Hardening de Windows

Autor: Diego Bruno

Edición y Corrección: Lic. Cristian Borghello, MVP - CISSP

Fecha Publicación: 13 de febrero de 2010

Publicado en [Segu-Info]

SHARE AND NTFS PERMISSIONS

Windows tiene ambos tipos de permisos, los permisos a nivel de share fólder (Carpeta compartida) o los permisos NTFS (Permisos de file system) para proteger el acceso a los objetos que se contengan y se quieran proteger. Cada permiso se puede configurar para permitir (Allow) o denegar (Deny). Deny sobrescribe al permiso de Allow si ambos están seteados a un mismo nivel. Los permisos pueden estar seteados por archivos (excepto en los shares), por carpetas, por usuario y por grupo.

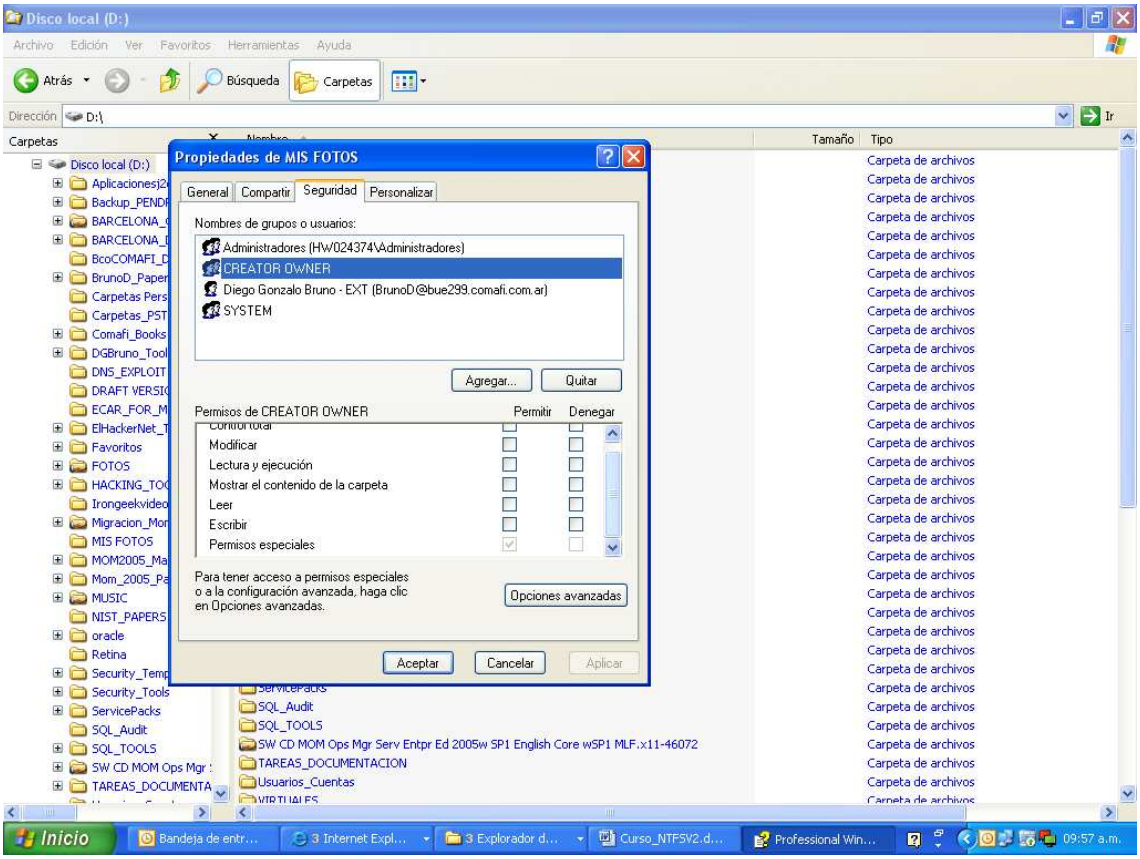
Contrariamente a la creencia popular, Microsoft Windows tiene uno de los mas granulares sistemas de seguridad en el mercado, que alguno me nombre cualquier otro sistema operativo de los popularmente conocidos que tenga 28 tipos de permisos diferentes para setear por archivos o por carpeta. Lo que Microsoft SI no tiene perfecto es la configuración de seguridad por default que trae y por el lado del usuario el no conocimiento de cómo trabaja el sistema NTFS. Esta sección describe en detalle los permisos NTFS y de File Share.

Share Permissions

Los permisos de share se aplican sólo cuando un recurso es contactado a través de una conexión de red, o incluso cuando es accedido localmente a través de Netbios o network neighborhood. Los share permissions están disponibles ya sea que el sistema de archivos sea NTFS o FAT, pero sólo pueden ser seteados a nivel de carpeta o impresora. Si se quisiera compartir un archivo o grupo de archivos se los debería primero poner una carpeta y luego compartirlo. Si no se usa el sistema de archivos NTFS los permisos de share son los únicos con los que contaríamos.

