# Documentation montage d'un AD sans GUI

- ## Préparation d'une VM :

  Pour lancer l'AD on vas utiliser un VM sur VMWare
  Pour cela on vas créer un VM avec l'ISO Windows server 2022 avec comme config 4000 ram, 4 CPU et 100G de mémoire
  Apres avoir lancer la VM on modifie l'adresse IP avec cette commande :

```
PS C:\Users\Administrateur> New-NetIPAddress -InterfaceAlias Ethernet -IPAddress 192.168.56.21 -PrefixLength 24


IPAddress         : 192.168.56.21
InterfaceIndex    : 3
InterfaceAlias    : Ethernet
AddressFamily     : IPv4
Type              : Unicast
PrefixLength      : 24
PrefixOrigin      : Manual
SuffixOrigin      : Manual
AddressState      : Tentative
ValidLifetime     : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
SkipAsSource      : False
PolicyStore       : ActiveStore

IPAddress         : 192.168.56.21
InterfaceIndex    : 3
InterfaceAlias    : Ethernet
AddressFamily     : IPv4
Type              : Unicast
PrefixLength      : 24
PrefixOrigin      : Manual
SuffixOrigin      : Manual
AddressState      : Invalid
ValidLifetime     : Infinite ([TimeSpan]::MaxValue)
PreferredLifetime : Infinite ([TimeSpan]::MaxValue)
```

  Juste après on vas rajouter un server DNS grâce a :

```
Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddress 192.168.56.21 , 8.8.8.8
```

  Il faut aussi également télécharger l'AD avant restart :

```
Install-WindowsFeature -Name AD-Domain-Services -IncludeManagementTools
```

  Ensuite on modifie le nom du computer grâce a la commande
  « Rename-Computer -NewName  AD1 »
  Il faut par contre redémarrer le PC avec « Restart-Computer

  Je les mit en script pour plus de faciliter pour l'AD2

```
New-NetIPAddress -InterfaceAlias Ethernet -IPAddress
192.168.56.22 -PrefixLength 24

Set-DnsClientServerAddress -InterfaceAlias Ethernet -
ServerAddress 192.168.56.21 , 8.8.8.8

Install-WindowsFeature -Name AD-Domain-Services -
IncludeManagementTools

Rename-Computer -NewName AD2

Restart-Computer
```

- Mise en place du AD :

Pour créer l'AD1 on vas utiliser le script suivant :

```
$domain = (Get-WmiObject -Class Win32
_ComputerSystem).Domain
if ($domain -ne "WORKGROUP") {
    Write-Host "L'ordinateur est déjà membre d'un
domaine. L'ajout d'AD sera arrêté."
    exit
}
Import-Module ADDSDeployment
Install-ADDSForest `
-CreateDnsDelegation:$false `
-DatabasePath "C:\Windows\NTDS" `
-DomainMode "WinThreshold" `
-DomainName "charlyne.local" `
-DomainNetbiosName "CHARLYNE" `
-ForestMode "WinThreshold" `
-InstallDns:$true `
-LogPath "C:\Windows\NTDS" `
-NoRebootOnCompletion:$false `
-SysvolPath "C:\Windows\SYSVOL" `
-Force:$true
```

Le script comporte un partie permettant de vérifier si un AD existe déjà et donc arrêter le script si c'est le cas.

## -Montage AD2 :

Pour préparer la VM de l'AD2 on vas utiliser le script créer lors de la première partie

Et on vas utiliser le script suivant pour monter l'AD :

```
$domain = (Get-WmiObject -Class Win32
_ComputerSystem).Domain
if ($domain -ne "WORKGROUP") {
    Write-Host "L'ordinateur est déjà membre d'un
domaine. L'ajout d'AD sera arrêté."
    exit
}
Import-Module ADDSDeployment
Install-ADDSDomainController `
-NoGlobalCatalog:$false `
-CreateDnsDelegation:$false `
-Credential (Get-Credential) `
-CriticalReplicationOnly:$false `
-DatabasePath "C:\Windows\NTDS" `
-DomainName "charlyne.local" `
-InstallDns:$true `
-LogPath "C:\Windows\NTDS" `
-NoRebootOnCompletion:$false `
-SiteName "Default-First-Site-Name" `
-SysvolPath "C:\Windows\SYSVOL" `
-Force:$true
```

