

Chapitre 6

Les contrôles d'accès

1. Définitions

1.1 Objets

Un objet dans le monde Microsoft est simplement un élément qui peut être sécurisé et protégé par des permissions. Il peut s'agir d'un fichier, d'un répertoire, d'une clé de registre, ou d'un objet système ou noyau comme un canal nommé, un socket, un processus, un thread, etc. Microsoft ne fournit pas en standard d'outils pour gérer les permissions sur tous les objets. Bien souvent, c'est à la charge du développeur de gérer ses permissions programmatiquement lorsqu'il crée et manipule un objet.

Un utilitaire est cependant disponible en téléchargement qui permet de lister et d'afficher les propriétés et sécurités des objets du système d'exploitation. Il s'agit de winobj.exe dont le lien pour le téléchargement est fourni en annexe.

1.2 ACL

Une ACL est une liste de contrôle d'accès qui régit les permissions d'accès aux objets du système. Elle est composée d'entrées de contrôles d'accès ACE (*Access Control Entry*). Il existe deux types de liste de contrôles d'accès, celles dites système et celles dites discrètes.

1.3 ACE

Une entrée de contrôle d'accès ACE (*Access Control Entry*) est un élément dans une ACL. L'ACE est la structure qui stocke les permissions pour un compte de sécurité donné et définit également les propriétés d'audit pour ce compte sur l'objet en question. Une ACL peut posséder zéro ou plusieurs ACE.

Une ACE est composée d'un SID, d'un masque d'accès qui définit les permissions accordées au SID (ex : Lecture, Ecriture, etc.), un drapeau qui détermine le type d'ACE et un dernier drapeau qui détermine à quels objets sont ajustées ces permissions (à l'objet en question ou à ses enfants, voir la notion d'héritage plus bas dans ce chapitre).

Il existe trois types d'ACE, une ACE de type Accès accordé, une ACE de type Accès refusé et une ACE de type Audit System.

1.4 Descripteur de sécurité

Le descripteur de sécurité SD (*Security Descriptor*) est la structure rattachée à tout objet sécurisable qui contient le SID du propriétaire de l'objet, son groupe primaire, et les deux listes d'ACL : le DACL qui définit les permissions et le SACL qui définit l'audit sur l'objet.

1.5 DACL

Une DACL (*Discretionary Access Control List*) est une liste de contrôle d'accès dite discrétionnaire car contrôlée par le propriétaire de l'objet, en d'autres termes, laissée à la discrétion du propriétaire de l'objet. La DACL est la liste de contrôles d'accès qui gère les permissions sur un objet sécurisé : elle spécifie qui est autorisé et interdit d'accéder à cet objet.

1.6 SACL

Une SACL (*System Access Control List*) est une liste de contrôle d'accès qui contrôle la génération de messages d'audit lors des tentatives d'accès à un objet sécurisé. Il faut avoir le privilège SE_SECURITY_NAME pour modifier une SACL.

1.7 SDDL

Le langage de définition des descripteurs de sécurité SDDL (*Security Descriptor Definition Language*) est un langage qui permet de définir et transporter des informations stockées dans un descripteur de sécurité au format texte. Cela peut être utile pour scripter ou automatiser l'application de sécurités sur un ensemble de données à partir d'un fichier d'entrée au format texte.

Par exemple, la commande **sc sdshow cryptsvc** affiche le descripteur de sécurité attaché au service au format SDDL :

```
▣ sc sdshow cryptsvc
```

```
▣ D:(A;;CCLCSWRPWPDTLOCRRC;;;SY)(A;;CCDCLCSWRPWPDTLOCRSDRC  
WDWO;;;BA)(A;;CCLCSWLOCRRC;;;IU)(A;;CCLCSWLOCRRC;;;SU)
```

SDDL est un langage à la syntaxe complexe qui peut facilement être source d'erreurs. Il vaut mieux éviter de construire des permissions à la main au format SDDL. En cas de besoin, il est plus facile d'appliquer des permissions sur un environnement de test et d'exporter ces permissions au format SDDL à l'aide d'utilitaires pour ensuite les appliquer sur un environnement cible. Le MSDN reste cependant la meilleure source d'informations pour l'utilisateur à la recherche d'informations supplémentaires sur la syntaxe SDDL. Un lien sur la documentation SDDL est fourni en annexe.