Existen tres tipos de permisos de share fólder, los cuales se muestran a continuación junto a la siguiente figura ilustrativa:

- Full Control
- Change
- Read



Cuando los permisos están seteados a nivel de carpeta los mismos son heredados por los objetos contenidos en la carpeta y así hacia abajo por default, pero los permisos de objeto pueden ser seteados para que difieran, y, si existieran permisos NTFS los mismos se aplicarían también.

La mayoría de las personas piensa que el permiso de READ sólo permite a un security principal a ver un objeto, cuando en realidad se puede hacer mucho más. El permiso de READ le permite a un security principal ver, copiar e imprimir (Si aplica) el recurso. Si el Share contiene un programa, el security principal puede ejecutar el programa. El permiso de READ también le permite al usuario a realizar traverse fólder para obtener otra carpeta que esté por debajo de la original.

Si un usuario tiene permiso de READ a un objeto, entonces ese usuario puede hacer una copia de dicho objeto en una nueva locación, y, en muchos casos obtener entonces el permiso de full control sobre la copia nueva, y esto porque sucede? Bueno sencillo porque cuando se copia un archivo a una nueva locación del filesystem, la misma en la mayoría de los casos tiene el permiso creator owner con full control, por lo cual el security principal que copia el archivo pasa a ser el creator owner de dicha copia y asignarse los permisos Full Control e inclusive una vez obtenido el permiso Full control poder cambiar el ownership del archivo.

La mayoría de los administradores se sorprenden que el permiso READ permite al usuario ganar el permiso full control sobre la copia, por lo tanto cuando se setean permisos de filesystem inclusive a través de GPO, siempre hay que tener muy en cuenta este punto.

El permiso de CHANGE permite al objeto relacionado, cómo cualquier subfolder o archivo, ver, agregar, copiar, imprimir, modificar y borrar. Hasta aca

sin sorpresas, muchas veces los administradores no están del todo seguros si el permiso de change permite eliminar o no el objeto, créanme, lo permite.

El permiso de Full Control le permite a un security principal modificar los permisos del share y a todos los archivos o carpetas contenidas en el mismo, pero el usuario debe ser miembro debe pertenecer al grupo power users o administrator para acceder a los permisos del share, con lo cual usualmente esto no es un problema. El permiso de Full Control en los share también permite al usuario tomar ownership del objeto si los permisos NTFS que estan por debajo también lo permiten.

Nuevos Shares

Un share puede ser creado por un administrador o un miembro del grupo power users haciendo solamente un clic derecho sobre la carpeta en cuestión o la impresora en cuestión. Luego una vez compartido puede asignar los permisos de Share y limitar o no las conexiones concurrentes al mismo. En los sistemas operativos de Workstation está limitada a 10 conexiones concurrentes por default. En las versiones home de Windows estas conexiones concurrentes están limitadas a un máximo de 5 conexiones.

Los usuarios finales regulares no pueden crear shares. Para poder compartir carpetas (Sharing) es necesario que los protocolos NetBios y de File and Printing Sharing estén habilitados.

Al grupo Everyone se le asigna en forma automática el permiso de READ en los nuevos shares que se crean en Windows XP y superiores. En Windows 2000 y NT, el grupo Everyone obtenía cuando se creaba un nuevo share el permiso de Full control, además en todo caso, los permisos NTFS que pudieran estar seteados en los archivos y subfolders contenidos, controlan los permisos efectivos asignados a un security principal. Otro mal creencia popular que se suele manejar es que los usuarios obtienen en forma automática el permiso de Full Control a todos los archivos y carpetas en en equipo Windows. Eso no fue nunca verdad, cómo dijimos recién más arriba, si fue verdad para los permisos de File Share en Windows NT y 2000, pero nunca para los permisos NTFS. Si no hay seteos de permisos NTFS, entonces lo normal por herencia es READ and EXECUTE. Si bien este permiso por defecto puede ser mucho más de lo que un administrador querría dar, es por lejos diferente al full control en todos los archivos y carpetas en un equipo Windows.

Nota: En un volumen formateado en FAT no hay obviamente seteos de permisos NTFS por lo cual el permiso Full Control de cualquier share se aplicaría.

