Utilisateurs et groupes

Récupération d'information sur les utilisateurs et groupes locaux

Get-LocalUser

Une commande simple comme la suivante nous permettrait de vérifier quel utilisateur a un profil local sur une machine ainsi que sa dernière date de connexion. Ce genre d'information est souvent utilisé pour nettoyer les profils locaux d'utilisateurs Active Directory inactifs.

```
PS C:\WINDOWS\system32> Get-LocalUser | Select-Object -Property Name,Enabled,LastLogon


Name               Enabled LastLogon
----               ------- ---------
Administrateur       False
DefaultAccount       False
Info                  True 2025-01-19 19:36:18
Invité               False
WDAGUtilityAccount   False
```

Les informations récupérables par cette commande sont toutefois limitées comme le démontre la commande suivante.

```
PS C:\WINDOWS\system32> Get-LocalUser | Where-Object Name -like Info | Select-Object -Property
*


AccountExpires         :
Description            :
Enabled                : True
FullName               :
PasswordChangeableDate :
PasswordExpires        :
UserMayChangePassword  : True
PasswordRequired       : False
PasswordLastSet        :
LastLogon              : 2025-01-19 19:36:18
```

```
Name                  : Info
SID                   : S-1-5-21-2716239665-1816042345-2264835360-1001
PrincipalSource       : Local
ObjectClass           : Utilisateur
```

Les dates d'expiration et de changement de mot de passe sont habituellement utilisées pour soit alerter les utilisateurs de l'expiration imminente de leur mot de passe ou encore pour effectuer des recommendations en terme de pratique de cybersécurité.

Get-Process

Quoi que Get-Process soit en lien avec l'administration du système plutôt que de ses utilisateurs, elle peut être utilisée pour déterminer quel utilisateur a une session active sur un poste de travail en regardant quelles instances d'un processus commun à tous tel que explorer.exe sont en exécution!

```
PS C:\WINDOWS\system32> Get-Process -IncludeUserName | Where-Object Name -like explorer* |
Select-Object -Property UserName,StartTime

UserName                       StartTime
--------                       ---------
Vidange-Windose\Administrateur 2025-01-19 19:59:14
Vidange-Windose\Info           2025-01-19 18:08:00
```

Get-LocalGroup

La commande Get-LocalGroup nous permet d'interroger le système au sujet des groupes locaux mais tout comme Get-LocalUsers, la commande n'est pas très bavarde.

```
PS C:\WINDOWS\system32> Get-LocalGroup -Name "Administrateurs" | Select-Object -Property *

Description     : Les membres du groupe Administrateurs disposent d’un accès complet et
illimité à l’ordinateur et au
                  domaine
Name            : Administrateurs
SID             : S-1-5-32-544
PrincipalSource : Local
ObjectClass     : Groupe
```

Get-LocalGroupMember

C'est plutôt la commande Get-LocalGroupMember qui nous permet de voir l'association entre les utilisateurs et les groupes.

```
PS C:\WINDOWS\system32> Get-LocalGroupMember -Name "Administrateurs" | Select-Object -Property
*

Name                           SID                                               PrincipalSource
ObjectClass
----                           ---                                               ---------------
-----------
Vidange-Windose\Administrateur S-1-5-21-2716239665-1816042345-2264835360-500           Local
Utilisateur
Vidange-Windose\Info           S-1-5-21-2716239665-1816042345-2264835360-1001          Local
Utilisateur
```

