Leçon 3 : Administration locale

3.1 Introduction

Dans cette leçon nous allons aborder les éléments d'administration locale. Ainsi, nous allons étudier comment nous pouvons administrer un serveur *autonome* ne servant pas à authentifier des machines et des utilisateurs sur le réseau. Bon nombre des éléments vus ici sont également applicables aux versions desktop *professionnelles* des systèmes d'exploitation Microsoft comme par exemple *Windows 10/8.1/7 Professionnel*.

Les éléments abordés dans cette leçon sont :

- Les utilisateurs, les groupes locaux et les profiles
- Le système de fichiers, les quotas et la sécurité NTFS
- La stratégie locale
- Divers : la console MMC, l'observateur d'événements, les tâches planifiées

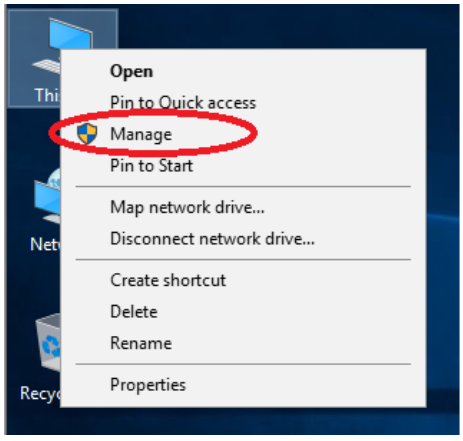
Référence bibliographique

Ce chapitre se base sur la référence bibliographique suivante :

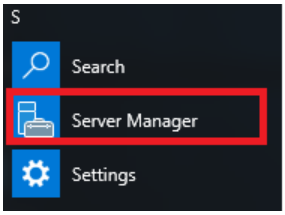
[1] C. Zacker, *Exam Ref 70-740 Installation, Storage and Compute with Windows Server 2016*, Microsoft Press, 19 January 2017.

Nous utiliserons beaucoup l'outil **Server Manager**. On peut procéder de 2 manières :

- Cliquer sur **Démarrer** puis faire un **clic-droit** sur **Ordinateur** (cette méthode est compatible avec Windows 10 et Windows 8.1/7) et choisir l'option **Manage** :



- Cliquer sur l'icône qui se trouve dans le menu démarrer.



L'interface de gestion du serveur démarre. Cette interface permet de configurer le serveur et ses ressources locales.

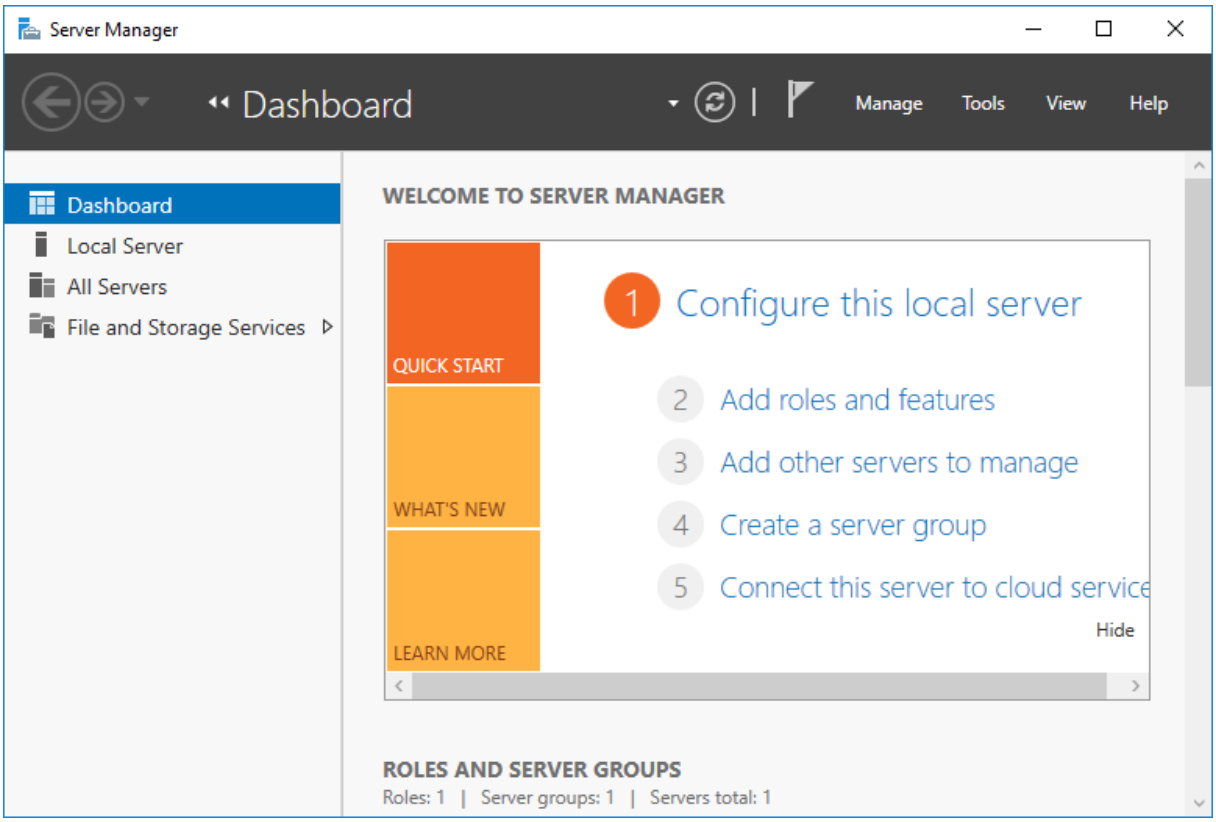


Figure 3.1 : Interface de gestion du serveur

Sur la figure 3.1, nous pouvons voir l'interface de gestion du serveur qui se compose d'un menu à gauche aves les options **Dashboard, Local Serveur, All Servers, …** et un menu en haut à droite composé des options **Manage, Tools, View et Help**.

Le menu **Manage** permet, notamment, d'**Ajouter des rôles** (*Add Roles and Features*) :

- **Roles** : permet d'ajouter des *rôles* au serveur courant. Les rôles principaux sont :
  - Des services Active Directory (notamment, AD Certificate Service, AD Domain Service, AD Federation Services, AD Lightweight Directory Services, AD Right Management Services)
  - DHCP Server (qui permet au serveur de distribuer des adresses IP sur le réseau),
  - DNS Server (qui permet d'installer un serveur de nom et de gérer des noms DNS),
  - Fax server (qui autorise le serveur à envoyer et recevoir des fax),
  - *Hyper-V* (gestion de machines virtuelles),
  - *Web Server (IIS)* (qui permet d'héberger des sites web et déployer des applications .NET),
  - ...
- **Features** : permet d'ajouter une fonctionnalité au serveur autonome. La distinction entre les fonctionnalités et les rôles est relativement difficile à faire étant donné qu'on y retrouve par exemple le *SMTP Serveur* (envoi de courriers).

Parmis les outils importants dans l'administration locale d'un serveur, il y a l'outil de *Gestion de l'ordinateur*, accessible depuis le **Server Manager > menu Tools > Computer Management**. On y trouve :

- **System Tools**. On y trouve les outils suivants :
  - Le *Task Scheduler* qui permet de démarrer automatiquement des jobs sur le serveur
  - L'*Event Viewer* qui reprend les fichiers journaux du système. Ainsi, les erreurs consignées par le serveur ou les applications peuvent se trouver ici.
  - Les *Shared Folders* qui reprennent les partages actifs (ie. les dossiers partagés) sur le serveur. Intéressant pour visualiser tous les partages en cours.
  - Les *Local Users and Groups* qui permet d'ajouter et de gérer les utilisateurs et les groupes locaux au serveur
  - L'outil *Performance* qui consigne les éléments des rapports et journaux concernant les performances du système.
  - Le *Device Manager* reprenant la configuration matérielle du serveur.
- **Storage**. On y trouve les éléments suivants :
  - *Windows Server Backup* qui permet de réaliser des backups du système
  - *Disk Management* qui permet de gérer les disques, partitions et systèmes de fichiers et les lettres attribuées à chaques lecteur.
- **Services and Applications**. On y trouve les éléments suivants :
  - L'option *Routing and Remote Access* permet d'ajouter des options NAT ou encore configurer un serveur VPN sur Windows Serveur.
  - L'option *Services* qui permettent de démarrer des programmes serveurs et de les arrêter.