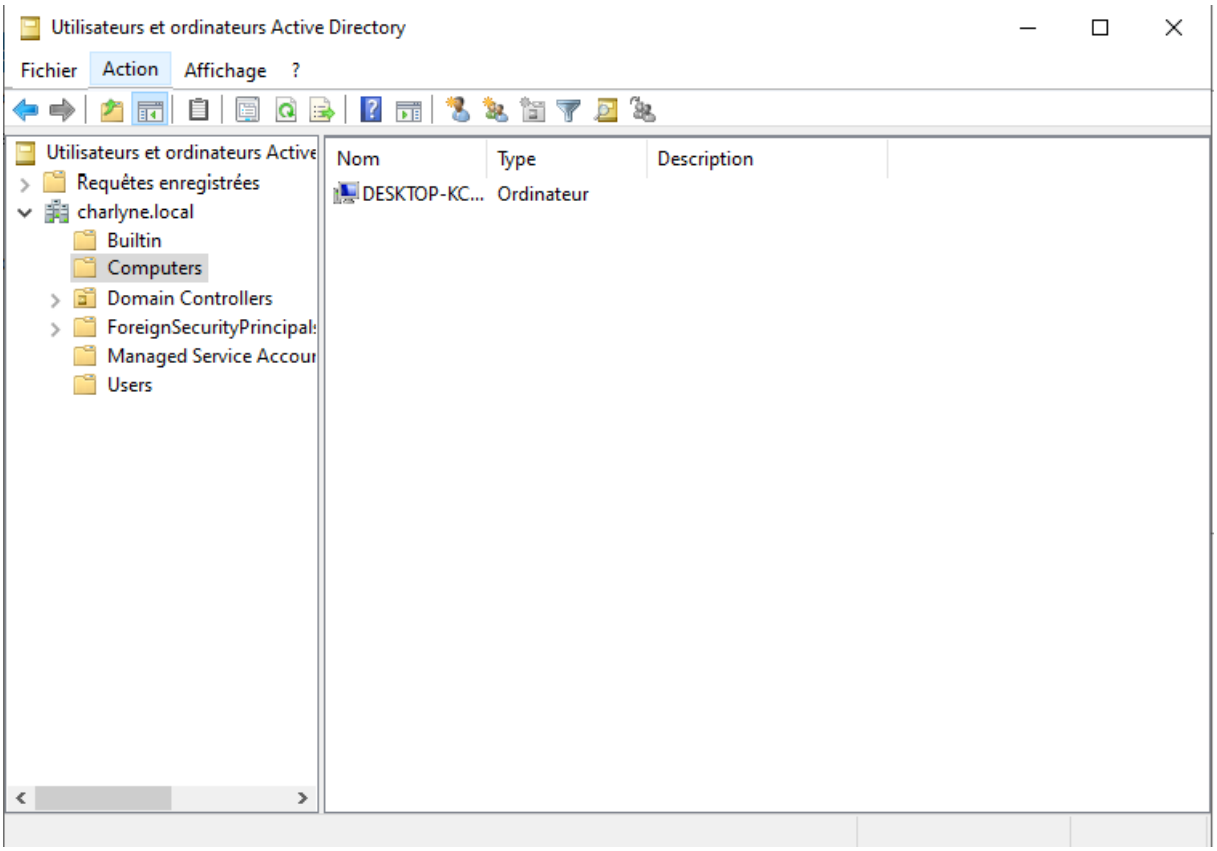
Comme pour le script précédent il y a une partie pour vérifier si il existe ou nom un AD

## -Montage Windows Client (Windows 11) :

Pour commencer on utiliser un script_prep comme celui du début masi avec quelque modifications :

```
Set-ExecutionPolicy Unrestricted -Scope CurrentUser

New-NetIPAddress -InterfaceAlias Ethernet -IPAddress
192.168.56.31 -PrefixLength 24

Set-DnsClientServerAddress -InterfaceAlias Ethernet -
ServerAddress 192.168.56.21 , 192.168.56.22

Rename-Computer -NewName client_1

Add-Computer -Domain charlyne.local

Restart-Computer
```

Voici le script complet pour la préparations d'un pc et la mise en place dans le domaine




Voici la preuve on voit que le serveur AD reconnait l'ordinateur

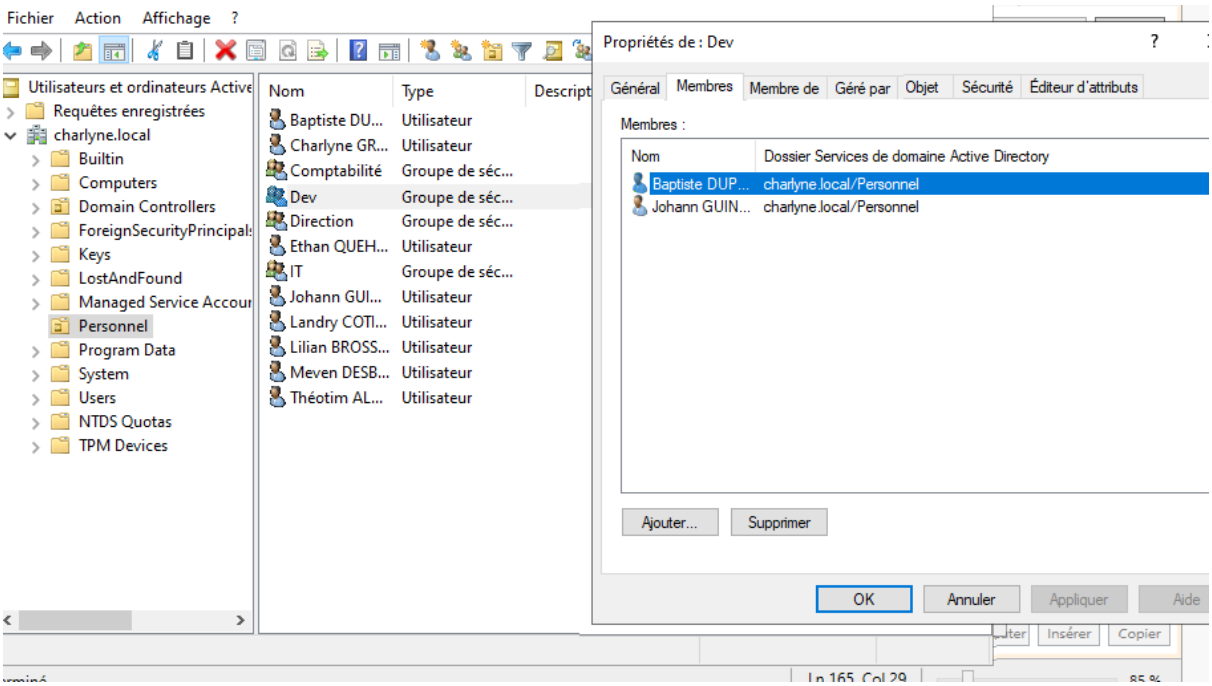
# Création d’un dossier OU , d’un groupe et des utilisateurs