Pour vérifier la liste des groupes auxquels un utilisateur appartient, il faut boucler dans tous les utilisateurs et tous les groupes avec un script tel que celui-ci.

```
PS C:\WINDOWS\system32> Get-LocalUser | ForEach-Object {
>>   $user = $_
>>   return [PSCustomObject]@{
>>     "User" = $user.Name
>>     "Groups" = Get-LocalGroup | Where-Object {
>>       $user.SID -in ($_ | Get-LocalGroupMember | Select-Object -ExpandProperty "SID") |
Select-Object -Property "Name"
>>     }
>>   }
>> }

User               Groups
----               ------
Administrateur     {Administrateurs, Administrateurs Hyper-V, Duplicateurs, IIS_IUSRS...}
DefaultAccount     {Administrateurs, Administrateurs Hyper-V, Duplicateurs, IIS_IUSRS...}
Info               {Administrateurs, Administrateurs Hyper-V, Duplicateurs, IIS_IUSRS...}
Invité             {Administrateurs, Administrateurs Hyper-V, Duplicateurs, IIS_IUSRS...}
WDAGUtilityAccount {Administrateurs, Administrateurs Hyper-V, Duplicateurs, IIS_IUSRS...}
```

Si un utilisateur était mentionné, la première boucle ne serait pas nécessaire.

Gestion des utilisateurs et des groupes locaux

Puisque les objets sont simples, leur création et leur gestion l'est aussi.

New-LocalUser

En se fiant à la commande Get-LocalUser mentionnée plus haut, on peut déterminer les paramètres à saisir pour créer un nouvel utilisateur à l'aide de la commande New-LocalUser.

Si un mot de passe doit être précisé, il doit être mentionné sous forme d'une "SecureString". Pour faciliter la structure de ces commandes, on peut stocker le mot de passe dans une variable plutôt que de faire un "one-liner" mais il est aussi possible de tout effectuer dans la même commande.

```
PS C:\WINDOWS\system32> Get-LocalUser

Name               Enabled Description
----               ------- -----------
Administrateur     True    Compte d’utilisateur d’administration
DefaultAccount     False   Compte utilisateur géré par le système.
Info               True
Invité             False   Compte d’utilisateur invité
WDAGUtilityAccount False   Compte d’utilisateur géré et utilisé par le système pour les
scénarios Windows Defender Applic...

PS C:\WINDOWS\system32> $pwd = ConvertTo-SecureString -String "bobettes" -AsPlainText -Force
PS C:\WINDOWS\system32> New-LocalUser -Name "bob" -FullName "Bobby" -Password $pwd

Name Enabled Description
---- ------- -----------
bob  True

PS C:\WINDOWS\system32> Get-LocalUser

Name               Enabled Description
----               ------- -----------
Administrateur     True    Compte d’utilisateur d’administration
bob                True
DefaultAccount     False   Compte utilisateur géré par le système.
Info               True
Invité             False   Compte d’utilisateur invité
WDAGUtilityAccount False   Compte d’utilisateur géré et utilisé par le système pour les
scénarios Windows Defender Applic...
```

Set-LocalUser

La commande "Set-LocalUser" fonctionne de la même façon que la commande "New-LocalUser" à l'exception que le paramètre "-Name" qui sera mentionné représentera le compte à modifier.

```
PS C:\WINDOWS\system32> Get-LocalUser bob

Name Enabled Description
---- ------- -----------
bob  True


PS C:\WINDOWS\system32> Set-LocalUser -Name bob -Description "Ta mère"
PS C:\WINDOWS\system32> Get-LocalUser bob

Name Enabled Description
---- ------- -----------
bob  True    Ta mère
```