3.2 Les utilisateurs et groupes locaux

Le serveur **dispose d'une base de données locale des utilisateurs**. Il est possible d'installer une base de données **globale** si l'on installe Active Directory *Directory Service*. Une base de données globale sert à identifier les utilisateurs au travers d'un réseau tandis que la base de données des utilisateurs locale est uniquement utilisée par le serveur courant.

Gérer des utilisateurs et la sécurité qui leur est attachée est une des occupations de l'administrateur système. En effet, gérer efficacement les ressources partagées entre tous n'est pas une tâche toujours facile.

Pour **visualiser les utilisateurs locaux** configurés sur le système, il suffit de démarrer le *Server Manager* et aller dans **Tools > Computer Management > Local Users and Groups > Users**[8]. On peut ainsi voir que seul 3 utilisateurs sont présents : *Administrator, DefaultAccount* et *Guest*. La flèche vers le bas, probablement présente sur les comptes *DefaultAccount* et *Guest* mentionne que ce compte **est désactivé.** Nous reparlerons du compte *Guest* plus tard, qui dispose de droits limités.

Pour **visualiser les groupes de sécurité locaux** configurés sur le système, il faut aller dans **Computer Management > Local Users and Groups > Groups**. Il y a déjà bon nombre de groupes de sécurité présents. Parmi ceux-ci, pointons :

- Administrators : Tous les utilisateurs membres de ce groupe sont administrateurs du serveur.
- Guests : Tous les utilisateurs membres de ce groupe sont *Invités*.

[8] Windows donne l'impression d'identifier les utilisateurs sur base de leur nom d'utilisateur. Ce n'est pas la réalité, les utilisateurs sont identifiés sur base de leur SID. Le SID d'un utilisateur est composé du SID de la machine, complété par l'utilisateur. Ainsi, ce dernier est unique.

- Users : Tous les utilisateurs membres de ce groupe sont des *utilisateurs standards* du serveur
- Power Users : Tous les utilisateurs membre de ce groupe sont des *utilisateurs avancés* (ils ont plus de droit que les utilisateurs standard). **Ne devrait plus être utilisé !**

Pour **créer un utilisateur**, il suffit de faire un clic-droit et choisir l'option **New User** (également présente par le menu *More Actions* à droite). Il faut mentionner les données suivantes :

- User name : c'est son login, la façon dont celui-ci pourra se connecter au système
- Full name : c'est le nom qui apparaîtra dans l'interface
- Description : un texte facultatif (p.ex. moment de la création, ...)
- Password : le mot de passe attribué à cet utilisateur
- Confirm password : afin de vérifier si le mot de passe est correct
- Certaines options :
    - *User must change password at next login :* impose que l'utilisateur modifie le mot de passe attribué par l'administrateur
    - *User cannot change password :* interdit les modifications de mot de passe par l'utilisateur
    - *Password never expires :* certaines politiques de sécurité imposent des changements réguliers du mot de passe.
    - *Account is disabled :* empêche toute connexion via ce mot de passe

Par défaut, quand un utilisateur est créé, il est placé dans le groupe Users. Il est donc considéré comme *un utilisateur standard* du système. Si cet utilisateur doit administrer le serveur, il est nécessaire qu'il soit *membre* du groupe de sécurité *Administrators*.

Si l'on **examine les propriétés d'un utilisateur** (clic-droit sur l'utilisateur concerné puis **propriétés**), on peut voir toutes les informations mémorisées pour un utilisateur.

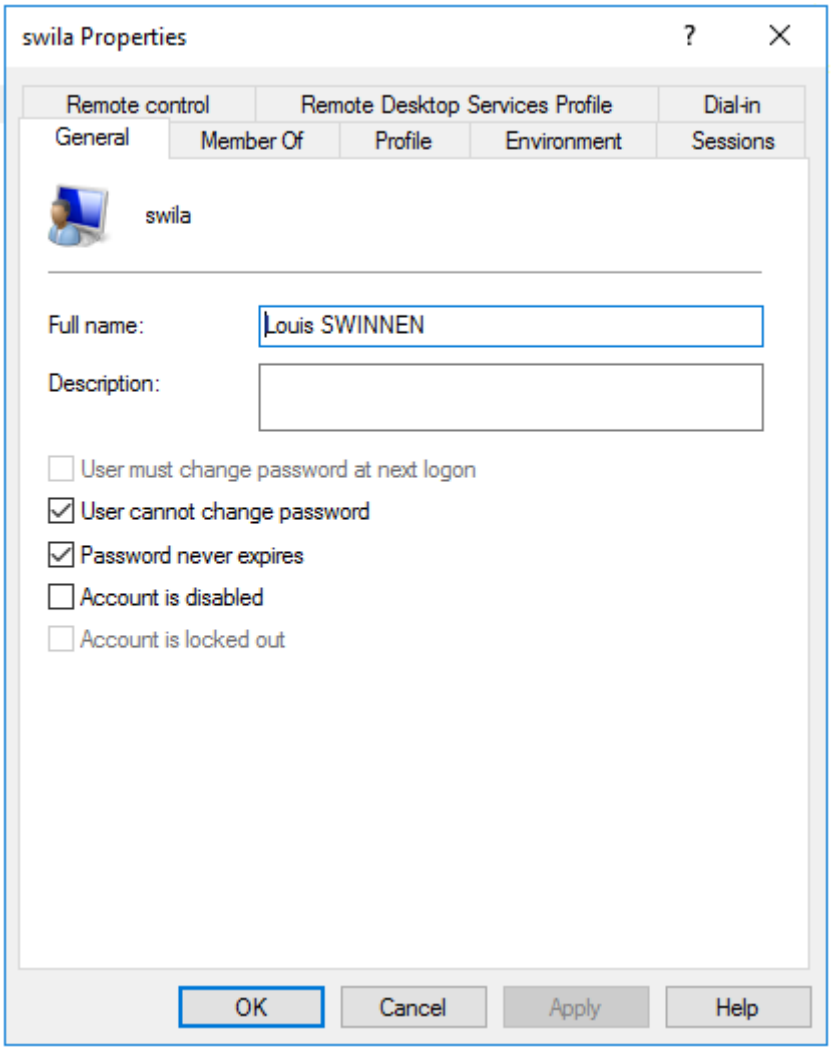


Figure 3.2 : Propriétés de l'utilisateur « swila »

Comme nous pouvons le voir sur la figure 3.2, il y a plusieurs onglets et donc beaucoup d'informations mémorisées par utilisateur. L'onglet *General* reprend les informations remplies lors de la création de l'utilisateur. L'onglet *Member Of* reprend les groupes auxquels cet utilisateur appartient. Il est possible d'ajouter des groupes directement par cet onglet. L'onglet *Profile* reprend les informations concernant *le profil* de cet utilisateur. Il est courant de spécifier le chemin dans lequel ce profil sera enregistré. Ainsi, dans notre exemple, on aurait pu mentionner le chemin suivant :

- Local path : `C:\profile\swila`
  Ce chemin mentionne l'endoit où les fichiers de l'utilisateur seront sauvegardés. La plupart des applications proposent par défaut cet endroit pour enregistrer ou charger un document.
- Profile path : `C:\profile\swila\ntprof`
  Ce chemin mentionne l'endroit où le *profil* de l'utilisateur est enregistré. Le profil comprend sa configuration (fond d'écran, icônes du bureau, contenu des « Documents »).

On mentionne dès lors que le **dossier de base** se trouve dans un chemin d'accès local (stocké sur le disque local dans le dossier *profile*) et que **le profil** de l'utilisateur se trouve dans un dossier *ntprof* à l'intérieur du chemin d'accès local. Ces options seront d'une très grande importance lors de l'étude d'Active Directory.

