

Anwenden des Befehls »fileacl«

Für wen ist diese Anleitung gedacht?

Diese Anleitung ist für diejenigen gedacht, die größere (oder auch kleinere) Datenmengen unter Windows mit Berechtigungen versehen müssen. Hierfür gibt es ja bekanntlich unterschiedliche Möglichkeiten. Diese sind jedoch im ersten Absatz dieser Anleitung aufgeführt. Wir haben uns aus verschiedenen Gründen für den Einsatz von "fileacl" entschieden und für uns selbst eine kleine Dokumentation erstellt, die wir gerne mit anderen Anwendern teilen möchten. In diesem Sinne:

Viel Spaß beim "Berechtigten"

Das TDWsoft-Team

1 Hintergrundinformationen

Wie in anderen Bereichen auch führen bei der Vergabe von Berechtigungen unter Windows viele Wege nach Rom. Dennoch sind manche Wege kürzer und daher vorzuziehen. Der Windows Explorer bietet zwar die Möglichkeit einer Benutzeroberfläche, doch meist bietet ein Kommandozeilen-Werkzeug mehr Optionen und ist zudem wesentlich schneller in der Ausführung. Der Faktor Zeit spielt nämlich bei großen Datenmengen durchaus eine gewisse Rolle. Zudem könnte es nicht schaden, wenn man aus Gründen der Nachvollziehbarkeit nach dem Ausführen eines Befehls eine Logdatei vorliegen hätte.

Der wohl wichtigste Vorteil der Befehle gegenüber dem Windows Explorer ist jedoch die häufig angebotene Option einen Befehl beim Auftreten von "Fehlern" fortzusetzen. Der Windows Explorer bricht in aller Regel die Ausführung von Vorgängen immer dann ab, sobald der erste Fehler (z.B. eine Zugriffsverletzung aufgrund mangelnder Berechtigung) auftritt. Bei der Anwendung von Berechtigungen auf Dateien bzw. auf Verzeichnisse kann das sehr zeitraubend sein.

Für das Setzen (oder Auslesen) von Berechtigungen unter Windows kann der (unter XP bzw. Server 2003 standardmäßig vorhandene) Befehl "cacls" verwendet werden. Dieser Befehl hat jedoch gegenüber dem Werkzeug "fileacl" den Nachteil, dass man mit ihm z.B. nicht den Besitzer zuordnen kann.

Dieses Problem könnte man auch durch die Installation von "subinacl" lösen. Doch dessen Optionsvielfalt ist für sporadische Anwender eher verwirrend.

2 Anwendung

2.1 Betriebsarten

Der Befehl fileacl kennt (wie viele andere Befehle aus dem Berechtigungs-Bereich auch) zwei Betriebs-Modi:

- Anzeigen von Berechtigungen ("show mode")
- Ändern von Berechtigungen ("change mode")

Gibt man lediglich ein Datei- bzw. Verzeichnisname an, gibt fileacl die aktuelle vorhandenen Berechtigungen aus:

```
fileacl <Datei/Verzeichnis>
```

Für das Setzen von Berechtigungen muss man etwas mehr Optionen angeben:

```
fileacl <Datei/Verzeichnis> <Berechtigungsoptionen> <Ausführungsoptionen>
```

z.B.

```
fileacl c:\tmp /o administrator /files /sub
```

Obiger Befehl ändert den Besitzer [/o für "owner"] aller Dateien [/files] und aller Unterverzeichnisse [/sub] unterhalb des Verzeichnisses "c:\tmp" auf "Administrator" (wobei das Verzeichnis "c:\tmp" selbst auch betroffen ist).

2.2 Berechtigungsoptionen

Neben dem Setzen des Besitzes gibt es zwei wichtige Optionen für die Vergabe von Berechtigungen:

- /g (g steht für "Grant" was soviel heisst, wie "zulassen")
- /s (s steht für "Set", was in diesem Zusammenhang "Er-Setzen" bedeutet)

Die Syntax ist für beide Optionen gleich:

- /g <Benutzer>:<Berechtigung>
- /s <Benutzer>:<Berechtigung>

Während die Option "/g" die nachfolgend aufgeführte Berechtigung zu den bestehenden Berechtigungen des angegebenen Besitzers **hinzufügt**, wird durch /s die Berechtigung des Benutzers **ersetzt**.