Pour créer un dossier OU des groupes et des user on vas utiliser le script suivant :

```powershell
if (-not (Get-ADOrganizationalUnit -Filter {Name -eq "Personnel"})) {
    New-ADOrganizationalUnit -Name "Personnel" -Path "DC=charlyne,DC=local"
}


$GroupNames = @("Dev", "Direction", "IT", "Comptabilité")
foreach ($GroupName in $GroupNames) {
    if (-not (Get-ADGroup -Filter {Name -eq $GroupName})) {
        New-ADGroup -Name $GroupName -Path "OU=Personnel,DC=charlyne,DC=local" -GroupScope Global
    }
}

function Create-UserIfNotExists {
    param (
        [string]$SamAccountName,
        [string]$UserPrincipalName,
        [string]$Name,
        [string]$GivenName,
        [string]$Surname,
        [string]$DisplayName,
        [string]$Password
    )

    if (-not (Get-ADUser -Filter {SamAccountName -eq $SamAccountName})) {
        New-ADUser -SamAccountName $SamAccountName `
                   -UserPrincipalName $UserPrincipalName `
                   -Name $Name `
                   -GivenName $GivenName `
                   -Surname $Surname `
                   -DisplayName $DisplayName `
                   -Path "OU=Personnel,DC=charlyne,DC=local" `
                   -AccountPassword (ConvertTo-SecureString $Password -AsPlainText -Force) `
                   -Enabled $true `
                   -Server "charlyne.local"
    }
}

$Users = @(
    @{ SamAccountName = "baptiste.D"; UserPrincipalName = "baptistedupuis@charlyne.local"; Name = "Baptiste DUPUIS"; GivenName = "Baptiste"; Surname = "DUPUIS"; DisplayName = "Baptiste
DUPUIS"; Password = "Azerty14/04"; Group = "Dev" },
```

```powershell
$Users = @(
    @{ SamAccountName = "baptiste.D"; UserPrincipalName = "baptistedupuis@charlyne.local"; Name = "Baptiste DUPUIS"; GivenName = "Baptiste"; Surname = "DUPUIS"; DisplayName = "Baptiste
DUPUIS"; Password = "Azerty14/04"; Group = "Dev" },
    @{ SamAccountName = "Johann.G"; UserPrincipalName = "johannguinberteau@charlyne.local"; Name = "Johann GUINBERTEAU"; GivenName = "Johann"; Surname = "GUINBRTEAU"; DisplayName = "Johann
GUINBERTEAU"; Password = "Azerty14/04"; Group = "Dev" },
    @{ SamAccountName = "charlyne.G"; UserPrincipalName = "charlynegraincourtmerieau@charlyne.local"; Name = "Charlyne GRAINCOURT-MERIEAU"; GivenName = "Charlyne"; Surname = "GRAINCOURT-
MERIEAU"; DisplayName = "Charlyne GRAINCOURT-MERIEAU"; Password = "Azerty14/04"; Group = "Direction" },
    @{ SamAccountName = "ethan.Q"; UserPrincipalName = "ethanquehelecozler@charlyne.local"; Name = "Ethan QUEHE-LE COZLER"; GivenName = "Ethan"; Surname = "QUEHE-LE COZLER"; DisplayName =
"Ethan QUEHE-LE COZLER"; Password = "Azerty14/04"; Group = "Direction" },
    @{ SamAccountName = "meven.D"; UserPrincipalName = "mevendesbois@charlyne.local"; Name = "Meven DESBOIS"; GivenName = "Meven"; Surname = "DESBOIS"; DisplayName = "Meven DESBOIS";
Password = "Azerty14/04"; Group = "IT" },
    @{ SamAccountName = "landry.C"; UserPrincipalName = "landrycotillon@charlyne.local"; Name = "Landry COTILLON"; GivenName = "Landry"; Surname = "COTILLON"; DisplayName = "Landry
COTILLON"; Password = "Azerty14/04"; Group = "IT" },
    @{ SamAccountName = "lilian.B"; UserPrincipalName = "lilianbrosset@charlyne.local"; Name = "Lilian BROSSET"; GivenName = "Lilian"; Surname = "BROSSET"; DisplayName = "Lilian BROSSET";
Password = "Azerty14/04"; Group = "Comptabilité" },
    @{ SamAccountName = "théotim.A"; UserPrincipalName = "théotimalberteau@charlyne.local"; Name = "Théotim ALBERTAU"; GivenName = "Théotim"; Surname = "ALBERTAU"; DisplayName = "Théotim
ALBERTAU"; Password = "Azerty14/04"; Group = "Comptabilité" }
)

foreach ($user in $Users) {
    Create-UserIfNotExists -SamAccountName $user.SamAccountName `
                           -UserPrincipalName $user.UserPrincipalName `
                           -Name $user.Name `
                           -GivenName $user.GivenName `
                           -Surname $user.Surname `
                           -DisplayName $user.DisplayName `
                           -Password $user.Password

    if (-not (Get-ADGroupMember -Identity $user.Group | Where-Object { $_.SamAccountName -eq $user.SamAccountName })) {
        Add-ADGroupMember -Identity $user.Group -Members $user.SamAccountName
    }
}
```

Voici le résultat obtenu avec ce script on vois bine que les groupes et user on été créer dans le dossier Personnel et que tout le monde est dans leur groupe