L'onglet *Environnement* permet de déterminer les programmes à lancer lorsque l'utilisateur ouvre sa session. Les autres onglets seront soit vus plus tard ou pas du tout. En effet, bon nombre des paramètres ici sont **gérés globalement via Active Directory**.

3.2.1 Création des groupes et utilisateurs en Powershell

Avant Windows Server 2016, il était nécessaire d'utiliser les objets ADSI pour réaliser les tâches d'administration locales. Cependant, toutes les possibilités ne sont pas encore implémentées. Ainsi, pour l'administration locale des utilisateurs, il faut faire un mélange des options, nous verrons cela dans la suite. Pour créer un groupe, **sous Windows Server 2016** :

```
$group = New-LocalGroup -Name "MonGroupe" -Description "Groupe de test"
```

Dans le script ci-dessus, nous créons un groupe de sécurité très simplement. Il siffit de passer les paramètres souhaités et ce groupe apparaîtra alors dans les *groupes locaux*.

Une méthode analogue peut être utilisée pour créer des utilisateurs et fixer certains paramètres[9] :

```
$user = New-LocalUser -AccountNeverExpires -PasswordNeverExpires `
    -FullName "Super SWILA" -name "Powerswila"  `
    -Password (ConvertTo-SecureString -AsPlainText "P@ssw0rd" -Force) `
    -UserMayNotChangePassword

Add-LocalGroupMember -Group "Users" -Member $user
Add-LocalGroupMember -Group "MonGroupe" -Member $user

<# Section ADSI #>
$userADSI = [ADSI] "WinNT://$env:computername/Powerswila"
$userADSI.Profile = "C:\profiles\Powerswila\ntprof"
$userADSI.HomeDirectory = "C:\profiles\Powerswila"
$userADSI.SetInfo()
```

Le script ci-dessus permet la création d'un utilisateur. L'utilisateur créé a, comme login, `Powerswila` et comme mot de passe `P@ssw0rd`, il ne peut modifier son mot de passe et celui-ci n'expire jamais. Ensuite, cet utilisateur est ajouté aux groupes *Users* et *MonGroupe* (par défaut il n'appartient à aucun groupe). Enfin, on modifie le chemin vers son profil et son chemin de base. Cette dernière partie est réalisée avec des objets ADSI puisque cette possibilité ne semble pas (encore) possible avec la nouvelles *cmdlet New-LocalUser*. Il ne faut surtout pas oublier d'appeler la méthode *SetInfo* qui va fixer la nouvelle valeur des paramètres.

3.2.2 Le profil

Le profil de l'utilisateur est un espace disque où toutes les informations le concernant (fichiers personnels, fichiers de configuration, fichiers systèmes, ...) y sont stockées. Il s'agit donc d'un élément important puisque ces données constituent le paramétrage de cet utilisateur. On y trouve également le dossier spécifique contenant son bureau, ses documents, ses images, son fond d'écran, etc.

Nous verrons que, dès qu'on utilise une authentification centralisée, l'emplacement du profil est un élément important.

[9] basé sur : http://powershell.com/cs/forums/t/6215.aspx?PageIndex=1

Création du profil

Lors de la première connexion d'un utilisateur, le système va lui créer un nouveau profil en prenant le profil par défaut comme exemple. Une fois le profil crée, il devient celui de cet utilisateur et seul ce dernier peut en modifier les paramètres.

Par défaut, les profils sont conservés dans le dossier `C:\Users`. On y trouve les profils suivants :

Dossiers	Signification
Administrateur	Profil de l'administrateur système dont le login est *Administrateur*.
Public	Contient les données pour tous les utilisateurs (icônes sur le bureau de tous les utilisateurs, menu démarrer partagé pour tous les utilisateurs, ...)
Default User	Contient le profil « type » utilisé pour la duplication et la création d'un nouveau profil utilisateur (lors de sa première connexion).
All Users	Est maintenu pour une compatibilité avec les systèmes pré-Vista.

Types de profil

Le profil de l'utilisateur peut être :

Type	Description
Local	Les informations de l'utilisateur sont stockées dans le dossier associé à cet utilisateur. Toutes les modifications réalisées par l'utilisateur sont enregistrées dans son profil.
Obligatoire	Le profil est *pré-configuré* et **ne peut pas être modifié par l'utilisateur**. Ainsi, toute modification du profil est perdue. Ce type de profil est utile lorsqu'un compte est partagé entre plusieurs personnes (par exemple : compte *guest*).
Temporaire	Un profil *temporaire* est créé par le système lorsqu'il y a un problème d'accès au profil de l'utilisateur. Le profil temporaire est détruit après la fermeture de la session. A l'inverse d'un profil obligatoire, il ne s'agit pas ici d'un profil pré-configuré mais bien d'un profil créé de manière temporaire.

Par défaut, le profil est *local*. Si l'on veut obtenir un profil *obligatoire*, il faut **renommer,** une fois la configuration terminée, le fichier `NTUSER.DAT` en `NTUSER.MAN`. Ce profil devient alors *obligatoire* et tout changement réalisé par l'utilisateur dans son profil (bureau, paramètres, …) est perdu.

Le dossier contenant le profil dépend des paramètres ajoutés au compte utilisateur. Si l'utilisateur est *local* et qu'aucun dossier n'a été mentionné dans son compte, son profil est sauvegardé dans le dossier local `C:\Users`. Si un dossier est précisé dans le compte utilisateur, le profil sera stocké dans le dossier mentionné **suffixé soit** par « `.V2` » (Windows < 10 version 1607), **soit** par « `.V6` » (Windows 10 >= 1607, Windows Server 2016)[10]. En effet, le dossier mentionné peut contenir un profil « ancienne génération » (pré-Vista sans le suffixe, …) tandis que le dossier suffixé par « `.V6` » contient les informations pour les versions les plus récentes de Windows 10.

[10] Une modification du registre peut causer la création de multiples versions du dossier profil en fonction du système d'exploitation client : extension « `.V2` » (Windows Vista, Windows 7) , « `.V3` » (Windows 8), « `.V4` » (Windows 8.1), « `.V5` » (Windows 10 version < 1607) ou « `.V6` » (Windows 10, version >= 1607).

3.3 Le système de fichiers

Les serveurs Windows peuvent utiliser plusieurs disques (ou volumes). Les volumes pris en charge peuvent être des disques *simples*, des disques *en RAID*, ... Le matériel supporté est relativement vaste.

Chaque volume peut être découpé en partition précise. La gestion des disques et des partitions est accessible depuis l'élément **Storage > Disk Management** de l'outil **Computer Manegement** disponible dans le **Server Management**. Grâce à cette interface, on peut :

- Voir tous les disques connectés au système
- Agrandir / Réduire un volume
- Formater un volume

Il faut être prudent car il y a des risques de perte d'information en cas de mauvaises manipulations.

Lors du formatage d'un volume (disque local, clé USB, ...) il faut mentionner **le système de fichiers** à utiliser. Microsoft utilise, depuis de nombreuses années, le système de fichiers NTFS. Ce système de fichiers, à l'inverse des systèmes FAT, **permet d'ajouter des permissions à chaque objet** présent sur le système. Ainsi, il est possible de limiter l'accès à des fichiers et/ou des dossiers en fonction de l'utilisateur connecté.

Pour ce faire, Windows utilise des ACL (liste de contrôle d'accès) pour limiter et contrôler les droits qui sont positionnés. En plus, chaque objet **hérite** des droits d'accès de son conteneur parent. Ainsi, si je crée un fichier dans un dossier qui n'est pas accessible à l'utilisateur *Powerswila*, le fichier hérite de cette propriété automatiquement.

En plus des permissions de type ACL, le système de fichiers permet d'ajouter **des attributs**. Ainsi, chaque objet sur le système de fichier peut posséder l'attribut *Lecture seule* (qui s'applique uniquement aux fichiers) qui empêche toute modification du fichier, *caché* qui cache le dossier sauf si l'utilisateur a mentionné qu'il souhaitait les voir, *système* qui mentionne que le fichier/dossier est de type système (utilisé par le système d'exploitation seulement), *archive* (qui était utilisé pour déterminer les fichiers qui avaient été modifiés depuis la dernière sauvegarde).

