

Systemes Microsoft Windows

Initiation aux scripts Powershell

Davia Moujabber-EXSI3P

Partie 1 : Interrogation du niveau de stratégie d'exécution de Powershell

```
PS C:\testPowershell> Get-ExecutionPolicy
Unrestricted
PS C:\testPowershell>
```

Partie 2 : Modification du niveau de stratégie d'exécution de Powershell

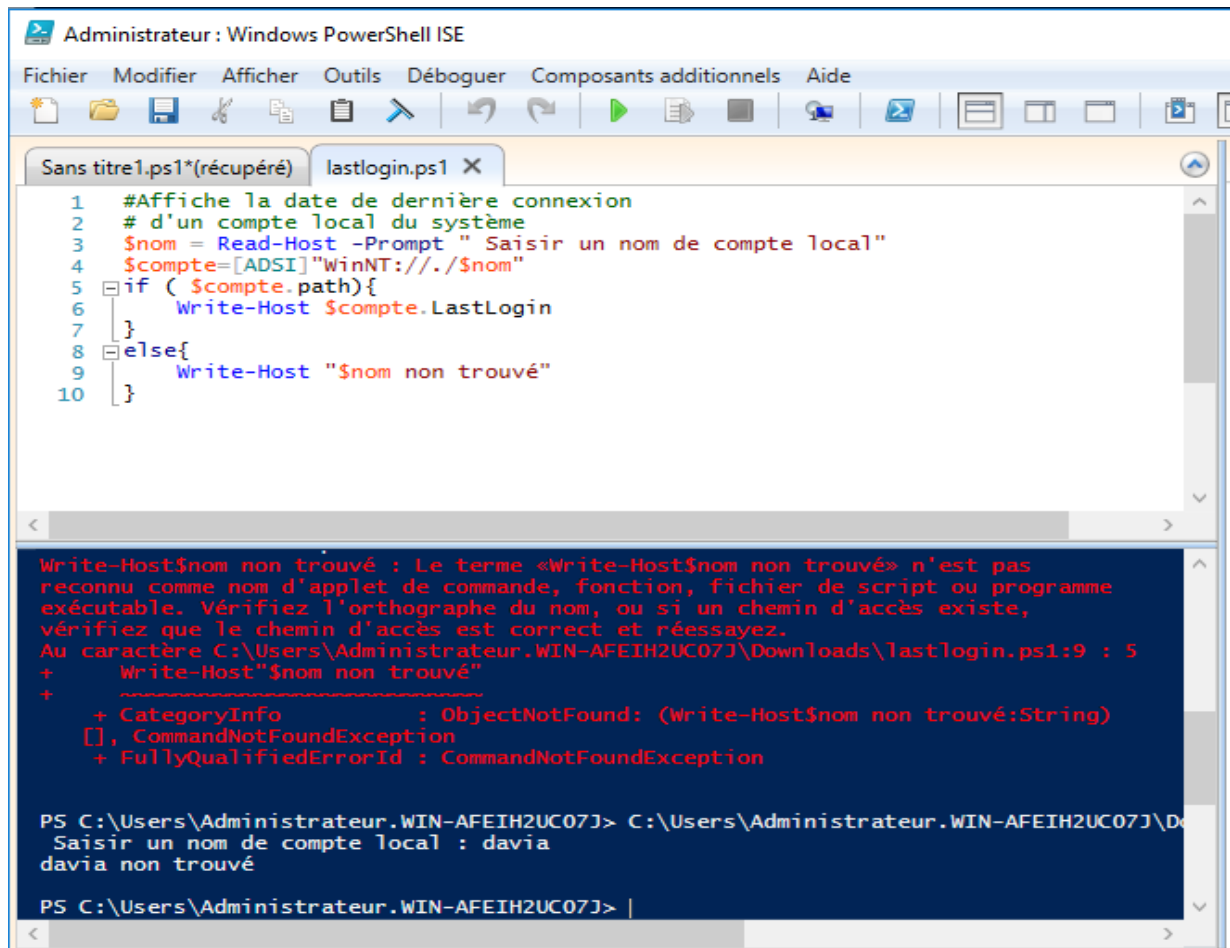
```
PS C:\WINDOWS\system32> Set-ExecutionPolicy

applet de commande Set-ExecutionPolicy à la position 1 du pipeline de la commande
Fournissez des valeurs pour les paramètres suivants :
ExecutionPolicy: RemoteSigned

Modification de la stratégie d'exécution
La stratégie d'exécution permet de vous prémunir contre les scripts que vous jugez non fiables. En modifiant la
stratégie d'exécution, vous vous exposez aux risques de sécurité décrits dans la rubrique d'aide
about_Execution_Policies à l'adresse https://go.microsoft.com/fwlink/?LinkID=135170. Voulez-vous modifier la stratégie
d'exécution ?
[0] Oui [T] Oui pour tout [N] Non [U] Non pour tout [S] Suspendre [?] Aide (la valeur par défaut est « N ») : o
PS C:\WINDOWS\system32>
```