Recomendaciones para un nuevo Share

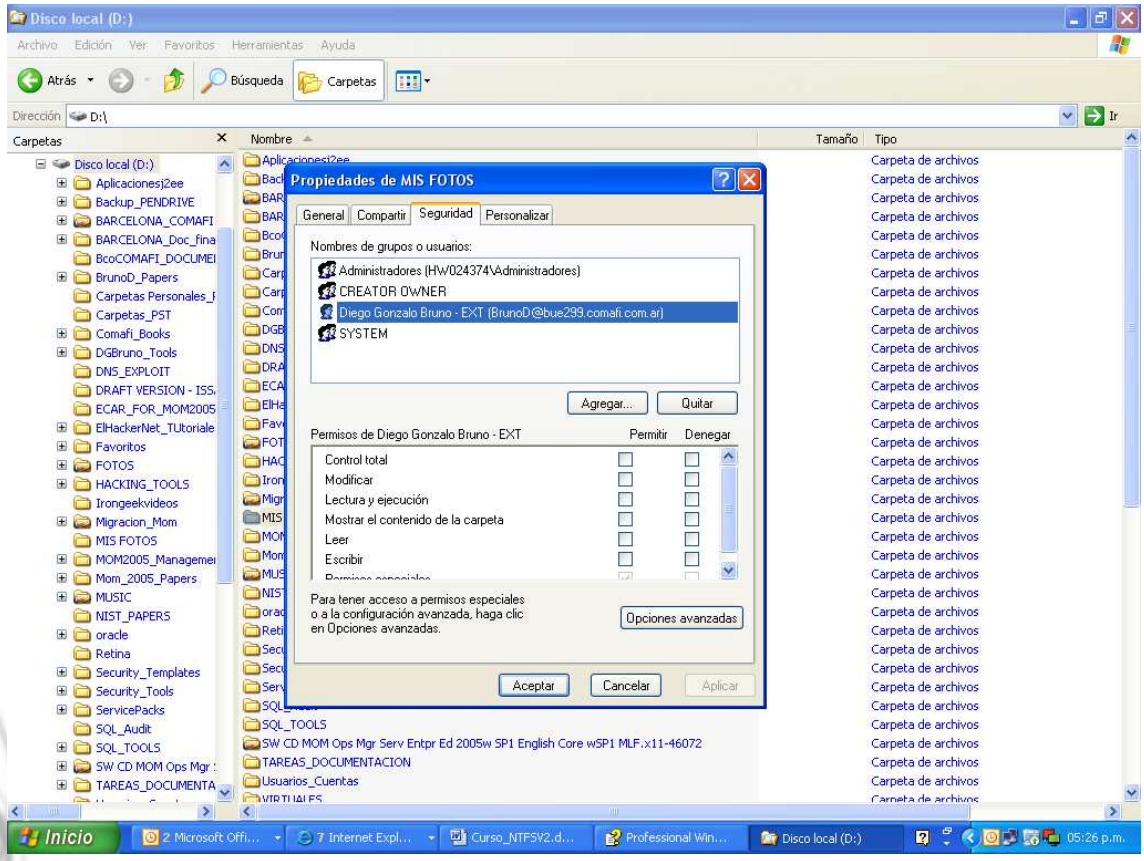
Cualquiera sea la locación en la cual se cree un nuevo share, los administradores deberían remover el grupo Everyone o eliminar en todo caso el permiso de READ y reemplazarlo por los mímimos privilegios necesarios. Se deberán de acuerdo al requerimiento realizar un ajuste a los mínimos privilegios necesarios tanto a nivel de permisos de share como de NTFS.