Avant de continuer, il est bon d'activer la visualisation des fichiers cachés et systèmes comme suit : dans l'explorateur, choisir le menu **View** puis **Options (à droite)** puis **Change folder and search options**. Choisir ensuite l'onglet **View** et ajuster les options *Show hidden files, folders and drives*, désactiver *Hide extensions for known file types* mais aussi *Hide protected operating system files*. Ainsi tous les fichiers/dossiers apparaitront dans l'explorateur de fichier.

3.3.1 Visualiser les attributs et permissions

Pour visualiser les permissions actives sur un objet, il faut simplement sélectionner celui-ci puis faire un **clic-droit** et choisir **Properties**.

L'onglet **General** reprend les informations courantes du fichier / dossier mais également ses attributs. L'onglet **Security** reprend les permissions ACL de ce fichier / dossier.

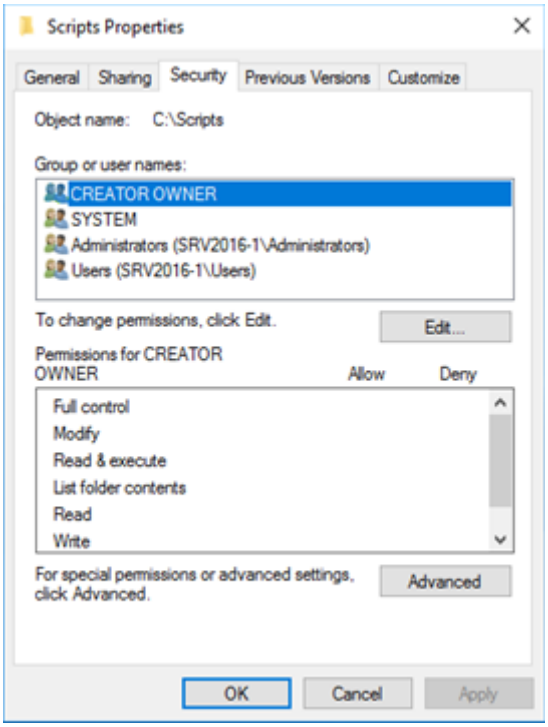


Figure 3.3 : Exemple de permissions sur le dossier *C:\Scripts*

Comme nous pouvons le voir sur la figure 3.3, la fenêtre reprenant les permissions est présentée en deux panneaux : le panneau supérieur reprend les utilisateurs disposant de permissions ou autorisations particulières. Le volet inférieur mentionne les autorisations accordées pour cet utilisateur.

Comme nous pouvons le voir, il y a des utilisateurs et des groupes courants :

Désignation	Explication
CREATOR OWNER	Cet intitulé reprend le créateur de l'objet en question (scripts dans notre exemple). Le propriétaire a, par défaut, des droits complets sur les objets qu'il possède.
SYSTEM	Cet utilisateur est utilisé par le système d'exploitation pour des tâches spécifiques. Il convient de ne pas toucher à ces permissions
Administrators	Reprend les autorisations associées au **groupe** des administrator**s** (remarquez le « s », >< de l'utilisateur administrator). Il est possible de limiter les droits pour ce groupe. Cependant, un administrateur peut toujours reprendre la propriété d'un objet et, dès ce moment, en modifier les permissions.
Users	Reprend les autorisations associées au groupe des utilisateurs.

Si l'on clique sur le bouton **Advanced** en bas de la fenêtre, on arrive sur une autre fenêtre qui décrit plus complètement les permissions et d'où elles viennent.

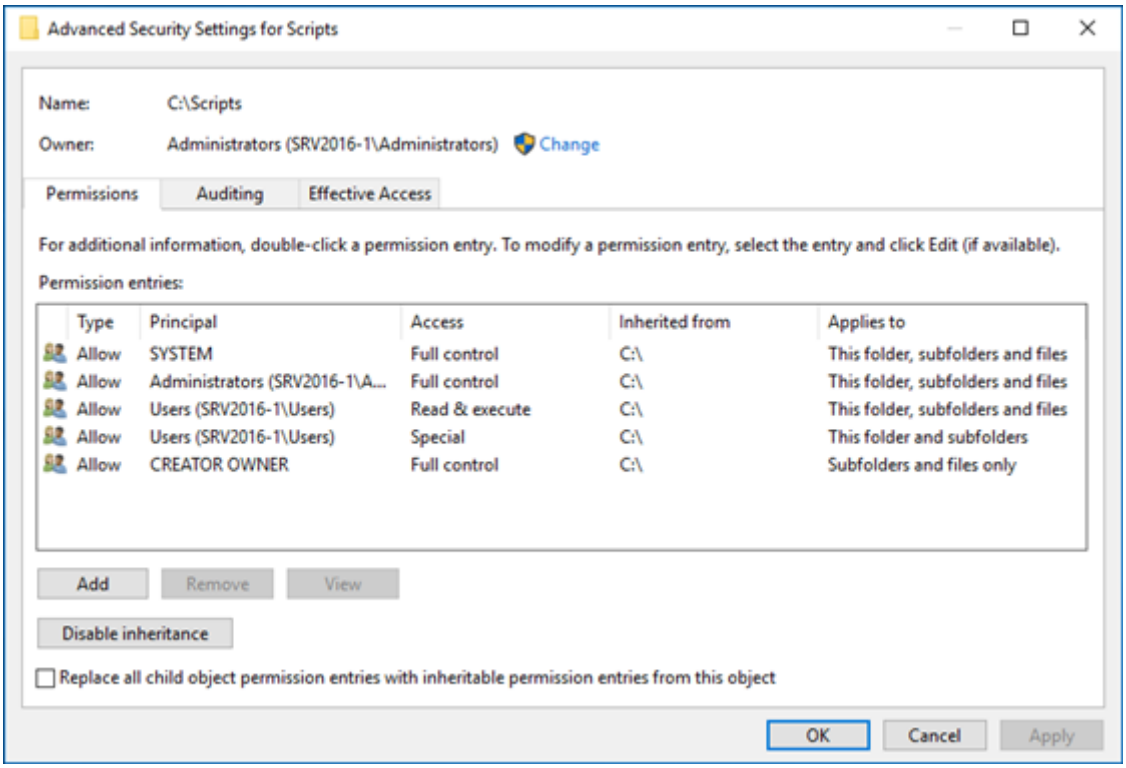


Figure 3.4 : Permissions avancées pour le dossier *C:\scripts*

La figure 3.4 nous montre comment les *permissions avancées* sont présentées. On peut y voir que, par exemple, *CREATOR OWNER*, *SYSTEM* et *Administrators* (2ème colonne) ont une autorisation de type *Full control* (3ème colonne) sur *ce dossier, les sous-dossiers et fichiers* (5ème colonne : *This older, subfolders and files*).

Les membres de *Users* (2ème colonne) ont une autorisation *lecture et exécution* (3ème colonne : *Read & execute*) sur *ce dossier, les sous-dossiers et fichiers* (5ème colonne). On remarque également que les membres de *Users* ont des autorisations *spéciales* (notée *Special*). De plus, ces autorisations *spéciales* portent sur *ce dossier, les sous-dossiers* (5ème colonne : *This folder and subfolders*).

Enfin, on peut lire que toutes ces autorisations sont **héritées** de C:\ (colonne *Inherited from*).

Comment comprendre ces autorisations ?

