active Directory, en centralisant les actions dans un fichier CSV et en assurant à la fois la cohérence des données, la gestion des groupes et la génération de rapports exploitables.

La phase de création permet d'ajouter automatiquement un nouvel utilisateur dans l'annuaire Active Directory à partir des informations présentes dans le fichier CSV.

Le script utilise la commande `New-ADUser` en récupérant dynamiquement les attributs tels que le nom, le prénom, le login (SamAccountName), ainsi que l'unité d'organisation (OU) dans laquelle l'utilisateur sera créé.

Le mot de passe est converti en format sécurisé grâce à `ConvertTo-SecureString`, puis le compte est activé selon les paramètres définis dans le fichier CSV.

```
>> foreach ($u in users) {
>> # Action de base (créer, modifier, supprimer)
>> switch ($u.Action) {
>> "Create" {
>> -New-ADUser -SamAccountName $u.SamAccountName `
>> -UserPrincipalName $u.UserPrincipalName `
>> -Name $u.DisplayName `
>> -GivenName $u.GivenName `
>> -surname $u.Surname
>> -Path $u.OU `
>> -AccountPassword (ConvertTo-SecureString $u.Password -AsPlainText -Force) `
>> -Enabled ([bool]::Parse($u.Enabled)) `
>> -ChangePasswordAtLogon ([bool]::Parse($u.ChangePasswordAtLogon)) `
>> -PassThru | Out-Null
>> Write-Host " Utilisateur $($u.SamAccountName) créé avec succès."
>> -
>> }
>> }
```

La phase de modification permet de mettre à jour les informations d'un utilisateur déjà existant dans Active Directory.

Le script identifie l'utilisateur grâce à son `SamAccountName`, puis applique les modifications nécessaires à l'aide de la commande `Set-ADUser`.

```
>> "Modify" {
>> Set-ADUser -Identity $u.SamAccount `
>> -GivenName $u.GivenName `
>> -Surname $u.Surname `
>> -DisplayName $u.DisplayName
>> Write-Host "Utilisateur $($u.SamAccountName) modifié."
>> }
>> }
```

La phase de suppression permet de retirer définitivement un utilisateur de l'annuaire Active Directory.

Le script utilise la commande `Remove-ADUser` en se basant sur le `SamAccountName` pour identifier le compte à supprimer.

L'option `-Confirm:$false` permet de désactiver la confirmation manuelle, ce qui rend le processus entièrement automatisé.

```
>> >> "Delete" {
>> >> Remove-ADUser -Identity $u.SamAccountName -Confirm:$false
>> Write-Host "Utilisateur $($u.SamAccountName) supprimé."
>> }
>> default {
>> Write-Warning "Action inconnue '$($u.Action)' pour =(=u.SamAccountName)."
>> }
>> }
```

Commandes PowerShell utilisées

Le projet repose sur les commandes PowerShell suivantes :

- `Import-Module ActiveDirectory` -> charge le module AD
- `New-ADUser` / `Set-ADUser` / `Remove-ADUser` -> manipulation des comptes utilisateurs
- `Add-ADGroupMember` -> gestion des groupes
- `Export-Csv` : génération de rapports.

• Résultats et livrables

Le script a fonctionné conformément aux attentes. Les comptes ont été créés automatiquement, leurs groupes associés ont été correctement attribués, et le rapport final a été généré.

Livrables produits :

- Script PowerShell AD-UserMgmt.ps1
- Fichier CSV de création utilisateur

```
PS C:\Scripts\AD-Powershell> Import-Csv .\users.csv -Delimiter ';' | Format-Table
Action SamAccountName GivenName Surname DisplayName UserPrincipalName OU
-----
Create mkonate Mamady Konate Mamady Konate mkonate@isitech.local OU=Utilisateurs,DC=isitech...
Modify mpaul Marie Paul Marie Paul mpaul@isitech.local OU=Utilisateurs,DC=isitech...
```

- **Conclusion et perspectives**

Ce projet a permis d'acquérir une solide expérience dans l'automatisation des tâches Active Directory à l'aide de PowerShell.

Les outils mis en place pourraient être enrichis de nouvelles fonctions, comme la détection automatique des comptes inactifs.