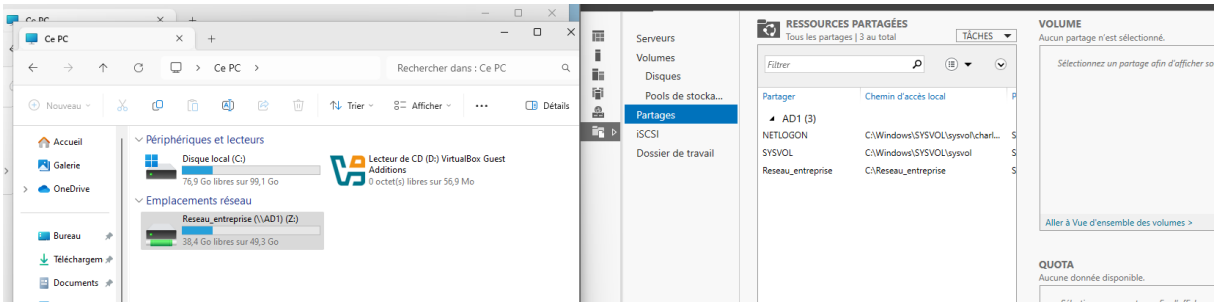


# Créations d’un dossier partager :

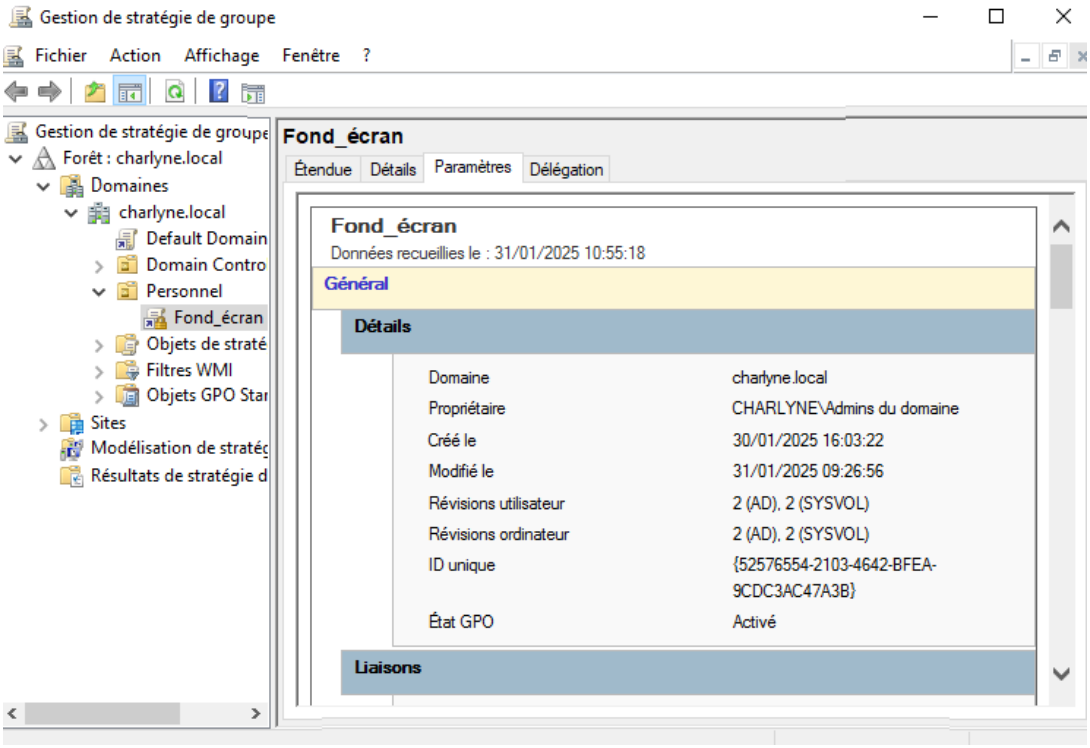
Voici le script pour créer un dossier SMB partagé:

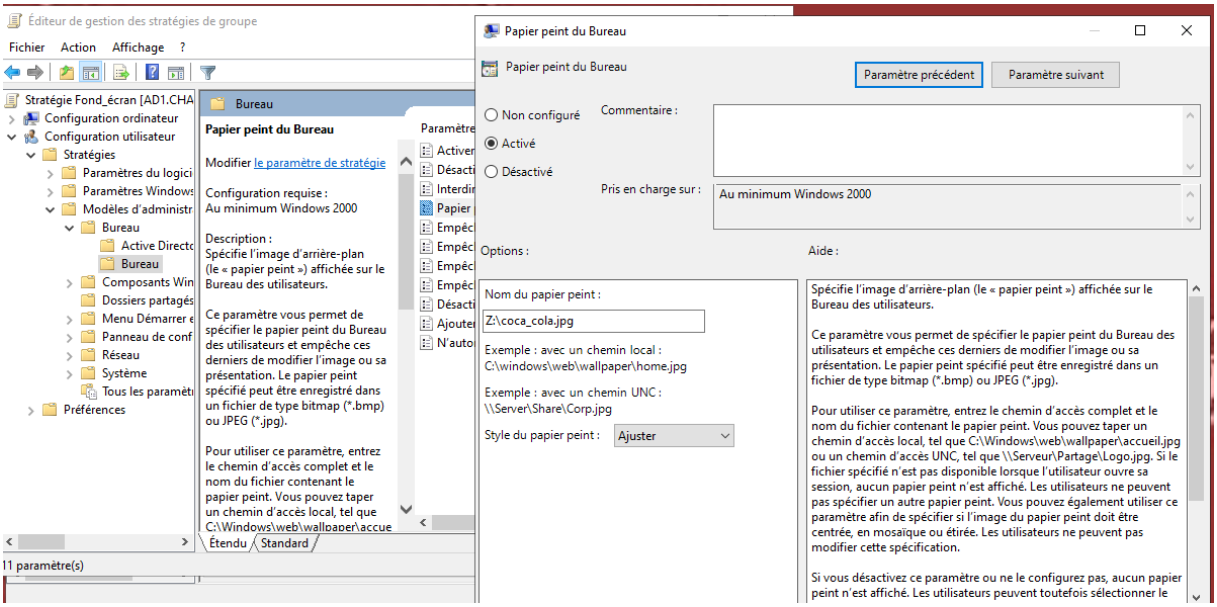
```
Install-WindowsFeature FS-FileServer


if (-not (Test-Path "C:\Reseaux_entreprise")) {
    New-Item -Path "C:\Reseaux_entreprise" -ItemType
Directory
}
else{
        Write-host "le dossier existe déjà"
}

$shareName = "Reseaux_entreprise"
if (-not (Get-SmbShare -Name $shareName -ErrorAction
SilentlyContinue)) {
    New-SmbShare -Name $shareName -Path "C:
\Reseaux_entreprise" -FullAccess "Direction", "Dev", "IT",
"Comptabilité", "Administrateur"
}
else{
    Write-Host "le dossier partager"
}
```

