**Les propriétés des utilisateurs dans l'Active Directory**

Ce laboratoire doit être fait individuellement sur le SERVEUR2

**Objectifs**

* Utiliser l’onglet "Éditeur d’attribut"
* Comprendre la différence entre le nom des attributs et les paramètres des cmdlets

Site qui affiche la liste complète des attributs de l'Active Directory avec des explications pour chaque attribut.

**Active Directory Schema**

https://docs.microsoft.com/en-us/windows/win32/adschema/attributes-all

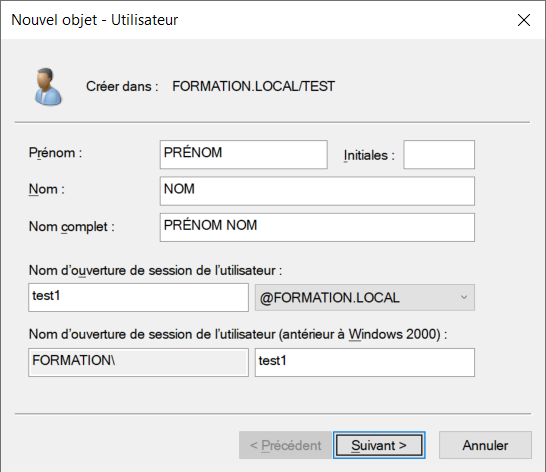
https://docs.microsoft.com/fr-fr/windows/win32/adschema/attributes-all

**Étape 1.1 - Les propriétés de bases d'un utilisateur**

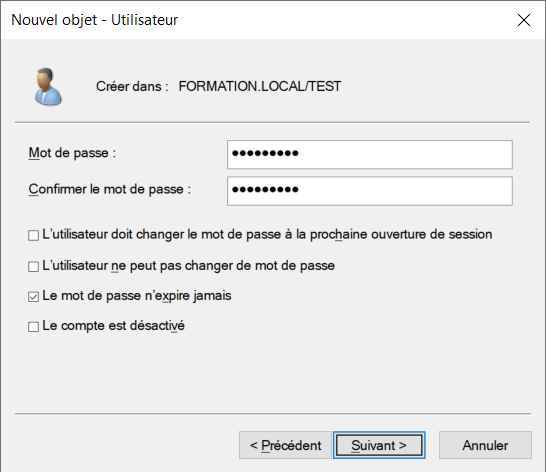
Ouvrir la console UOAD et vérifier que votre affichage est en "Fonctionnalités Avancées"

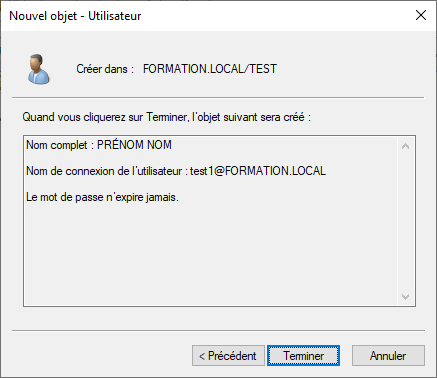
Dans l’unité d’organisation **TEST** qui est directement sous le domaine FORMATION.LOCAL

* Créer l'utilisateur en utilisant les paramètres
  + Prénom: **PRÉNOM**
  + Nom: **NOM**
  + Nom complet: **PRÉNOM NOM**
  + Nom d'ouverture de session de l'utilisateur: **test1@FORMATION.LOCAL**
  + Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000): **FORMATION\test1**
  + Mot de passe: AAAaaa111
  + Le mot de passe n'expire jamais



**Le nom de connexion de l'utilisateur (nom d'ouverture de session) doit être unique dans tout le domaine.**



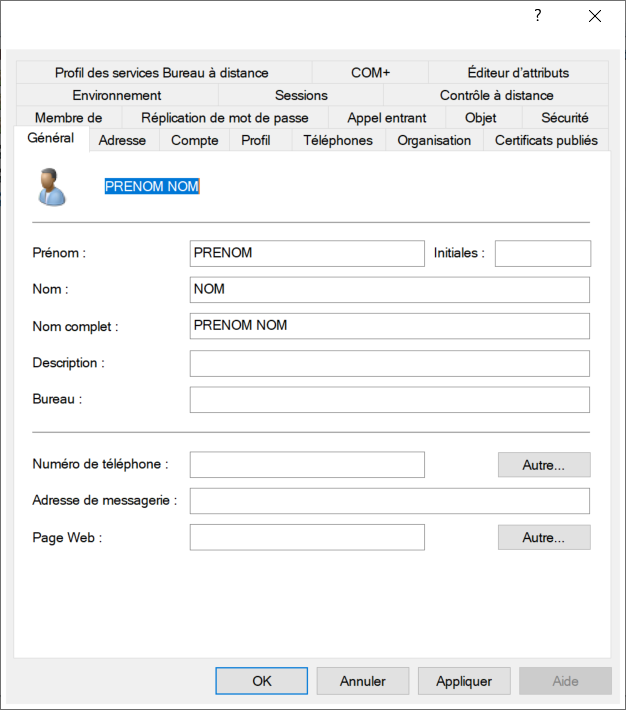


L’utilisateur est créé dans l’unité d’organisation "**TEST**"

