O'REILLY®

7. Auflage

Windows-Befehle für Server 2025 und Windows 11 kurz & gut

Mit PowerShell-Alternativen

Olaf Engelke

7. AUFLAGE

Windows-Befehle für Server 2025 und Windows 11

kurz & gut

Olaf Engelke

O'REILLY®

Olaf Engelke

Lektorat: Alexandra Follenius
Copy-Editing: Sibylle Feldmann, *www.richtiger-text.de*
Satz: III-satz, *www.drei-satz.de*
Herstellung: Stefanie Weidner
Umschlaggestaltung: Karen Montgomery, Michael Oréal, *www.oreal.de*

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über *https://www.dnb.de/* abrufbar.

ISBN:
Print 978-3-96009-146-2
PDF 978-3-96010-871-9
ePub 978-3-96010-872-6

7. Auflage 2025

Wieblinger Weg 17
69123 Heidelberg
E-Mail: *hallo@dpunkt.de*

Dieses Buch erscheint in Kooperation mit O'Reilly Media, Inc. unter dem Imprint »O'REILLY«.
O'REILLY ist ein Markenzeichen und eine eingetragene Marke von O'Reilly Media, Inc. und wird mit Einwilligung des Eigentümers verwendet.

Schreiben Sie uns:
Falls Sie Anregungen, Wünsche und Kommentare haben, lassen Sie es uns wissen: *hallo@dpunkt.de*.

noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

Inhalt

Windows-Befehle für Server 2025 und Windows 11

Einführung

Am Anfang war die Dunkelheit. Ein meist schwarzer Bildschirm mit weiß, bernsteingelb oder grün leuchtender Befehlszeile, in der man dem Computer mit mehr oder weniger mühsam erlernten Befehlen und Tastenkombinationen sagen konnte, was er zu tun hatte, war die dominierende Schnittstelle für jeden, der schon mit dem Computer arbeiten oder gar spielen durfte. Es sollte einige Zeit dauern, bis grafische Benutzeroberflächen, geeignet auch für eine Mausbedienung, die Eingabeaufforderung ablösten.

Bei modernen Windows-Versionen wie Windows 11 tritt diese Eingabeaufforderung kaum noch in Erscheinung, stattdessen wird sie mehr denn je vor dem Auge des Anwenders verborgen. Das liegt nicht daran, dass sie plötzlich bedeutungslos wäre. Vielmehr geht Microsoft wohl davon aus, dass die meisten Anwender ohnehin zu den in der grafischen Benutzeroberfläche bereitgestellten Werkzeugen greifen und Profis sich mit der PowerShell arrangieren, die bereits seit Windows 7 integraler Bestandteil des Betriebssystems ist und in den aktuellen Betriebssystemversionen zunehmend die alte Eingabeaufforderung ersetzt. Das kann man deutlich daran erkennen, dass in den aktuellen Versionen von Windows als veraltet deklarierte Befehle nach und nach durch PowerShell-Cmdlets ersetzt wurden und hinzugefügte oder erweiterte Features wie beispielsweise die Virtualisierung maßgeblich oder ausschließlich über PowerShell verwaltbar sind.

Denn anscheinend kehrt bei den Windows-Server-Versionen mit der PowerShell die Dunkelheit zurück. Zwischenversionen, die analog zu Windows-11-Funktionsupdates für Software-Assurance-Kunden veröffentlicht werden, lassen den auch bei vielen Administratoren beliebten Desktop gänzlich vermissen.

Das zeigt, dass Microsoft der Administration mit Windows PowerShell eine hohe Bedeutung beimisst, während gleichzeitig neue Befehle für die klassische Eingabeaufforderung mit der Lupe zu suchen sind, bestehende teils für obsolet erklärt werden oder schlicht nicht mehr funktionieren. Dieser Tatsache versucht diese Auflage erneut Rechnung zu tragen und die Leser bei der Transformation zu PowerShell durch den einen oder anderen Fingerzeig zu unterstützen.

Für wen dieses Buch gedacht ist

Diese Referenz beschreibt die meisten Befehle der Windows-Eingabeaufforderung (die oft auch als Befehlszeile, Kommandozeile, Konsole oder DOS-Prompt bezeichnet wird) in der aktuellen Version von Windows und benennt, sofern es sich anbietet, Äquivalente in Windows PowerShell. Sie ist nicht nur für Systemadministratoren gedacht, sondern auch für normale Windows-Anwender, die jenseits der grafischen Benutzeroberfläche nicht hilflos dastehen möchten. Enthalten ist die Mehrzahl der Windows-Befehle in den zum Zeitpunkt der Entstehung dieser Auflage aktuellen Versionen von Client und Server. Etliche der Befehle haben eine weit zurückreichende Geschichte und sind in gleicher oder abgewandelter Syntax auch in früheren Fassungen von Windows vorhanden.

Was dieses Buch nicht enthält

Aufgrund des kompakten Formats der Reihe »kurz & gut« wurden selten genutzte und spezielle Befehle und Parameter zum Teil nicht in diese Referenz aufgenommen. Einige weitere Befehle werden nicht behandelt, weil

ihre Funktion bereits komplett von einem anderen Befehl übernommen wurde oder sie veraltet sind. Daher sind Befehle, die es ausschließlich für frühere Windows-Versionen gab, nur noch in Ausnahmen in dieser Ausgabe aufgeführt. Zudem wurden die Informationen zu einigen Befehlen gekürzt, um Raum für PowerShell-Alternativen zu schaffen und das Buch dabei weiter möglichst kompakt zu halten. Teils fehlen Befehle, die nur nach der Installation spezieller Rollen oder Funktionen des Betriebssystems verfügbar sind, was besonders bei den Serverversionen von Windows häufiger der Fall ist – speziell auch Befehle, die in Verbindung mit Microsoft Azure eingeführt wurden.

Ebenfalls nicht enthalten sind Programme mit grafischer Benutzeroberfläche, da sich dieses Buch auf Befehle beschränkt, die in der Eingabeaufforderung bzw. in Skripten nutzbar sind. Ausnahmen bilden lediglich solche Programme, die sowohl über eine grafische Oberfläche verfügen als auch weitgehend aus der Eingabeaufforderung heraus gesteuert werden können.

Auf Linux-Befehle wird nicht eingegangen, wenngleich dieses Betriebssystem mit dem als Feature installierbaren Windows-Subsystem für Linux ebenfalls die Windows-Arena betreten hat.

Aufbau

Die Befehle sind nach Funktionsgruppen geordnet und innerhalb dieser Gruppen weitgehend alphabetisch sortiert. Einen bestimmten Befehl finden Sie am einfachsten über den Index. Befehlsoptionen sind nach ihren Funktionen und nach Wichtigkeit angeordnet. Weniger wichtige Parameter werden nicht immer aufgeführt. Einige Befehle bieten derart zahlreiche und umfassende Optionen, dass es aufgrund des kompakt zu haltenden Buchformats unmöglich ist, alle aufzuführen und zu erläutern.

Viele der in diesem Buch beschriebenen Befehle lassen sich unter allen gegenwärtig eingesetzten Windows-Versionen verwenden, auch wenn sie nur mit einem Vorgänger oder Nachfolger des von Ihnen eingesetzten Betriebssystems mitgeliefert werden. Einige Tools benötigen jedoch

zwingend ein bestimmtes Serverbetriebssystem oder eine Mindestversion des Betriebssystems, weil sie API-Funktionen verwenden, die von Microsoft nicht für alle Versionen des Betriebssystems bereitgestellt wurden. Andere Befehle wurden von Microsoft stetig verbessert und verändert, sodass z. B. die Syntax und der Funktionsumfang des in der aktuellen Version von Windows 11 mitgelieferten Befehls anders sind als jene des gleichnamigen Befehls in einer Vorgängerversion. In solchen Fällen wird die zum Zeitpunkt der Drucklegung aktuelle Version beschrieben.

Hinweis: Mit Windows 11/Server 2025 sind all jene Befehle weggefallen, die für die Kompatibilität mit 16-Bit-Anwendungen noch enthalten waren. Diese Befehle bezogen sich bereits in der Vergangenheit auf 32-Bit-Versionen von Windows, die mit den neuen Betriebssystemen nicht mehr verfügbar sind.

Unterscheiden sich Elemente zwischen Windows 11 und Windows Server 2025, wird die Windows-11-Variante beschrieben.

Konventionen

`Fett`
: Kennzeichnet Windows-Befehle und Optionen.

`GROSSBUCHSTABEN UND FETT`
: Kennzeichnet interne Befehle des Befehlszeileninterpreters *cmd.exe*. Diese Befehle sind bei Nutzung einer alternativen Shell, z.B. in PowerShell, nicht unmittelbar verfügbar.

`Kursiv`
: Kennzeichnet Parameter, die Sie selbst eingeben müssen.

`[...]`
: Kennzeichnet optionale Befehlsteile.

a | *b*
: Bedeutet, dass entweder *a* oder *b* eingesetzt werden kann.

{a | b***}***

Bedeutet, dass entweder *a* oder *b* eingesetzt werden muss.

HKLM
HKCU

Kennzeichnen die Registrierungsbäume (Hives) *HKEY_LOCAL_MACHINE* und *HKEY_CURRENT_USER*.

Menüname ⟶ *Menüname*

Der Pfeil (⟶) in Verbindung mit kursiver Schrift beschreibt die Navigation innerhalb eines Menüs.

PoSh:

Hinweise zu PowerShell-Alternativen

Die Eingabeaufforderung

Starten unter Windows 11/Server 2025

In den aktuellen Versionen von Windows-Client und Server (mit Ausnahme von Windows Server in der Core-Installation ohne grafische Benutzeroberfläche) ist die Eingabeaufforderung im Startmenü unter *Alle Apps* ⟶ *Windows-Tools* versteckt. Am schnellsten erreichen Sie diese, indem Sie im Startmenü von Windows 11 oder Server 2025 einfach drauflostippen und entweder `cmd` oder `Eingabeaufforderung` eingeben. Sobald Windows, oft bereits während des Tippens, fündig geworden ist, können Sie das Symbol zur unmittelbaren Ausführung anklicken. Diese Variante hat einen Nachteil, denn die Eingabeaufforderung wird in diesem Fall mit den eingeschränkten Berechtigungen eines Standardbenutzers aufgerufen. Da viele der in diesem Buch beschriebenen Befehle Administratorrechte benötigen und zudem die standardmäßig aktivierte Benutzerkontensteuerung Zugriffe beschränkt, reichen diese Berechtigungen oft nicht, selbst wenn das Benutzerkonto der Gruppe der Administratoren angehört.

Klicken Sie daher das gefundene Symbol der Eingabeaufforderung mit der rechten Maustaste an und dann mit der linken Maustaste auf *Als Administrator ausführen*, um ein Fenster der Eingabeaufforderung mit erhöhten Rechten zu öffnen. Alternativ funktioniert die Tastenkombination *Strg+Umschalt+Enter* bei ausgewähltem Symbol sowie *Alt+J* zum Bestätigen der Sicherheitsabfrage.

Soll häufig auf die Eingabeaufforderung zurückgegriffen werden, ist diese Methode nicht sonderlich praktikabel. Es empfiehlt sich das Anlegen einer Verknüpfung. Wo sollte diese liegen? Windows 11 bietet bei Rechtsklick die Erstellung einer Verknüpfung im Startmenü (*An Start anheften*) oder in der Taskleiste des Desktops (*An Taskleiste anheften*) an, wobei Letzteres

erfahrungsgemäß die praktischere Variante ist. Durch Rechtsklick auf das angeheftete Symbol, nochmaligen Rechtsklick auf den Menüeintrag *Eingabeaufforderung* und Anklicken von *Eigenschaften* im Kontextmenü können Sie einige Einstellungen für verbesserten Komfort anpassen:

Ausführen in:

Hier können Sie einen alternativen Pfad eingeben, in dem sich die Eingabeaufforderung öffnen soll, beispielsweise einen Pfad, in dem sich Ihre Skriptdateien oder regelmäßig zu bearbeitende Dateien befinden.

Tastenkombination:

Legen Sie hier eine Tastenkombination fest, mit der die Eingabeaufforderung gestartet werden soll. Verwenden Sie dafür keine anderweitig im System oder in ständig laufenden Anwendungsprogrammen verwendete Kombination.

Durch Klick auf die Schaltfläche *Erweitert* öffnet sich das Dialogfeld *Erweiterte Eigenschaften*, in dem Sie über das Kontrollkästchen *Als Administrator ausführen* festlegen, dass bei jeder Ausführung der Verknüpfung die Eingabeaufforderung mit erhöhten Rechten (und entsprechender Sicherheitsabfrage) geöffnet wird.

Weitere Einstellungsmöglichkeiten betreffen unter anderem Schriftart und -größe, den verfügbaren Puffer zur Zwischenspeicherung von Befehlen und die Fenstergröße. Eine Eingabeaufforderung im Vollbildmodus steht schon seit mehreren Windows-Versionen nicht mehr zur Verfügung, allerdings kommen Sie mit der Tastenkombination *Alt+Enter* dieser recht nahe.