Ein einfaches Beispiel soll den Unterschied verdeutlichen:

```
fileacl c:\tmp /g maier:rx /files /sub
```

Die Anwendung von /g hängt an die bereits bestehenden Berechtigungen, welche "maier" hat die Berechtigung für das Lesen und Ausführen an. Angenommen der "Benutzer" "maier" hatte keinerlei Berechtigungen auf Dateien, dann hat er nach der Ausführung des Befehls zumindest einige Basisberechtigungen bekommen. Hatte der Benutzer jedoch bereits Berechtigungen und man möchte diese **einschränken**, dann ist die Option "/s" anzuwenden. Denn die Option "/g" entzieht dem Benutzer "maier" ein eventuell vorhandenes Schreibrecht auf eine Datei oder ein Verzeichnis **nicht**.

Dagegen führt die Ausführung des Befehls

```
fileacl c:\tmp /s maier:rx /files /sub
```

dazu, dass der Benutzer "maier" auf alle Dateien und Unterverzeichnisse des Verzeichnisses "c:\tmp" zukünftig nur noch Lese- bzw. Ausführungsberechtigung haben wird. Denn alle Berechtigungen, welche der Benutzer "maier" vor der Ausführung des Befehls hatte, werden durch die Verwendung der Option "/s" **ersetzt**.

Die Optionen "/g" bzw. "/s" können mehrfach hintereinander angewandt werden, um mit einer Ausführung gleich mehrere Berechtigungen auf einmal zu setzen:

```
fileacl c:\tmp /s maier:rx /s administrator:f /files /sub
```

Folgende Berechtigungen können gesetzt werden:

Berechtigung	Bedeutung
F	Vollzugriff
C	Ändern
R	Lesen
X	Ausführen (Dateien) / in das Verzeichnis wechseln (Verzeichnisse)
W	Schreiben
D	Löschen
O	Besitz übernehmen bzw. übergeben
P	Lesen von Berechtigungen

Normalerweise kommt man mit den ersten drei Standard-Berechtigungen aus der Tabelle zurecht. Die in der Tabelle grau hinterlegten Berechtigungen sind so genannte Spezial-Berechtigungen, welche man für komplexere Berechtigungs-Konzepte einsetzen kann.

Beispiele

```
fileacl c:\tmp /s everyone:r /files /sub
```

Jeder soll sowohl die Dateien lesen, als auch in (Unter-)Verzeichnisse wechseln können.

```
fileacl c:\tmp /s everyone:c /files /sub
```

Jeder soll ändern können (beinhaltet auch das Löschen von Dateien bzw. Verzeichnissen)

Der Unterschied zwischen "Ändern" und "Vollzugriff" besteht in der Übernahme des Besitzes und dem Ändern von Berechtigungen.

2.3 Ausführungsoptionen

Während die Berechtigungsoptionen festlegen WER WELCHE Berechtigungen bekommt, legen die Ausführungsoptionen fest, WIE der Befehl ausgeführt werden soll (Anwendungstiefe bei Verzeichnissen, mit Debugging oder ohne, nur Dateien oder nur Verzeichnisse).