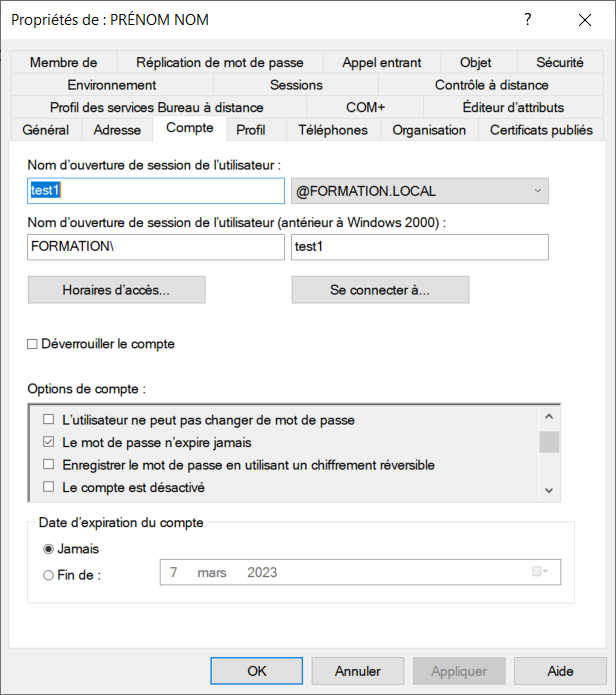


**Vérification de la configuration de l’utilisateur après sa création**

L’onglet "**Général**" après la création de l’utilisateur

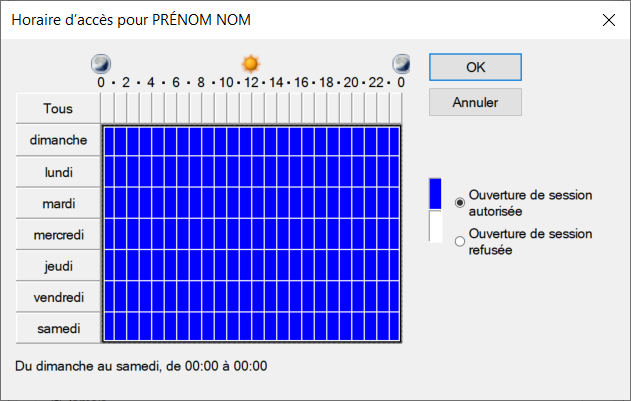


L’onglet "**Compte**" après la création de l’utilisateur

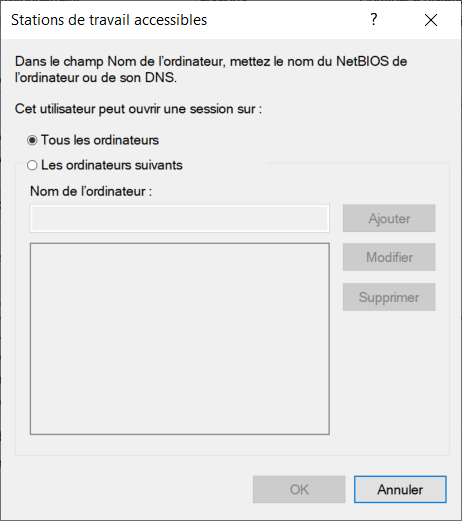


"**Date d'expiration du compte**" Par défaut, le compte d'un utilisateur n'expire jamais.

"**Horaire d'accès…**" Par défaut, un utilisateur n'a pas de restriction pour les heures d'accès.



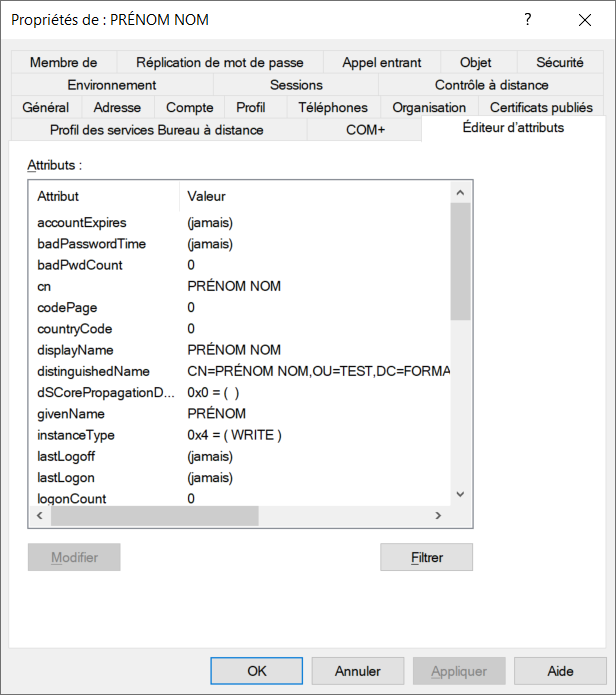
"**Se connecter à…**" Par défaut, un utilisateur peut ouvrir une session sur tous les ordinateurs.