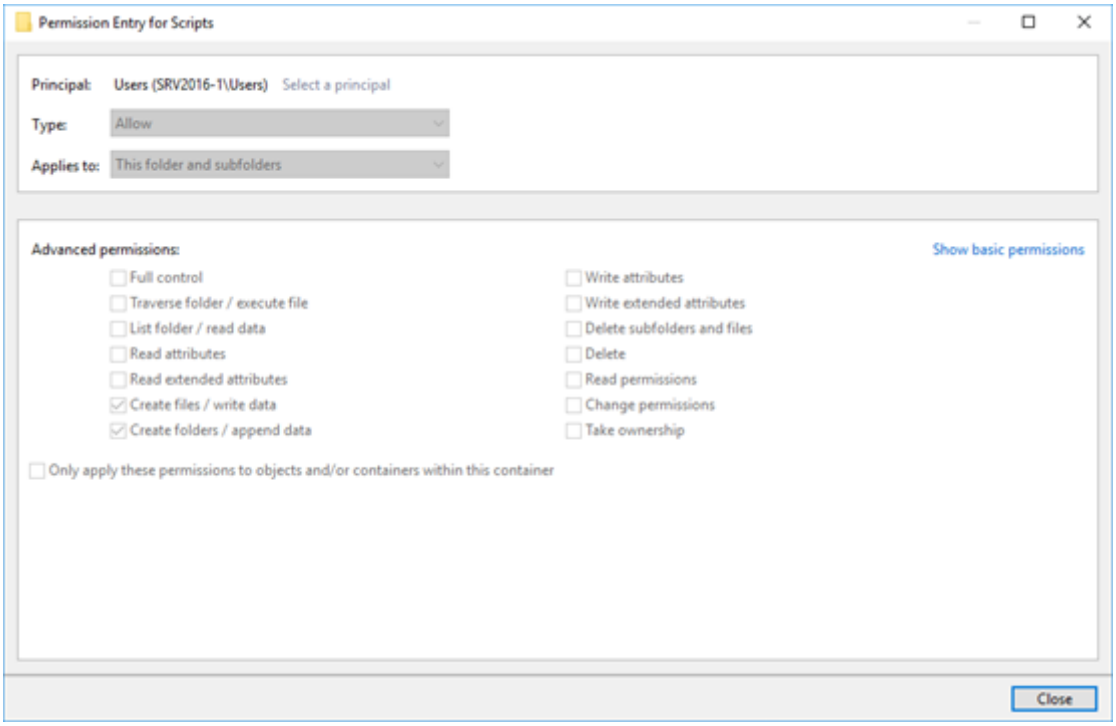
Tout d'abord, faisons un rapide état des lieux des autorisations courantes (extrait de [1][11]) :

Autorisation	Description
Read	Peut voir le contenu d'un dossier et ouvrir des fichiers. Il ne peut pas exécuter les programmes présents.
Read & Execute	Peut voir le contenu d'un dossier et ouvrir des fichiers. En outre, il peut exécuter les programmes présents.
Write	Peut créer des fichiers dans un dossier mais pas nécessairement les lire. Cette permission est intéressante pour la création d'un dossier dans lequel plusieurs utilisateurs peuvent déposer des fichiers sans avoir accès aux fichiers déposés par les autres (ex. : depot-labo)
Modify	Peut lire, modifier et supprimer des fichiers et dossiers (ex. C:\AdmSys)
Full Control	Peut réaliser toutes les opérations, y compris changer les permissions.

[11] Voir « Référence bibliographique » au début de la leçon.

Special	Autorisation personnalisée

Les *autorisations spéciales* sont donc une personnalisation. Voici quelques éléments :



Ainsi, l'autorisation *Read & execute* reprend les éléments suivants : *PTraverse folder / execute file, list folder / read data, Read attributes, Read extended attributes* et *Read permissions*.

Une autorisation peut s'appliquer à :

Désignation	Explication
This folder, subfolders and files	L'autorisation s'applique à ce dossier et sera héritée par tous les objets (dossiers et fichiers) créés dans ce dossier.
This folder only	L'autorisation s'applique à ce dossier seulement. Elle ne sera pas héritée par les objets présents dans le dossier
This folder and subfolders	L'autorisation s'applique à ce dossier et sera héritée par tous les dossiers qui seront créés dans ce dossier
This folder and files	L'autorisation s'applique à ce dossier et sera héritée par tous les fichiers présents dans ce dossier
Subfolders and files only	L'autorisation sera héritée par tous les objets (dossiers et fichiers) créés dans ce dossier. Cependant, elle ne s'applique pas au dossier lui-même.
Subfolders only	L'autorisation sera héritée par tous les dossiers créés dans ce dossier. Cependant, elle ne s'applique pas au dossier lui-même.
Files only	L'autorisation sera héritée par tous les fichiers présents dans ce dossier. Cependant, elle ne s'applique pas au dossier lui-même.

Comme nous pouvons le voir, la compréhension des permissions n'est pas aisée car il y a beaucoup (trop) de possibilités.

3.3.2 Modification des permissions

La modification des permissions peut prendre deux formes très différentes :

1. Ajouter/Retirer des autorisations pour des utilisateurs donnés. Donc *aucune modification aux permissions héritées*.
2. Modifier les permissions héritées

Ajouter / modifier des autorisations

S'il s'agit d'ajouter/retirer des autorisations, cela peut se faire très facilement en suivant les étapes suivantes :

1. **Clic-droit** sur l'objet pour lequel des permissions doivent être ajoutées, puis **Properties**
2. Onglet **Security** et choisir le bouton **Edit**.
3. Choisir **Add** puis entrer le nom du groupe ou de l'utilisateur dont il faut adapter les permissions puis choisir **Check names** afin de s'assurer que celui-ci existe, puis choisir **OK**[12]
4. Les autorisations pour l'élément ajouté peuvent ensuite être précisées dans le panneau du bas.

Modifier les autorisations héritées

Modifier des autorisations héritées impose une étape supplémentaire. En effet, il faut **bloquer l'héritage** des permissions avant de pouvoir réaliser une quelconque modification. Pour ce faire, il faut suivre les étapes suivantes :

1. **Clic-droit** sur l'objet pour lequel des permissions doivent être ajoutées, puis **Properties**
2. Onglet **Security** et choisir le bouton **Advanced**.
3. Cliquer sur le bouton **Disable inheritance**.
    a. Le système vous informe que dans ce cas, les autorisations ne seront pas héritées et vous demande ce qu'il doit faire des autorisations actuelles.
    b. Le plus souvent, il suffit de choisir l'option **Convert inherited permissions into explicit permissions on this object** qui va ainsi copier les permissions héritées et permettre de les modifier. Le bouton **Remove all inherited permissions from this object** retire toutes les autorisations héritées et l'administrateur doit alors les définir à nouveau.
4. Appuyer sur le bouton **Apply** puis **OK** pour fermer la fenêtre *Advanced Security Settings*.
5. Cliquer sur le bouton **Edit** dans le panneau du haut
6. Il est désormais possible de modifier les permissions des utilisateurs

[12] Vous pouvez également choisir **Avancé** et préciser les paramètres de votre recherche et choisir **Rechercher**. Vous pouvez alors effectuer votre choix parmi la liste proposée.

Gérer le propriétaire d'un objet

Le propriétaire (ou CREATOR OWNER) est visible et peut être modifié comme suit :

1. **Clic-droit** sur l'objet pour lequel des permissions doivent être ajoutées, puis **Properties**
2. Onglet **Security** et choisir le bouton **Advanced**.
3. Dans le panneau du haut, l'option **Owner** montre le propriétaire actuel de l'objet et le bouton **Change** permet de changer celui-ci.

Ainsi, il est possible de s'approprier un objet. Cette option est indispensable dans certains cas quand l'administrateur doit reprendre la main sur un objet pour lequel tous les droits d'accès lui ont été supprimés.

3.3.3 Commandes et scripting

La gestion des ACL (et donc la modification des permissions ou autorisations) peut se faire au moyen du programme **icacls.exe.** Ce programme permet de réaliser bon nombre d'opérations, l'aide est disponible en tapant **icacls.exe /?**.

Ainsi, si nous créons le dossier `C:\testACL`, il hérite par défaut des ACL placées sur le disque `C:\`. On peut donc bloquer l'héritage et configurer des ACL particulières pour l'utilisateur `powerswila` :

`C:\>` **`icacls testACL /inheritance:d`**
Cette commande bloque l'héritage et copie les ACL

`C:\>` **`icacls testACL /grant powerswila:(M)`**
Cette commande ajoute l'autorisation en modification sur le dossier seulement. Voir l'aide de `icacls` pour plus d'information.

`C:\>` **`icacls testACL /grant powerswila:(OI)(CI)(F)`**
Cette commande ajoute l'autorisation contrôle total sur le dossier, les sous-dossiers et les fichiers.