Option	Bedeutung
/sub[:n]	Anwenden des Befehls auf Unterverzeichnisse bis Ebene [n] n steht in eckigen Klammern was soviel bedeutet wie "diese Option kann angegeben werden, sie muss aber nicht". D.h. ohne die Angabe der Anwendungstiefe schreibt man /sub . Wenn der Befehl auf eine Verzeichnisebene unter dem angegebenen Verzeichnis angewandt werden soll, schreibt man /sub:1 WICHTIG: Die Angabe von einer Verzeichnisebene wird bei der Verwendung der Option "/nodirs" ignoriert!
/files	Die Berechtigungen der in den Unterverzeichnissen enthaltenen Dateien ebenfalls ändern (ohne /files werden lediglich die Verzeichnisse berücksichtigt)
/nodirs	Kann angewandt werden, wenn nur Dateien von der Aktion betroffen sein sollen. Diese Option enthält bereits den Schalter "/files". Ihre Anwendung erspart jedoch nicht die Anwendung von "/sub", wenn man möchte, dass alle Dateien unterhalb des angegebenen Verzeichnisses bearbeitet werden sollen.
/replace	Ersetzt die an den Objekten vorhanden Zugriffsberechtigungen
/noroot	Nicht auf das angegebene Verzeichnis anwenden, sondern auf die untergeordneten Verzeichnisse.
/verbose	Sorgt für die Ausgabe von etwas mehr Informationen
/quote	Maskiert die Datei- bzw. Verzeichnisnamen mit Hochkommata (wird erfahrungsgemäß nicht benötigt)
/silent	Keinerlei Ausgabe auf STDOUT produzieren
/line	Alle Berechtigungen eines Objektes innerhalb einer Zeile ausgeben. Diese Option ist nützlich, um den momentanen Stand der Berechtigungen zu dokumentieren, da man das Ergebnis (nach der Umleitung in eine Datei) z.B. in Excel importieren kann.*
/owner	Zeigt den Besitzer des Objekts an*
/batch	Die wohl mächtigste Option, wenn man sich vor dem Ändern von Berechtigungen eine Rückfahrkarte erstellen möchte. Man leitet das Ergebnis in eine Datei um und schon hat man eine Batchdatei, mit der man die Berechtigungen wieder jederzeit so herstellen kann, wie sie waren. Man sollte für eine Komplettsicherung jedoch die Optionen "/sub" bzw. "/files" nicht vergessen.*
/noinherited	Die von oben vererbten Berechtigungen nicht anzeigen
/batchreal	Ähnlich wie "/batch", jedoch erzeugt diese Option zusätzliche Befehle für das angegebene Verzeichnis

*Die grau hinterlegten Optionen sind nur für die Anzeige von Berechtigungen relevant

Beispiele

```
fileacl c:\daten /files /sub /batch > c:\tmp\rollback.cmd
```

Erklärung:

Erstellt die Batchdatei "c:\tmp\rollback.cmd", mit der die Berechtigungen auf das Verzeichnis "c:\daten" (inklusive aller Unterverzeichnisse und Dateien) wiederhergestellt werden können.

```
fileacl c:\daten /s Jeder:c /s administrator:f /nodirs /verbose
```

Erklärung:

Setze die Berechtigungen auf **alle Dateien** innerhalb von "c:\daten" [/nodirs] so, dass "Jeder" ändern darf und der Administrator Vollzugriff besitzt.

```
fileacl c:\daten /o administrator /nodirs /verbose
```

Erklärung:

Setze den Besitzer für alle Dateien innerhalb von "c:\daten" auf "Administrator". Die Aktion betrifft lediglich die Dateien innerhalb des angegebenen Verzeichnisses. Denn die Option "/sub" wurde nicht angegeben. Würde sie hinzugefügt, wären alle Dateien betroffen

```
fileacl c:\daten /s jeder:f /sub /replace /verbose
```

Erklärung:

(Er-)Setze [/replace] die Berechtigungen **aller Unterordner** [denn Option "/files" fehlt!] unterhalb von "c:\daten" so, dass jeder Vollzugriff auf diese Ordner hat (ein "c" für "Change", also ändern dürfte auch reichen).

```
fileacl c:\daten /s jeder:r /sub /nodirs /replace /verbose
```

Erklärung:

(Er-)Setze [/replace] die Berechtigungen **aller Dateien** [/sub mit /nodirs] unterhalb von "c:\daten", so dass "Jeder" nur noch Leseberechtigung besitzt.