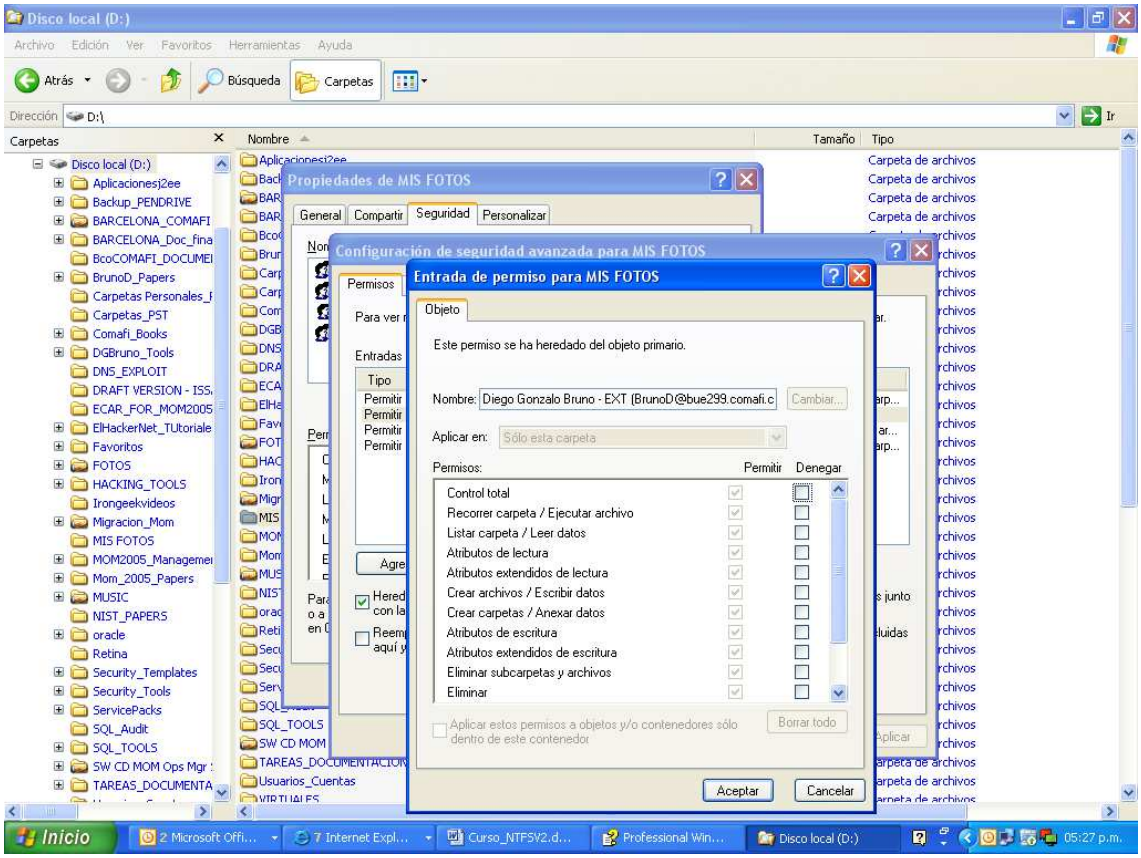
NTFS PERMISSIONS

Los permisos NTFS se utilizan para proteger archivos, claves de regedit, carpetas y otros objetos en un filesystem que este formateado en NTFS. Si el volumen está cómo NTFS los permisos se aplicarán ya sea que el acceso al recurso sea en forma local o remota. NTFS debería ser instalado para poder habilitar muchos de los features que trae incluyéndo:

- NTFS permissions — A nivel de archive y carpeta
- Auditing (Auditoría)
- Active Directory
- Disk quotas (Cuotas de disco)
- EFS encryption (Encripción de archivos y carpetas)
- Compression (Compresión)

Para modificar permisos NTFS, simplemente botón derecho sobre el objeto y elegimos *Properties* ⇨ *Security*. Existen 7 tipos de permisos NTFS de file and fólder creados por una combinación de 14 permisos especiales, los cuales ya se verán a continuación.





La sumarización de los siete permisos de archivo y carpeta de describen en la siguiente tabla a continuación:

Permiso	**Efecto**
Read	Ver, copiar, imprimir y renombrar archivos, carpetas y objeto. No puede levantar ejecutables de programas a menos que corras un archivo de batch que levante a su vez un ejecutable con permisos de lectura y escritura.
Puede leer los permisos del objeto, leer atributos de objeto y atributos extendidos (Ej., EFS, etc.). Listar archivos y subcarpetas en la carpeta.	
Write	Leer permisos, y además crear y sobrescribir archivos y carpetas.
List (Folders Only)	Permite visualizar los nombres de archivos y carpetas en el recurso.
Read and Execute	Leer permisos y la ejecución de archivos de programa.
Modify	Permite todos los permisos excepto la habilidad de hacer un Take Ownership y

	modificar permisos. Permite leer, eliminar, cambiar y sobrescribir archivos y carpetas.
Full Control	Lo mismo que modify pero con el plus de poder tomar posesión del archivo y setear permisos.
Special	Esta opción se utiliza para seleccionar algunas de las 14 opciones de permisos mas granulares que existen y con los cuales se puede crear un efecto de permiso diferente a los primeros 6. Además incluye el permiso de sincronización.

La siguiente tabla a continuación discute lo que permite cada uno de los 14 permisos especiales:

Permission	**Effect**
Traverse Folder/Execute File	Traverse Folder permite moverse a través de carpetas para alcanzar otros archivos o subcarpetas, inclusive cuando el security principal no tiene los permisos para hacer traverse fólder (Aplica sólo a carpetas).
Traverse folder toma efecto solo cuando el security principal no tiene granteado el privilegio de Bypass traverse checking (El cual el grupo tiene por default).	
Para Archivos: Execute File permite o deniega correr archivos de programa.	
Setear el permiso de Traverse Folder en un folder no implica que se setee automaticamente el permiso de Execute File en todos los archivos contenidos en el folder.	
List Folder/Read Data	List Folder permite ver nombres de archivos y subcarpetas dentro del folder.
List Folder solo afecta al contenido de ese folder en el cual se setea.	
Read Data permite o deniega, ver, copiar e imprimir archivos.	
Read Attributes	Permite o deniega a un security