**Vérification des attributs lorsqu’un utilisateur est créé avec la console UOAD.**

En utilisant l’onglet "Éditeur d’attributs", trouvez le nom des attributs qui contiennent vos valeurs.

note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"



**cn PRÉNOM NOM**

**displayName PRÉNOM NOM**

**distinguishedName CN=PRÉNOM NOM,OU=TEST,DC=FORMATION,DC=LOCAL**

**givenName PRÉNOM**

**Lors de la création d'un utilisateur en utilisant l'environnement graphique.**

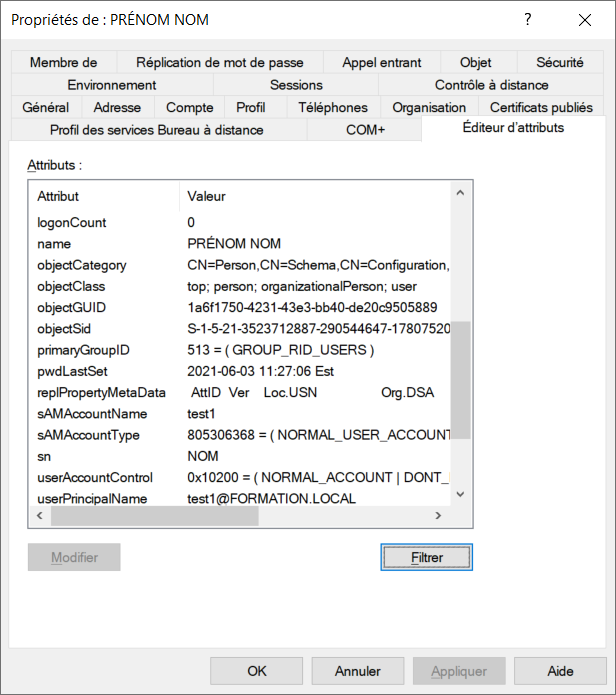
**L'attribut "CN" contient le nom complet de l'utilisateur.**

**L'attribut "CN" est unique dans une unité d'organisation.**

**L'attribut "distinguishedName" est unique dans le domaine.**

**L'attribut "distinguishedName" débute par l'attribut "CN".**

**Pour être capable de créer un utilisateur qui a le même prénom et le même nom qu'un autre utilisateur qui est dans la même unité d'organisation, il faut simplement modifier le nom complet.**



**name PRÉNOM NOM**

**sAMAccountName test1**

**sn NOM**

**userPrincipalName test1@FORMATION.LOCAL**

**L'attribut "name" correspond au nom du compte de l'utilisateur.**

**Lors de la création d'un utilisateur en utilisant l'environnement graphique l'attribut "sAMAccountName" contient le nom d'ouverture de session (antérieur à Windows 2000) de l'utilisateur.**

|  |  |
| --- | --- |
| **Nom du champ**  **lors de la création de l'utilisateur** | **Nom de l'attribut** |
| Prénom | **givenName** |
| Nom | **sn** |
| Nom Complet | **displayName** |
| Nom d'ouverture de session de l'utilisateur | **userPrincipalName** |
| Nom d'ouverture de session de l'utilisateur  (antérieur à Windows 2000) | **sAMAccountName** |

**Vérification des attributs par programmation PowerShell**

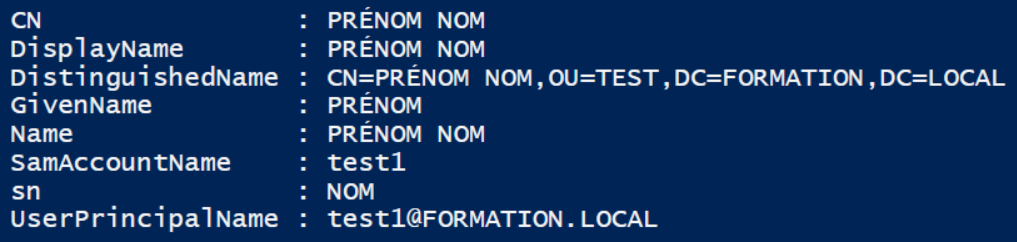
**Voici la commande PowerShell qui affiche la liste des attributs qui correspondent aux propriétés de bases lorsqu’un utilisateur est créé avec la console UOAD.**