Ob die gestartete App mit erhöhten Rechten ausgeführt wird, können Sie mit einem schnellen Blick feststellen: Die Titelzeile des Fensters beginnt in diesem Fall mit *Administrator:*.

Eine aus Sicherheitsgründen wenig empfehlenswerte Alternative zur generellen Ausführung der Eingabeaufforderung und anderer Anwendungen mit erhöhten Rechten ist die Deaktivierung der Benutzerkontensteuerung des Betriebssystems bei gleichzeitiger Anmeldung mit einem Benutzerkonto der Administratorengruppe.

Auf klassische Art können Sie eine Verknüpfung auf dem Desktop erzeugen, indem Sie mit der rechten Maustaste auf eine beliebige freie Stelle des Desktops klicken, aus dem Kontextmenü den Eintrag *Neu* ⟶ *Verknüpfung* wählen, als *Speicherort des Elements* `cmd` eingeben und einen beliebigen Namen festlegen. Nach Beendigung des Assistenten können Sie die Eigenschaften der Verknüpfung bearbeiten, wie es bereits für das an die Taskleiste angeheftete Symbol beschrieben wurde.

Terminal

Schließlich können Sie auch unter Windows 11/Server 2025 die altbekannte Tastenkombination *Win+X* verwenden. Freilich dürften Sie darüber stolpern, dass hier weder Eingabeaufforderung noch PowerShell angeboten wird – stattdessen erscheinen in dem sich links unten auf dem Desktop öffnenden Menü die Einträge *Terminal* und *Terminal (Administrator)*, die Sie mit der Maus auswählen können.

Bei der *Terminal*-App handelt es sich um eine minimalistische grafische Benutzeroberfläche, in der sich sowohl Instanzen der Eingabeaufforderung als auch von PowerShell in Registerkarten einbetten lassen. Wurde das Windows-Subsystem für Linux installiert, ist dessen Nutzung ebenfalls von hier aus möglich. Über Profile bestehen zahlreiche Anpassungsmöglichkeiten des Verhaltens der eingebundenen textbasierten Programme.

Möchten Sie aus Kompatibilitätsgründen die Standard-App für die Ausführung von `cmd`-basierten Befehlen fest auf die klassische *Eingabeaufforderung* ändern, öffnen Sie die Windows-Einstellungen, navigieren dort zum Abschnitt *System* ⟶ *Für Entwickler und wählen unter Terminal im Drop-down-Menü den* Eintrag *Windows-Konsolenhost* aus.

Eingabe von Befehlen

Befehle werden durch Tippen in die Befehlszeile eingegeben und mit der

Enter-Taste, oft auch als Eingabetaste bezeichnet, ausgeführt. Dabei kann man eigentlich nicht viel falsch machen, abgesehen von Syntaxfehlern, Tippfehlern und der Verwendung von nicht zur Aufgabenstellung passenden Befehlen. Selbst ohne diese Hindernisse sind Befehlseingaben in ihren Grundanforderungen nicht immer konsistent, sodass es sich lohnen kann, im Fehlerfall die Hilfe zum jeweiligen Befehl etwas ausführlicher zu lesen:

- Befehle und Befehlsoptionen können normalerweise in Groß- oder Kleinbuchstaben eingegeben werden. Ist das der Fall, werden sie in diesem Buch kleingeschrieben. Nur Optionen, die großgeschrieben werden *müssen*, werden in Großbuchstaben geschrieben.
- Befehlsoptionen werden normalerweise durch einen Schrägstrich eingeleitet: `/x`. In vielen Fällen kann der Schrägstrich durch ein Minuszeichen ersetzt werden. Einige Befehle akzeptieren nur das Minuszeichen, vereinzelt werden sogar deren zwei benötigt.
- Die Reihenfolge der Optionen ist nicht einheitlich und nicht immer beliebig. Bitte entnehmen Sie die korrekte Reihenfolge der Syntax des jeweiligen Befehls.
- Einzelne Parameter werden abhängig vom Befehl durch Leerzeichen, Kommata oder Semikola voneinander getrennt.
- Parameter mit Leerzeichen, beispielsweise Verzeichnispfade und Dateinamen, müssen in der Regel von Anführungszeichen umschlossen werden.
- Regions- und Spracheinstellungen sowie die Sprachversion des Betriebssystems können die Syntax beeinflussen. In diesem Buch wird die deutschsprachige Version des Betriebssystems mit den Einstellungen für Deutschland verwendet.
- Befehle können in der nachfolgenden Zeile fortgesetzt werden, wenn die vorherige Zeile mit dem Zeichen `^` beendet wurde.
- Durch ein vorangestelltes `^`-Zeichen wird zudem verhindert, dass der Befehlsinterpreter das folgende Zeichen interpretiert. Solche Zeichen werden auch »Escapezeichen« genannt.

- Zeilenumbrüche bei im Buch dargestellten Befehlsaufrufen sind in der Regel der begrenzten Zeilenlänge geschuldet und werden aufgrund der Lesbarkeit nicht explizit hervorgehoben. Sie sind daher bei der Übertragung in die Eingabeaufforderung nicht zu berücksichtigen und gegebenenfalls durch Leerzeichen zu ersetzen. Ausnahmen sind Skriptbeispiele oder Beispiele, bei denen explizit erwähnt wird, dass es sich um Mehrzeiler handelt.
- Mehrere Befehle können mit dem &-Zeichen verknüpft werden:
  `Befehl1 & Befehl2`
  Die Befehle werden der Reihe nach ausgeführt.
- Die Ausführung eines Befehls kann davon abhängig gemacht werden, ob der vorangegangene Befehl erfolgreich ausgeführt wurde. Dazu werden die Befehle mit && bzw. || verknüpft:
  `Befehl1 && Befehl2`
  `Befehl2` wird nur ausgeführt, wenn `Befehl1` erfolgreich ausgeführt werden konnte.
  `Befehl1 || Befehl2`
  `Befehl2` wird nur ausgeführt, wenn `Befehl1` nicht erfolgreich ausgeführt werden konnte.

Umleitung der Ein- und Ausgabe

<Datei

Liest die Standardeingabe aus einer Datei statt von der Tastatur.

>Datei

1>Datei

Schreibt die Standardausgabe in eine Datei statt auf den Bildschirm.

>>Datei

1>>Datei

Hängt die Standardausgabe an eine Datei an. Ist keine Datei vorhanden, wird diese erzeugt.

2>Datei

Schreibt die Standardfehlerausgabe in eine Datei.

2>>Datei

Hängt die Standardfehlerausgabe an eine Datei an.

>Datei
2>&1

Schreibt die Standardausgabe in eine Datei und leitet die Standardfehlerausgabe zur Standardausgabe um. Damit werden Standardausgabe und Standardfehlerausgabe in dieselbe Datei geschrieben.

Befehl1 | Befehl2

Stellt eine Verknüpfung zwischen der Standardausgabe des ersten Befehls (`Befehl1`) und der Standardeingabe des zweiten Befehls (`BefXEhl2`) her.

In Einzelfällen, beispielsweise für die Wiederverwendung von komplexen Werten wie GUIDs zur Vermeidung von Tippfehlern, ist es hilfreich, den gewünschten Ausschnitt der Eingabeaufforderung in die Zwischenablage zu kopieren und von dort entweder direkt oder nach Bearbeitung im Editor wieder einzufügen. Dafür können Sie auf althergebrachte Art in der Titelzeile des Fensters der Eingabeaufforderung das Systemmenü oben links nutzen: also auf *Bearbeiten* ⟶ *Markieren* klicken und dann mit der Maus oder mit der Tastatur unter Zuhilfenahme der Pfeiltasten und der gedrückten *Umschalt*-Taste den zu kopierenden Bereich in Rechteckform auswählen. Sobald Sie die Auswahl abgeschlossen haben, drücken Sie die *Enter*-Taste, um den Text des ausgewählten Bereichs in die Zwischenablage zu kopieren.

Zum Einfügen aus der Zwischenablage verwenden Sie wiederum den Menüeintrag *Bearbeiten* ⟶ *Einfügen* aus dem Systemmenü.
Oder aber Sie nutzen die aus vielen anderen Windows-Anwendungen vertrauten Methoden mithilfe von Tastatur oder Maus, die in den aktuellen Windows-Versionen auch in der Eingabeaufforderung und im Terminal verfügbar sind, zum Beispiel:

Umschalt+Pfeil links/Pfeil rechts: Markiert Zeichen links oder rechts von der Cursorposition (jedes erneute Betätigen der Pfeiltaste markiert ein weiteres Zeichen).

Umschalt+Strg+Pfeil links/Pfeil rechts: Markiert nach Markierung eines Zeichens Wörter links oder rechts von der Cursorposition (jedes erneute Betätigen der Pfeiltaste markiert ein weiteres Wort).

Strg+C: Kopiert die markierten Zeichen in die Zwischenablage.

Strg+V: Fügt Inhalte aus der Zwischenablage an der Cursorposition ein.

Ziehen mit gedrückter linker Maustaste: Markiert den ausgewählten Bereich.

Rechtsklick: Kopiert einen markierten Bereich als Text in die Zwischenablage oder fügt den Inhalt der Zwischenablage an der Cursorposition ein.

Umgebungsvariablen

Dieser Abschnitt erklärt einige wichtige Windows-Umgebungsvariablen. Variablen werden in die Eingabeaufforderung und Batchdateien mit dem Prozentzeichen (%) ausgelesen. Zum Beispiel zeigt der Befehl `echo %SystemRoot%` den Pfad des Windows-Ordners. Damit können Anwendungen und Skripte auf unterschiedlich eingerichteten Systemen ausgeführt werden, ohne dass eine abweichende Ordnerstruktur für die Aufrufe berücksichtigt werden müsste. Die aktiven System- und Benutzervariablen können mit `set` angezeigt und temporär angepasst sowie mit `setx` dauerhaft geändert werden.

PoSh: In PowerShell befinden sich die Umgebungsvariablen im Container `env:`.

`gci env:systemroot` zeigt die obige Variable unter PowerShell.

APPDATA

Enthält den Pfad zu *AppData\Roaming* im Profilordner des Benutzers.

COMPUTERNAME

Enthält den Computernamen der Windows-Installation.

HOMEDRIVE und HOMEPATH

Das Laufwerk und der Pfad auf diesem Laufwerk zum Benutzerprofilordner des angemeldeten Benutzers.

PATH

Mehrere durch Semikola getrennte Verzeichnisse, die in dieser Reihenfolge nach Befehlen durchsucht werden, wenn sie ohne vorangestellten Pfad aufgerufen werden.

PATHEXT

Mehrere durch Semikola getrennte Dateierweiterungen, die in dieser Reihenfolge an einen Befehl ohne Erweiterung angehängt werden, um die ausführbare Datei für den Befehl zu finden. Diese Erweiterungen müssen den voranstehenden Punkt beinhalten, wie *.exe* oder *.cmd*.

ProgramFiles

Der Verzeichnisname des Programmordners (normalerweise *C:\Program Files*). Bei 64-Bit-Versionen des Betriebssystems handelt es sich um den Programmordner für 64-Bit-Software.

ProgramFiles(x86)

Der Verzeichnisname des Programmordners für 32-Bit-Anwendungen.

SystemRoot und windir

Der Pfad des Windows-Systemverzeichnisses (normalerweise *C:\WINDOWS*).

TEMP und TMP

Der komplette Pfad eines Verzeichnisses, das von Anwendungen und vom Betriebssystem zur Ablage temporärer Dateien verwendet wird.

USERNAME

Der Name des angemeldeten Benutzers.

USERPROFILE

Der Pfad zum Profilverzeichnis des angemeldeten Benutzers.

PoSh: Einer der Vorteil der PowerShell ist, dass mit Platzhaltern gearbeitet werden kann. So zeigt `gci env:*user*` alle Variablen an, die das Wort `user` im Namen enthalten.

Installation zusätzlicher Administrationstools

Nützliche Programme werden durch Microsoft direkt veröffentlicht, beispielsweise die Werkzeuge der Sysinternals-Suite. Auf einige dieser Programme wird im Buch eingegangen.

Im Unternehmensumfeld sind zudem die »Remoteserver-Verwaltungstools« (*Remote Server Administration Tools*, RSAT) zur Verwaltung von Servern von Bedeutung. Diese dienen der Verwaltung von Servern und sind in Windows 11 als Feature über *Einstellungen/System/Optionale Features/Optionales Feature hinzufügen* einzubinden, unter Windows Server 2025 im Servermanager über das Menü *Verwalten/Rollen und Features hinzufügen/Features – Remoteserver-Verwaltungstools*. Alternativ sind sie unter Windows Server mittels PowerShell installierbar.

Für die Unterstützung der automatisierten Installation von aktuellen Windows-Versionen in Unternehmen stellt Microsoft das *Windows Assessment and Deployment Kit* zum Herunterladen bereit, das ebenfalls einige nützliche Werkzeuge für die Eingabeaufforderung für EDV-Profis enthält.

Auch befehlszeilenbasierte Werkzeuge von Drittanbietern haben nach wie vor ihre Berechtigung und füllen die eine oder andere Lücke.

Hilfebefehle und -dateien

`help` **`Befehl`**

Zeigt die Hilfe für viele Windows-Standardbefehle an.