Partie 4 : Initiation aux scripts PS via ISE

1. Accès aux comptes locaux du système
 - a. Créer et enregistrer le script lastlogin.ps1 qui permet d'afficher la date de dernière connexion d'un compte local du système.



The screenshot shows the Windows PowerShell ISE interface. The title bar reads "Administrateur : Windows PowerShell ISE". The menu bar includes "Fichier", "Modifier", "Afficher", "Outils", "Débuguer", "Composants additionnels", and "Aide". The toolbar contains icons for file operations and execution. Two tabs are open: "Sans titre1.ps1*(récupéré)" and "lastlogin.ps1 X". The script in the editor is as follows:

```
1 #Affiche la date de dernière connexion
2 # d'un compte local du système
3 $nom = Read-Host -Prompt " Saisir un nom de compte local"
4 $compte=[ADSI]"WinNT://./$nom"
5 if ( $compte.path){
6     Write-Host $compte.LastLogin
7 }
8 else{
9     Write-Host "$nom non trouvé"
10 }
```

The console output shows an error message in red text:

```
Write-Host$nom non trouvé : Le terme «Write-Host$nom non trouvé» n'est pas
reconnu comme nom d'applet de commande, fonction, fichier de script ou programme
exécutable. Vérifiez l'orthographe du nom, ou si un chemin d'accès existe,
vérifiez que le chemin d'accès est correct et réessayez.
Au caractère C:\Users\Administrateur.WIN-AFEIH2UC07J\Downloads\lastlogin.ps1:9 : 5
+ Write-Host"$nom non trouvé"
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (Write-Host$nom non trouvé:String)
[] , CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException
```

Below the error, the prompt shows the user input "davia" and the output "davia non trouvé".

```
PS C:\Users\Administrateur.WIN-AFEIH2UC07J> C:\Users\Administrateur.WIN-AFEIH2UC07J\Downloads\lastlogin.ps1
Saisir un nom de compte local : davia
davia non trouvé
PS C:\Users\Administrateur.WIN-AFEIH2UC07J> |
```

B.

Administrateur : Windows PowerShell ISE

Fichier Modifier Afficher Outils Débugger Composants additionnels Aide

Sans titre1.ps1*(récupéré) lastlogin.ps1 X

```
1 #Affiche la date de dernière connexion
2 # d'un compte local du système
3 $nom = Read-Host -Prompt " Saisir un nom de compte local"
4 $compte=[ADSI]"WinNT://./$nom"
5 if ( $compte.path){
6     Write-Host $compte.LastLogin
7 }
8 else{
9     Write-Host "$nom non trouvé"
10 }
```

PS C:\Users\Administrateur.WIN-AFEIH2UC07J> C:\Users\Administrateur.WIN-AFEIH2UC07J\Documents> Saisir un nom de compte local : davia
davia non trouvé

PS C:\Users\Administrateur.WIN-AFEIH2UC07J> C:\Users\Administrateur.WIN-AFEIH2UC07J\Documents> Saisir un nom de compte local : Administrateur
28/01/2021 09:56:20