**Get-ADuser -Identity test1 `**

**-Properties \* | `**

**Select-Object CN,DisplayName,DistinguishedName,GivenName,**

**Name,SamAccountName,sn,UserPrincipalName**



**Étape 1.2 - Programmation d'un utilisateur avec PowerShell ISE**

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

* Get-ADUser
* New-ADUser
* Remove-ADUser
* Set-ADUser

|  |  |
| --- | --- |
| **Nom du champ**  **lors de la création de l'utilisateur** | **Nom de la propriété dans PowerShell**  **pour le cmdlet New-ADUser** |
| Prénom | **-GivenName** |
| Nom | **-SurName** |
| Nom Complet | **-DisplayName** |
| Nom d'ouverture de session de l'utilisateur | **-UserPrincipalName** |
| Nom d'ouverture de session de l'utilisateur  (antérieur à Windows 2000) | **-SamAccountName** |

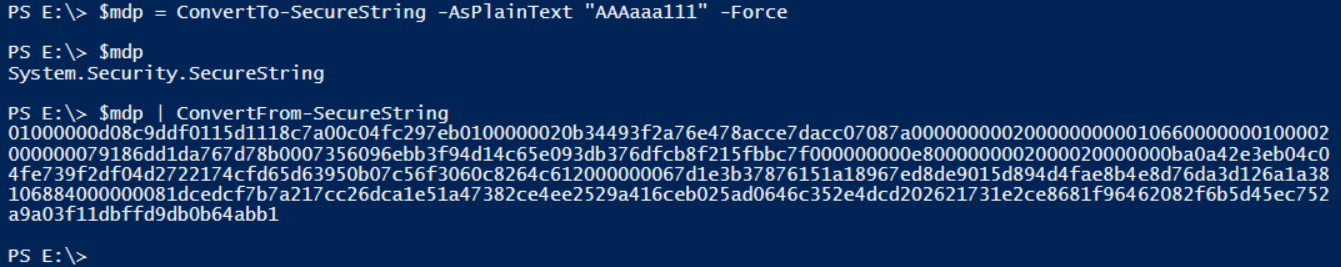
Exemple de création d’un utilisateur avec PowerShell.

**# Supprime l'utilisateur test1 sans aucune confirmation**

**Remove-ADUser -Identity test1 `**

**-Confirm:$false**

**$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force**



**# Ce code permet de recréer l'utilisateur test1 avec les mêmes propriétés qu'à l'étape 1.1.**

**New-ADUser -Name "PRÉNOM NOM" `**

**-SamAccountName test1 `**

**-UserPrincipalName "test1@formation.local" `**

**-Path "OU=TEST,DC=FORMATION,DC=LOCAL" `**

**-GivenName "PRÉNOM" `**

**-Surname "NOM" `**

**-DisplayName "PRÉNOM NOM" `**

**-AccountPassword $mdp `**

**-PasswordNeverExpires $true `**

**-Enabled $true**

L’utilisateur est créé dans l’unité d’organisation "**TEST**"



**Le résultat final semble identique.**

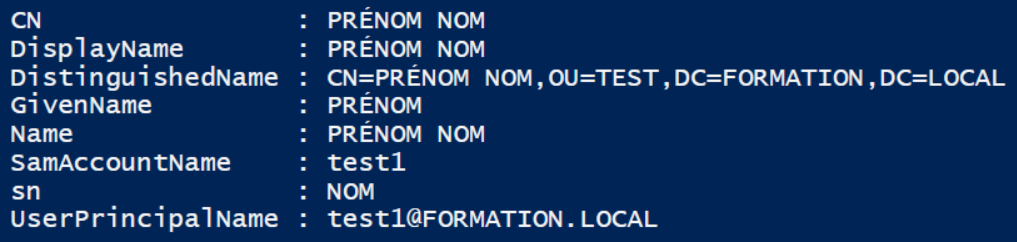
**Vérification des attribututs par programmation PowerShell**

