

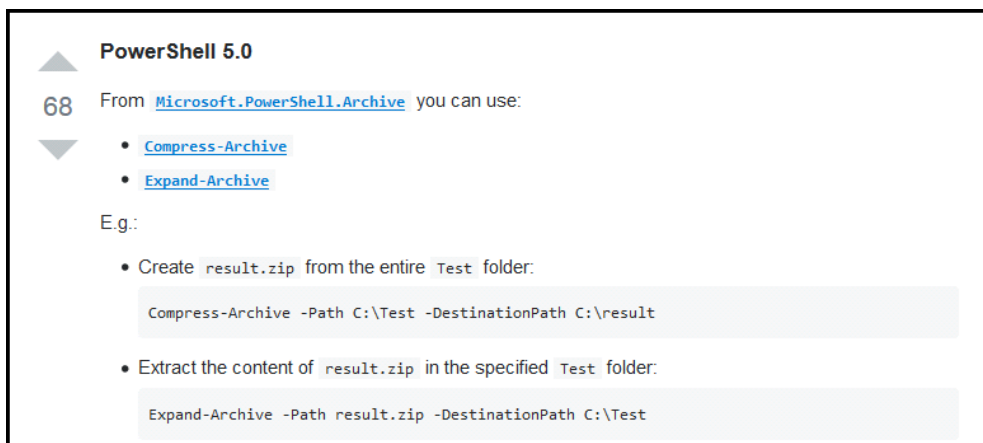
Kann man über die PowerShell auch zippen?

Hin und wieder gibt es Anwendungen, die so gestrickt sind, daß sie jeden Tag Tausende kleinster Dateien, z.B. Log- und Handshake-Dateien, erzeugen. Angeblich braucht die Anwendung diese Dateien, um bestimmte Arbeitsschritte lücklos nachweisen/dokumentieren zu können. Bereits direkt nach ihrer Erstellung müßten diese Dateien eigentlich auf ein Archiv ausgelagert werden; doch da dies in der Regel extra kostet, wird dieses 'Archiv' lieber dem Backup zugeschlagen.

Bereits nach wenigen Wochen ist das allerdings nicht mehr lustig, denn eine solche Verzeichnis-Struktur ist praktisch nicht mehr sicherbar - wegen der vielen Dateien und dem schlechten Verhältnis von Nutz- zu Metadaten geht der Durchsatz beim Backup entsprechend in die Knie und die Sicherung dauert entsprechend lange. Aus diesem Grund macht es durchaus Sinn, die Dateien einer logischen Periode (z.B. der eines Monats) vor dem Backup zu zippen.

Das kann man manuell durchführen. Viel eleganter wäre es jedoch, wenn man es automatisieren könnte - z.B. mit einem PowerShell Script, das schließlich durch den Windows Task Scheduler gestartet wird.

Hierfür hält die Windows PowerShell ab der Version 5.0 diese zwei Befehle bereit, mit denen Sie die Komprimierung direkt erreichen bzw. wieder rückgängig machen können:



PowerShell 5.0

68 From [Microsoft.PowerShell.Archive](#) you can use:

- [Compress-Archive](#)
- [Expand-Archive](#)

E.g.:

- Create `result.zip` from the entire `Test` folder:

```
Compress-Archive -Path C:\Test -DestinationPath C:\result
```
- Extract the content of `result.zip` in the specified `Test` folder:

```
Expand-Archive -Path result.zip -DestinationPath C:\Test
```

Wenn sie in dem PowerShell Script vor allem `Compress-Archive` verwenden und zum Schluß noch eine 'Fertigmeldung' per Email erzeugen, haben Sie eine einfache und elegante Methode gefunden, das Backup solcher Verzeichnisbäume bereits mit Hilfe des Betriebssystems und vor den Backup zu optimieren.