`Befehl /?`

Zeigt in vielen Fällen einen Hilfetext an.

`net help` **`Befehl`**

Zeigt die Hilfe für einen der `net`-Befehle an. Beispielsweise erklärt `net help user` die Optionen des Befehls `net user`.

`net helpmsg` **`nnnn`**

Zeigt den Fehlertext zum Windows-Fehler mit der Nummer `nnnn` an.

Windows Assessment and Deployment Kit

Informationen zur Verwendung der zusätzlichen Werkzeuge sind zum Teil nur in englischer Sprache verfügbar und müssen gegebenenfalls online recherchiert werden. Zumindest zu den Befehlszeilenwerkzeugen kann oftmals eine rudimentäre Hilfe mittels `/?` hinter dem Befehlswort aufgerufen werden.

PowerShell

Seit PowerShell im Jahr 2007 erstmals veröffentlicht wurde, ist ihre Integration in Microsoft-Betriebssysteme und anwendungen weit vorangeschritten. In Windows Server 2025 und Windows 11 24H2 ist Version 5.1 der PowerShell installiert. Bei PowerShell handelt es sich um eine leistungsfähige Skriptumgebung, die sowohl die klassische Eingabeaufforderung als auch früher beliebte Skriptsprachen wie VBScript beerbt hat und um die gerade im Serverumfeld mittlerweile kein Weg mehr vorbeiführt.

PowerShell arbeitet nicht textbasiert wie andere befehlszeilenorientierte Benutzeroberflächen, sondern mit Objekten, die bei Bedarf über Pipelines (|) von einem Befehl zum anderen weitergegeben werden. Die Benennung der als »Cmdlets« bezeichneten Befehle folgt dem Schema *Verb-Nomen*, z. B. `get-command` zur Auflistung aller Befehle. PowerShell verwendet ein erweiterbares Providermodell, um neben dem Dateisystem beispielsweise folgende Datenspeicher als Laufwerk anzusprechen: Registrierung,

Zertifikatsspeicher, Umgebungsvariablen, Aliase, Variablen und Funktionen.

Jede neue Version von PowerShell brachte erhebliche Steigerungen des Funktionsumfangs mit sich. IntelliSense und Tab-Vervollständigung helfen bei der Erkundung und Eingabe von Befehlen. Eine Hilfe, die online aktualisiert werden kann, trägt zur Funktionalität bei.

Um zu ermitteln, welche Version der PowerShell aktuell auf Ihrem System installiert ist, führen Sie innerhalb von PowerShell den folgenden Befehl aus:

```
$psversiontable.psversion
```

Die Ausgabe zeigt in der Spalte *Major* die Versionsnummer an.

Auch wenn es nach wie vor Anwendungsfälle gibt, in denen PowerShell im Vergleich zu den spezialisierten Befehlen der Eingabeaufforderung zu komplex ist, um mal schnell ein bestimmtes Ziel zu erreichen (und sei es nur die manchmal gefühlt ellenlange Tipperei), gibt es nur wenige in den Bereich der Eingabeaufforderung fallende Aufgaben, die sich mit PowerShell nicht lösen oder damit kombinieren ließen.

Ergänzend bietet PowerShell zahllose Möglichkeiten, die mit den Standardwerkzeugen der Eingabeaufforderung nicht und auch in anderen Skriptsprachen oft nicht trivial umzusetzen sind. Da fast alle Befehle der Eingabeaufforderung – mit Ausnahme der Parameterübergabe für interne Befehle – unmittelbar in der PowerShell-Konsole und im Terminal funktionieren, besteht kaum mehr ein Grund, noch aus der Eingabeaufforderung heraus zu arbeiten. Lediglich die ISE (*Integrated Scripting Environment*, integrierte Skriptumgebung) weist bei der Unterstützung von einzelnen Befehlen der Eingabeaufforderung noch Mankos auf, die teilweise vom Terminal geerbt wurden. Die in der ISE integrierte umfassende Hilfe unterstützt dafür den Anwender gut dabei, die anfangs steil erscheinende Lernkurve beim Erlernen der PowerShell-Befehls- und Skriptsprache zu bewältigen.

Im weiteren Verlauf dieses Buchs werden zu den Befehlen der Eingabeaufforderung teilweise PowerShell-Alternativen oder eine Kombination aus PowerShell und Befehlszeile gezeigt. Praxisnahe Beispiele

vermitteln die Stärken von PowerShell auch in Alltagssituationen. Auf PowerShell bezogene Abschnitte werden mit dem Kürzel **PoSh** gekennzeichnet. Die aufgeführten Beispiele und Hinweise sollen nicht perfekt sein, sondern vielmehr die vielfältigen Möglichkeiten der Problemlösung mit PowerShell vermitteln und beim selbstständigen Finden und Ausprobieren dafür relevanter Cmdlets helfen. Zudem gibt es bei der Verwendung von PowerShell meist nicht nur den einen Weg, ein Ziel zu erreichen, und Perfektion ist nur selten wirklich erforderlich.

PowerShell 7

Bei den der Version 5 nachfolgenden PowerShell-Versionen – zum Zeitpunkt des Erscheinens dieser Auflage aktuell ist Version 7.5 – gab es verschiedene Änderungen der technischen Grundlagen, um Interoperabilität mit anderen Betriebssystemen zu erreichen. Deshalb werden diese nicht automatisch als Update verteilt, sondern können separat heruntergeladen und parallel zur betriebssystemeigenen PowerShell-Installation eingerichtet werden.

Für die Installation der aktuellen Version wird auf Clientsystemen der Windows-Paket-Manager *winget.exe* empfohlen.

Nach verfügbaren Versionen von Microsoft PowerShell suchen Sie mittels:

```
winget search Microsoft.PowerShell
```

Als Ergebnis werden Name, ID, Version und Quelle der online verfügbaren PowerShell-Versionen aufgelistet.

Zum Installieren rufen Sie `winget` mit den Parametern `id` und `source` auf. So lädt der folgende Befehl die aktuelle Version mit der ID `Microsoft.PowerShell` als MSI-Paket herunter und installiert sie:

```
winget install --id Microsoft.Powershell --source winget
```

Die neu installierte Version verfügt über eine eigenständige Verknüpfung im Startmenü und lässt sich zudem wahlweise als zusätzliches Profil im Windows-Terminal oder mit dem Befehl `pwsh` starten. Die Mehrzahl der in früheren Versionen von PowerShell unterstützten Module und Befehle ist mit

PowerShell 7 kompatibel, aber es gibt auch Ausnahmen.

Windows-Subsystem für Linux

Die Installation des Linux-Subsystems unter Windows lässt sich mithilfe von *wsl.exe* durchführen. Eine Besonderheit dieses Befehls ist die Verwendung eines doppelten Bindestrichs für die meisten Optionen.

Die einfachste Methode zur WSL-Installation ist der Aufruf des Befehls

`wsl --install`

Dieser Befehl, in einer administrativen Eingabeaufforderung ausgeführt, installiert die Voraussetzungen für WSL und die standardmäßige Ubuntu-Distribution von Linux.

Hinweis: Auf dem verwendeten System muss das Windows-Feature *VM-Plattform* installiert sein, und gegebenenfalls muss im BIOS die Virtualisierungsunterstützung aktiviert werden.

`wsl --list --online`

Mit diesem Befehl wird eine Liste der verfügbaren Distributionen angezeigt.

`wsl --install [Distributionsname]`

Mit der Eingabe eines Distributionsnamens aus der Liste installieren Sie die gewünschte Distribution. Eine parallele Installation mehrerer Distributionen ist ebenfalls möglich.

Hinweis: Je nach Windows-Version müssen Sie dem Distributionsnamen im Befehlsaufruf gegebenenfalls den Parameter `-d` voranstellen.

`wsl --set --default [Distributionsname]`

Hiermit ändern Sie die Standarddistribution, die durch `wsl` gestartet wird.

`wsl --distribution [Distributionsname] --user [Benutzername]`

Mit diesem beispielsweise im Terminal eingebundenen Befehl legen Sie individuell fest, welche Distribution mit welchem Linux-Benutzernamen aufgerufen wird.

Allgemeine Befehle

clip

`Befehl | clip`
`clip < `*`Datei`*

Der Befehl `clip` ermöglicht die Umlenkung der Bildschirmausgabe eines Befehls in die Zwischenablage, um deren Inhalt in eine Windows-Anwendung einzufügen.

PoSh: Mit `clip` lassen sich auch PowerShell-Bildschirmausgaben einfach als Text in die Zwischenablage kopieren.

cmd

`cmd [`*`Optionen`*`] [[/c | /k] [/s] `*`Befehl`*`]`

Startet eine neue Instanz des Windows-Befehlsinterpreters. Wurde ein Befehl angegeben, wird er ausgeführt. Verwenden Sie die mit `cmd /?` aufrufbare ausführliche Hilfe, um Informationen zu weiteren Features zu erhalten (z.B. der automatischen Vervollständigung von Pfaden und Befehlen oder der verzögerten Expansion von Variablen). Über die Eigenschaften eines Kommandozeilenfensters (Klick auf das Systemmenü in der linken Ecke des Titelfensters und dann auf *Eigenschaften*) können Sie sein Aussehen und Verhalten in weiten Bereichen beeinflussen, wobei diese Anpassung im Gegensatz zu den Eigenschaften der Verknüpfung nur für das aktive Fenster gilt. Insbesondere die Vergrößerung des Fensterpuffers, der das Scrollen in den nicht mehr am Bildschirm sichtbaren Zeilen ermöglicht, ist oft sinnvoll. Verwenden Sie die *Terminal*-App, werden deren Einstellungen verwendet, die Sie ausführlich an Ihre Bedürfnisse anpassen können.

Der Befehl `exit` beendet den Kommandozeileninterpreter.

Optionen

`[/c | /k] [/s]`

Der Interpreter führt den angegebenen Befehl aus und bleibt nach Beendigung des Befehls aktiv (`/k`) oder beendet sich (`/c`). Die Option `/s` veranlasst den Befehlsinterpreter, den Befehl umschließende Anführungszeichen vor Ausführung des Befehls zu entfernen (normalerweise werden diese beibehalten). Das gilt jeweils in Verbindung mit `/c` oder `/k`.

`/q`

Schaltet die Befehlsausgabe ab (siehe `echo off`).

`/e:{`*`on`*` | `*`off`*`}`

Aktiviert oder deaktiviert die Erweiterungen des Befehlsinterpreters. Der Standardwert wird durch den Registrierungswert *\Software\Microsoft\Command Processor\Enable Extensions* unter *HKCU* oder *HKLM* bestimmt. Im Auslieferungszustand von Windows sind die Erweiterungen aktiviert.

`/a | /u`

Die Ausgabe von internen Befehlen erfolgt im ANSI- (Standard) bzw. im Unicode-Format.

`/d`

Deaktiviert die Autorun-Einträge in der Registrierung unter *\Software\Microsoft\Command Processor\Autorun* in *HKLM* und *HKCU*.

`/f:{`*`on`*` | `*`off`*`}`

Aktiviert bzw. deaktiviert die Ergänzung von Datei- und Verzeichnisnamen mit der *Tabulatortaste*.

`/v:{`*`on`*` |`*`off`*`}`

Aktiviert bzw. deaktiviert die verzögerte Erweiterung von Variablen. Damit werden entsprechende Variablen erst zur Laufzeit beispielsweise einer Schleife aufgelöst. Solche Variablen werden mit der Syntax `!variable!` angesprochen.

Weitere Parameter entnehmen Sie dem Aufruf von `cmd /?` in einem offenen Befehlszeilenfenster.

`cmd /u /? /c >` *`c:\temp\help.txt`*

Leitet unter Verwendung des Unicode-Formats (womit deutsche Sonderzeichen in der erzeugten Datei in Windows-Anwendungen wie Notepad korrekt dargestellt werden) die Hilfeinformationen zum Befehlszeileninterpreter in die Datei *help.txt* im gegebenenfalls vorher anzulegenden Ordner *c:\temp* um.

PoSh: Da interne Befehle des Befehlszeileninterpreters wie beispielsweise `DIR` in der PowerShell nur nachgebildet sind und diese Nachbildungen gebräuchliche Parameter nicht unterstützen, kann durch expliziten Aufruf von `cmd` in der PowerShell-Konsole oder im Terminal auf das Original des Befehls zurückgegriffen werden:

`cmd /c dir /p`

Gibt den Inhalt des aktuellen Verzeichnisses in einer PowerShell-Sitzung mit dem klassischen `DIR`-Befehl seitenweise am Bildschirm aus und gibt anschließend an PowerShell zurück.

COLOR

`color` ***`Farbcode1 Farbcode2`***

Stellt die Hintergrundfarbe und die Schriftfarbe in der aktuell geöffneten Konsole um. Die Farbattribute sind als hexadezimale Werte anzugeben, die durch die Hilfe angezeigt werden. So stellt `color` *`1f`* den Hintergrund auf Blau und die Schrift auf Weiß ein. Ohne Angabe von Farbcodes wird die Standardanzeige wiederhergestellt. In der *Terminal*-App können Sie unter *Einstellungen* ⟶ *Eingabeaufforderung* ⟶ *Zusätzliche Einstellungen* ⟶ *Darstellung* ein Farbschema definieren.