Résultat :

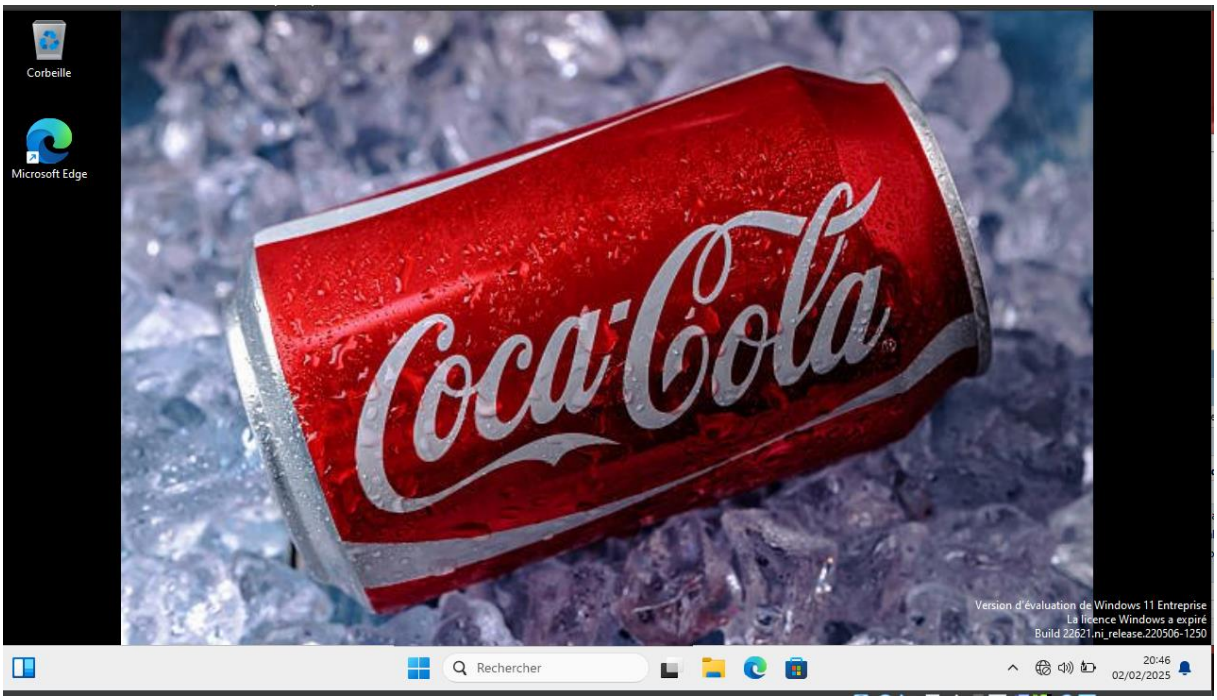


# Création d’une GPO :

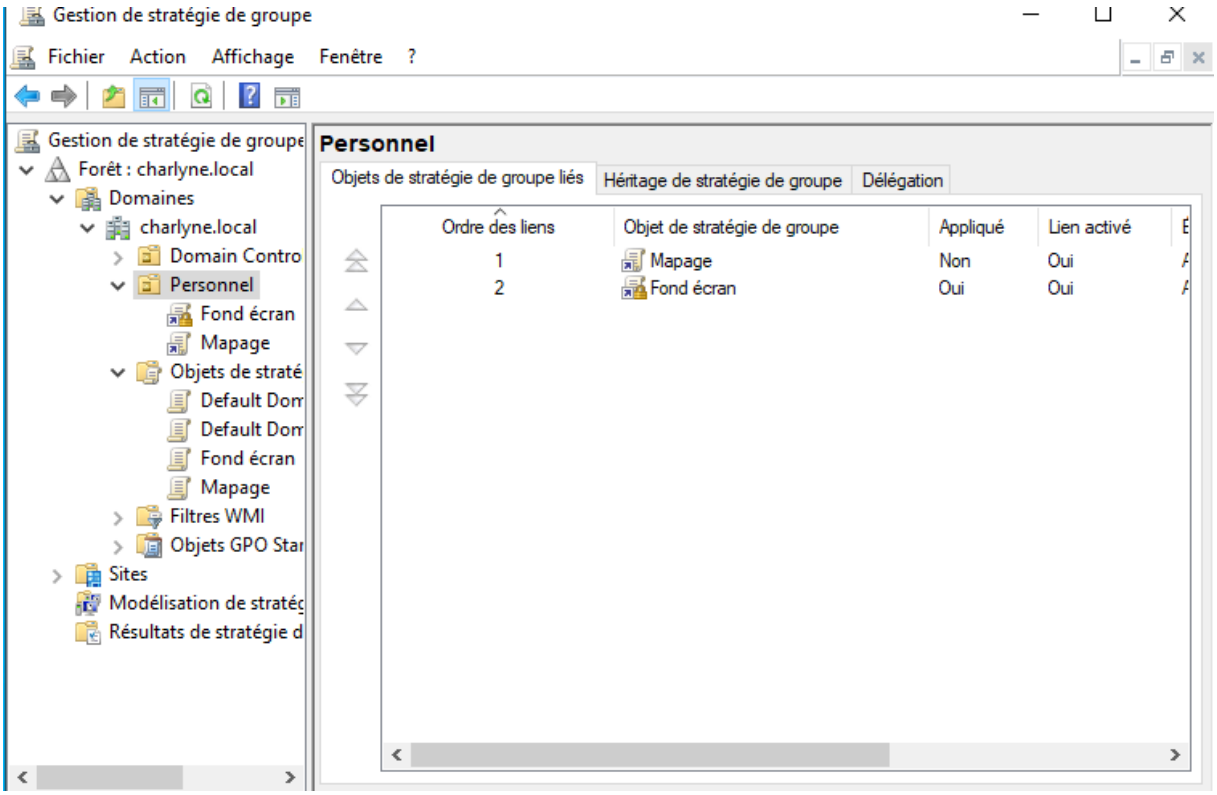


