Robocopy

- Daten kopieren mit NTFS Berechtigungen mit robocopy

Daten kopieren mit NTFS Berechtigungen mit robocopy

Beschreibung:

Will man von einem Server auf den anderen Verzeichnisse/Freigaben übertragen mittels rechten. Dann ist robocopy das ideale Werkzeug was in jeder Windows installation von ahus aus, dabei ist Das kann nämlich alle Rechte übertragen.

Parameter von Robocopy:

Parameter	Bedeutung
`/E`	Alle Unterordner (auch leere)
`/SEC`	Kopiert NTFS-Berechtigungen (DACL)
`/COPYALL`	Alles: Daten, Attribute, Timestamps, Security, Owner, Auditing
`/R:1 /W:1`	Nur 1 Retry, 1 Sekunde warten (sonst hängt er ewig bei Fehlern)
`/LOG:C:\robocopy.log`	Optional: Log-Datei
`/MT:16`	Optional: Multithreaded (schneller bei vielen Dateien)

Kopiert von Server B auf Server B:
Wenn der Befehl auf Server B ausgeführt wird, was ja meistens , Sinn macht, dann ist das Zielverzeichnis natürlich ein lokales Verzeichnis.
Beispiel:

```
robocopy "\\quellserver\c$\Pfad" "\\zielserver\c$\Pfad" /E /COPYALL /R:1 /W:1
```

Unglücklicherweise, kopiert robocopy immer nur den Inhalt des Quellverzeichnisses
Das heißt robocopy sieht das Quellverzeichnis so:

```
robocopy "\\quellserver\c$\Pfad\*.*"
```

Man möchte ja aber gern den Pfad samt berechtigungen kopieren.
Robocopy legt Zielverzeichnisse an die nicht bestehen, aber unglücklicherweise, hat das Verzeichnis dann nicht das recht wie auf dem Server. Dazu holhen wir uns wenn der kopiervorgang fertig ist die rechte vom alten sevrver und kopieren diese auf das neue Verzeichnis aufm Zielserver.

Es ist übrigend egal, in welcherer Reihenfolge ob wir erst das Verzeichnis erstellen dann die Rechte kopieren und dann erst robocopy
losrennen lassen oder ob wir erst robocopy und dann die Rechte kopieren.
Ich nehme variante zwei, erspart mir das Verzeichnis erstellen.

Befehl um rechte zu kopieren, die rechte werden in einer variabeln $acl gespeichert mit dem Befehl Get und wieder aus der Variablen mit Set in das Zielverzeichnis geschrieben.

```
powershell -Command "$acl = Get-Acl '\\quellserver\c$\Pfad'; Set-Acl '\\zielserver\c$\Pfad' $acl"
```

Hier ein Script das Quelle Ziel setzt und zum Schluss die Rechte anpasst.
Wir brauchen nur Quelle und Ziel anpassen.
Einfach als bat Datei

Ein kleines Script:

```
@echo off
set "QUELLE=\\quellserver\c$\Daten\Projekte"
set "ZIEL=\\zielserver\c$\Backup\Projekte"

robocopy "%QUELLE%" "%ZIEL%" /E /COPYALL /R:1 /W:1

powershell -Command "$acl = Get-Acl '%QUELLE%'; Set-Acl '%ZIEL%' $acl"

echo Fertig - inkl. Berechtigungen des Wurzelordners!
```