1.8 Héritage

L'héritage est une notion fondamentale et importante dans l'application de permissions sur des objets sécurisable. L'héritage permet la propagation d'ACL positionnées sur des objets conteneur parents à leurs enfants. Ceci est particulièrement utilisé pour sécuriser des branches de clé de registre ou des fichiers et sous-répertoires de répertoires parents. Lors de la création d'une ACE, il est possible de préciser si l'ACE en question s'applique à l'objet, à ses enfants seulement (drapeau héritage seulement ou *Inherit Only*) ou à l'objet lui-même et à ses enfants. Pour un répertoire par exemple, il est également possible de préciser que l'ACE ne s'applique qu'à elle-même et aux fichiers qu'elle contient mais pas aux sous-répertoires et sous-fichiers.

Il est bien entendu possible de bloquer la propagation des ACE à partir d'un niveau de sous-répertoire donné.

1.9 Ordre et priorité d'application des ACE

Au travers des définitions présentées ci-dessus, nous avons donc vu qu'il existe des ACE qui octroient des permissions, d'autres qui interdisent l'accès, qu'il existe des ACE positionnées explicitement sur un objet et d'autres héritées de conteneurs parents. Voyons maintenant quelles sont les permissions effectives pour un objet qui pourrait contenir plusieurs de ces entrées.

Les différents systèmes d'exploitation de Microsoft ne réordonnent pas systématiquement les ACE contenues dans les ACL. Cette charge est laissée au développeur qui doit s'assurer qu'il a positionné les ACE dans un ordre correct pour aboutir au résultat escompté. L'éditeur graphique de sécurité du système d'exploitation ordonne cependant correctement les permissions et prévient l'utilisateur lorsqu'il se rend compte que certaines ACE par exemples positionnées par des outils tiers ne sont pas dans le bon ordre.

L'ordre d'interprétation des permissions est ou devrait être le suivant :

- Accès refusés explicites (DENY ONLY).
- Accès autorisés explicites (ALLOW ONLY).
- Accès refusés hérités.
- Accès autorisés hérités.

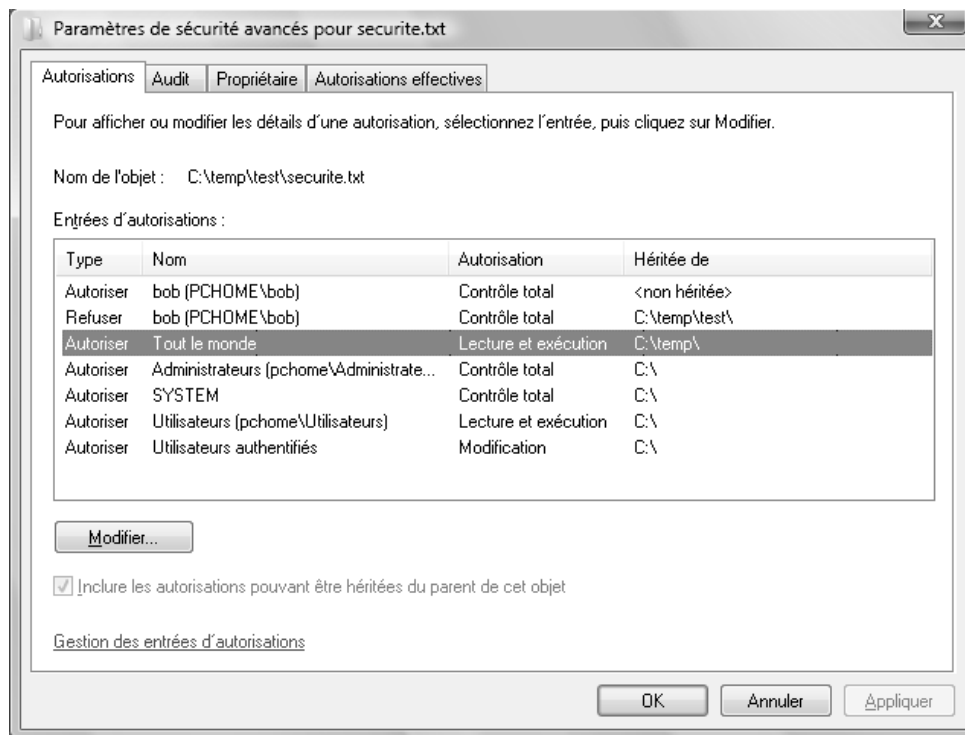
Lors du contrôle des permissions, un compte de sécurité qui a un accès refusé explicitement positionné (non hérité) sur un objet se verra refusé l'accès même si des droits lui sont accordés par héritage ou explicitement.