**Get-ADuser -Identity test1 `**

**-Properties \* | `**

**Select-Object CN,DisplayName,DistinguishedName,GivenName,**

**Name,SamAccountName,sn,UserPrincipalName**



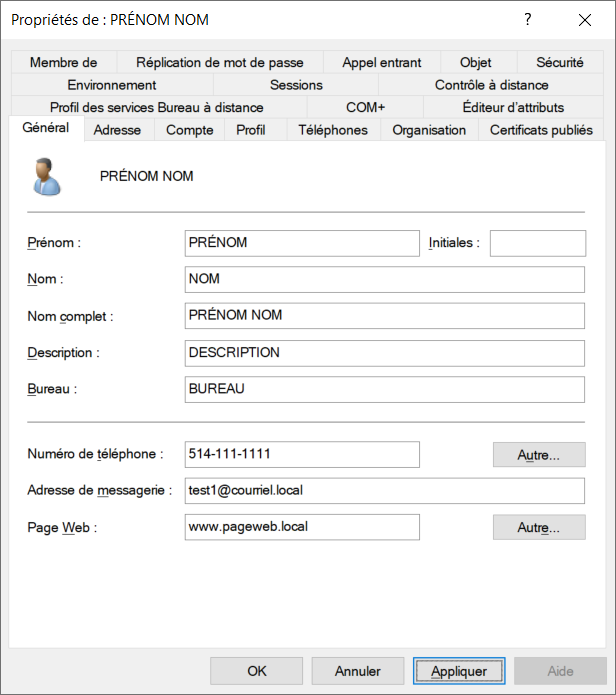
Les attributs sont identiques à ceux dont l’utilisateur a été créé avec la console UOAD.

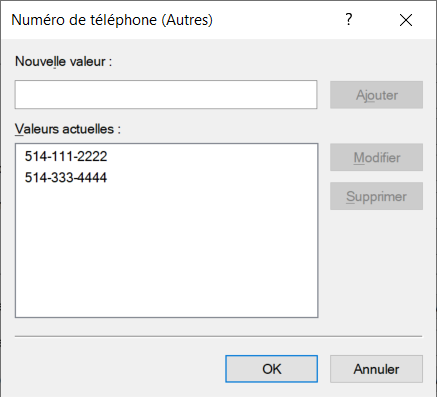
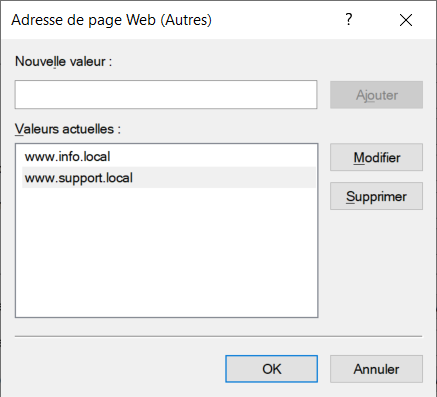
Lors de la création d’un utilisateur dans l’Active Directory avec la commande New-ADUser, vous devez vous assurer de configurer correctement les paramètres **-Name**, **-SamAccountName** et **-UserPrincipalName**.

Le paramètre **-Name** est obligatoire.

**Étape 2.1 - Les attributs d'un utilisateur dans l'onglet "Général"**

Dans l’onglet "Général", remplir les propriétés en y inscrivant les valeurs suivantes:



Ajouter des numéros de téléphones Ajouter des pages web

En utilisant l’onglet "Éditeur d’attributs", trouvez le nom des attributs qui contiennent vos valeurs.

note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Général"** | **Nom de l'attribut** |
| Prénom | **givenName** |
| Nom | **sn** |
| Nom complet | **displayName** |
| Description | **description** |
| Bureau | **physicalDeliveryOfficeName** |
| Numéro de telephone  Autre… | **telephoneNumber**  **otherTelephone** |
| Adresse de messagerie | **mail** |
| Page Web  Autre… | **wWWHomePage**  **url** |

**Étape 2.2 - Programmation d'un utilisateur avec PowerShell ISE**

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

* Get-ADUser
* New-ADUser
* Remove-ADUser
* Set-ADUser

|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Général"** | **Nom de la propriété dans PowerShell**  **pour le cmdlet New-ADUser** |
| Prénom | **-GivenName** |
| Nom | **-SurName** |
| Nom complet | **-DisplayName** |
| Description | **-Description** |
| Bureau | **-Office** |
| Numéro de telephone  Autre… | **-OfficePhone**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherTelephone.** |
| Adresse de messagerie | **-EmailAddress** |
| Page Web  Autre… | **-HomePage**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut url.** |

Exemple de création d’un utilisateur avec PowerShell.