En PowerShell, deux possibilités s'offrent à nous : **l'exécution directe** d'une commande externe ou l'utilisation de cmdlets particulier. Pour *l'exécution directe* (i.e. appel à la commande *icacls* depuis un script PowerShell), il suffit de faire comme suit :

```
$resultat= &"icacls" "testACL" "/grant" "powerswila:(OI)(CI)(F)"
```

Bien sûr, il est possible d'utiliser directement des objets particuliers, disponibles en PowerShell pour modifier les ACL.

Voici un premier script introductif[13] de modification des ACL pour le dossier `C:\testACL` :

```
$autorisation= [System.Security.AccessControl.FileSystemRights]"Modify"
$heritage= [System.Security.AccessControl.InheritanceFlags]::None
$propagation= [System.Security.AccessControl.PropagationFlags]::None
$decision= [System.Security.AccessControl.AccessControlType]::Allow

$utilisateur = New-Object System.Security.Principal.NTAccount("powerswila")

$acl= Get-Acl "C:\testACL"
$ace= New-Object Security.AccessControl.FileSystemAccessRule($utilisateur,`
                    $autorisation, $heritage, $propagation, $decision)
```

[13] Extrait de http://technet.microsoft.com/en-us/library/ff730951.aspx

```
$acl.AddAccessRule($ace)
Set-Acl "C:\testACL" $acl
```

Dans ce script, nous **ajoutons** une entrée ACL (appelée dans le script $ace) au dossier `C:\testACL`. Cette entrée concerne :

- Un utilisateur `System.Security.Principal.NTAccount("powerswila")`
- Une autorisation `[System.Security.AccessControl.FileSystemRights]"Modify"`
- Aucune option d'héritage, ni de propagation :
  `[System.Security.AccessControl.InheritanceFlags]::None`
  `[System.Security.AccessControl.PropagationFlags]::None`
- Une décision `[System.Security.AccessControl.AccessControlType]::Allow`

Après exécution de ce script, l'utilisateur `PowerSwila` se voit *ajouter* une autorisation de *modification* sur *le dossier uniquement* `C:\testACL`.

Voici les valeurs possibles pour les **autorisations** (`FileSystemRights`) : *AppendData, ChangePermissions, CreateDirectories, CreateFiles, Delete, DeleteSubdirectoriesAndFiles, ExecuteFile,* ***FullControl****, ListDirectory,* ***Modify****,* ***Read****,* ***ReadAndExecute****, ReadAttributes, ReadData, ReadExtendedAttributes, ReadPermissions, Synchronize, TakeOwnership, Traverse,* ***Write****, WriteAttributes, WriteData, WriteExtendedAttributes*

Les valeurs pour la **propagation** sont : `InheritOnly`, `NoPropagateInherit` et `None`. Pour **l'héritage**, les valeurs possibles sont : `ContainerInherit`, `ObjectInherit` et `None`. En fonction de la portée souhaitée, il faut combiner[14] ces deux valeurs *propagations* et *héritage* comme suit :

	Héritage	Propagation
This folder only	None	None
This folder, Subfolders and files	Container|Object	None
This folder and subfolders	Container	None
This folder and files	Object	None
Subfolders and files only	Container|Object	InheritOnly
Subfolders only	Container	InheritOnly
Files only	Object	InheritOnly

Dans le tableau précédent, il faut comprendre `Container|Object` comme étant une opération OU bit à bit. En PowerShell, cela se traduirait par une ligne du type :

```
$heritage= `
 [System.Security.AccessControl.InheritanceFlags]::ContainerInherit -bor `
 [System.Security.AccessControl.InheritanceFlags]::ObjectInherit
```

3.3.4 La gestion des quotas

Windows Server 2016 supporte deux types de quota :

- **Les quotas disques** : il s'applique sur un volume complet prennent en compte l'occupation de l'espace par les utilisateurs. Des entrées de quota, par utilisateur, peuvent être configurées.

[14] Repris de http://stackoverflow.com/questions/3282656/settings-inheritance-and-propagation-flags-with-set-acl-and-powershell

- **Les quotas sur un chemin donné** : il s'applique à un dossier ou des sous-dossiers. La portée de ces quotas est plus fine. Il est possible de définir un *modèle de quota* qui s'appliquera sur le dossier en question.

Pour configurer les quotas sur base d'un chemin donné, il faut **installer un rôle supplémentaire**, pour ce faire, aller dans **le Server Manager**, choisir **Manage** > **Add Roles and Features**. Dans la liste des rôles disponibles, il faut déployer **File and Storage Services** et, ensuite, déployer **Files and iSCSI Services**, il convient d'installer le **File Server Ressource Manager** (et accepter toutes les dépendances). Une fois ce rôle ajouté, une nouvelle option apparaît dans le menu **Tools** s'appelant **File Server Resource Manager**.

Les quotas disques

Il est possible de définir des quotas sur le disque dur. Les quotas sont des limites imposées par utilisateur quant à l'espace disque. Ils sont particulièrement intéressants pour s'assurer que l'utilisateur ne dépasse pas une limite donnée. Ils sont indispensables pour les ressources partagées entre plusieurs utilisateurs.

Pour **activer** les quotas, il faut aller dans **Démarrer** > **This PC**. Il faut, ensuite, faire un **clic-droit** sur le volume concerné (C: par exemple) et choisir **Properties**.

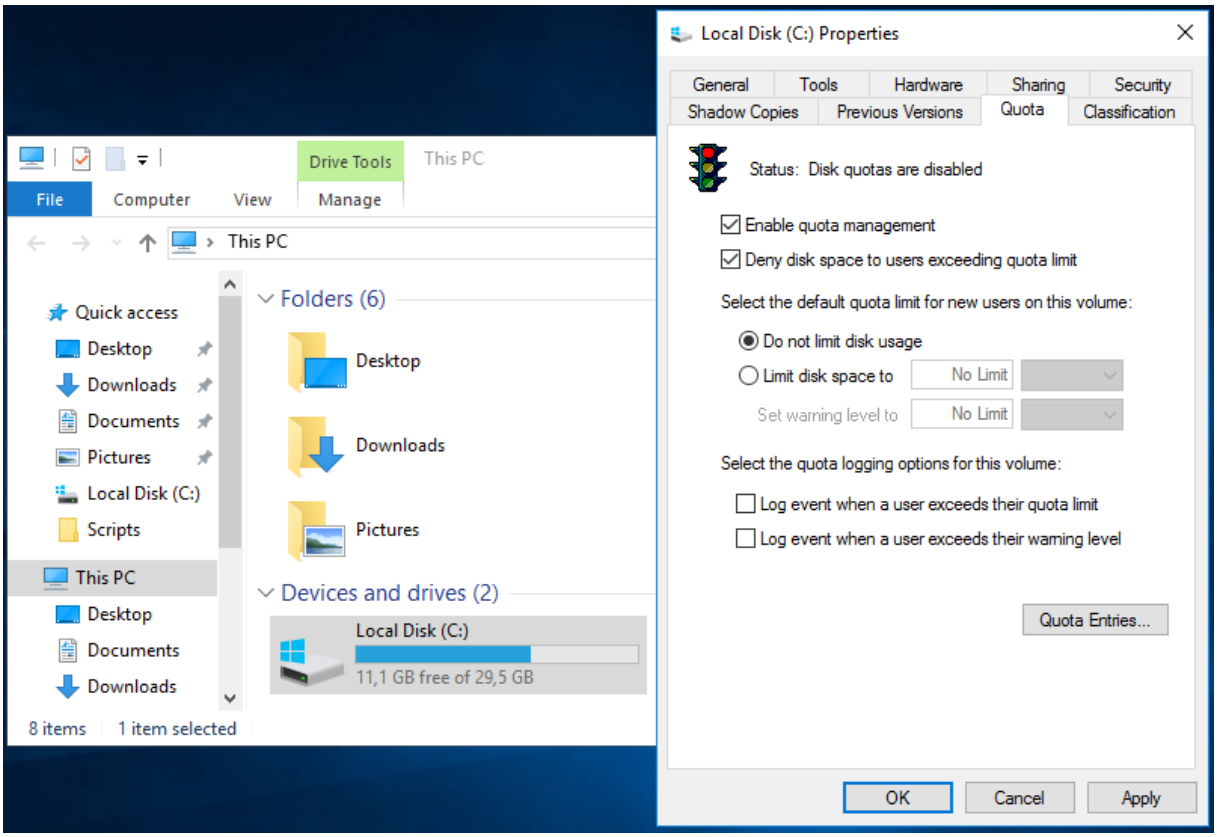


Figure 3.5 : Gestion des quotas

Il faut alors choisir **l'onglet Quota** (voir figure 3.5) et cocher **Enable quota management** ainsi que **Deny disk space to users exceeding quota limit**. Une fois que l'on confirme la gestion des quotas, ceux-ci sont actifs sur le disque concerné.