PoSh: Auch die Farbgebung der PowerShell-Konsole lässt sich mit dem Befehl `cmd /c color` *`xy`* anpassen. Alternativ lassen sich folgende PowerShell-Befehle verwenden:

```
$HOST.UI.RawUI.BackgroundColor = "DarkBlue"
$HOST.UI.RawUI.ForegroundColor = "White"
```

---

date

`date [`*`tt.mm.jj`*`]`*`jj`*`] [/t]`

Stellt das angegebene Datum ein oder fragt danach, wenn es nicht angegeben wurde. Mit der Option `/t` wird das Datum angezeigt, ohne es zu ändern, was sich mit Ausgabeumleitung für selbst erstellte Logdateien eignet.

PoSh: `get-date -displayhint date` zeigt das Systemdatum ohne Uhrzeit. Um das Datum in Kurzform angezeigt zu bekommen, verwenden Sie `get-date -format dd.MM.yyyy`.

Das Cmdlet `set-date` erlaubt hingegen, ein neues Systemdatum zu setzen. Beachten Sie, dass dabei die Uhrzeit auf 00:00 gesetzt wird, sofern sie nicht explizit angegeben wird.

---

doskey

`doskey [`*`Optionen`*`]`

Erlaubt den Zugriff auf bereits ausgeführte Befehle mit den Tasten *Pfeil hoch* und *Pfeil runter* und die Erstellung von Makros (Aliasdefinitionen).

Befehlshistorie und Editieroptionen

`/history`

Zeigt die vollständige Befehlshistorie an. Durch Ausgabeumleitung mit > gefolgt von einem Dateinamen können Sie diese Liste für die künftige Verwendung als Makrodatei abspeichern.

`/listsize=`*`n`*

Stellt die Größe der Befehlshistorie auf *n* Einträge ein.

`/insert | /overstrike`

Stellt den Bearbeitungsmodus für aus der Historie abgerufene Befehle auf *Einfügen* bzw. auf *Überschreiben*. Die Standardeinstellung ist *Einfügen*. Durch Drücken der Taste *Einfg* können Sie den Modus ebenfalls wechseln.

Makrooptionen

`Makroname=`*`Befehl`*

Definiert ein Makro. Innerhalb des Befehls können die folgenden Variablen verwendet werden: `$t` fügt ein Trennzeichen ein, `$1` bis `$9` erlauben den Zugriff auf einzelne Parameter, `$*` fügt alle eingegebenen Parameter ein. `$G` ersetzt das Zeichen für die Ausgabeumleitung:

```
doskey liste=dir /b $1 $g dateien.txt $t notepad.exe dateien.txt
```

`/macros`

Zeigt alle vorhandenen `doskey`-Makros an.

`/macros:all`

Zeigt zusätzlich die ausführbaren Dateien zugewiesenen `doskey`-Makros an.

`/macrofile=`*`Datei`*

Aktiviert alle in der angegebenen Datei enthaltenen Makros.

`/exename=`*`EXE-Datei`*

Erlaubt die Zuordnung einer ausführbaren Datei zu dem Makro, das soeben definiert wird.

`/macros:`*`EXE-Datei`*

Zeigt alle vorhandenen Makros an, die der angegebenen ausführbaren Datei zugeordnet sind.

Die Tastenkombination *Alt+F10* löscht alle definierten Makros.

Weitere Tastenkombinationen sind unter anderem:

F7 zeigt gespeicherte Befehle, *Alt+F7* löscht sie.

F8 durchsucht die gespeicherten Befehle.

F9 wählt einen Befehl durch seine Nummer aus.

PoSh: Standardmäßig werden `doskey`-Makros dem Befehlszeileninterpreter *cmd.exe* zugeordnet. Mithilfe des Parameters `/exename=powershell.exe` können Sie `doskey`-Makrodateien für die Ausführung von PowerShell-Befehlen verwenden. Dabei beachten Sie bitte, dass die Syntax der Befehle den Anforderungen von PowerShell entsprechen muss – so wäre z.B. die Variable `%WINDIR%` durch den PowerShell-Ausdruck `$env:WINDIR` zu ersetzen.

---

find

`find [`*`Optionen`*`] "`*`Zeichenfolge`*`" [`*`Dateien`*`]`

Sucht in den angegebenen Dateien, in einem über die Tastatur eingegebenen Text oder über eine Pipeline in der Standardeingabe nach der von Anführungszeichen umschlossenen Zeichenfolge und gibt Zeilen bzw. deren Anzahl aus, die die Zeichenfolge enthalten.

Eine beliebte Anwendung von `find` ist die Filterung der Ausgabe eines anderen Befehls. Das folgende Beispiel gibt unter Windows 11 diejenigen Ordner innerhalb des Benutzerprofils aus, die als Junction (Soft Link) auf ein anderes Verzeichnis zeigen:

```
dir /ad %userprofile% | find /i "junction"
```

Optionen

`/v`

Zeigt nur die Zeilen an, in denen die Zeichenfolge nicht vorkommt.

`/i`

Ignoriert Groß-/Kleinschreibung beim Vergleich.

`/c`

Zeigt nur die Anzahl der übereinstimmenden Zeilen an.

`/n`

Zeigt vor jeder Zeile die Zeilennummer an.

/offline

Schließt bei der Suche in Dateien auch Offlinedateien ein.

---

findstr

findstr [*Optionen*] [/c:*Zeichenfolge* | /g:*Datei* | *Zeichenfolgen*] [*Dateien*]

Sucht in den angegebenen Dateien nach einer oder mehreren Zeichenfolgen oder regulären Ausdrücken und gibt übereinstimmende Zeilen aus. Wurden keine Dateien angegeben, wird die Standardeingabe durchsucht. Falls Sie mehrere Suchbegriffe verwenden wollen, müssen Sie sie gemeinsam in Anführungszeichen einschließen.

Optionen

/b | /e

Übereinstimmende Zeilen werden nur dann ausgegeben, wenn die Übereinstimmung am Anfang (/b) oder am Ende (/e) der Zeile auftritt. Es kann nur eine dieser beiden Optionen verwendet werden.

/c:*Zeichenfolge*

Kennzeichnet die angegebene Zeichenfolge als Suchbegriff. Diese Option ist insbesondere bei Leerzeichen im Suchbegriff hilfreich, da ohne sie jedes Wort der Zeichenfolge einzeln gesucht wird.

/d:*Verzeichnisliste*

Durchsucht die Dateien in der angegebenen Verzeichnisliste. Verzeichnispfade werden durch Semikola voneinander getrennt.

/f:*Datei*

Liest die Dateiliste aus der angegebenen Datei. Ein Schrägstrich anstelle des Dateinamens bedeutet, dass der Dateiname in der Eingabeaufforderung abgefragt wird.

/g:*Datei*

Liest die Suchausdrücke aus der angegebenen Datei. Ein Schrägstrich anstelle des Dateinamens bedeutet, dass der Dateiname in der

Eingabeaufforderung abgefragt wird.

/i

Ignoriert Groß-/Kleinschreibung.

/l

Interpretiert die Zeichenfolge buchstabengetreu.

/m

Zeigt nur die Namen der Dateien mit gefundener Übereinstimmung.

/n | /o

Zeigt die Zeilennummer (/n) oder die Anzahl der Zeichen vom Dateianfang bis zur Übereinstimmung (/o) für jede Fundstelle an.

/offline

Schließt bei der Suche in Dateien auch Offlinedateien ein.

/p

Überspringt Dateien mit nicht druckbaren Zeichen.

/r

Interpretiert die Zeichenfolge als regulären Ausdruck.

/s

Durchsucht die angegebenen Dateien auch in Unterverzeichnissen.

/v

Gibt Zeilen ohne die gesuchte Zeichenfolge aus.

/x

Zeigt nur exakt übereinstimmende Zeilen an.

Bestandteile von regulären Ausdrücken

.

Ein beliebiges Zeichen.

^ | $

Der Anfang bzw. das Ende einer Zeile.

\< | \>

Der Anfang bzw. das Ende eines Worts.

\x

Zeichen `x` verwenden, auch wenn es ein Metazeichen ist (z.B. bedeutet `\$`, dass das Dollarzeichen als Suchbegriff verwendet wird).

`[Zeichenklasse]`

Ein beliebiges Zeichen aus einem Zeichensatz.

`[^Zeichenklasse]`

Ein beliebiges Zeichen, das nicht im Zeichensatz enthalten ist.

`[a–z]`

Ein beliebiges Zeichen aus dem angegebenen Bereich. Es können mehrere Bereiche und Listen von Zeichen in den Klammern angegeben werden.

`*`

Keines oder mehrere der Zeichen/Klassen aus der angegebenen Liste. Zum Beispiel bedeutet `[0–9]*` keine oder mehrere Zahlen, und `.*` bedeutet keines oder mehrere beliebige Zeichen.

PoSh: PowerShell verfügt über leistungsstarke Suchfunktionen und lässt sich deutlich komfortabler nutzen und anpassen. Folgendes Kommando durchsucht alle Dateien mit der Erweiterung *.log* im angegebenen Ordner und in angegebenen Unterverzeichnissen und zeigt Dateinamen, Zeilennummer und Inhalt aller Zeilen mit der Zeichenkette »error«:

```
gci C:\Skript\*.log -rec | Select-String -pattern "error"
```

Sollen lediglich die Namen derjenigen Dateien aufgeführt werden, in denen der Suchbegriff vorkommt, erweitern Sie den obigen Befehl mit:

```
-list | Select-Object Path | ft -AutoSize
```

Dabei bewirkt der Parameter `–list` die Unterdrückung von Duplikaten; und da es sich bei den Ergebnissen von `Select-String` nicht um bloßen Text, sondern um Objekte handelt, lässt sich aus diesen Objekten der Dateipfad durch Weitergabe an `Select-Object` extrahieren. `ft` sorgt für eine tabellarische Anzeige, und `AutoSize` verhindert, dass die angezeigten Pfade abgeschnitten werden.

---

more

***Befehl* | more [*Optionen*]**
more [*Optionen*] [*Dateien*]

Zeigt die Ausgabe eines Befehls oder die angegebene(n) Datei(en) seitenweise an. Wird häufig verwendet, um überlange Befehlsausgaben bequem lesen zu können.

Optionen

/c
: Löscht den Bildschirm, bevor die erste Seite angezeigt wird.

/e
: Aktiviert die erweiterten Features.

/s
: Zeigt statt mehrerer aufeinanderfolgender Leerzeilen nur eine an.

/t*n*
: Konvertiert Tabulatoren in *n* Leerzeichen. Der Standardwert für *n* sind acht Leerzeichen.

+*n*
: Beginnt mit der Anzeige in Zeile *n*.

`more` verwendet alle Optionen, die gegebenenfalls in der Umgebungsvariablen `MORE` gesetzt sind.

Die Hilfe beschreibt zusätzlich die an der Eingabeaufforderung »-- Fortsetzung --« akzeptierten Befehle zur Steuerung der Anzeige.

Mit der *Leertaste* blättern Sie eine Seite vor, mit der *Enter*-Taste eine Zeile. Mit der Taste *Q* oder durch Betätigung der Tastenkombination *Strg+C* brechen Sie die Ausgabe ab. *P* gefolgt von einer Zahl *n* scrollt um *n* Zeichen weiter.

PoSh: PowerShell bietet als Äquivalent die Möglichkeit, die Ausgabe über den Pipelineoperator an den Befehl `out-host` weiterzureichen. Die Ausgabe des PowerShell-Befehls

```
get-childitem -rec | out-host –p
```

verhält sich grundsätzlich wie jene mit `more`: Durch Betätigung der *Leertaste* wird die Ausgabe um eine Bildschirmseite weitergeblättert, mit der *Enter*-Taste zeilenweise. Das funktioniert ebenso in der *Terminal*-App, jedoch nicht in der ISE-Umgebung.

Durch den Befehl `more` werden umgekehrt auch PowerShell-Bildschirmausgaben akzeptiert.

PATH

`PATH [`*`Pfad`*`]`

Zeigt den Suchpfad (eine durch Semikola getrennte Verzeichnisliste, in der die Eingabeaufforderung nach aufgerufenen Programmen sucht) an oder verändert ihn. Die Umgebungsvariable `%path%` kann verwendet werden, um den aktuellen Suchpfad in einen veränderten Suchpfad einzufügen:

`path %path%;`*`C:\NeuerOrdner\`*

Ein mit dem Befehl `path` geänderter Suchpfad wird nicht gespeichert. Die Änderung gilt nur für die aktuelle Instanz des Befehlszeileninterpreters und von dieser aus aufgerufene Instanzen. Für dauerhafte Änderungen verwenden Sie den Befehl `setx`.

PoSh: `$env:path` zeigt den aktuellen Pfad an.

Hinweis: Das vorübergehende Ändern der Pfadvariablen mit PowerShell-eigenen Befehlen wird beim Befehl `set` erläutert.