New-LocalGroup

Encore une fois, en se fiant à Get-LocalGroup, on peut déterminer l'ensemble des paramètres qui seront à mentionner à la création d'un groupe.

```
PS C:\WINDOWS\system32> Get-LocalGroup

Name                                Description
----                                -----------
Administrateurs                     Les membres du groupe Administrateurs disposent
d’un accès complet et illimit...
Utilisateurs                        Les utilisateurs ne peuvent pas effectuer de
modifications accidentelles ou i...
Utilisateurs de gestion à distance  Les membres de ce groupe ont accès aux ressources
WMI via des protocoles de g...
Utilisateurs du Bureau à distance   Les membres de ce groupe disposent des droits
nécessaires pour ouvrir une ses...
Utilisateurs OpenSSH                Les membres de ce groupe peuvent se connecter à
cet ordinateur à l’aide de SSH.


PS C:\WINDOWS\system32> New-LocalGroup -Name "Yo Les Jeunes" -Description "Le club secret à
Bob"

Name          Description
----          -----------
```

```
Yo Les Jeunes Le club secret à Bob



PS C:\WINDOWS\system32> Get-LocalGroup

Name                                  Description
----                                  -----------
Yo Les Jeunes                         Le club secret à Bob
Administrateurs                       Les membres du groupe Administrateurs disposent
d’un accès complet et illimit...
Utilisateurs                          Les utilisateurs ne peuvent pas effectuer de
modifications accidentelles ou i...
Utilisateurs de gestion à distance    Les membres de ce groupe ont accès aux ressources
WMI via des protocoles de g...
Utilisateurs du Bureau à distance     Les membres de ce groupe disposent des droits
nécessaires pour ouvrir une ses...
Utilisateurs OpenSSH                  Les membres de ce groupe peuvent se connecter à
cet ordinateur à l’aide de SSH.
```

Set-LocalGroup

Tout comme pour Set-LocalUser, Set-LocalGroup utilise le paramètre "Name" pour déterminer le groupe à modifier.

```
PS C:\WINDOWS\system32> Get-LocalGroup -Name "Yo Les Jeunes"

Name          Description
----          -----------
Yo Les Jeunes Le club secret à Bob



PS C:\WINDOWS\system32> Set-LocalGroup -Name "Yo Les Jeunes" -Description "Le club plus tant
secret à Bob"
PS C:\WINDOWS\system32> Get-LocalGroup -Name "Yo Les Jeunes"

Name          Description
----          -----------
Yo Les Jeunes Le club plus tant secret à Bob
```

Add-LocalGroupMember

Pour attribuer des utilisateurs ou des groupes à un groupe (oui, on peut imbriquer des groupes), la commande Add-LocalGroupMember est aussi simple que de mentionner le nom du membre qui peut être le nom d'un utilisateur ou d'un groupe et de mentionner le groupe auquel assigner le membre.

On assigne ici l'utilisateur "bob" à son groupe

```
PS C:\WINDOWS\system32> Get-LocalGroupMember "Yo Les Jeunes"
PS C:\WINDOWS\system32> Add-LocalGroupMember -Group "Yo Les Jeunes" -Member "bob"
PS C:\WINDOWS\system32> Get-LocalGroupMember "Yo Les Jeunes"

ObjectClass Name                PrincipalSource
----------- ----                ---------------
Utilisateur Vidange-Windose\bob Local
```

Le principe est le même pour imbriquer un groupe dans un autre

```
PS C:\WINDOWS\system32> Add-LocalGroupMember -Group "Yo Les Jeunes" -Member "Invités"
PS C:\WINDOWS\system32> Get-LocalGroupMember "Yo Les Jeunes"

ObjectClass Name                PrincipalSource
----------- ----                ---------------
Groupe      BUILTIN\Invités     Local
Utilisateur Vidange-Windose\bob Local
```

Remove-LocalGroupMember

Le principe ici est le même que pour l'ajout!

```
PS C:\WINDOWS\system32> Get-LocalGroupMember "Yo Les Jeunes"

ObjectClass Name                PrincipalSource
----------- ----                ---------------
Groupe      BUILTIN\Invités     Local
Utilisateur Vidange-Windose\bob Local



PS C:\WINDOWS\system32> Remove-LocalGroupMember -Group "Yo Les Jeunes" -Member "bob"
PS C:\WINDOWS\system32> Get-LocalGroupMember "Yo Les Jeunes"

ObjectClass Name            PrincipalSource
----------- ----            ---------------
```

```
Groupe      BUILTIN\Invités Local
```

Revision #10
Created 2025-01-19 20:05:26 UTC by Alexandre Arsenault-Jetté
Updated 2025-01-21 03:35:59 UTC by Alexandre Arsenault-Jetté