Il faut remarquer que si l'on ne définit pas d'**entrées de quota** (figure 3.5, bouton en bas à droite, *Quota Entries…*), aucune limite n'est observée sur le système.

Il est possible de définir une *limite par défaut* (figure 3.5, *limit disk space to*) pour les nouveaux utilisateurs. Cette limite s'applique alors à tous les nouveaux utilisateurs déposant un fichier sur le disque concerné. Ensuite, il est possible de spécifier des entrées particulières, par utilisateur, grâce au bouton **Quota Entries**.

Une fois dans la fenêtre montrant les ***entrées de quotas*****,** il faut choisir l'option **New quota entry** et fixer les limites voulues. Le quota est alors actif pour cet utilisateur **sur tout le volume**.

En ligne de commande, il est possible de fixer les quotas disques en utilisant la commande **fsutil quota**. Cette commande permet de :

- Activer/Désactiver les quotas sur un volume
- Activer l'application des quotas
- Ajouter ou modifier des entrées de quota

Cette dernière option est particulièrement intéressante lorsqu'on crée des scripts d'ajout des utilisateurs, on peut ainsi créer les entrées de quotas correspondantes. Surtout si la politique de quota est différente en fonction des utilisateurs (un utilisateur d'une orientation donnée pourrait ne pas avoir les mêmes quotas qu'un autre).

```
C:\> fsutil quota modify /?
```

Cette commande explique comment utiliser cette option pour ajouter ou modifier une entrée de quota. Par exemple, si nous voulons ajouter une entrée de quota pour l'utilisateur *powerswila* sur le disque *C:*, il faut entrer la commande suivante :

```
C:\> fsutil quota modify C: 900000000 1000000000 powerswila
```

Cette commande permet d'ajouter une entrée de quota pour l'utilisateur *powerswila* sur le volume *C:* avec un seuil d'avertissement autour[15] de 900 Mo et une limite autour de 1000 Mo.

Powershell ! Il est facile d'intégrer cette commande dans un script *Powershell* **en exécution directe**. Pour ce faire, il faut simplement précéder la commande du symbole « & ». A l'intérieur d'un script Powershell, nous obtiendrions :

```
$resultat = &"fsutil" "quota" "modify" "C:" "900000000" "1000000000"
                    "powerswila"
```

Tous les arguments sont séparés par des espaces et des guillemets. Le résultat de la commande se trouve dans la variable `$resultat`.

Les quotas sur un chemin donné

L'interface permettant de définir *les quotas sur un chemin donné* se trouve dans **Server Manager > Tools > File Server Resource Manager.** Avant de pouvoir fixer un quota sur un chemin déterminé, il convient de déterminer **la politique spécifique** à appliquer (au moyen d'un *template*). La présence des

[15] Par facilité, j'ai compté ici le Mo à 1000 000 o ce qui n'est pas correct. Pour rappel, 1 Mo = 1024*1024 octets

modèles de quota (**Quota Templates**) sont une réelle avancée en termes de granularité (**Quota Management > Quota Templates**).

Un modèle de quota peut reprendre :

- Une limite **soft** (qui est principalement utilisée pour les notifications et les rapports) ou une **limite hard** (qui ne peut être dépassée)
- Des *seuils de notification* (**Notifications thresholds**) exprimés en pourcentage de la limite. A chaque seuil, on peut définir :
    - Un mail qui est envoyé automatiquement à l'utilisateur pour lui notifier que le seuil a été atteint (colonne E-mail)
    - Un événement qui sera consigné dans le journal pour permettre aux administrateurs de suivre l'évolution (colonne Event)
    - Une commande à exécuter qui permet à l'administrateur de faire tout ce qu'il veut (colonne Command). Ainsi, certains modèles de quota proposent une extension unique de 50 Mo par l'exécution de la commande dirquota (voir plus loin).

Comme nous pouvons le voir, il est facile de définir un modèle de quota. Une fois les modèles nécessaires définis, il faut les faire appliquer à des chemins particuliers.

Pour ce faire, il faut aller dans **Quota** pour **Create Quota**. Lors de l'ajout, il faut mentionner :

- *Quota path* – le dossier qui est visé par le quota que nous allons créer.
- S'il s'agit d'un *Quota on a path* (s'applique donc au dossier mentionné, dans sa globalité) ou s'il s'agit d'un quota automatique *Auto apply template and create quotas on existing and new subfolders* (auquel cas, tous les dossiers existants et nouveaux se verront définir le quota en fonction du modèle choisi).
- Lier le modèle de quota définissant la politique choisie (via *Derive properties from this quota template*)

Dès qu'on appuie sur **Create**, l'entrée est ajoutée et active.

Par exemple, si je crée le dossier `C:\testQuota`, je peux lui attribuer le quota comme suit :

- *Quota path* : `C:\testQuota`
- *Create quota on path*
- *Derives properties from this quota template* : *200 Mb Limit with 50 MB Extension*

Dans ce mode, le dossier `C:\testQuota` peut occuper une taille maximale de 250 Mo.

Si on supprime l'entrée précédente et que nous la remplaçons par la suivante :

- *Quota path* : `C:\testQuota`
- *Auto apply template and create quotas on existing and new subfolders*
- *Derives properties from this quota template* : *200 Mb Limit with 50 MB Extension*

Dans ce mode, tous les dossiers présents ou créés dans `C:\testQuota` se verront attribuer un quota automatiquement et chacun pourra occuper une taille maximale de 250 Mo. Intéressant pour fixer un quota sur le dossier contenant l'ensemble des répertoires personnels des utilisateurs.

En ligne de commande, il est possible de définir des quotas sur un chemin déterminé au moyen de la commande `dirquota`.

Pour les quotas qui s'appliquent sur le dossier directement :

```
C:\> dirquota quota /?
```
Cette commande affiche l'aide en ligne pour ce type de quota.

Pour les quotas automatiques (quotas créés automatiquement sur les sous-dossiers) :

```
C:\> dirquota autoquota /?
```
Cette commande affiche l'aide en ligne pour ce type de quota.

Dans la suite, vous trouverez quelques exemples d'utilisation, pour plus d'information, veuillez-vous reporter à la documentation.

```
C:\> dirquota quota add /path:C:\testQuota /SourceTemplate:"200 Mb Limit with 50 MB Extension"
```
Cette commande ajoute une entrée de quota pour le dossier C:\testQuota en lui appliquant le modèle mentionné.

```
C:\> dirquota autoquota add /path:C:\testQuota /SourceTemplate:"200 Mb Limit with 50 MB Extension"
```
Cette commande ajoute un quota automatique pour le dossier C:\testQuota en lui appliquant le modèle mentionné.

En **Powershell**, il est possible de scripter facilement ce type de quotas. Pour les quotas définis sur un chemin précis, il suffit[16] de faire comme suit :

```
$fqtm = New-Object -com Fsrm.FsrmQuotaManager
$quota = $fqtm.CreateQuota("C:\testQuota")
$quota.ApplyTemplate("200 Mb Limit with 50 MB Extension")
$quota.Commit()
```

Pour les quotas automatiques, c'est tout aussi simple[17] :

```
$fqtm = New-Object -com Fsrm.FsrmQuotaManager
$quota = $fqtm.CreateAutoApplyQuota("200 Mb Limit with 50 MB Extension",
        "C:\testQuota")
$quota.Commit()
```

Bien sûr, il est aussi possible de faire appel à la commande `dirquota` depuis *Powershell* **en exécution directe** comme suit :

```
$resultat = &"dirquota" "autoquota" "add" "/path:C:\testQuota"
        "/SourceTemplate: 200 Mb Limit with 50 MB Extension"
```

3.4 La stratégie locale

Comme nous l'avons vu dans la leçon 1 concernant l'installation initiale, il existe une **stratégie locale** définissant beaucoup de paramètres concernant la *politique locale* du serveur. On y trouve,

[16] Extrait de : http://blog.dboden.be/2009/03/managing-fsrm-by-using-powershell/
[17] Extrait de : http://www.simple-talk.com/sysadmin/exchange/implementing-windows-server-2008-file-system-quotas/

notamment, la *stratégie concernant les mots de passe*. Nous allons maintenant explorer d'autres éléments de cette stratégie locale.