	principal la capacidad de ver el atributo de un objeto (Ej. Read-only, System, Hidden, etc.)
Read Extended Attributes	Permite o deniega a un security principal la capacidad de poder ver atributos extendidos de un objeto (Ej. cómo EFS, Compression, etc.)
Create Files/Write Data	Create Files permite o deniega la creación de archivos dentro del folder. (Se aplica solo a carpetas)
Write Data permite o deniega hacer cambios a un archivo y sobrescribir contenido ya existente (Se aplica sólo a archivos)	
Create Folders/Append Data	Create Folders permite o deniega crear folders dentro del folder. (Se aplica solo a Folders.)
Append Data permite o deniega realizar cambios al final del archive pero no cambiar, eliminar o sobrescribir datos ya existentes (Se aplica a los archivos solamente).	
Write Attributes	Determina cuando un security principal puede escribir o modificar atributos standard (Ej, Read-only, System, Hidden, etc.) sobre archivos y carpetas. NO escribe el archive o la carpeta solo sus atributos.
Write Extended Attributes	Determina cuando un security principal puede escribir o modificar atributos extendidos (Ej. EFS, Compression, etc.) en archivos y carpetas.
Delete Subfolders and Files	Permite o deniega eliminar subfolders y archivos, incluso si el permiso de delete no ha sido granteado en el subfolder o archivo.
Delete	Permite o Deniega eliminar el archivo o la carpeta. Si no se tiene el permiso de delete en un archivo o carpeta, aún así se los puede eliminar si se tiene granteado el permiso de Delete Subfolders and Files en la carpeta padre o raíz.
Read Permissions	Permite o deniega leer los permisos del archive o carpeta, como Full Control,

	Read, and Write. No envuelve el poder leer el archivo en si mismo.
Change Permissions	Permite o Deniega modificar los permisos de archivos o carpetas, cómo Full Control, Read, and Write.
Take Ownership	Determina quién puede tomar posesión de un archivo o carpeta. Los Owners de dichos objetos pueden siempre tener Full Control, y sus permisos al archivo o carpeta no pueden ser removidos permanentemente a menos que se les quite el ownership de dicho objeto también.
Synchronize	No es manipulado por muchos administradores. Trabaja con la sincronización de items relacionados al multiprocesamiento de hilos y otros. No es relativo a nuestro tema de seguridad.

En la tabla que se muestra a continuación podemos ver un resumen de cómo los 14 permisos especiales realizan la sumatoria de los 7 permisos iniciales:

Permission	**Full Control**	**Modify**	**Read & Execute**	**List**	**Read**	**Write**
Traverse Folder/ Execute File	X	X	X	X		
List Folder/Read Data	X	X	X	X	X	
Read Attributes	X	X	X	X	X	
Read Extended Attributes	X	X	X	X	X	
Create Files/Write Data	X	X				X
Create Folders/ Append Data	X	X				X
Write Attributes	X	X				X

Write Extended Attributes	X	X				X
Delete Subfolders and Files	X					
Delete	X	X				
Read Permissions	X	X	X	X	X	X
Change Permissions	X					
Take Ownership	X					
Synchronize	X	X	X	X	X	X

INTERACCIONES DE PERMISOS INTERESANTES

Vamos a citar algunas interacciones interesantes de permisos que probablemente UD no conozca:

Read Permissions vs. Read Attributes vs. Read Data

La mayoría de los administradores tienen confusión con las etiquetas de READ en los permisos NTFS. El permiso de READ permite a un security principal (O a un programa que lo impersonalize) leer los permisos NTFS del archivo o carpeta y sólo los permisos. Sin ese permiso no se pueden leer los permisos del objeto, la solapa de seguridad queda Griselda. Los atributos de Read Attributes and Read Extended Attributes se refieren a cuando un security principal puede leer los atributos NTFS atachados a un objeto en particular. Dichos atributos atachados a un objeto cambian según el objeto. Los atributos de un archivo son cosas cómo seteos Read-only, EFS y archivos marcados como archivos de sistema. READ DATA es el permiso al cual la mayoría de la gente se refiere cuando le intentan dar a un security principal acceso de lectura al contenido de un objeto. Lo mismo aplica para los permisos de WRITE (Cómo por ejemplo Write Data, Write Attributes, Write Permissions), asi que a no confundirse!

Read vs. Read & Execute

El permiso de Read solo en si mismo no permite a un programa ser ejecutado (Cómo ya habíamos explicado anteriormente en la tabla de los permisos]) a menos que sea un script o un archivo de datos asociado con otro programa en el cual el security principal si tiene los permisos de Read y Execute. Por ejemplo, si en usuario sólo tiene el permiso de READ en una carpeta pero esa carpeta contiene archivos de scripts VBS, al hacer doble clic en el archivo se va a leer en memoria y será cargado el programa asociado para ejecutarlo, Wscript.exe. Esta regla se aplica a IIS también. Los archivos de script no necesitan tener permisos de Read & Execute, sólo de READ.

¿Qué significa realmente Full control?

Cuando al usuario se le dá el permiso de Full control sobre un objeto, el usuario podrá hacer take ownership del mismo y cambiar los permisos y lo mismo con cualquier otro usuario que tenga ese permiso. Es muy común, a veces por temas operacionales, otorgales a usuarios no administradores permisos de Full Control en shares, archivos, carpetas sus archivos de directorio home, etc. Cómo administrador uno no va a querer que los usuarios puedan cambiar los permisos de estos directorios pero pueden, e incluso remover los del administrador (Al menos temporalmente hasta que el mismo tenga que tomar el Ownership nuevamente del objeto y modificar los permisos).