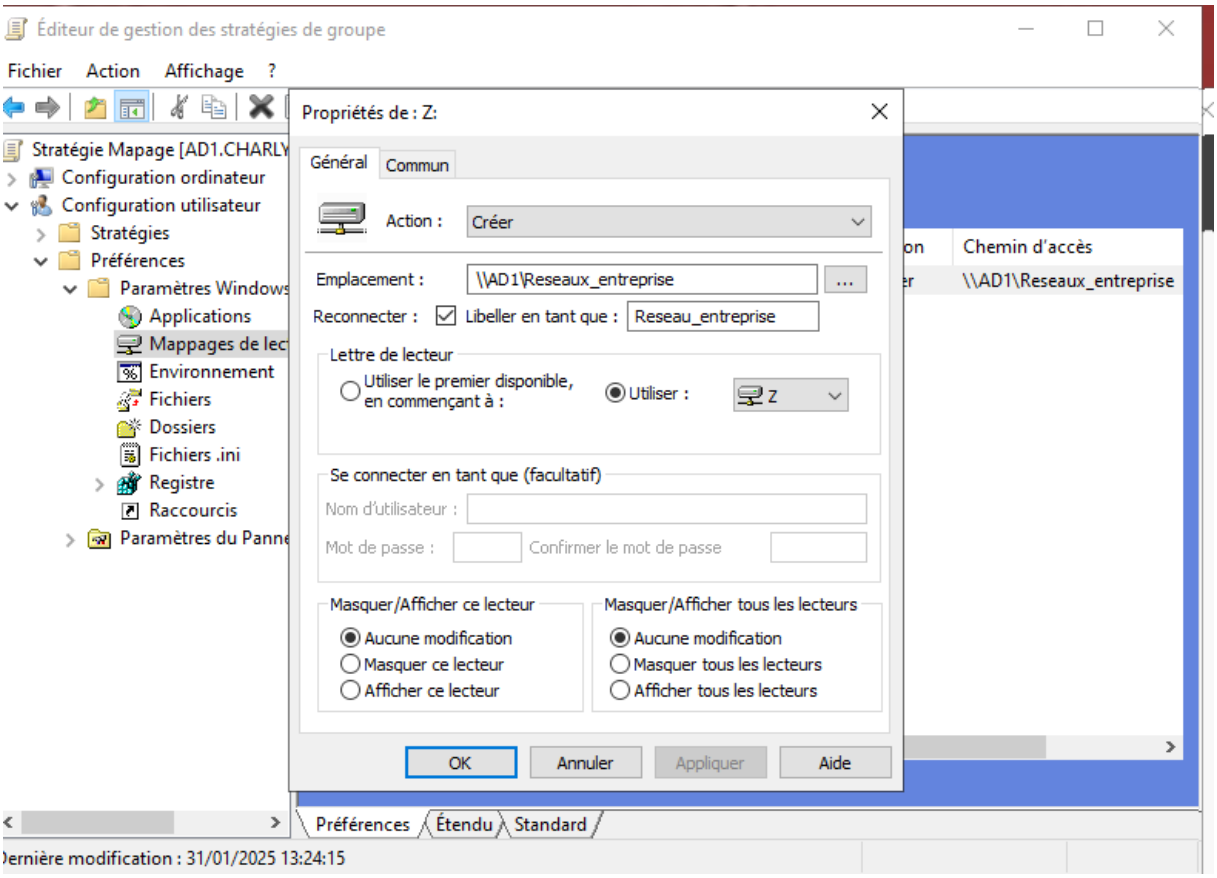




Cette GPO permet de modifier le fond d’écran de l’ordinateur client voici le résultat