---

set

`set [/a] [/p] [Variable=[`*`Zeichenfolge`*`]]`

Mit dem Befehl `set` lassen sich Variablen für das aktive Befehlszeilenfenster anzeigen, setzen und ändern. Wird das Fenster geschlossen, werden die Änderungen verworfen.

`Variable`

Der Name der Variablen. Wird keine Variable angegeben, werden alle aktuellen Umgebungsvariablen angezeigt. Werden anstelle einer Variablen einzelne Zeichen oder Zeichenketten angegeben, werden diejenigen Variablen gezeigt, die mit diesen Zeichen beginnen.

`Zeichenfolge`

Der Wert, den die Variable annehmen soll.

`set currentpath="C:\Mein Programm;%PATH%"`

Setzt die Variable `currentpath` und verknüpft als deren Wert den Pfad *C:\Mein Programm* mit dem Inhalt der aktuellen `PATH`-Variablen.

`/a`

Gibt an, dass die Zeichen rechts des Gleichheitszeichens einen numerischen Ausdruck darstellen. Sie müssen in Anführungszeichen eingeschlossen, numerisch und mit gültigen Operatoren verknüpft sein. Die verfügbaren Operatoren sind ausführlich in der Hilfe beschrieben, zum Beispiel:

```
set /a test="2*4"
```

`/p`

Ermöglicht es, eine Variable durch eine Benutzereingabe zu definieren, zum Beispiel:

```
set /p keyword=
```

Mithilfe von `set` lässt sich so per Batchdatei eine speziell für eine Konsolenanwendung definierte Umgebung schaffen, die andere Bereiche von Windows nicht beeinflusst.

Tipp: Ausgabe einer Zeichenfolge mit vorangestelltem Zeitstempel:

Zunächst setzen Sie die Variable `jetzt`, die Datum, Uhrzeit und eine beliebige Zeichenfolge beinhaltet, mit folgendem Befehl:

`set jetzt=%date% %time% "`*`Zeichenfolge`*`"`

Mit `echo %jetzt%` lässt sich die so gesetzte Variable am Bildschirm anzeigen und mit den Operatoren `>` und `>>` in eine neue Datei schreiben bzw. an eine vorhandene anhängen.

PoSh: Das Auslesen der Umgebungsvariablen kann mit folgendem Befehl

geschehen:

```
get-childitem env:
```

Um eine einzelne Variable wie beispielsweise `username` auszulesen, verwenden Sie:

```
echo $env:username
```

Eine Variante zur Erweiterung der Pfadvariablen der aktuellen Umgebung um einen Ordner ist:

```
Set-Item –path env:path –value ($env:path + ";C:\Skript")
```

---

setx

setx [/s *System* **[/u [***Domäne***|***Computer***\]***Benutzer* **[/p [***Kennwort***]]]] var** *Wert* **[/m]**

setx [/s *System* **[/u [***Domäne***|***Computer***\]***Benutzer* **[/p [***Kennwort***]]]] var /k** *Registrierungspfad* **[/m]**

setx [/s *System* **[/u [***Domäne***|***Computer***\]***Benutzer* **[/p [***Kennwort***]]]] /f** *Datei* **{var {/a x,y | /r x,y** *Zeichenfolge***}[/m] | /x} [/d** *Trennzeichen*]

Setzt den Wert einer Umgebungsvariablen für den angemeldeten Benutzer oder die ganze Maschine (`/m`) auf dem lokalen oder einem entfernten Computer (`/s`). Der neue Wert der Variablen kann entweder direkt angegeben oder aus einem Registrierungswert (`/k`) gelesen werden. Die Variablen werden in die Registrierung geschrieben. Mit dem Befehl `setx` konfigurierte Variablen werden dauerhaft geändert, allerdings werden sie lokal erst in einer neu gestarteten Instanz des Befehlszeilenfensters aktiv, bei entfernten Systemen ist die Neuanmeldung des Benutzers erforderlich.

`setx` verarbeitet auch Dateien. Verwenden Sie `setx /?` für weitere Informationen.

Beispiele

Löschen einer Variablen:

```
setx Variablenname ""
```

Setzen einer Variablen:

```
setx Variablenname "Wert"
```

Setzen einer Variablen in der Systemumgebung:

```
setx Variablenname "Wert" /m
```

PoSh: Mit PowerShell greifen Sie auf Umgebungsvariablen über das Laufwerk `env:` zu:

```
get-childitem env:computername
```

zeigt den Wert der Variablen `computername` an.

Einfaches Voranstellen eines Dollarzeichens zeigt PowerShell an, dass es sich um eine Variable handelt:

```
$env:computername
```

Alternativ greifen Sie über .NET auf die Umgebungsvariablen zu, wodurch eine einfache Trennung von System- und Benutzervariablen ermöglicht wird:

```
[environment]::GetEnvironmentVariable("Tmp","machine")
[environment]::GetEnvironmentVariable("Tmp","user")
```

Um eine Variable zu ändern, muss die laufende PowerShell-Konsolensitzung unter einem Benutzerkonto mit ausreichenden Berechtigungen laufen, also gegebenenfalls als Administrator gestartet sein.

Eine einfache Form ist das Setzen von Variablen mit dem bereits angesprochenen Ausdruck ohne jegliches Cmdlet:

```
$env:variablenname = "Wert"
```

Diese Variante ist analog zum `set`-Befehl der Eingabeaufforderung auf die laufende Sitzung begrenzt.

Dauerhaft ändern können Sie eine Variable entweder durch Bearbeiten der Registrierung oder mit der Methode `SetEnvironment-Variable` des .NET Framework:

```
[Environment]::SetEnvironmentVariable("NeueVariable", "Neuer
```

Wert", "*user*")

Um auf diese Weise gesetzte Variablen in der Windows PowerShell zu sehen, kann es nötig sein, eine neue PowerShell-Sitzung zu starten.

Das letzte Beispiel demonstriert das Löschen einer Umgebungsvariablen:

```
[Environment]::SetEnvironmentVariable("Variablenname",$null,"User
```

Die in PowerShell 7 enthaltenen Befehle `get-variable` und `set-variable` sind bislang leider auf die Handhabung von PowerShell-eigenen Variablen beschränkt.

---

shutdown

shutdown [/i | /l | /s | /sg | /r | /g | /a | /p | /h | /e | /o] [/hybrid] [/soft] [/f] [/fw] [/m *Computer*] [/t *xxx*] [/d [p|u:]xx:yy [/c "*Kommentar*"]]

Der Befehl `shutdown` ermöglicht das kontrollierte Herunterfahren und den Neustart eines Computers, die Abmeldung des aufrufenden lokalen Benutzers und die Dokumentation des Vorgangs in der Ereignisanzeige.

Optionen

/i

Zeigt eine grafische Benutzeroberfläche an. Falls diese verwendet werden soll, muss das der erste Parameter sein.

/l

Meldet den aufrufenden Benutzer lokal ab – das entspricht dem Befehl `logoff`.

/s

Fährt das Betriebssystem herunter. Sie können auch eine Verknüpfung auf dem Desktop erzeugen, die den Befehl `shutdown /s /t 0` zum unmittelbaren Herunterfahren des Rechners aufruft.

/r

Startet das Betriebssystem neu.

/g

Startet das Betriebssystem neu und lädt danach die registrierten Anwendungen.

/p

Schaltet den lokalen Computer ohne Zeitüberschreitungswarnung ab.

/f

Erzwingt das Schließen offener Anwendungen ohne Warnung des Benutzers, wenn der Wert für die Zeitüberschreitung (Parameter `/t`) größer als 0 ist.

/fw

Bereitet den Computer in Kombination mit einer Option zum Herunterfahren für die Ausführung des nächsten Starts in der Firmwareoberfläche vor.

In der Zeit `xxx` bis zum Start des Vorgangs kann lokal angemeldeten Benutzern eine Meldung angezeigt und mit `/d` ein Grund in der Ereignisanzeige eingetragen werden.

Tipp: Dieser Befehl kann praktisch sein, wenn Sie beispielsweise in einer Remotesitzung einen Netzwerkkartentreiber aktualisieren, bei dessen Update die Gefahr besteht, die Verbindung bis zum nächsten Neustart des Systems zu verlieren. In dem Fall rufen Sie vor dem Installieren des Treiberupdates `shutdown /r /t` *`xxx`* mit einem ausreichenden Zeitpuffer *`xxx`* auf, und wenn alles gut gegangen ist, brechen Sie den Neustart mit `shutdown /a` ab. Andernfalls warten Sie einfach bis zum getriggerten Neustart und haben danach (hoffentlich) wieder Zugriff.

PoSh: Versucht man, einen nur noch eingeschränkt erreichbaren Remotecomputer neu zu starten, kann es passieren, dass *shutdown.exe* beispielsweise aufgrund eines RPC-Fehlers nicht mehr mit diesem kommunizieren kann. In einigen dieser Fälle hilft der Einsatz eines der folgenden PowerShell-Kommandos weiter, vorausgesetzt, der lokale Anwender hat auf dem Remotesystem Administratorrechte. Andernfalls können Anmeldedaten übergeben werden:

```
restart-computer -computer Computername -force
```

oder über WMI:

```
(gwmi win32_operatingsystem -ComputerName
Computername).Win32Shutdown(6)
```

Sollte das alles ebenfalls nicht funktionieren, hilft nur Turnschuhadministration, ein Anruf vor Ort oder ein eingerichtetes Intel ME für den Fernzugriff auf BIOS-Ebene, sofern der zu unterstützende Rechner Intel vPro im Lieferumfang beinhaltet, um den Stromschalter zu bedienen.

Das Cmdlet `Stop-Computer` dient dem Herunterfahren des lokalen Computers und entfernter Rechner.

---

sort

`sort [`*`Optionen`*`] [`*`Datei`***`]`**

Sortiert von der Standardeingabe oder aus dem Inhalt von `Datei` gelesene Textzeilen. Nach Groß- und Kleinschreibung wird dabei nicht unterschieden.

Optionen

`/l` *`Gebietsschema`*

Nutzt das angegebene Gebietsschema anstelle der Systemeinstellung. Der Wert `C` bietet die beste Performance und ist zum Zeitpunkt des Erscheinens dieses Buchs der einzige unterstützte Wert dieser Option. Nach Groß- und Kleinschreibung wird nicht unterschieden.

`/m` *`Kilobytes`*

Gibt die für die Ausführung des Befehls zu verwendende Menge an Arbeitsspeicher fest an, wobei das Minimum 160 KByte nicht unterschreiten kann. Die beste Performance wird in der Regel ohne Angabe einer solchen Begrenzung erzielt.

`/+n`

Beginnt ab dem `n`-ten Zeichen mit der Sortierung (die ersten Zeichen jeder Zeile werden dann ignoriert).

`/r`

Kehrt die Sortierreihenfolge um (also von Z bis A und dann von 9 bis 0).

/rec *n*

Definiert die maximale Anzahl von Zeichen (*n*) pro Datensatz. Der Standardwert ist 4.096, der Maximalwert 65.535.

[Laufwerk1:][Pfad1]Datei1

Gibt die zu sortierende Datei an. Ist diese nicht angegeben, wird die Standardeingabe als Quelle der zu sortierenden Zeilen verwendet.

/t *[Laufwerk2:][Pfad2]*

Verwendet das als Pfad angegebene Verzeichnis anstelle des Temp-Verzeichnisses als Arbeitsverzeichnis des Befehls.

/o *Pfad-der-Ausgabedatei*

Schreibt die sortierten Daten in die angegebene Datei anstatt in die Standardausgabe.

PoSh: Mit dem Befehl `sort-object` verfügt auch PowerShell über eine leistungsstarke Funktion zum Sortieren nicht nur von Text. Während der `sort`-Befehl der Eingabeaufforderung nicht in der Lage ist, Zahlen korrekt zu sortieren (beispielsweise würde eine aufsteigende Sortierfolge *1,11,2,20* statt *1,2,11,20* lauten), ist das mit PowerShell machbar. Der Befehl

```
get-content C:\Temp\sort.txt | sort-object {[void]($_ -match
'\d*');[int]$matches[0]}
```

leitet den Inhalt der Datei *sort.txt* an `sort-object` weiter und sorgt für eine numerisch korrekte Sortierung der Zeilen.

TIME

TIME [*std***[:***min***[:***sec***[.***hd***]]]] [/t]**

Stellt die angegebene Uhrzeit ein oder fragt danach, wenn nichts angegeben wurde. Mit dem Parameter `/t` wird die aktuelle Zeit nur angezeigt, was sich zur Ausgabeumleitung innerhalb von Skripten verwenden lässt.

PoSh: Das Auslesen und Konfigurieren der Systemzeit erfolgt in PowerShell mit `get-date` und `set-date`.

`get-date -format T`

Gibt die Systemzeit sekundengenau aus, wie auch `get-date displayhint time`.

`set-date hh:mm`

Setzt die Systemzeit, ohne das Datum zu ändern.

`set-date -adjust -0:01 -displayhint time`

Setzt die Systemzeit um eine Minute zurück und zeigt die neue Uhrzeit an.

VER

VER

Zeigt die Version des Betriebssystems an.