¿Qué es File Traverse?

File transversal permite a un security principal ver o ejecutar un archivo en un fólder en el cual puede no tener permisos. El security principal o proceso debe conocer la ruta exacta de la carpeta y el archive para poder alcanzarlo. Por default el grupo Everyone tiene seteado el derecho de file and fólder transversal. Este privilegio no debería ser modificado a menos que Ud. Realmente sepa lo que

hace. El derecho de file traverse es útil cuando se quiere permitir a los usuarios ver contenidos de un subfolder sin ver la información contenida en el fólder raíz o padre.

A continuación se muestra una tabla extraída textual de una Technet de Microsoft la cual muestra los permisos de traversal que automáticamente son asignados al grupo Users para los directorios \Windows\Task y \Windows\Temp. Debido a que estos directorios son áreas que suelen agrupar normalmente una gran cantidad de usuarios, Windows trata de mantener a los usuarios no administradores de que puedan ver otros contenidos. La misma nos sirve sólo a modo de ejemplo y se la puede obviamente modificar de acuerdo a los permisos que se quieran setear en forma granular:

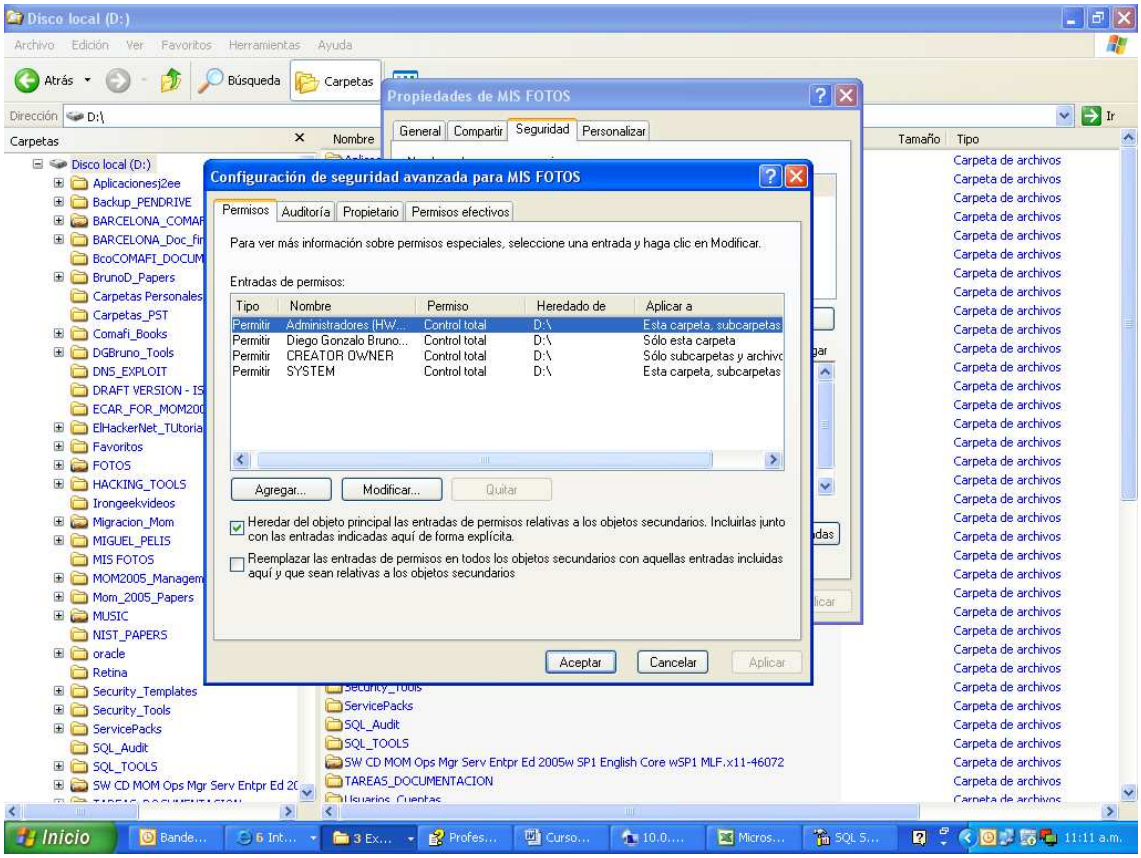
File or Folder	**Default Permissions (beyond Administrators, System, and Creator Owners having Full Control)**
%SystemDrive% (i.e., C:\ most often)	Users-Read & Execute, Create Folders-This folder and below
Users-Create Files-Subfolders only	
Everyone-Read & Execute-This folder only	
\Autoexec.bat	Users-Read & Execute
\Config.sys	Power Users-Modify
\Boot.ini	Power Users-Read & Execute
\Ntbootdd.sys	Users have no permissions to these files
\Ntdetect.com	
\Ntldr	
\Documents and Settings	Users, and Everyone-Read & Execute, Read, List Folder Contents
Power Users-all permissions but Full Control	
\Documents and Settings\Administrator	Administrator, Administrators, System-Full Control
\Documents and Settings\All Users	Users and Everyone-Read & Execute, Read, List Folder Contents
Power Users-Modify, Read & Execute, List Folder Contents, Read, and Write	
\Documents and Settings\Default User	Everyone- Read & Execute, Read, List Folder Contents (inherited)
No Creator Owner permissions	
\Documents and Settings\%UserName%	%Username%-Full Control No Creator Owner permissions
\Program Files	Power Users and Terminal Server User-