PS C:\Users\Administrateur.WIN-AFEIH2UC07J> \$compte | Get-Member

TypeName : System.DirectoryServices.DirectoryEntry

Name	MemberType	Definition
----	-----	-----

Terminé | Ln 83 Col 45

```
PS C:\Users\Administrateur.WIN-AFEIH2UC07J> $compte | Get-Member
```

TypeName : System.DirectoryServices.DirectoryEntry

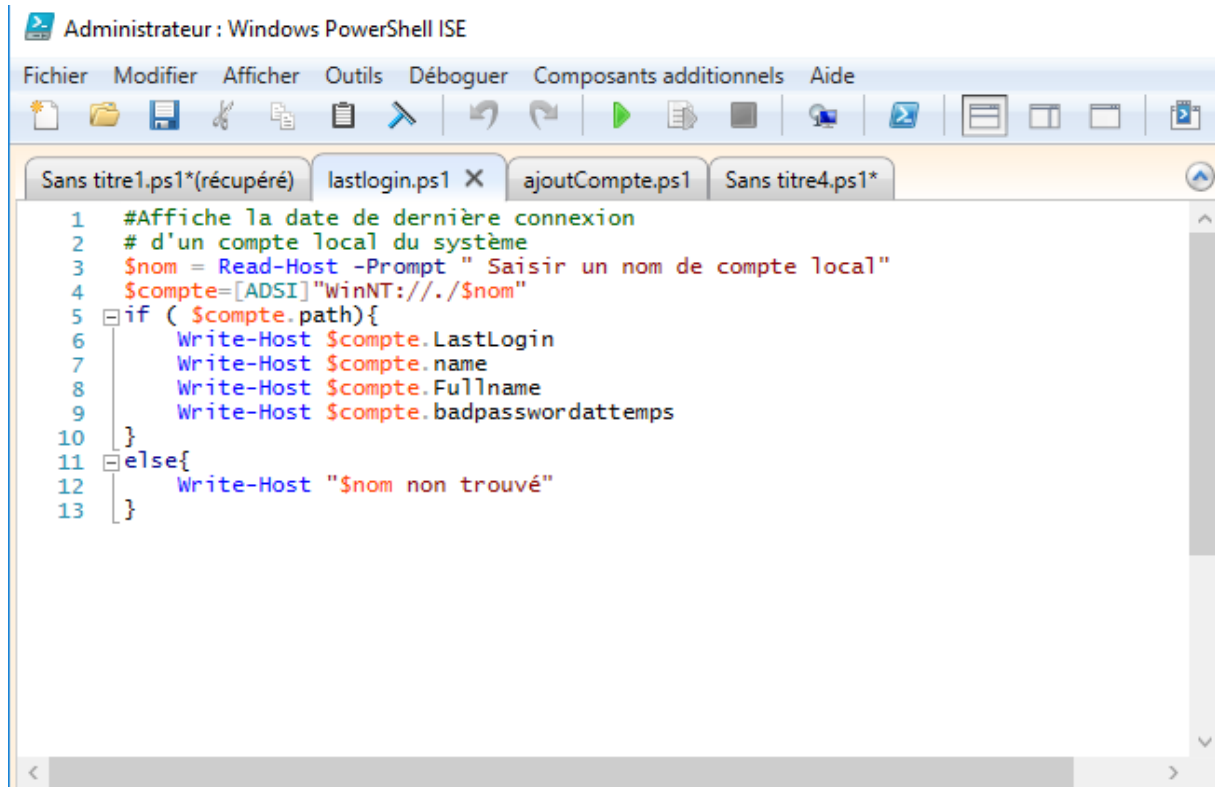
Name	MemberType	Definition
ConvertDNWithBinaryToString	CodeMethod	static string ConvertDNWithBinaryToStrin...
ConvertLargeIntegerToInt64	CodeMethod	static long ConvertLargeIntegerToInt64(p...
AutoUnlockInterval	Property	System.DirectoryServices.PropertyValueCo...
BadPasswordAttempts	Property	System.DirectoryServices.PropertyValueCo...
Description	Property	System.DirectoryServices.PropertyValueCo...
FullName	Property	System.DirectoryServices.PropertyValueCo...
HomeDirDrive	Property	System.DirectoryServices.PropertyValueCo...
HomeDirectory	Property	System.DirectoryServices.PropertyValueCo...
LastLogin	Property	System.DirectoryServices.PropertyValueCo...
LockoutObservationInterval	Property	System.DirectoryServices.PropertyValueCo...
LoginHours	Property	System.DirectoryServices.PropertyValueCo...
LoginScript	Property	System.DirectoryServices.PropertyValueCo...
MaxBadPasswordsAllowed	Property	System.DirectoryServices.PropertyValueCo...
MaxPasswordAge	Property	System.DirectoryServices.PropertyValueCo...
MaxStorage	Property	System.DirectoryServices.PropertyValueCo...
MinPasswordAge	Property	System.DirectoryServices.PropertyValueCo...
MinPasswordLength	Property	System.DirectoryServices.PropertyValueCo...
Name	Property	System.DirectoryServices.PropertyValueCo...
objectSid	Property	System.DirectoryServices.PropertyValueCo...
Parameters	Property	System.DirectoryServices.PropertyValueCo...
PasswordAge	Property	System.DirectoryServices.PropertyValueCo...
PasswordExpired	Property	System.DirectoryServices.PropertyValueCo...
PasswordHistoryLength	Property	System.DirectoryServices.PropertyValueCo...