Zusammenfassung

Ziel	Option[en]
Alle Dateien innerhalb des angegebenen Verzeichnisses (nicht jedoch die Dateien in den Unterverzeichnissen und nicht die Unterverzeichnisse selbst)	/nodirs
Alle Dateien und alle Unterverzeichnisse	/sub /files
Nur die Verzeichnisse innerhalb des angegebenen Verzeichnisses und eine Ebene tiefer (und keine Dateien)	/sub:2
Alle Dateien unterhalb des angegebenen Verzeichnisses (und keine Unterverzeichnisse)	/sub /nodirs
Nur das angegebene Verzeichnis (oder die angegebene Datei) selbst	Keine Option notwendig (außer z.B. "/verbose)
Ersetzen der vorhandenen Berechtigungen (muss zusätzlich zu den obigen Optionen angegeben werden)	/replace

3 Für Fortgeschrittene

3.1 Grundlagen der Vererbung

Ein sehr wichtiger Aspekt wurde bisher aus Gründen der Verständlichkeit ignoriert: Die Vererbung. Die Vererbung funktioniert denkbar einfach:

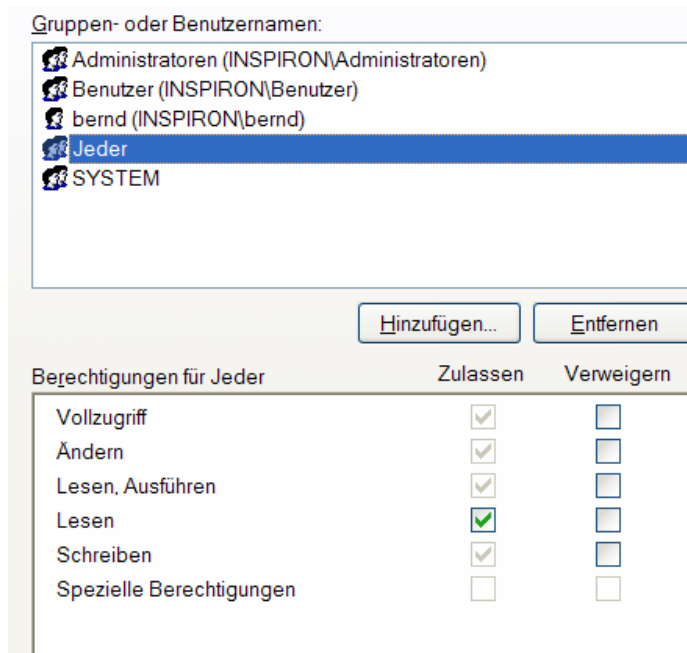
Eine Datei oder ein Verzeichnis erben die Berechtigungen des übergeordneten Verzeichnisses.

Ist also die Vererbung "eingeschaltet" und es werden die Berechtigungen am übergeordneten Verzeichnis geändert, ändern sich die Berechtigungen der Datei ebenfalls.

Doch damit nicht genug. Eine Datei oder ein Verzeichnis kann zwei Arten von Berechtigungen haben:

- Direkt angebrachte Berechtigungen
- Ererbte Berechtigungen

Die ererbten Berechtigungen erkennt man im Eigenschaften-Menü (Windows-Explorer) daran, dass die ererbten Berechtigungen ausgegraut dargestellt sind:



Unter "Erweitert" bietet die Spalte "Geerbt von" tiefere Einblicke in die Herkunft der Berechtigungen:

Berechtigungseinträge:			
Typ	Name	Berechtigung	Geerbt von
Zulassen	Jeder	Lesen	<nicht geerbt>
Zulassen	Jeder	Vollzugriff	D:\tmp\rechte\Untergeordnet

Bei der Auflistung der Berechtigungen über die Kommandozeile wird die Vererbung mit der Kennung [I] gekennzeichnet:

```
D:\daten;Jeder:F[I]
```

Fazit

Im vorliegenden Fall besitzt die Datei zwar die Berechtigung "Lesen" für "Jeder", aber von oben herab wurde der Vollzugriff für "Jeder" ererbt.

**Die Summe der Berechtigungen für die betroffene Datei bzw. für das betroffene Verzeichnis lautet:
Jeder hat Vollzugriff**

Mit den bisher gezeigten Optionen können nur die direkt angebrachten Berechtigungen manipuliert werden. Die ererbten Berechtigungen bleiben unangetastet (auch dann, wenn man die Option "/replace" verwendet).