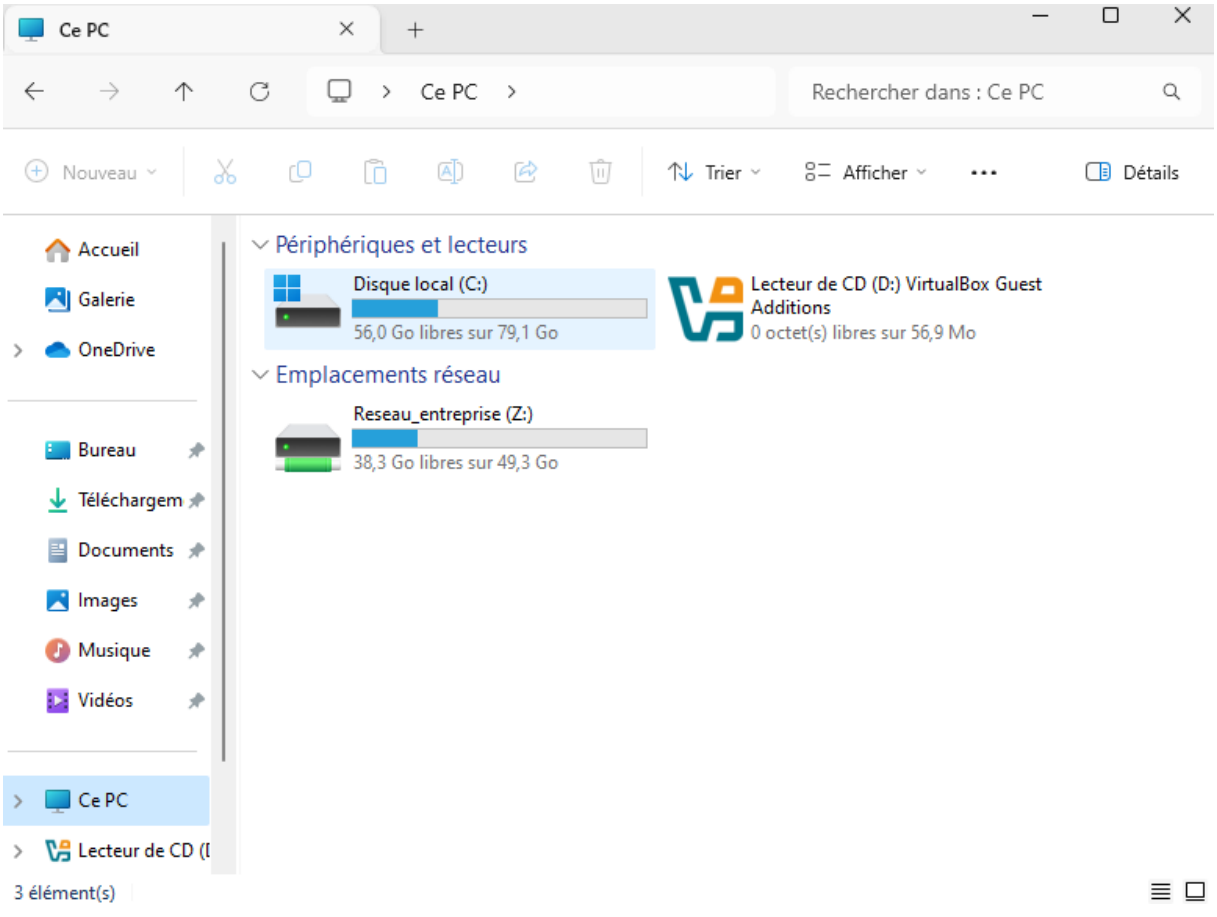



Une autre GPO pour le mappage automatique du dossier partager :