Nom: Write-Host \$compte.name

Nom complet : Write-Host \$compte.FullName

Le nombre de tentatives de connexion avec un mauvais password : Write-Host
\$compte.badpasswordattempts

Tester le script avec les propriétés

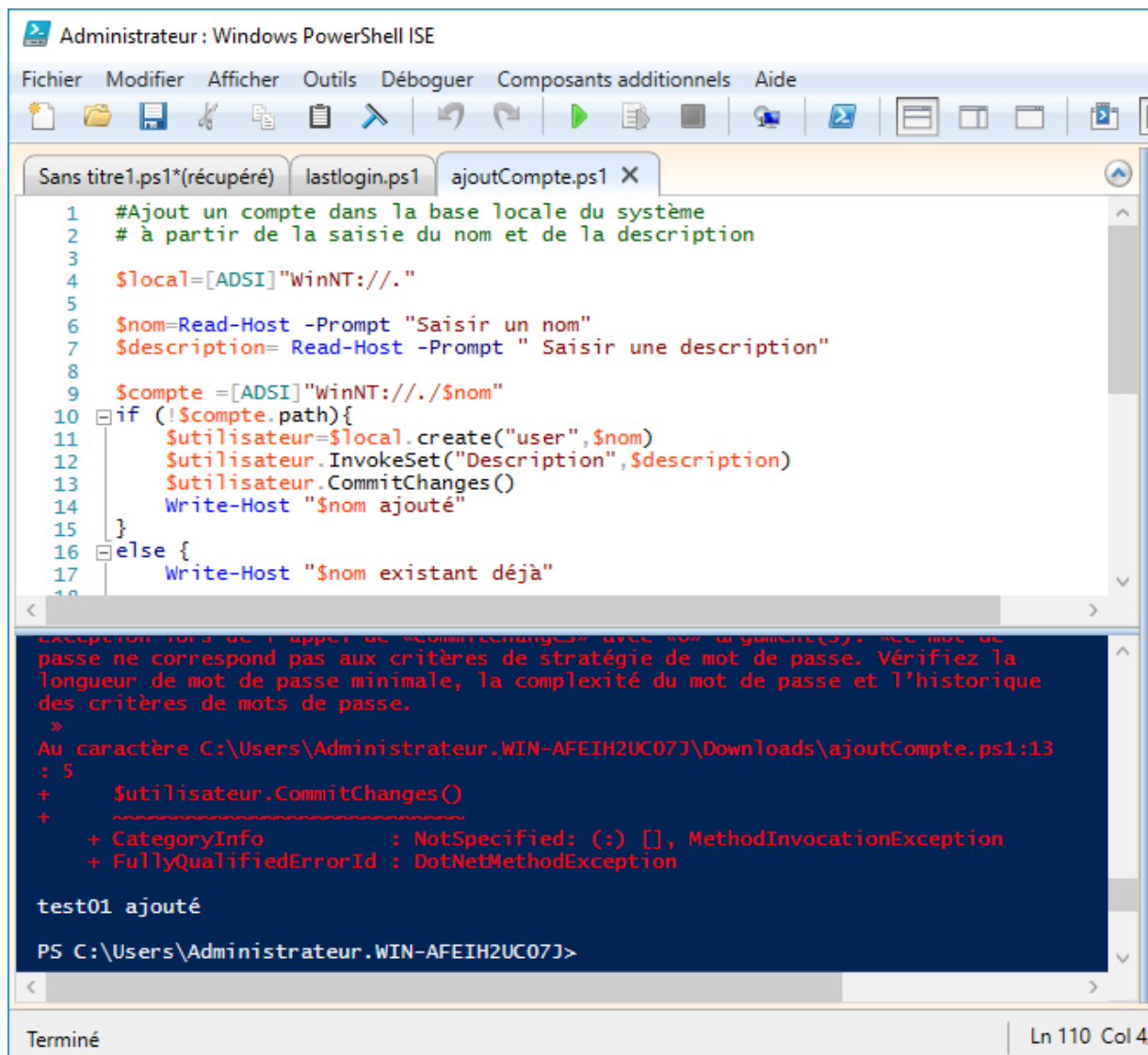


The screenshot shows the Windows PowerShell ISE interface. The title bar reads "Administrateur : Windows PowerShell ISE". The menu bar includes "Fichier", "Modifier", "Afficher", "Outils", "Déboguer", "Composants additionnels", and "Aide". The toolbar contains icons for file operations and execution. The tab bar shows four tabs: "Sans titre1.ps1*(récupéré)", "lastlogin.ps1", "ajoutCompte.ps1", and "Sans titre4.ps1*". The active tab is "ajoutCompte.ps1", which contains the following PowerShell script:

```
1 #Affiche la date de dernière connexion
2 # d'un compte local du système
3 $nom = Read-Host -Prompt " Saisir un nom de compte local"
4 $compte=[ADSI]"WinNT://./$nom"
5 if ( $compte.path){
6     Write-Host $compte.LastLogin
7     Write-Host $compte.name
8     Write-Host $compte.Fullname
9     Write-Host $compte.badpasswordattempts
10 }
11 else{
12     Write-Host "$nom non trouvé"
13 }
```

2. Ajout d'un compte local du système

a.



```