Beispiel

Angenommen der Benutzer "maier" besitzt auf das übergeordnete Verzeichnis Vollzugriff und die Vererbung ist eingeschaltet. Dann würde der folgende Befehl seine Berechtigungen überhaupt nicht einschränken:

```
fileacl c:\daten /s maier:r /sub /files /replace /verbose
```

So lange der Benutzer "maier" auf das Verzeichnis "c:\daten" Vollzugriff hat und die Vererbung auf Dateien bzw. Verzeichnisse eingeschaltet ist, hat der Benutzer auf alle Dateien und Unterordner Vollzugriff.

3.2 Arbeiten mit der Vererbung

Für das Ein- bzw. Ausschalten der Vererbung kennt fileacl die folgenden Optionen:

- /protect (schütze bzw. "immunisiere" die Datei bzw. das Verzeichnis gegen Berechtigungen von oben)
- /inherit (erbe die Berechtigungen von oben)

Im einfachsten Fall der Vererbung (wenn es nur sie gibt) genügt es, die Berechtigungen des obersten Verzeichnisses zu ändern und schon "schlagen" die Berechtigungen nach unten durch. Erreichen kann man dies mit dem folgenden Befehl:

```
fileacl c:\daten /inherit /replace /sub /files
```

Erklärung:

Die Vererbung wird mit der Option "/inherit" veranlasst, während die Option "/sub" für die Bearbeitung aller Unterverzeichnisse sorgt. Schließlich teilt die Option "/files" dem Befehl mit, dass er auch die Dateien in den Unterverzeichnissen berücksichtigen soll. Die Option "/replace" sorgt für das Ersetzen aller bisher vorhandenen Berechtigungen. Somit sind alle Berechtigungen an den betroffenen Dateien bzw. Verzeichnissen vom Stammverzeichnis "c:\daten" ererbt, so dass keinerlei direkten Berechtigungen mehr an den Objekten vorhanden sind.

Das "Abhängen" eines Verzeichnisses von der Vererbung kann mit der Option "/protect" erreicht werden. Hierbei wird das Verzeichnis von der Vererbung ausgeschlossen und die momentan vorhandenen Berechtigungen werden kopiert. D.h. die "von oben" ererbten Berechtigungen werden zu direkten Berechtigungen umgewandelt.

Angenommen das Verzeichnis "c:\daten\ablage" hat sich folgende Berechtigungen von oben ererbt:

```
Jeder:R / Administrator:F
```

Dann besitzt das Verzeichnis nach der Anwendung der Option "/protect" immer noch genau dieselben Berechtigungen. Der Unterschied zu vorher besteht darin, dass bei einer Änderung der Berechtigungen am übergeordneten Verzeichnis ("c:\daten") sich bei "c:\daten\ablage" nichts mehr ändert. Denn es wurde von der Vererbung ausgeschlossen:

```
Fileacl c:\daten /protect
```

Möchte man nun für die Dateien eines Verzeichnisses erreichen, dass diese wirklich für alle nur lesbar sind, so kann man dies damit erreichen, dass man das für die Vererbung maßgebliche Verzeichnis für alle mit der Berechtigung "lesen" versieht. Allerdings führt die Vergabe der Berechtigung "lesen" auf Verzeichnisse dazu, dass niemand mehr neue Dateien erzeugen kann. Daher bietet sich in diesem Fall das "abhängen" der Dateien von der Vererbung an:

```
fileacl c:\daten /s jeder:r /s administrator:F /nodirs /protect /replace /verbose
```

Erklärung:

Dieser Befehl hängt alle Dateien (/nodirs ohne /sub) innerhalb des Verzeichnisses "c:\daten" von der Vererbung ab (/protect). Zusätzlich sorgt die Option "/replace" für das Entfernen aller vorherigen Berechtigungen)

Das Ergebnis:

Berechtigungseinträge:

Typ	Name	Berechtigung	Geerbt von
Zulassen	Administrator (INSPIRON\Administrator)	Vollzugriff	<nicht geerbt>
Zulassen	Jeder	Lesen	<nicht geerbt>

Sollten alle Dateien unterhalb von "c:\daten" so bearbeitet werden, muss man folgenden Befehl verwenden:

```
fileacl c:\daten /s jeder:r /s administrator:F /nodirs /protect /replace /verbose /sub
```