	Modify
\Program Files\Common Files	Users-Read & Execute, List Folder Contents
\Program Files\Windows Update	Power Users=Read & Execute, Write
\RECYCLER	Users-Read & Execute, List Folder Contents (inherited)
\System Volume Information	None for non-admin users
\Windows	Users-Read & Execute, List Folder Contents Power Users-Modify
In W2K, the Everyone group has Read & Execute	
\Windows\System32	Users-Read & Execute, List Folder Contents Power Users-Modify
\Windows\System32\Drivers	Users and Power Users-Read & Execute, List Folder Contents
\Windows\System32\Netmon\ Netmon.exe	Everyone, Users, and Power Users-Read & Execute
\Windows\System32\Spool\Printers	Users and Power Users-Traverse Folder/Execute File, Create Files, Read
\Windows\Tasks	Backup Operators-Traverse Folder\Execute Files, Read, Create Files
In XP and later, non-admin users can't create tasks. In W2K, non-admin users can create tasks.	
\Windows\Temp	Users-Traverse Folder\Execute Files, Create files and folders
Power Users-Modify	
\Windows\Regedit.exe	Users and Power Users-Read & Execute
\Windows\Repair	Users-Read & Execute
Power Users-Modify	
\Windows\Security	Power Users and Users-Read, Traverse Folder and Execute File
\Windows\System.ini	Users and Power Users-Read & Execute
\Windows\Prefetch	No access for non-admin users
\Windows\Help	Power Users-Modify
Users-Read & Execute	
Terminal Server User-Read & Execute, Write	
\Windows\System32\Config	Users and Power Users-Read & Execute

\Windows\System32\Catroot	
\Windows\System32\DHCP	
\Windows\System32\Drivers	
\Windows\System32\Logfiles	
\Windows\Application Compatibility Scripts	Interactive, Batch, Service-Read & Execute

HERENCIA DE PERMISOS

Por default, todos los archivos, carpetas y claves del registro heredan los permisos del contenedor padre a menos que se deshabilite la herencia. La herencia de permisos se puede deshabilitar en un archivo, carpeta o incluso clave de regedit, y se puede ajustar de una manera más granular a uno o más security principals. Muchas veces los administradores no saben que pueden deshabilitar o habilitar la herencia de permisos en forma selectiva archivo por archivo y hacerla aplicar por usuario inclusive. En la figura que se muestra a continuación de una DACL se puede ver la solapa Apply To field en cual se puede determinar la herencia al objeto que se la configura.



PERMISOS EFECTIVOS

Para refrescar entonces un poco todo lo aprendido hasta ahora veamos un poco en forma resumida lo que estuvimos viendo.

Cuando un usuario intenta acceder a un objeto protegido, todos los SID pertenecientes al usuario, cómo los SID de los grupos a los que el usuario pertenece son colocados dentro del token de acceso de seguridad (Security Access Token), junto a otra información complementaria también. El usuario o el proceso que lo impersonaliza entrega el token al proceso Windows security referente monitor, el cual es un subproceso manejado por el Local Security Authority (Lsass.exe). El Security Referente monitor compara los SID otorgados por el usuario con la DACL del objeto la cual dentro contiene los ACE (Access control Entries) con los SID de cuentas y los permisos que permiten o deniegan.

Todos los permisos asignados a todos los SID's del usuario son colectados todos juntos. Si existe un permiso de Deny seteado al mismo nivel que uno de Allow. El Deny normalmente gana. Si un permiso Full control-Deny gana, el usuario normalmente no puede acceder al objeto. Por defecto cuando los permisos NTFS y de Share se ven envueltos (Cómo por ejemplo un usuario conectándose a un recurso a través de un network share) el subsistema Security Referente Monitor (SRM) debe colectar todos los permisos de Share y de NTFS. Hasta este punto, todos los permisos son recolectados en forma conjunta y acumulativamente, pero los permisos de share y de NTFS son conservados por separado