Administrateur : Windows PowerShell ISE
Fichier  Modifier  Afficher  Outils  Déboguer  Composants additionnels  Aide

Sans titre1.ps1*(récupéré)  lastlogin.ps1  ajoutCompte.ps1 X

1  #Ajout un compte dans la base locale du système
2  # à partir de la saisie du nom et de la description
3
4  $local=[ADSI]"WinNT://."
5
6  $nom=Read-Host -Prompt "Saisir un nom"
7  $description= Read-Host -Prompt " Saisir une description"
8
9  $compte =[ADSI]"WinNT://./$nom"
10 if (!$compte.path){
11     $utilisateur=$local.create("user",$nom)
12     $utilisateur.InvokeSet("Description",$description)
13     $utilisateur.CommitChanges()
14     Write-Host "$nom ajouté"
15 }
16 else {
17     Write-Host "$nom existant déjà"
18 }

Exception lors de l'appel de «CommitChanges» avec «0» argument(s). Le mot de
passe ne correspond pas aux critères de stratégie de mot de passe. Vérifiez la
longueur de mot de passe minimale, la complexité du mot de passe et l'historique
des critères de mots de passe.
»
Au caractère C:\Users\Administrateur.WIN-AFEIH2UC07J\Downloads\ajoutCompte.ps1:13
: 5
+      $utilisateur.CommitChanges()
+      ~~~~~
+ CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
+ FullyQualifiedErrorId : DotNetMethodException