Nun könnte man noch die (Unter-)Verzeichnisse von "c:\daten" auf die Vererbung einstellen und für jeden das Schreibrecht vergeben, damit die bestehenden Dateien geschützt und neue Dateien erstellt werden können:

```
fileacl c:\daten /s jeder:f /inherit /sub /replace /verbose
```

3.3 Feintuning

Wie beschrieben, erhalten neu zu erstellende Dateien bzw. Verzeichnisse bei eingeschalteter (Standard-)Vererbung immer die Berechtigungen des übergeordneten Verzeichnisses. Genauer gesagt: Sie erhalten die Berechtigungen des für die Vererbung maßgeblichen Verzeichnisses. Denn: Das übergeordnete Verzeichnis kann ja seine Berechtigungen wieder von seinem übergeordneten Verzeichnis erhalten haben.

Es gibt jedoch Situationen, bei denen die neu erstellten Dateien bzw. Verzeichnisse zu viele Berechtigungen "von oben" vererbt bekommen.

Angenommen es soll ein eine Art Austauschverzeichnis für viele Anwender eingerichtet werden. D.h. eine große Anwenderzahl soll Schreibrecht auf dieses Verzeichnis haben. Was allerdings noch lange nicht heißt, dass alle Anwender auf die neu erstellten Dateien bzw. Unterordner ebenfalls Schreibrecht besitzen sollen. Jeder Anwender sollte die von den Kollegen erstellten Objekte lediglich lesen können.

Im Klartext:

Ordner: Jeder (besser: Domänen-Benutzer): Ändern – Administrator: Vollzugriff

(neu zu erstellende) Dateien und Unterordner: Jeder: Lesen – Ersteller-Besitzer: Vollzugriff – Administrator: Vollzugriff

Im vorliegenden Fall unterscheiden sich die Berechtigungen für neu zu erstellende Ordner bzw. Dateien von den Berechtigungen des übergeordneten Ordners. Mit anderen Worten: Es muss etwas anderes nach unten vererbt werden.

Für das Einstellen solcher Berechtigungen bietet Fileacl die passenden Optionen:

```
/s <Anwender>:<A>/<B>/<C>
```

Man gibt (nach der Angabe des Benutzerkontos) anstelle einer einzelnen Berechtigung drei Berechtigungen an, welche mit einem Schrägstrich voneinander getrennt werden.

Die angegebenen Einzelberechtigungen werden folgendermaßen angewandt:

- <A>: Zugriffsberechtigung auf das Verzeichnis selbst
- : Zugriffsberechtigung für **zukünftig** zu erstellende Dateien
- <C>: Zugriffsberechtigung für **zukünftig** zu erstellende Verzeichnisse

Möchte man die Berechtigung für einen Bereich nicht setzen, so kann man anstelle einer Berechtigung einfach "U" (für "unspecified") angeben.

Im genannten Beispiel müsste man also folgendermaßen vorgehen, um die dort aufgeführten Anforderungen erfüllen zu können (der Verständlichkeit halber wurde das Setzen der Berechtigungen auf Einzelschritte unterteilt, obwohl alles in einem Schritt ausgeführt werden könnte:

1. Schritt: Verzeichnis von der Vererbung abhängen

```
fileacl h:\austausch /protect
```

2. Schritt: Setzen der Berechtigungen auf das Verzeichnis selbst

```
fileacl h:\austausch /s jeder:C/U/U /s administrator:F/U/U
```

"/U/U" bedeutet: Hier werden keine Angaben für neu zu erstellende Dateien () und neu zu erstellende Ordner (<C>) gemacht. Es werden nur die Berechtigungen auf das Verzeichnis selbst (nämlich für <A> gemacht).