**# Permet de recréer l'utilisateur test1**

**# avec les mêmes propriétés qu'à l'étape 1.1 (rouge sur fond jaune)**

**# et avec les propriétés de l’étape 2.1 (rouge sur fond noir)**

**Remove-ADUser -Identity test1 `**

**-Confirm:$false**

**$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force**

**New-ADUser -Name "PRÉNOM NOM" `**

**-SamAccountName test1 `**

**-UserPrincipalName "test1@formation.local" `**

**-Path "OU=TEST,DC=FORMATION,DC=LOCAL" `**

**-GivenName "PRÉNOM" `**

**-Surname "NOM" `**

**-DisplayName "PRÉNOM NOM" `**

**-Description "DESCRIPTION" `**

**-Office "BUREAU" `**

**-OfficePhone "514-111-1111" `**

**-EmailAddress "test1@courriel.local" `**

**-HomePage "www.pageweb.local" `**

**-OtherAttributes @{'otherTelephone'="514-111-2222","514-333-4444";**

**'url'="www.info.local","www.support.local"} `**

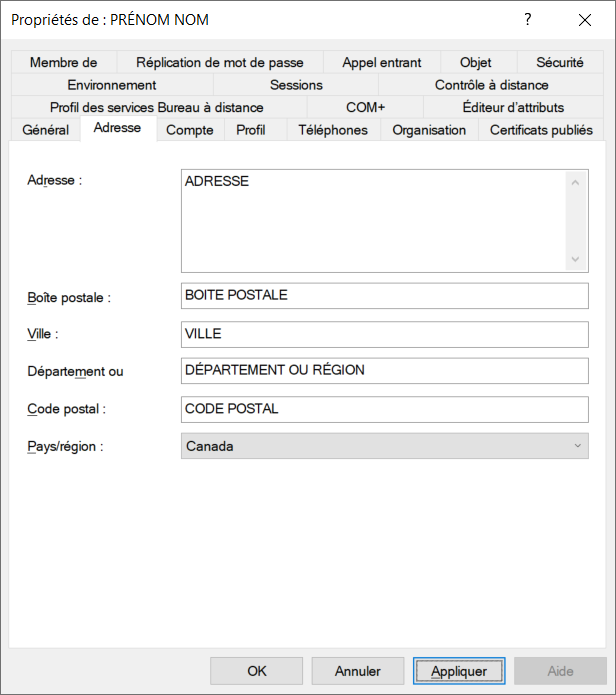
**-AccountPassword $mdp `**

**-PasswordNeverExpires $true `**

**-Enabled $true**

**Étape 3.1 - Les attributs d'un utilisateur dans l'onglet "Adresse"**

Dans l’onglet "Adresse", remplir les propriétés en y inscrivant les valeurs suivantes:



En utilisant l’onglet "Éditeur d’attributs", trouvez le nom des attributs qui contiennent vos valeurs.

note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Adresse"** | **Nom de l'attribut** |
| Adresse | **streetAddress** |
| Boîte postale | **postOfficeBox** |
| Ville | **|** |
| Département ou région | **st** |
| Code postal | **postalCode** |
| Pays/région | **note: il y a trois attributs par pays**  **c=CA**  **co=Canada**  **countryCode=124** |

**Étape 3.2 - Programmation d'un utilisateur avec PowerShell ISE**

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

* Get-ADUser
* New-ADUser
* Remove-ADUser
* Set-ADUser

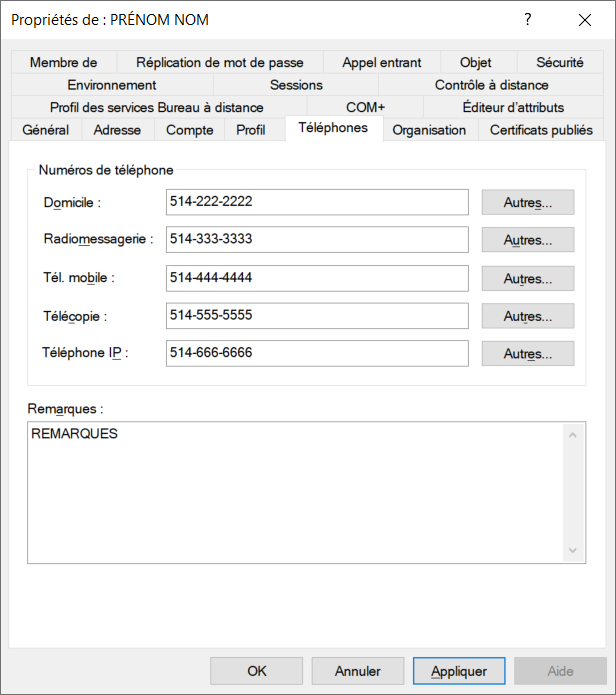
|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Adresse"** | **Nom de la propriété dans PowerShell**  **pour le cmdlet New-ADUser** |
| Adresse | **-StreetAddress** |
| Boîte postale | **-POBox** |
| Ville | **-City** |
| Département ou région | **-State** |
| Code postal | **-PostalCode** |
| Pays/région | **-Country**  **Il est préférable d'utiliser le paramètre -OtherAttributes**  **avec les trois attributs c, co et countryCode.** |

**Exemple pour configurer le pays avec le paramètre -OtherAttributes**

**-OtherAttributes @{'c'="CA";'co'="Canada";'countryCode'=124}**

**Étape 4.1 - Les attributs d'un utilisateur dans l'onglet "Téléphones"**

Dans l’onglet "Téléphones", remplir les propriétés en y inscrivant les valeurs suivantes:



En utilisant l’onglet "Éditeur d’attributs", trouvez le nom des attributs qui contiennent vos valeurs.