test01 ajouté
PS C:\Users\Administrateur.WIN-AFEIH2UC07J>
```

Terminé | Ln 110 Col 4

B.

The screenshot shows the Windows PowerShell ISE interface. The title bar reads "Administrateur : Windows PowerShell ISE". The menu bar includes "Fichier", "Modifier", "Afficher", "Outils", "Déboguer", "Composants additionnels", and "Aide". The toolbar contains icons for file operations and execution. The script editor shows a file named "ajoutCompte.ps1" with the following code:

```
1 #Ajout un compte dans la base locale du système
2 # à partir de la saisie du nom et de la description
3
4 $local=[ADSI]"WinNT://."
5
6 $nom=Read-Host -Prompt "Saisir un nom"
7 $nomcomplet=Read-Host -Prompt " Saisir un nom complet "
8 $description= Read-Host -Prompt " Saisir une description"
9
10 $compte =[ADSI]"WinNT://./$nom"
11 if (!$compte.path){
12     $utilisateur=$local.create("user",$nom)
13     $utilisateur.InvokeSet("FullName",$nomcomplet)
14     $utilisateur.InvokeSet("Description",$description)
15     $utilisateur.CommitChanges()
16     Write-Host "$nom ajouté"
17 }
18 else {
19     Write-Host "$nom existant déjà"
20 }
21 }
```

The console window shows the output of the script execution:

```
+ ~~~~~
+ CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
+ FullyQualifiedErrorId : DotNetMethodException

test02 ajouté

PS C:\Users\Administrateur.WIN-AFEIH2UC07J>
```