PoSh: Die magere Ausgabe des `VER`-Befehls zu übertrumpfen, verlangt von PowerShell keine besonderen Anstrengungen. Um die Bezeichnung des Betriebssystems und der Architektur, die Versionsnummer und das Servicepack auszugeben, genügen diese Befehle:

```
get-ciminstance win32_operatingsystem | select
Caption,OSArchitecture,Version
```

Alternativ ermitteln Sie das installierte Funktionsupdate über die Registrierung:

```
(Get-ItemProperty "HKLM:\SOFTWARE\Microsoft\Windows
NT\CurrentVersion").DisplayVersion
```

Zudem können Sie folgenden Befehl verwenden, um die Windows-Edition, also z.B. Professional oder Enterprise, festzustellen:

```
get-windowsedition -online
```

Dateien und Verzeichnisse

Dateien und Verzeichnisse sind bis heute Kernbestandteile der auf Windows-Computern gespeicherten Programme und Daten. Auch wenn Softwareanbieter mehr und mehr persönliche Daten in die Cloud auslagern oder auf verschiedenen Systemen synchronisieren, hilft nicht nur bei Problemfällen eine gute Kenntnis der Wege zum Finden und Behandeln der eigenen Dateien auf dem Computer.

ASSOC – nicht funktional

`ASSOC [`*`.erw=Dateityp`***`]`**

Assoziiert eine Dateierweiterung mit einem Dateityp (beispielsweise *txtfile*). Bei einem Aufruf ohne Parameter werden die aktuellen Zuordnungen aller Dateierweiterungen angezeigt, bei Aufruf mit einer Erweiterung deren spezielle Zuordnung, bei Aufruf mit Erweiterung, Gleichheitszeichen und ohne Typ wird die Zuordnung der Erweiterung gelöscht. Siehe auch `ftype`.

Warnung: Dieser Befehl wie auch `FTYPE` funktionieren beim Ändern von bestehenden Zuordnungen auf Clientsystemen derzeit nicht mehr korrekt. Grund ist, dass Microsoft zum einen den hierfür genutzten Pfad in der Registrierung geändert hat und zum anderen jeder vom Benutzer in der GUI ausgewählten Zuordnung ein Hash beigefügt wird. Da danach der Hash nicht mehr zur Zuordnung passt oder fehlt, funktioniert auch ein direktes Ändern des Eintrags in der Registrierung nicht. Eine bei kommerzieller Nutzung kostenpflichtige Alternative bietet das Tool *SetUserFTA.exe*.

PoSh: Das Auslesen und Konfigurieren von Zuordnungen ist aus obigem Grund mit PowerShell durch Bearbeitung der zugeordneten Einträge in der Registrierung nicht mehr möglich, sofern dem Skript nicht ein funktionierender Algorithmus zum Berechnen des Hashs mitgegeben werden kann.

attrib

attrib [*Optionen*] [*Dateien*]

Zeigt (bei Aufruf ohne Optionen) bzw. ändert die Dateiattribute angegebener Dateien, der Dateien im aktuellen Verzeichnis und von Verzeichnissen.

Optionen

+*x* | -*x*

Setzt oder löscht ein Attribut. *x* kann dabei einen der folgenden Werte annehmen: `r` (Schreibschutz), `h` (versteckt), `s` (System), `a` (Archiv), `o` (offline), `i` (nicht indiziert), `x` (keine Bereinigungsdatei), `v` (Integrität), `p` (angeheftet), u (losgelöst) sowie `b` (Attribut für SMR-BLOB).

/s

Der Befehl verarbeitet auch Dateien in allen Unterverzeichnissen.

/d

Verzeichnisse werden ebenfalls verarbeitet.

/l

Die Attribute von symbolischen Links werden anstelle der Attribute der Linkziele verändert.

PoSh: `Attrib` ist einer der Befehle der Eingabeaufforderung, bei dem das Ersetzen durch einen PowerShell-Befehl für Standardsituationen kaum sinnvoll erscheint, da er mit wenig Aufwand seine Aufgaben erfüllt. Das Kombinieren mit PowerShell kann jedoch hilfreich sein, um beispielsweise deren einzigartige Filter- und Weiterverarbeitungsmöglichkeiten zu nutzen.

Um die Attribute einer Datei direkt in PowerShell auszulesen, können Sie auf das Cmdlet `get-itemproperty` zurückgreifen. Dieses vielseitig einsetzbare Kommando lässt sich unter anderem auf Dateien anwenden:

```
get-itemproperty .\*.txt | format-table -property
fullname,attributes -auto
```

zeigt den vollen Pfad aller Textdateien im aktuellen Verzeichnis und deren Attribute an.

Um versteckte Dateien einschließlich derjenigen in Unterverzeichnissen mit anzuzeigen, lässt sich das Ergebnis von `get-childitem-recurse force` an `get-itemproperty` weiterleiten:

```
get-childitem *.txt -recurse -force | get-itemproperty |
format-list -property fullname,attributes
```

Ausschließlich Dateien mit dem Attribut »versteckt« (`hidden`) zeigen Sie mit folgendem Befehl an:

```
get-childitem -recurse -force | where {
($_.attributes.tostring() -split ", ") -contains "hidden"
} | select fullname
```

Um alle Attribute einer Datei zu entfernen, können Sie folgendes Konstrukt nutzen:

```
$(get-item .\dateiname.txt -force) .attributes='normal'
```

Statt `normal` können Sie durch Kommata voneinander getrennt die Attribute `Archive`, `ReadOnly`, `Hidden` und `System` zuweisen. Zuvor gesetzte Attribute werden im Gegensatz zum `attrib`-Befehl überschrieben. Um im aktuellen Verzeichnis alle Dateien mit der Erweiterung *.txt* mit dem Attribut »schreibgeschützt« zu versehen, kann eine `foreach`-Schleife zum Einsatz kommen:

```
get-item *.txt | foreach {$_.attributes='ReadOnly'}
```

CD
CHDIR

`CD [/d] [`*`Pfad`*`]`

Zeigt bzw. setzt das aktuelle Arbeitsverzeichnis. Wenn `Pfad` keinen Laufwerkbuchstaben enthält, wird das aktuelle Laufwerk verwendet. Die Option `/d` bewirkt, dass das aktuelle Arbeitsverzeichnis auf das angegebene Laufwerk gesetzt wird, anstatt nur das Arbeitsverzeichnis auf dem gegenwärtig ausgewählten Laufwerk einzustellen. Zwei Punkte (`..`) werden verwendet, um das darüberliegende Verzeichnis anzusprechen.

`CHDIR ist ein Synonym für CD.`

PoSh: In PowerShell gibt es den Alias `CD` für das Cmdlet `set-location`.

```
set-location -path c:\windows\system32
```

wechselt von einem beliebigen Laufwerk aus direkt in den Ordner *c:\windows\system32*, wie auch:

```
cd c:\windows\system32
```

---

cipher

cipher [/e | /d | /c] [/s:*Verzeichnis*] [/b] [/h] [*Pfadname* [...]]

Ermöglicht die Anzeige bzw. Änderung der Verschlüsselung von Ordnern oder Dateien auf NTFS-Partitionen (EFS, *Encrypted File System*).

Optionen

cipher [/c] *Pfad*

Zeigt den Verschlüsselungsstatus von `Pfad` an, bei Angabe von `/c` detailliert.

cipher {/e | /d} *Pfad*

Verschlüsselt (`/e`) oder entschlüsselt (`/d`) eine Datei oder ein Verzeichnis. Dateien, die verschlüsselten Verzeichnissen später hinzugefügt werden, werden ebenfalls verschlüsselt. Ist das übergeordnete Verzeichnis nicht verschlüsselt, können verschlüsselte Dateien nach Änderungen unentschlüsselt abgespeichert werden.

cipher {/e | /d} /s:*Verzeichnis*

Ver- bzw. entschlüsselt alle Unterverzeichnisse von `Verzeichnis`.

cipher /k [/ecc:*256*|*384*|*521*]

Erstellt ein neues EFS-Zertifikat samt Schlüssel, die den aktuellen Gruppenrichtlinien entsprechen. Bei Verwendung dieser Option werden alle anderen Optionen ignoriert. Bei Angabe von `ecc` erstellt `cipher` ein selbst signiertes Zertifikat mit der angegebenen bzw. verfügbaren

Schlüsselgröße.

Hinweis: Wenngleich bei **`/ecc`** von der Logik her 512 als Schlüsselgröße zu erwarten wäre, so ist hier tatsächlich der ebenfalls in der Hilfe aufgeführte Wert 521 als zugehöriger Parameter implementiert.

`cipher /r` *`Dateiname`* **`[/smartcard]`**

Generiert einen privaten Schlüssel und ein Zertifikat für den EFS-Wiederherstellungsagenten. Das Zertifikat wird in *Dateiname.cer* gespeichert. Falls die Option `/smartcard` nicht angegeben wurde, wird der Schlüssel in *Dateiname.pfx* gespeichert, ansonsten wird er ausschließlich zusammen mit dem Zertifikat auf einer Smartcard abgelegt.

`cipher /n /u`

Zeigt alle verschlüsselten Dateien auf den lokalen Laufwerken an. `/n` verhindert dabei, dass deren Schlüssel aktualisiert werden. Diese zwei Optionen können nicht mit weiteren kombiniert werden.

`cipher /u`

Aktualisiert die Schlüssel aller verschlüsselten Dateien mit den aktuellen Schlüsseln sowohl des Benutzers als auch des Wiederherstellungsagenten. Dies ist nach einer Schlüsseländerung notwendig, wenn beispielsweise der Wiederherstellungsagent geändert wurde.

`cipher /w:`*`Verzeichnis`*

Überschreibt alle gelöschten Daten im angegebenen Verzeichnis dreimal hintereinander mit Nullen, Einsen und Zufallszahlen. Wird als Zielverzeichnis nur der Laufwerkbuchstabe angegeben, werden alle freien Speicherbereiche des Laufwerks überschrieben. Dadurch wird eine Wiederherstellung von gelöschten Inhalten vom Datenträger mit Wiederherstellungswerkzeugen unmöglich gemacht.

`cipher /x[`*`:EFS-Datei`***`] [`***`Dateiname`***`]`**

Exportiert das EFS-Zertifikat und die Schlüssel in die angegebene Datei. Wird eine EFS-Datei (verschlüsselte Datei) angegeben, werden die zur Verschlüsselung verwendeten Zertifikate des angemeldeten Benutzers gesichert, ohne diese Angabe das aktuelle EFS-Zertifikat und die

Schlüssel des Benutzers.

`cipher /y`

Zeigt das aktuelle EFS-Zertifikat als Miniaturansicht auf dem Computer an.

Weitere Optionen

`/adduser [/certhash:`*`Hash`*` | /certfile:`*`Dateiname`*`] | /user:` *`Benutzername`*`]`

Fügt einen Benutzer den angegebenen verschlüsselten Dateien hinzu. Wird `/certhash` angegeben, sucht die Chiffrierung nach einem Zertifikat mit diesem Sha1-Hash. Wird `/certfile` angegeben, extrahiert `cipher` das Zertifikat aus der Datei. Bei Verwendung von `/user` wird versucht, das Benutzerzertifikat in den Active-Directory-Domänendiensten zu finden.

`/b`

Abbruch bei Fehler (standardmäßig wird im Fehlerfall fortgefahren).

`/h`

Bezieht auch versteckte Dateien und Systemdateien mit ein.

`/rekey`

Aktualisiert angegebene verschlüsselte Dateien für die Verwendung des konfigurierten und aktuellen Schlüssels.

`/removeuser /certhash:`*`hash`*

Entfernt einen Benutzer aus den angegebenen Dateien. `/cert-hash` muss der Sha1-Hash des zu entfernenden Zertifikats sein.

Werden keine Parameter angegeben, wird der Verschlüsselungsstatus des aktuellen Verzeichnisses und der darin befindlichen Dateien angezeigt. Weitere Optionen können der Hilfe des Befehls entnommen werden.

PoSh: PowerShell unterstützt ebenfalls den Umgang mit der Dateiverschlüsselung. Allerdings dürfte der hoch spezialisierte Befehl `cipher` in vielen Fällen vom Aufwand her sinnvoller einzusetzen sein, als auf Biegen und Brechen eine Alternative in PowerShell aufzubringen.

`encrypted` ist eines der Dateiattribute, die durch das Cmdlet `get-`

`itemproperty` mit angezeigt werden. Um also alle verschlüsselten Dateien innerhalb einer Verzeichnishierarchie anzuzeigen, genügt eine leichte Abwandlung des Beispiels zum Anzeigen versteckter Dateien:

```
get-childitem c:\efs -recurse -force | where {
($_.attributes.tostring() -split ", ") -contains
"encrypted" } | select fullname
```

Um eine Datei in Windows PowerShell zu ver- oder entschlüsseln, können Sie deren `encrypt`-Methode verwenden (vorausgesetzt, die EFS-Verschlüsselung ist in der installierten Betriebssystemversion/edition verfügbar):

```
(get-item c:\test\info.txt).encrypt()
```

Zum Entschlüsseln verwenden Sie die `decrypt`-Methode:

```
(get-item c:\test\info.txt).decrypt()
```

---

comp

comp [*Optionen*] *Dateigruppe1 Dateigruppe2*

Vergleicht zwei Gruppen von Dateien (oder einzelne Dateien). Wenn mehrere Dateien angegeben sind, werden Dateien mit jeweils gleichem Namen verglichen. Unterschiede werden nur für gleich große Dateien ausgegeben. Verwenden Sie `fc`, um Dateien unterschiedlicher Größe zu vergleichen.