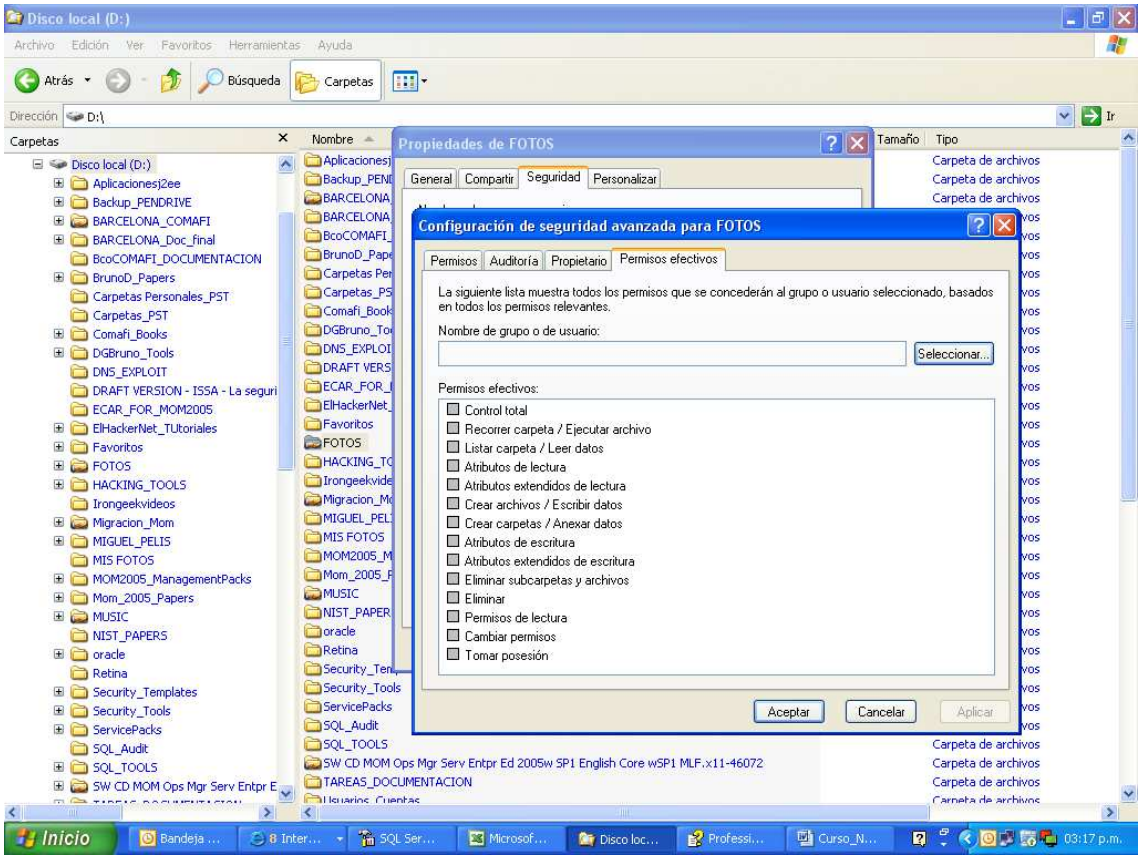
Los permisos efectivos del usuario entonces, son un set de los permisos de Share o de los permisos NTFS, cualquiera sea menos permisivo o más restrictivo. Este es otro punto muy importante ya que al momento de tratar de determinar los permisos efectivos, la salida es seteo entero de permisos, ya sea de Share o de NTFS. Un gran grupo de administradores cree que la salida de los permisos efectivos es el permiso menos permisivo lo cual es erróneo, en realidad es un seteo de los permisos menos permisivos ya sea de NTFS o de Share. En la siguiente figura se lo puede ver más claro:

PERMISO DE SHARE	PERMISO DE NTFS
Change	
Change	
Read	Modify
Read	
Read, Change	Read, Modify

En esta figura se puede ver que un usuario final termina con los permisos de share Read y Change y con los permisos NTFS de Read y modify. Cómo se puede observar en la misma figura, inicialmente el usuario tiene múltiples permisos de Read debido a que los permisos son acumulativos por todos los grupos a los que pertenece el usuario, además del cualquier otro permiso individual que se la haya podido otorgar a dicho usuario (Lo cual no es una buena práctica). Todos los permisos de share y de NTFS son acumulativos pero almacenados por separado. Entonces en este ejemplo que vimos son casi idénticos. Los permisos efectivos serían Read y Change/Modif..

SOLAPA DE PERMISOS EFECTIVOS

A partir de Windows XP y superiores Microsoft incorpora, cómo se puede ver en la figura más abajo, una nueva solapa para ver los permisos efectivos, con lo cual es más sencillo poder verlo. Para acceder a este nuevo feature hacemos botón derecho sobre el objeto, luego seleccionamos Propiedades, luego seguridad, luego Avanzada y finalmente la solapa "Permisos Efectivos".



Lamentablemente este feature nuevo sólo funciona para los permisos NTFS, o sea, no contempla a los permisos de Share que se pudieran sumar. Es una herramienta útil pero no siempre nos va a mostrar los verdaderos permisos efectivos, que podría un usuario tener en el caso de que existiera además permisos de Share sobre el objeto. Tenerlo en cuenta ante el análisis de seteos de permisos.