Partie 4 : Composition de scripts PS

B. Script pour ajouter un utilisateur dans l'ADDS

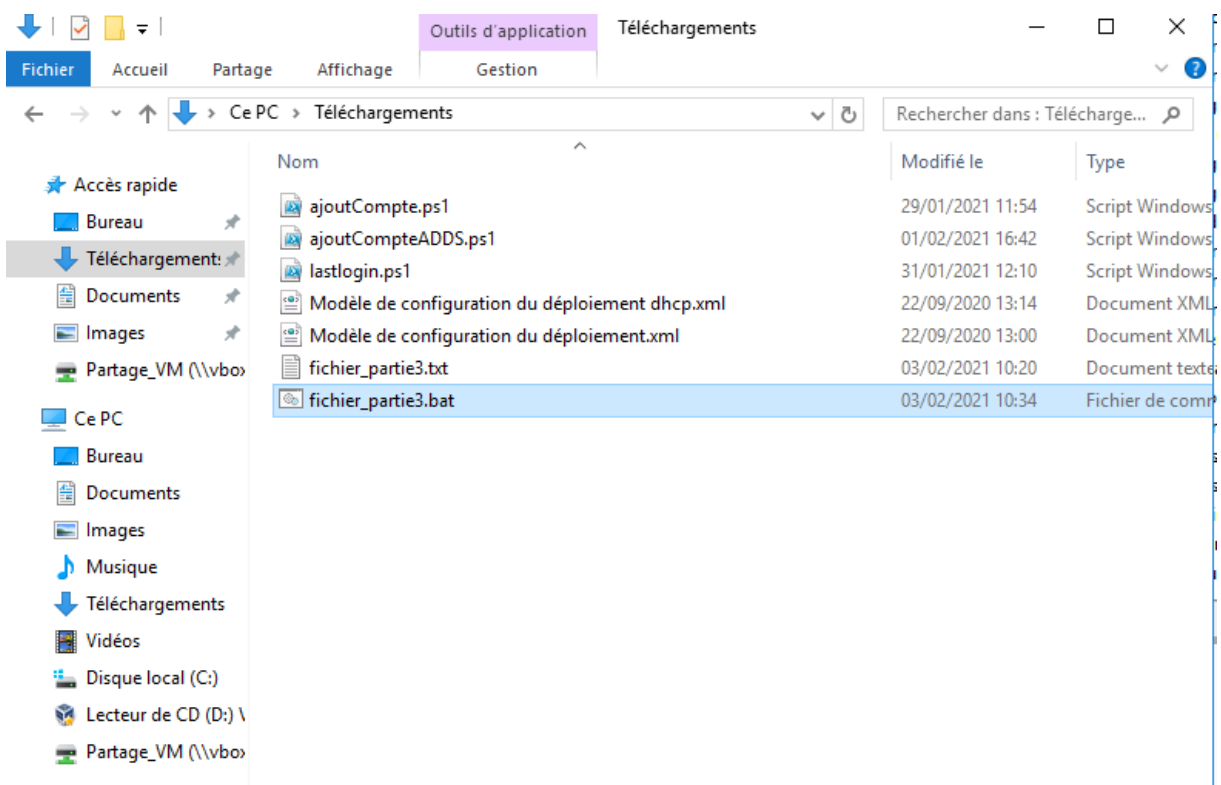
Administrateur : Windows PowerShell ISE

Fichier Modifier Afficher Outils Débugger Composants additionnels Aide

Sans titre1.ps1*(récupéré) lastlogin.ps1 ajoutCompte.ps1 ajoutCompteADDS.ps1 X Sans titre7.ps1*(récupéré) Sans titre8.ps1*

```
1 #Script d'ajout de compte sur l'AD
2 # à partir de la saisie du nom et de la description
3 Import-Module ActiveDirectory
4
5 $path =" OU=Users, DC=moujabber, DC=local"
6
7 $nom=Read-Host -Prompt "Saisir un nom" # Saisir un nom ou description
8 $nomcomplet=Read-Host -Prompt " Saisir un nom complet "
9 $description= Read-Host -Prompt " Saisir une description"
10
11
12 # si le compte existe il affiche un message comme quoi le compte existe déjà
13 if (Get-ADUser -Filter{SamAccountName -eq"$nom"}){
14     Write-Warning "$nom existe déjà dans l'AD"
15 }
16
17
18 else {
19     New-ADUser -Name $nom -SamAccountName $nomcomplet -Enabled $True
20     Write-Host "$nomcomplet a été créé"
21 }
22
23
24
```

Partie 3 : Exécution d'un script Powershell à l'aide d'un fichier Batch



Davia Moujabber
EXSI3P

Le Script a fonctionné en double cliquant et en mettant une pause avant l'exit

```
fichier_partie3.bat1.bat - Bloc-notes
Fichier Edition Format Affichage ?

@ echo off
Powershell Set-ExecutionPolicy RemoteSigned
Powershell ./ajoutCompteADDS.ps1
Powershell Set-ExecutionPolicy Restricted
Pause
Exit
```

```
Administrateur : C:\Windows\system32\cmd.exe
Saisir un nom: Moujabber
Saisir un nom complet : Davia Moujabber
Saisir une description: etudiante
New-ADUser : Le mot de passe ne répond pas aux spécifications de longueur, de complexité ou d'historique du domaine.
Au caractère C:\Users\Administrateur.WIN-AFEIH2UC07J\Downloads\ajoutCompteADDS.ps1:20 : 5
+ New-ADUser -Name $nom -SamAccountName $nomcomplet -Enabled $True
+ ~~~~~
+ CategoryInfo          : InvalidData : (CN=Moujabber,CN...jabber,DC=local:String) [New-ADUser], ADPasswordComplexityException
+ FullyQualifiedErrorId : ActiveDirectoryServer:1325,Microsoft.ActiveDirectory.Management.Commands.NewADUser

Davia Moujabber a été créé
Appuyez sur une touche pour continuer... █
```

Davia Moujabber
EXSI3P