Attention ! Les éléments de stratégies sont importants dans l'administration des serveurs Windows Server. Ainsi, lorsqu'un serveur intègre un réseau (précisément avec Active Directory), des stratégies globales peuvent s'appliquer sur la machine. Nous retrouverons donc les stratégies lors de notre étude d'Active Directory.

Pour modifier la stratégie locale, il faut aller dans le **Server Manager > Tools > Local Security Policy.**

Par exemple, il est possible de *ne pas afficher le dernier utilisateur qui s'est connecté* en modifiant la stratégie locale comme suit : **Security Settings > Local Policies > Security Options > Interactive login : Do not display last user name > Enabled**.

Il est également possible d'activer *un écran d'accueil* juste avant la procédure de connexion en modifiant les paramètres suivants : **Security Settings > Local Policies > Security Options > Interactive login : Message title for users attempting to log on** et **Interactive login : Message text for users attempting to log on**.

Enfin, on trouve également dans la stratégie de sécurité locale un dernier élément important : *la stratégie d'audit* (**Audit Policy**). La stratégie d'audit permet de superviser des éléments importants du système. Les événements audités sont consignés dans les journaux systèmes. On peut ainsi auditer l'ouverture de la connexion (aussi bien la réussite que l'échec), les événements systèmes (modification de l'heure, les tentatives de démarrage/d'arrêt de certains éléments critiques, …), les tentatives de modification des stratégies …

Pour modifier ces paramètres, il faut suivre le chemin **Security Settings > Local Policies > Audit Policy**. Il est alors aisé d'activer les stratégies voulues dans le panneau de droite.

3.5 Divers

3.5.1 La console MMC

L'outil de **Computer Manager** est un exemple de ce que peut être la console MMC. Il s'agit d'un composant à l'intérieur duquel l'administrateur peut ajouter des *Snap-in* (ie. Des composants). Ces composants sont prévus pour fonctionner dans la console. L'intérêt d'utiliser la console est double :

- L'administrateur peut personnaliser sa console avec les outils de gestion qu'il utilise
- La console peut lorsqu'elle ajoutée, se rapporter à l'ordinateur local, à un ordinateur distant ou parfois à un compte de service. Elle permet donc des modifications plus fines.

Pour démarrer la console MMC, il suffit de *rechercher* et *exécuter* mmc.exe. Une fois lancée, il faut choisir l'option *Add/Remove Snap-in* et choisir le composant souhaité, par exemple **Computer Management**. Lors de l'ajout, la console vous demande s'il faut réaliser cet ajout pour l'ordinateur local ou un ordinateur distant. Le choix par défaut nous convient. On se retrouve donc avec une console de gestion reprenant l'outil de gestion des serveurs. Il est possible d'ajouter plusieurs éléments à l'intérieur de la console afin de regrouper les éléments souvent utilisés par l'administrateur.

Lorsque vous quittez cette console, un message vous demandant si vous souhaitez sauvegarder cette dernière s'affiche (il ne s'agit pas des modifications qui ont pu être effectuées mais simplement de la configuration de la console avec les composants enfichables ajoutés).

3.5.2 L'observateur d'événements

Event Viewer est l'outil permettant de consulter les journaux systèmes. Les journaux sont une mine d'information en cas de problèmes sur un serveur car ils renseignent (de manière pas toujours claire) la nature de l'erreur rencontrée. Ce doit être le premier réflexe de l'administrateur système lorsqu'un événement étrange survient : il faut consulter ces journaux pour déterminer s'est passé.

En plus, en précisant la *stratégie d'audit* (**Audit Policy**) souhaitée, les journaux vont également se souvenir des événements importants pour l'administrateur comme les tentatives (réussies ou ratées) d'ouverture de sessions, …

Pour ouvrir l'observateur d'événements, il faut démarrer le **Server Manager> Tools > Event Viewer > Windows Logs**.

3.5.3 Planification de tâches

A l'instar des systèmes Linux, il existe également dans les systèmes Windows une possibilité pour planifier une tâche ponctuelle ou répétitive. Pour atteindre cet outil : **Server Manager > Tools > Task Scheduler**.

Il faut ensuite, dans la section **Task Scheduler Library**, créer une tâche et lui donner tous ses paramètres pour travailler.

Exercices

Création des comptes

1. En reprenant le fichier obtenu à l'exercice 1 de la leçon 2, écrivez un script Powershell permettant de créer les comptes des utilisateurs.
    - Le script fixera le mot de passe, le chemin d'accès local à `C:\UserData\<login>` ainsi que le chemin vers le profil à `C:\UserData\<login>\myprofile`.
    - Un groupe particulier sera créé par catégorie[18]. Tous les utilisateurs seront membres du groupe *Users* et du groupe spécifique correspondant à sa catégorie (*administratif-communication-comptabilité …*)
    - Les dossiers suivants seront créés :
        - `C:\UserData\<login>`
        - `C:\UserData\<login>\myprofile.V6`

      Vous ajouterez une autorisation de type *controle total* à l'utilisateur sur ces dossiers.
    - Vérifiez si tout s'est bien passé ! Tentez de vous connecter avec l'un des comptes créés.

(Conseil : testez votre script sur un nombre très réduit d'utilisateurs : les deux premiers par exemple)

[18] Pour déterminer si un groupe existe déjà, vous pourriez utiliser la cmdlet `Get-LocalGroup` avec un bloc try/catch.

2. Localisez le profil de l'utilisateur avec lequel vous vous êtes connectés. Jetez un œil aux données présentes à l'intérieur

3. Ecrivez le script qui permet de supprimer les comptes des utilisateurs créés précédemment. Pour ce faire, utilisez la cmdlet `Remove-LocalUser`

4. (Via l'interface graphique) Créer un compte *helmo*
   - Sans mot de passe.
   - Fixer le chemin le chemin d'accès local à `C:\UserData\helmo` et le profil dans `C:\UserData\helmo\preconfig`
   - Créer les dossiers `C:\UserData\helmo` et `C:\UserData\helmo\preconfig.V6`. Fixer les droits pour que *helmo* dispose d'un *contrôle total* sur ces dossiers
   - Connectez-vous avec ce login.
   - Créer un lien vers le serveur DATA (\\192.168.128.3) et placez ce raccourci sur le bureau.
   - Transformer ce profil standard en profil obligatoire.
   - Testez-le !

Les quotas

5. Modifier le script d'ajout des utilisateurs pour que des quotas soient ajoutés en même temps. **Prévoyez deux versions** : l'utilisation des quotas disques et l'utilisation des quotas basés sur le chemin. Les quotas à respecter par catégories sont les suivants :
   a. Administratif, social, comptabilité et direction : quota=400 Mo ; Alerte=390 Mo
   b. E-Learning, etudiant, juridique et travaux: quota = 300 Mo ; Alerte à 290 Mo
   c. Informatique, communication et personnel : quota = 800 Mo ; Alerte à 750 Mo

   Il sera peut être nécessaire de modifier le script suivant le type de quota à adapter.

6. Vérifiez que les quotas sont effectivement appliqués.

La stratégie locale

7. Activez la stratégie visant à *ne pas afficher le dernier utilisateur qui s'est connecté*. Redémarrez votre serveur. Quel changement observez-vous ?

8. Activez un écran et un message d'accueil comme suit :
   a. Titre : Welcome on Windows Server 2016
   b. Message : System Administrator : <VotreNom>

   Fermez votre session et connectez-vous à nouveau. Qu'observez-vous ?