note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Téléphones"** | **Nom de l'attribut** |
| Domicile  Autres… | **homePhone**  **otherHomePhone** |
| Radiomessagerie  Autres… | **pager**  **otherPager** |
| Tél. mobile  Autres… | **mobile**  **otherMobile** |
| Télécopie  Autres… | **facsimileTelephoneNumber**  **otherFacsimileTelephoneNumber** |
| Téléphone IP  Autres… | **ipPhone**  **otherIpPhone** |
| Remarques | **info** |

**Étape 4.2 - Programmation d'un utilisateur avec PowerShell ISE**

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

* Get-ADUser
* New-ADUser
* Remove-ADUser
* Set-ADUser

|  |  |
| --- | --- |
| **Nom du champ**  **dans l'onglet "Téléphones"** | **Nom de la propriété dans PowerShell**  **pour le cmdlet New-ADUser** |
| Domicile  Autres… | **-HomePhone**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherHomePhone.** |
| Radiomessagerie  Autres… | **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut Pager.**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherPager.** |
| Tél. mobile  Autres… | **-MobilePhone**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherMobile.** |
| Télécopie  Autres… | **-Fax**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherFacsimileTelephoneNumber.** |
| Téléphone IP  Autres… | **ipPhone**  **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut otherIpPhone.** |
| Remarques | **Il faut utiliser le paramètre -OtherAttributes**  **avec l'attribut info.** |

**Étape 5 - Autres attributs d'un utilisateur**

Trouvez les valeurs des attributs suivants pour l'utilisateur TECH

|  |  |
| --- | --- |
| **Nom de l'attribut** | **Valeur de l'attribut** |
| accountExpires | **(jamais)** |
| badPasswordTime | **(jamais)** |
| badPwdCount | **0** |
| DistinguishedName | **CN=Richard Jean,CN=users,DC=formation,DC=local** |
| lastLogoff | **(jamais)** |
| lastLogon | **2021-06-03 21:36:22 Est** |
| lastLogonTimestamp | **2021-06-01 09:44:58 Est** |
| logonCount | **7** |
| pwdLastSet | **2020-06-01 09:37:46 Est** |
| UserAccountControl | **0x10200 (NORMAL\_ACCOUNT|DONT\_EXPIRE\_PASSWORD)** |
| WhenChanged | **2020-06-01 09:53:13 Est** |
| WhenCreated | **2020-06-01 09:37:45 Est** |

**accountExpires La date d'expiration du compte**

**lastLogoff C'est un attribut que Microsoft n'a jamais utilisé.**

**logonCount Le nombre de fois que l'utilisateur s’est connecté avec succès.**

**accountExpires**

accountExpires est un entier qui représente le nombre de 100 nano secondes depuis le 1601/01/01 (UTC).

Si la valeur de **accountExpires** correspond à **0** ou **9223372036854775807** alors le compte n'expire jamais.

**9223372036854775807** correspond à la valeur hexadécimale **0x7FFFFFFFFFFFFFFF**

**[int64]::MaxValue = 9223372036854775807**

**lastLogon**

Cet attribut n'est jamais répliqué, ce qui veut dire que sa valeur est spécifique à chaque contrôleur de domaine.

LastLogon est un entier qui représente le nombre de 100 nanosecondes depuis le 1601/01/01 (UTC).

**lastLogonTimeStamp**

Cet attribut est répliqué sur chaque contrôleur de domaine mais seulement dans un intervalle de 9 à 14 jours.

LastLogonTimeStamp est un entier qui représente le nombre de 100 nano secondes depuis le 1601/01/01 (UTC).

**Exemple avec LastLogon**

**Get-ADUser -Identity TECH `**

**-Properties accountExpires,LastLogon,LastLogonDate,LastLogonTimeStamp**

**accountExpires : 9223372036854775807**

**DistinguishedName : CN=Richard Jean,CN=Users,DC=FORMATION,DC=LOCAL**

**Enabled : True**

**GivenName : Richard**

**LastLogon : 132672441821805799**

**LastLogonDate : 1 juin 2021 09:44:58**

**LastLogonTimeStamp : 132670286986966976**

**Name : Richard Jean**

**ObjectClass : user**

**ObjectGUID : 58aae375-36dd-432e-b2c6-20395ad28cb1**

**SamAccountName : TECH**

**SID : S-1-5-21-3523712887-290544647-1780752054-1104**

**Surname : Jean**

**UserPrincipalName : TECH@FORMATION.LOCAL**

**# Avec PowerShell, permet de convertir la valeur qui correspond au LastLogon**

**[DateTime]::FromFileTime(132672441821805799)**

**3 juin 2021 21:36:22**

**# Avec PowerShell, permet de convertir la valeur qui correspond au LastLogonTimeStamp**

**[DateTime]::FromFileTime(132670286986966976)**

**1 juin 2021 09:44:58**

LastLogonDate correspond à la date de LastLogonTimeStamp.