Optionen

/a

Gibt die Unterschiede in ASCII-Darstellung aus. Standard ist die Ausgabe in Dezimaldarstellung.

/c

Vergleicht, ohne auf Groß- und Kleinschreibung zu achten.

/d

Gibt die Unterschiede in dezimaler Darstellung aus.

/l

Zeigt die Zeilennummern der unterschiedlichen Zeilen an.

/m

Für den Vergleich werden keine weiteren Dateien angefordert.

/n=*n*

Vergleicht nur die ersten *n* Zeilen jeder Datei.

/off

Dateien mit gesetztem Offlineattribut werden nicht ausgeklammert.

PoSh: Das Vergleichen von Dateien in PowerShell kann mit dem Cmdlet `compare-object` realisiert werden. Das folgende Beispiel vergleicht die Inhalte der Dateien *info1.log* und *info2.log* und listet die unterschiedlichen Zeilen auf.

```
compare-object -referenceobject (get-content info1.log)
-differenceobject (get-content info2.log)
```

---

compact

compact [*Optionen*] [*Dateien* | *Verzeichnis*]

Verwaltet die NTFS-Dateikompression, (de)komprimiert Dateien, stellt den Standard für Verzeichnisse ein und zeigt den Komprimierungsgrad der Dateien an. Wenn keine Dateien angegeben wurden, wird das aktuelle Verzeichnis mit den darin enthaltenen Dateien verwendet.

Optionen

/c | /u

Gibt an, ob komprimiert (`/c`) oder dekomprimiert (`/u`) werden soll.

/s[:*Verzeichnis***]**

Der Befehl wirkt sich auch auf alle Unterverzeichnisse und die darin enthaltenen Dateien aus.

/f

Erzwingt die Komprimierung bereits komprimierter Dateien.

Standardmäßig werden diese Dateien übersprungen.

`/i`

Setzt den Vorgang fort, auch wenn ein Fehler auftritt.

`/q`

Deaktiviert den ausführlichen Anzeigemodus.

`/a`

Zeigt auch Dateien an, die die Eigenschaften »versteckt« (`H`) oder »System« (`S`) haben. Diese Dateien werden normalerweise nicht angezeigt, obwohl sich der Befehl auf sie auswirkt.

`/exe`

Verwendet eine für ausführbare Dateien optimierte Komprimierung.

`/compactos`

Legt den Zustand der Komprimierung des Betriebssystems fest oder fragt ihn ab.

`/windir`

Wird mit `/compactos:query` zur Abfrage eines Offlinebetriebssystems verwendet.

PoSh: PowerShell kann zwar den Komprimierungsstatus von Dateien über deren Attribut auslesen, beinhaltet aber derzeit keine native Funktion, dieses Attribut zu setzen. Folgerichtig lässt sich diese Aufgabe in Kombination mit dem `compact`-Befehl der Eingabeaufforderung realisieren:

```
get-childitem -file -path "C:\test" -recurse | ?
{$_.attributes -notlike "*compressed*" | foreach {compact
/C $_.FullName}
```

COPY

`COPY [`*`Optionen`*`]` *`Quelle Ziel`*

Kopiert Dateien von *`Quelle`* nach *`Ziel`*. Wenn *Ziel* eine einzelne Datei ist, werden alle in *`Quelle`* genannten Dateien aneinandergehängt. Das Aneinanderhängen von Dateien kann auch explizit eingestellt werden; verwenden Sie dazu diese Syntax für die Quelle: `Datei1 + Datei2 + ...`

Optionen

`/a | /b`

Kennzeichnet entweder ASCII- (`/a`) oder Binärdaten (`/b`) und wird einem Quelldateinamen vorangestellt bzw. an den Zieldateinamen angehängt.

`/v`

Überprüft, ob die kopierten Daten richtig geschrieben wurden.

`/n`

Verwendet den Kurznamen (wenn vorhanden) beim Kopieren einer Datei mit langem Dateinamen.

`/d`

Erlaubt die Entschlüsselung von mit EFS verschlüsselten Dateien beim Kopieren auf entfernte Systeme. Wird diese Option nicht angegeben oder ist dem Anwender die Entschlüsselung nicht erlaubt, werden verschlüsselte Dateien nicht auf entfernte Systeme kopiert.

`/z`

Kopiert Dateien im Netz; eine unterbrochene Operation kann fortgesetzt werden.

`/y | /-y`

Unterdrückt (`/y`) oder verlangt (`/-y`) eine Bestätigung beim Überschreiben von Dateien.

`/l`

Kopiert symbolische Verknüpfungen statt deren Ziele.

`COPY CON` *`Dateiname`* `übernimmt die Eingabe von der Tastatur in die Datei` *`Dateiname`*`; mit der Funktionstaste` *`F6`* `wird die Eingabe abgeschlossen und die Datei gespeichert.`

Das Anfordern der Bestätigung für das Überschreiben kann mit der Umgebungsvariablen `COPYCMD` eingestellt werden. Um ganze Verzeichnisstrukturen und Eigenschaften wie Dateiattribute zu kopieren, verwenden Sie die Befehle `xcopy` oder `robocopy`.

PoSh: Der PowerShell-Befehl `copy-item` lässt sich über den Alias `copy` in der PowerShell-Umgebung ansprechen. Im Unterschied zum `copy`-Befehl

der Eingabeaufforderung beherrscht dieser den Umgang mit Verzeichnissen und mit weiteren kopierbaren Objekten. Der folgende Befehl kopiert die Verzeichnisstruktur einschließlich der im Quellverzeichnis enthaltenen Dateien mit der Erweiterung *.txt*. Der Ordner *ziel* wird erzeugt, falls er nicht vorhanden ist.

```
copy-item -rec -filter *.txt c:\quelle d:\ziel
```

DEL

DEL [*Optionen*] *Dateien*

Löscht Dateien. ERASE ist ein Synonym für DEL.

Optionen

/s

Der Befehl wirkt sich auch auf die in Unterverzeichnissen enthaltenen Dateien aus.

/q

Unterdrückt alle Bestätigungsaufforderungen.

/p

Verlangt eine Bestätigung für jeden Löschvorgang.

/f

Erzwingt das Löschen schreibgeschützter Dateien.

/a:*Attribute*

Beschränkt die Operation auf Dateien mit den folgenden angegebenen Attributen: H (versteckt), S (System), R (schreibgeschützt), A (Archiv), I (nicht indiziert), L (Reparse Points). Wird dem Attribut ein - vorangestellt, werden die mit diesem Attribut versehenen Dateien von der Operation ausgeklammert.

Warnung: Mit DEL gelöschte Dateien lassen sich nicht über den Windows-Papierkorb wiederherstellen.

PoSh: PowerShell kennt den Alias `DEL` für das Cmdlet `remove-item`. Die Stärke dieses Befehls besteht darin, dass sich damit Dateien, Verzeichnisse und sogar Registrierungsschlüssel löschen lassen.

`remove-item -rec` *`c:\test`*

löscht die gesamte Ordnerstruktur *c:\test* ohne weitere Rückfrage. Dateiattribute wie `readonly` werden allerdings berücksichtigt, solange nicht der Parameter `-force` verwendet wird.

DIR

`DIR [`*`Optionen`*`] [`*`Pfad`***`]`**

Zeigt den Inhalt des aktuellen Verzeichnisses bzw. des angegebenen Pfads an.

Optionen

Optionen können in der Umgebungsvariablen `DIRCMD` angegeben werden. Um die dort angegebenen Optionen zu deaktivieren, geben Sie sie in der Befehlszeile mit einem führenden Minuszeichen an, z.B. `/-n`.

Tipp: Häufig sollen mit `dir` alle Dateien und Verzeichnisse angezeigt werden (auch versteckte usw.). Dazu kann der Befehl `dir /a [Pfad]` verwendet werden.

`/b`

Zeigt nur die Datei- bzw. Verzeichnisnamen an, Größen- und Datumsinformationen sowie Kopf- und Fußzeilen werden weggelassen; hilfreich für die Verwertung durch weitere Programme.

`/w | /d`

Zeigt mehrere Datei- bzw. Verzeichnisnamen pro Ausgabezeile an, sortiert in Zeilen (`/w`) oder Spalten (`/d`).

`/q`

Gibt den Besitzer der Dateien an.

/r

Zeigt alternative Datenströme (ADS) an.

/n

Ausführliche, UNIX-ähnliche Darstellung. Die Dateinamen stehen auf der rechten Seite. Das ist die Standardeinstellung.

/l

Zeigt alle Namen in Kleinbuchstaben an.

/x

Zeigt zusätzlich zu den langen Dateinamen auch die Namen im 8.3-Format an.

/s

Zeigt auch den Inhalt aller Unterverzeichnisse an.

/o:*Sortierkriterium*

Legt die Sortierreihenfolge der Verzeichniseinträge fest. Die folgenden Kürzel werden verwendet: `n` (Namen), `e` (Erweiterungen), `s` (Größe), `d` (Datum und Uhrzeit) und `g` (Verzeichnisse zuerst). Ein Minuszeichen vor dem jeweiligen Kürzel kehrt die normale Sortierreihenfolge um.

/t:*Zeittyp*

Definiert, welche Zeit angezeigt und zur Sortierung verwendet wird: `c` (Erstellung), `a` (letzter Zugriff) oder `w` (letzter Schreibzugriff, Standardwert).

/a:*Attribute*

Beschränkt die Anzeige auf Dateien mit den folgenden angegebenen Attributen: `D` (Verzeichnis), `H` (versteckt), `S` (System), `R` (schreibgeschützt), `A` (Archiv), `I` (nicht indiziert), `L` (Reparse Point). Um Dateien mit bestimmten Eigenschaften auszuschließen, geben Sie ein Minuszeichen vor dem Kürzel des entsprechenden Attributs an. Die Verwendung von `/a` ohne angegebene Attribute zeigt alle zutreffenden Dateien unabhängig vom Attribut an.

/c | /-c

Tausendertrennzeichen in Dateigrößen anzeigen (`/c`, Standardwert) bzw. nicht anzeigen (`/-c`).

/4

Zeigt das Jahr vierstellig an.

`/p`

Zeigt die Ausgabe seitenweise an.

PoSh: In PowerShell ruft der Alias `DIR` das Cmdlet `get-childitem` auf.

```
get-childitem -rec | out-host -p
```

listet alle (nicht versteckten) Dateien im aktuellen Verzeichnis einschließlich der in Unterverzeichnissen auf und wartet nach jeder Bildschirmseite auf einen Tastendruck.

---

dirquota

`dirquota` *`Hauptbefehl Unterbefehl`*`(e) [`*`Optionen`*`]`

Dieser als veraltet geltende Befehl verwaltet Dateisystemkontingente (»Quoten«). Die gemeinsamen Optionen mehrerer Unterbefehle sind unter *`Optionen`* aufgeführt. Auf die Beschreibung der globalen Optionen (`dirquota admin`) wird hier verzichtet. Stattdessen wurden einige Beispiele aufgenommen.

dirquota quota

Verwaltet Quoten, die Verzeichnissen individuell zugewiesen wurden.

`list`

Zeigt vorhandene Kontingente an, die optional unter anderem nach folgenden Kriterien gefiltert werden können: `/MinUsed`, `/Path`, `/SourceTemplate`, `/Status`, `/TemplateMatch`, `/Type`.

`add /Path:`*`Verzeichnis`* **`{/SourceTemplate:`***`Vorlage`* **`| /Limit:`***`Grenze`***`} [{/overwrite | /ignore}]`**

Fügt einem Verzeichnis ein Kontingent hinzu, dessen Einstellungen einer Vorlage entnommen oder direkt angegeben werden. Zusätzlich können unter anderem `/Type` und `/Status` spezifiziert werden. Falls das Kontingent schon vorhanden ist, kann es überschrieben (`/overwrite`) oder übersprungen (`/ignore`) werden.

modify /Path:*Verzeichnis*

Ändert das Kontingent auf *Verzeichnis*. Unter anderem können folgende Eigenschaften verändert werden: /Limit, /SourceTemplate, /Status, /Type.

delete /Path:*Verzeichnis* **[/quiet]**

Löscht das einem Verzeichnis zugewiesene Kontingent, bei Angabe von /quiet ohne Warnung.

dirquota autoquota

Verwaltet automatisch zugewiesene Quoten, die auf ein übergeordnetes Verzeichnis gesetzt und von dort auf Unterverzeichnisse vererbt werden.

list

Zeigt vorhandene Kontingente an, die optional unter anderem nach folgenden Kriterien gefiltert werden können: /Path, /Type, /SourceTemplate, /TemplateMatch.

add /Path:*Verzeichnis* **/SourceTemplate:***Vorlage*

Fügt einem Verzeichnis ein Kontingent hinzu, dessen Einstellungen einer Vorlage entnommen werden.

modify /Path:*Verzeichnis* **/SourceTemplate:***Vorlage*
[/Apply:{*none* | *all* | *matching*}]

Ändert die einem Verzeichnis zugewiesene Autoquota-Vorlage. Optional kann die neue Vorlage auf alle Unterverzeichnisse angewendet werden (/Apply:all) oder nur auf diejenigen, deren Kontingente mit der ursprünglichen Vorlage übereinstimmen.

delete /Path:*Verzeichnis* **[/quiet]**

Löscht das einem Verzeichnis zugewiesene Kontingent, bei Angabe von /quiet ohne Warnung.

scan /Path:*Verzeichnis*

Initiiert einen Scanvorgang zur Aktualisierung des belegten Speicherplatzes in Verzeichnissen mit Kontingenten.

freespace /Path:*Verzeichnis*

Zeigt den freien Speicherplatz in *Verzeichnis* unter Berücksichtigung

der darauf gesetzten Quoten an.

dirquota template

Verwaltet Kontingentvorlagen.