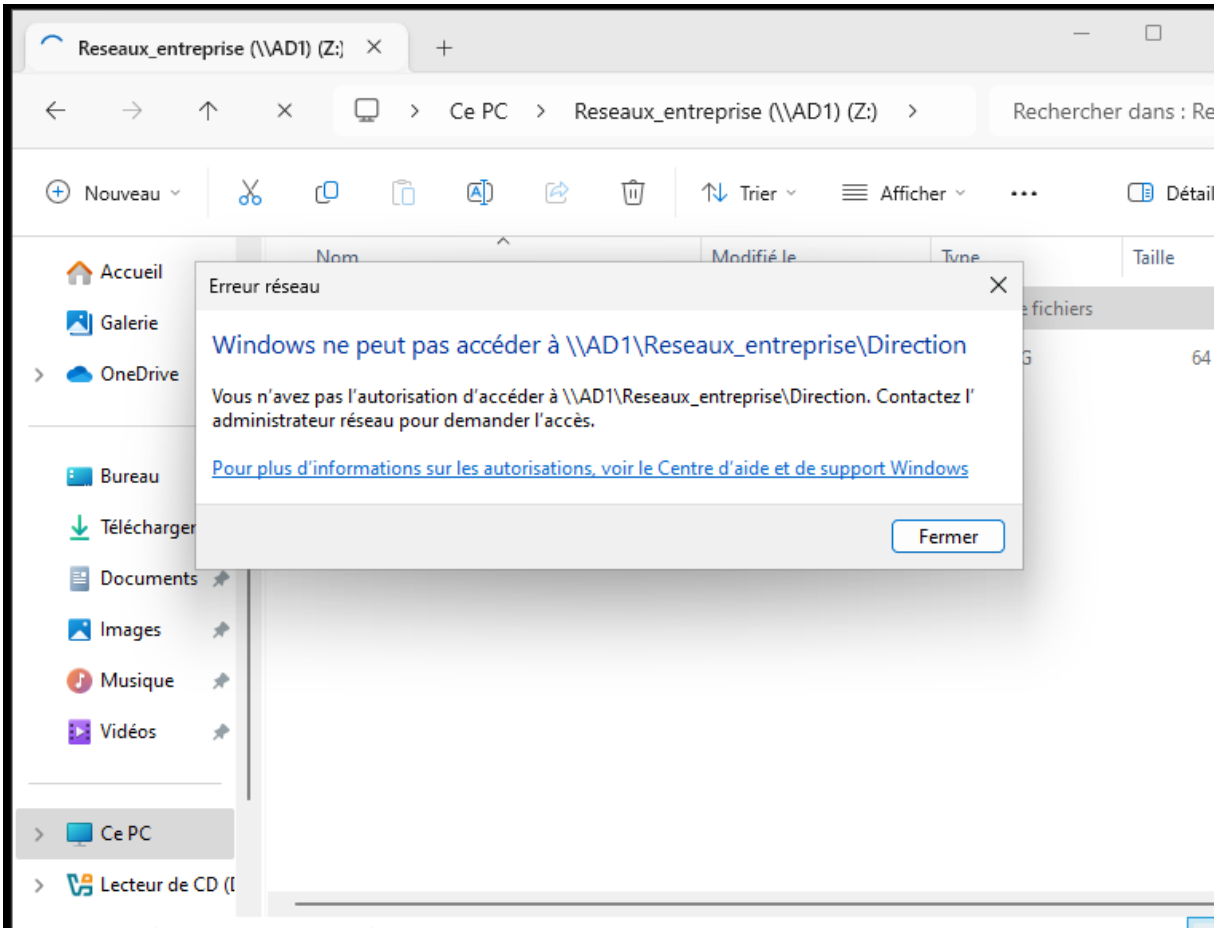

Voici le résultat



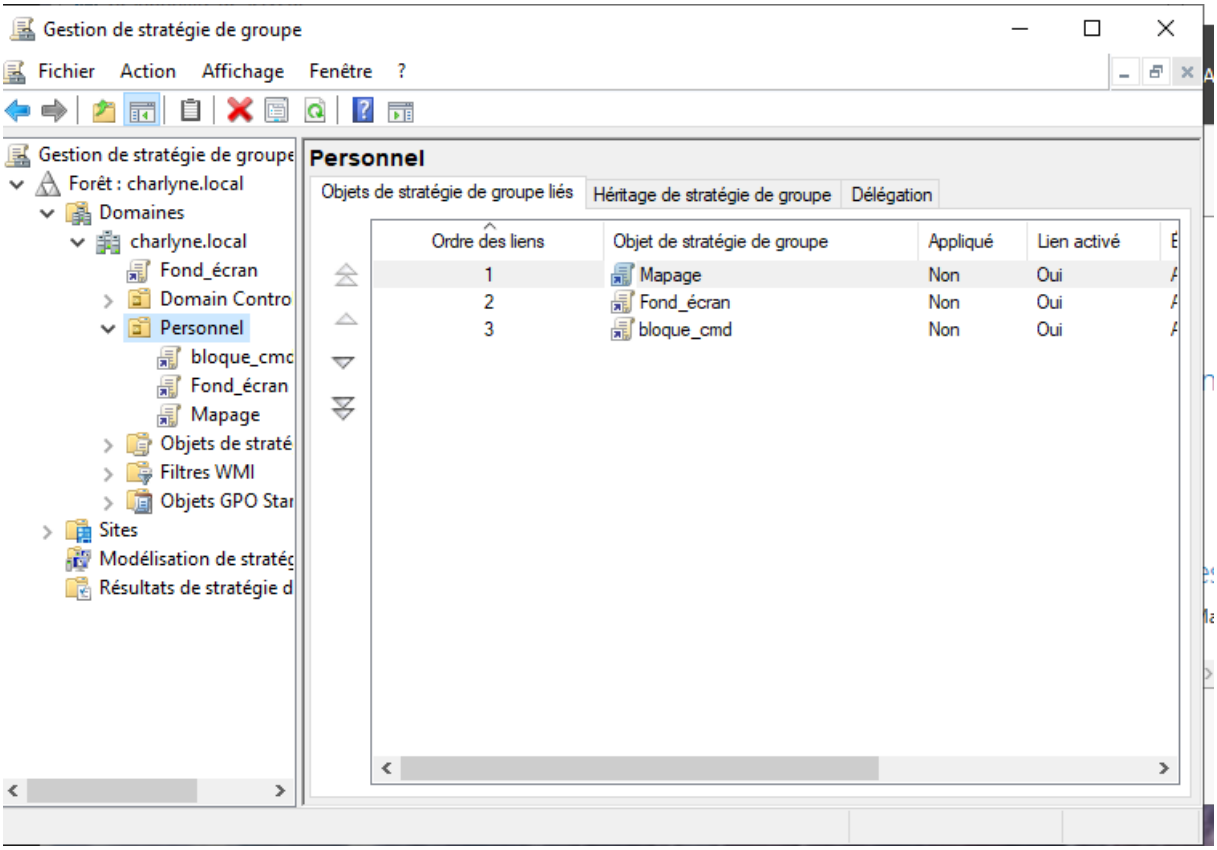
Création d’un script permettant de géré les permission sur un dossier

```
$dossier = "C:\Reseaux_entreprise\script"
$acl = Get-Acl $dossier


$accessRule = New-Object
System.Security.AccessControl.FileSystemAccessRule("Dev",'
FullControl", "Deny")

$acl.AddAccessRule($accessRule)

Set-Acl $dossier $acl
```

Voici le résultat de ce script :



Voici la GPO pour bloqués le cmd :




Voici le résultat