**Get-ADUser -Identity TECH `**

**-Properties \* | `**

**Select-Object Name,accountExpires,LastLogonTimeStamp,LastLogon,**

**@{label='accountExpires - DATE';**

**expression={if ( ($PSItem.accountExpires -eq 0) -or**

**($PSItem.accountExpires -eq 9223372036854775807) )**

**{**

**Write-Output "JAMAIS"**

**}**

**else**

**{**

**[DateTime]::FromFileTime($PSItem.accountExpires)**

**}**

**}**

**},**

**@{label='LastLogonTimeStamp - DATE';**

**expression={[DateTime]::FromFileTime($PSItem.LastLogonTimeStamp)}**

**},**

**@{label='LastLogon - DATE';**

**expression={[DateTime]::FromFileTime($PSItem.LastLogon)}**

**}**

**ANNEXE**

**L'attribut UserAccountControl**

Ce tableau présente les différentes valeurs de l'attribut **UserAccountControl**.

Les valeurs sont cumulatives.

|  |  |  |
| --- | --- | --- |
| Nom de la constante  pour la valeur | Valeur  hexadécimale | Explication de la valeur |
| **SCRIPT** | **0x1** | Exécution du script d'ouverture de session |
| **ACCOUNTDISABLE** | **0x2** | Désactivation du compte utilisateur |
| **HOMEDIR\_REQUIRED** | **0x8** | Dossier de base requis |
| **LOCKOUT** | **0x10** | Aucun mot de passe n'est requis. |
| **PASSWD\_NOTREQD** | **0x20** | Vous pouvez lire cette valeur, mais vous ne pouvez pas la définir directement. |
| **PASSWD\_CANT\_CHANGE** | **0x40** | Impossible de modifier le mot de passe. |
| **ENCRYPTED\_TEXT\_PWD\_ALLOWED** | **0x80** | L'utilisateur peut envoyer un message crypté. |
| **TEMP\_DUPLICATE\_ACCOUNT** | **0x100** | Compte pour les utilisateurs dont le compte principal se trouve dans un autre domaine. Ce compte fournit l'accès à ce domaine, mais pas à tous les domaines qui ont des relations d'approbation avec ce domaine. Il est parfois appelé compte utilisateur local. |
| **NORMAL\_ACCOUNT** | **0x200** | Type de compte par défaut représentant un utilisateur |
| **INTERDOMAIN\_TRUST\_ACCOUNT** | **0x800** | Autorisation d'approuver un compte pour un domaine du système qui a des relations d'approbation avec d'autres domaines. |
| **WORKSTATION\_TRUST\_ACCOUNT** | **0x1000** | Compte d'ordinateur |
| **SERVER\_TRUST\_ACCOUNT** | **0x2000** | Compte d'ordinateur d'un contrôleur de domaine membre de ce domaine. |
| **DONT\_EXPIRE\_PASSWORD** | **0x10000** | Représente le mot de passe, qui ne doit jamais expirer pour le compte. |
| **MNS\_LOGON\_ACCOUNT** | **0x20000** | Compte d'ouverture de session de jeu de nœuds majoritaire. |
| **SMARTCARD\_REQUIRED** | **0x40000** | Cette valeur force l'utilisateur à ouvrir une session avec une carte à puce. |
| **TRUSTED\_FOR\_DELEGATION** | **0x80000** | Le compte de service (compte d'utilisateur ou d'ordinateur) est approuvé pour la délégation Kerberos. N'importe lequel de ces services peut prendre l'identité d'un client demandant le service. |
| **NOT\_DELEGATED** | **0x100000** | Le contexte de sécurité de l'utilisateur n'est pas délégué à un service même si le compte de service est approuvé pour la délégation Kerberos. |
| **PASSWORD\_EXPIRED** | **0x800000** | Le mot de passe de l'utilisateur a expiré. |

**# Exemple pour trouver les utilisateurs avec "NORMAL\_ACCOUNT" et "DONT\_EXPIRE\_PASSWORD"**

**# -band est un "ET binaire"**

Clear-Host

$n = 0x10200

Get-ADuser -Filter {userAccountControl -band $n} `

-Properties \* `

| Sort-Object canonicalname `

| Format-Table CanonicalName,Name,userAccountControl `

-AutoSize

**# AMÉLIORATION de la commande précédente pour afficher la valeur**

**# du paramètre userAccountControl en hexadécimale sur 8 caractères (32 bits)**

**# et de l'aligner à droite.**

Clear-Host

$n = 0x10200

Get-ADuser -Filter {userAccountControl -band $n} `

-Properties \* `

| Sort-Object canonicalname `

| Format-Table CanonicalName, `

Name, `

@{Label="AccountControl (hex)"; `

Expression={'{0:x8}' -f ($PSItem.userAccountControl)}; `

Align="Right"} `

-AutoSize