In der Eigenschaften-Ansicht des Explorers sieht diese Einstellung folgendermaßen aus:

Berechtigungseinträge:

Typ	Name	Berechtigung	Geerbt von	Übernehmen für
Zulass...	Administrator ...	Vollzugriff	<nicht geerbt>	Nur diesen Ordner
Zulass...	Jeder	Ändern	<nicht geerbt>	Nur diesen Ordner

Zu beachten ist die Spalte "Übernehmen für". Dort wird angezeigt, welchen Geltungsbereich die aufgeführten Berechtigungen besitzen. Wenn man die Berechtigungen im Sicherheits-Dialog unter "Erweitert" bearbeitet (Doppelklick auf eine angezeigte Berechtigung ausführen, bekommt man ein Menü mit dessen Hilfe man den Geltungsbereich einstellen kann. Dieses Menü bietet nämlich (neben der sehr genauen Einstellmöglichkeit für Berechtigungen) das Einstellen des Geltungsbereichs der gewählten Berechtigung an (übernehmen für):

Objekt

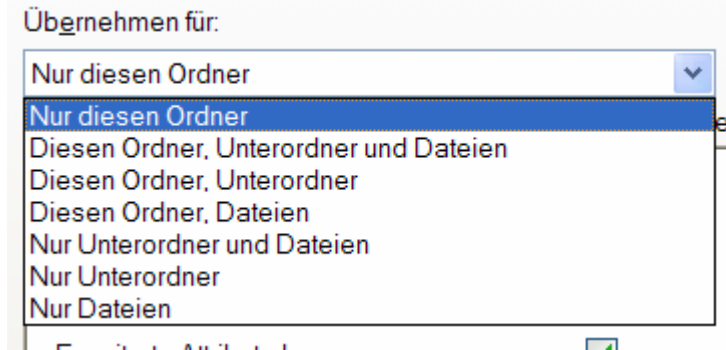
Name: Administrator (INSPIRON\Administrator) Ändern...

Übernehmen für:
 Nur diesen Ordner ▼

Berechtigungen:

	Zulassen	Verweigern
Vollzugriff	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Ordner durchsuchen / Datei ausführen	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Ordner auflisten / Daten lesen	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Die unter "Übernehmen für" angebotenen Optionen können allesamt auch mit Fileacl umgesetzt werden:



Geltungsbereich	Option[en]
Nur diesen Ordner	<A>/U/U
Diesen Ordner, Unterordner und Dateien (volle Vererbung)	<A>//<C>
Diesen Ordner, Unterordner	<A>/U/<C>
Diesen Ordner, Dateien	<A>//U
Nur Unterordner und Dateien	U//<C>
Nur Unterordner	U/U/<C>
Nur Dateien	U//U

3. Schritt: Setzen der Berechtigungen für zukünftig zu erstellende Dateien bzw. Verzeichnisse

```
fileacl h:\austausch /s jeder:U/R/R /s administrator:U/F/F /ersteller-besitzer:U/F/F
```

Wie man erkennen kann, ist also auch eine Unterscheidung zwischen neu zu erstellenden Dateien und neu zu erstellenden Verzeichnissen möglich. So könnte man z.B. festlegen, dass "Jeder" lediglich auf neu zu erstellende Dateien lesend zugreifen darf, während der Zugriff auf Verzeichnisse dem Ersteller nur dem Ersteller gewährt wird. In diesem Fall sähe die Kombination der Berechtigungen folgendermaßen aus:

```
fileacl h:\austausch /s jeder:U/R/U /s administrator:U/F/F /ersteller-besitzer:U/F/F
```

Durch die Anwendung des letzten "/U" bei den Berechtigungen für "Jeder" kann die oben aufgeführte Anforderung umgesetzt werden.

Stellt sich die Frage, was passieren würde, wenn man die Vererbung ausschalten würde und alle Berechtigungen nur auf das Verzeichnis setzen würde. D.h. man gäbe keine Berechtigungen für neu zu erstellende Dateien bzw. Verzeichnisse an:

```
fileacl d:\test /s jeder:F/U/U /protect /replace
```

Wenn man (nur) auf das Verzeichnis "d:\test" (F/U/U) Vollzugriff für alle gibt, so gibt es trotzdem eine Grundberechtigung auf neu erstellte Verzeichnisse bzw. Dateien. Erstellt ein Anwender in einem solchen Verzeichnis eine Datei bzw. ein Verzeichnis, so werden zwei Grundberechtigungen vergeben: Sowohl der Anwender, als auch das vordefinierte Konto "System" bekommen Vollzugriff auf die erstellten Objekte.