De même, un utilisateur qui a des droits d'accès explicitement positionnés sur un objet, se verra accorder l'accès même s'il a un refus positionné par héritage.

Finalement, les accès refusés par héritage sont prioritaires sur les autorisations par héritage.

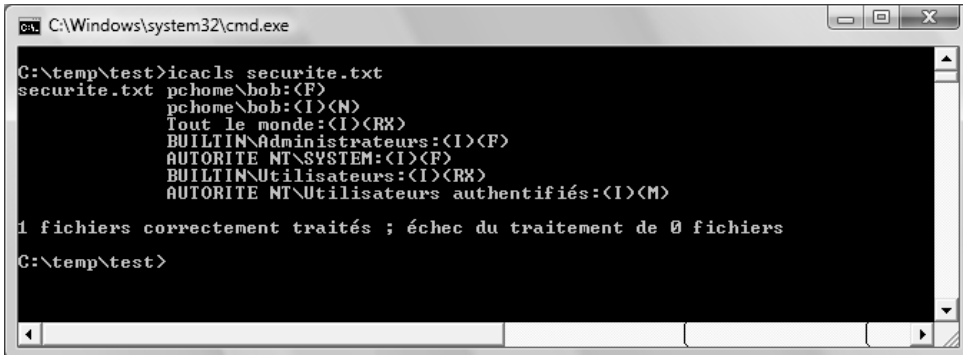
Pour illustrer ces propos, prenons un exemple. Soit un répertoire test sur lequel est positionnée une ACE de type refus pour l'utilisateur BOB et qui retire toutes les permissions (Contrôle total refusé). Dans ce répertoire, créons un fichier `securite.txt` et accordons à ce même utilisateur BOB un contrôle total.

Les permissions effectives sont illustrées dans la capture d'écran ci-dessous.



Il y a une ACE explicite pour BOB de type **Autoriser Contrôle total <non héritée>** en premier, suivi d'une ACE de type **Refuser Contrôle total <héritée>**. Dans les faits, l'utilisateur BOB a donc un contrôle total sur le fichier.

Les mêmes permissions sur le fichier affichées en ligne de commande à l'aide de l'utilitaire `icacls.exe` :



```
C:\Windows\system32\cmd.exe

C:\temp\test>icacls securite.txt
securite.txt pchome\bob:(F)
             pchome\bob:(I)<(N)
             Tout le monde:(I)<(RX)
             BUILTIN\Administrateurs:(I)<(F)
             AUTORITE NT\SYSTEM:(I)<(F)
             BUILTIN\Utilisateurs:(I)<(RX)
             AUTORITE NT\Utilisateurs authentifiés:(I)<(M)

1 fichiers correctement traités ; échec du traitement de 0 fichiers

C:\temp\test>
```

Bob a un contrôle total explicite représenté par (F) pour Full.

Bob a également un contrôle total refusé (N) et hérité (I).

Le résultat est un accès complet autorisé car les permissions explicites prévalent sur les refus hérités.

1.10 Création et application de permissions

Utilisons un autre exemple pour illustrer les différentes options de création de permissions. Par défaut, lorsqu'un utilisateur crée des permissions sur un répertoire, elles s'appliquent à ce répertoire et fichiers ainsi que par héritage à tous les sous-répertoires et fichiers enfants.

Créons un répertoire `test` et positionnons une ACE pour l'utilisateur Bob qui lui refuse toutes les permissions.