`list`

Zeigt vorhandene Vorlagen an, die optional unter anderem nach den Kriterien `/Template` und `/Type` gefiltert werden können.

`add /Template:`*`Vorlage`* **`{/SourceTemplate:`***`QuellVorlage`* **`| /Limit:`***`Grenze`***`}`**

Erstellt eine neue Vorlage, deren Einstellungen einer *`QuellVorlage`* entnommen oder direkt angegeben werden. Zusätzlich kann unter anderem `/Type` spezifiziert werden.

`modify /Template:`*`Vorlage`* **`[/New-Template:`***`NeuerName`***`]`**

Ändert eine Vorlage. Bei Angabe von `/New-Template` wird sie in *`NeuerName`* umbenannt. Zusätzlich können unter anderem `/Limit`, `/SourceTemplate` und `/Type` spezifiziert werden.

`delete /Template:`*`Vorlage`* **`[/quiet]`**

Löscht eine Vorlage, bei Angabe von `/quiet` ohne Warnung.

`export /File:`*`Datei`* **`[/Template:`***`Vorlage`***`]`**

Exportiert alle oder nur die angegebene Vorlage in eine Datei.

`import /File:`*`Datei`* **`[/Template:`***`Vorlage`***`] [{/overwrite | /ignore}]`**

Importiert alle oder nur die angegebene Vorlage aus einer Datei. Falls schon eine Vorlage vorhanden ist, kann sie überschrieben (`/overwrite`) oder übersprungen (`/ignore`) werden.

Optionen

`/Limit:`*`Zahl`***`{`*****`kb | mb | gb`*****`}`**

Gibt die Grenze (das Kontingent) an.

`/MinUsed:`*`Zahl`***`{% | `*****`kb | mb | gb`*****`}`**

Gibt den Minimalbedarf prozentual oder absolut an.

`/Path:`*`Verzeichnis`*

Spezifiziert ein Verzeichnis. Durch Anhängen von `\*` werden alle unmittelbaren Unterverzeichnisse eingeschlossen, durch `\...` alle Unterverzeichnisse rekursiv auf allen Ebenen.

/Remote:*Computer*

Führt den Vorgang auf einem entfernten Computer statt auf dem lokalen System aus.

/SourceTemplate:*Vorlage*

Verweist auf eine Vorlage. Vorlagen können mit `dirquota template` verwaltet werden.

/Status:{*enabled* | *disabled*}

Quoten haben den Status »aktiviert« bzw. »deaktiviert«.

/Template:*Vorlage*

Verweist auf eine Vorlage.

/TemplateMatch:{*yes* | *no*}

Legt fest, ob Kontingente mit der Quellvorlage übereinstimmen sollen.

/Type:{*hard* | *soft*}

Spezifiziert harte bzw. weiche Kontingente.

Beispiele

```
dirquota template add /template:"Home 10 GB" /limit:10GB
/type:hard
```

Erstellt eine Vorlage namens *Home 10 GB* mit einer festen Kontingentgrenze von zehn GByte.

```
dirquota quota add /path:"D:\Data\Users\Home\NutzerA"
/sourcetemplate:"Home 10 GB"
```

Weist dem Verzeichnis *NutzerA* die vorher erstellte Vorlage zu.

```
dirquota autoquota add /path:"D:\Data\Users\Home"
/sourcetemplate: "Home 10 GB"
```

Besser für Home-Verzeichnisse geeignete Variante des vorherigen Beispiels: Auf den Stamm der Home-Verzeichnisse wird ein sich automatisch auf die Unterverzeichnisse vererbendes Kontingent gesetzt. Auf diese Weise müssen keine Kontingente für die individuellen Benutzerverzeichnisse verwaltet werden.

Hinweis: Um auf diesen Befehl zugreifen zu können, muss unter Windows Server 2025 der Rollendienst *Ressourcen-Manager für Dateiserver* installiert sein. Unter Windows 11 ist das optionale Feature *RSAT: Tools für Dateidienste* hinzuzufügen.

PoSh: Die PowerShell-Cmdlets zur Verwaltung von Kontingenten finden Sie, indem Sie in PowerShell nach Cmdlets mit dem Namensbestandteil `*FSRM*` suchen.

Das folgende Beispiel erzeugt ein Kontingent für jedes existierende und künftig erstellte Unterverzeichnis von *D:\Home* basierend auf der Vorlage `2 GB erweiterte Grenze` und wendet dieses automatisch an:

```
new-fsrmautoquota -path "D:\Home" -Template "2 GB
erweiterte Grenze"
```

---

du WWW

du [-*c*] [-*l* *Tiefe* | -*n* | -*v*] [-*u*] [-*q*] *Verzeichnis*

Das Programm du (*Disk Usage*) von Sysinternals zeigt den vom angegebenen *Verzeichnis* belegten Speicherplatz an. Dabei können Sie mit `-c` die Ausgabe in einer Form darstellen, die durch Ausgabeumleitung geeignet für die Erzeugung einer CSV-Datei ist. Mit `-l` legen Sie fest, dass der durch Unterverzeichnisse bis in die durch *Tiefe* angegebene Hierarchieebene belegte Speicherplatz für jedes dieser Verzeichnisse angezeigt wird, `-n` unterbindet die Einbeziehung von Unterverzeichnissen, `-v` zeigt die Details aller Unterverzeichnisse. Mit `-q` unterbinden Sie die Anzeige der Informationen zum Programm.

---

expand

expand [*Optionen*] Quelle [*Ziel*]

Entpackt Dateien aus CAB-Archiven.

Optionen

`-r`

Entpackte Dateien werden umbenannt.

`-i`

Entpackte Dateien werden ins Zielverzeichnis entpackt, die ursprüngliche Verzeichnisstruktur des Archivs wird aber ignoriert.

`-d`

Zeigt in der Quelldatei enthaltene Dateien an.

`Quelle`

Gibt die Quelldatei an.

`-f:`*`Dateien`*

Name(n) der Datei(en), die aus dem Archiv entpackt werden soll(en).

`Ziel`

Zieldatei oder verzeichnis. Falls die Quelle mehrere Dateien umfasst und `-r` nicht angegeben ist, muss das Ziel ein Verzeichnis sein.

---

fc

`fc [`*`Optionen`*`]` *`Dateigruppe1 Dateigruppe2`*

Vergleicht Dateien oder Dateigruppen und zeigt die Unterschiede an. Wenn mehrere Quelldateien angegeben wurden, werden sie mit gleichnamigen Dateien der zweiten Dateigruppe verglichen.

Optionen

`/b | /l | /u`

Vergleicht die Dateien als Binärdateien (`/b`), als ASCII-Textdateien (`/l`) oder als Unicode-Textdateien (`/u`). Die Option `/b` kann mit keiner anderen Option kombiniert werden.

`/c`

Vergleicht, ohne auf Groß- und Kleinschreibung zu achten.

/w

Fasst mehrere aufeinanderfolgende Tabulator- bzw. Leerzeichen vor dem Vergleich zusammen.

/t

Wandelt Tabulatorzeichen nicht in Leerzeichen um.

/lb*n*

Stellt die Maximalanzahl aufeinanderfolgender ungleicher Zeilen ein.

/nnn

Stellt die Anzahl aufeinanderfolgender Zeilen ein, die nach einer Abweichung übereinstimmen müssen, bevor die Dateien wieder als synchron angesehen werden. Der Standardwert ist 2.

/a

Beschränkt die Anzeige auf die erste und letzte Zeile jeder Abweichung.

/n

Zeigt bei Textdateien Zeilennummern an.

/off

Schließt Offlinedateien mit ein.

PoSh: Für Vergleiche mit PowerShell können Sie auf das Cmdlet `compare-object` zurückgreifen.

Folgender Einzeiler listet Unterschiede der zwei genannten Textdateien auf:

```
compare-object $(get-content c:\daten\abc.txt) $(get-content c:\daten\def.txt)
```

Im Beispiel wird der Inhalt beider Textdateien miteinander verglichen. Allerdings werden keine Unterschiede gefunden, wenn Inhalte generell gleich und nur in der Reihenfolge der Zeilen vertauscht sind.

FTYPE

FTYPE [*Dateityp=Öffnen-Befehl*]

Bearbeitet die Zuordnungen von Dateitypen zu *Öffnen*-Befehlen. Ohne Angabe von Parametern werden die aktuellen Zuordnungen angezeigt. Bei

Angabe eines Dateityps wird dessen *Öffnen*-Befehl angezeigt, bei Angabe von *Öffnen-Befehl* wird dieser dauerhaft gesetzt. Siehe auch `assoc`.

Im Zusammenspiel mit `assoc` legen Sie mit `ftype` fest, welches Programm geöffnet wird, wenn eine Datei mit einer bestimmten Erweiterung per Doppelklick oder durch Eingabe des Namens in der Befehlszeile aufgerufen wird.

Beispiel: Erstellen einer neuen Zuordnung für die Dateierweiterung *.ulog*, die mit dem `type`-Befehl seitenweise am Bildschirm ausgegeben werden soll.

```
assoc .ulog=UlogDateiZeigen
ftype UlogDateiZeigen=cmd /c "type %1 | more"
```

---

makecab

makecab [/v[n]] [/d var=*Wert* **...] [/l dir] Quelle [***Ziel***]**
makecab [/v[*n***]] [/d var=***Wert* **...] /f** *Anweisungsdatei* **[...]**

Erzeugt aus der als Quelle angegebenen Datei eine komprimierte Datei. Wird kein Zieldateiname angegeben, wird der letzte Buchstabe der Dateinamenserweiterung durch einen Unterstrich ersetzt.

Optionen

/d

Legt den Wert einer Variablen `var` fest.

/f

Gibt eine Anweisungsdatei an, die wiederholt werden kann. Die Anweisungen sind im Microsoft Cabinet SDK erläutert.

/l

Gibt einen abweichenden Verzeichnispfad für die Zieldatei an. Standardmäßig wird das Verzeichnis der Quelldatei verwendet.

/v[*n***]**

Definiert die Ausführlichkeit der angezeigten Meldungen.

Platzhalter werden von `makecab` nicht unmittelbar unterstützt. Um dennoch mehrere Dateien in einem Rutsch zu komprimieren, können Sie den Aufruf von `makecab` in einer Batchdatei verpacken, deren Inhalt wie folgt beschaffen ist:

```
REM Dateiname compress.cmd
@echo off
for %%a in (%1) do MAKECAB %%a /l .\archiv
```

Die Stapeldatei wird mit einem Parameter für zu komprimierende Dateien aufgerufen, zum Beispiel `compress *.log`. Dieser Aufruf würde ein Unterverzeichnis *archiv* unterhalb des aktuellen Verzeichnisses erzeugen und darin die komprimierten Dateien ablegen.

Der Befehl `expand` kann zum Entpacken von mit `makecab` komprimierten Dateien genutzt werden.

PoSh: In Windows PowerShell stehen zur Erzeugung und zum Entpacken komprimierter Dateien die Cmdlets `compress-archive` und `expand-archive` bereit. Im Unterschied zu `makecab` werden durch die PowerShell-Werkzeuge *zip*-Dateien erstellt bzw. entpackt.

```
compress-archive -path C:\Logs\* -CompressionLevel
optimal -destinationpath C:\Archiv\log.zip
```

erzeugt aus alle Dateien im Verzeichnis *C:\Logs* und dessen Unterverzeichnissen eine komprimierte Datei im Verzeichnis *C:\Archiv*. Das Zielverzeichnis muss bereits bestehen.

```
expand-archive –literalpath C:\Archiv\Log.zip –
destinationpath C:\Logs
```

entpackt die Zipdatei in das mit `destinationpath` angegebene Zielverzeichnis.

MD

`MD` *`Verzeichnis`*

Legt das angegebene *`Verzeichnis`* und bei aktivierten

Befehlserweiterungen alle fehlenden dazwischenliegenden Unterverzeichnisse an. `MKDIR` ist ein Synonym für `MD`.

PoSh: Der Befehl `MD` wird als PowerShell-Alias bereitgestellt und kann analog verwendet werden.

Alternativ nutzen Sie den Befehl `new-item`, um ein neues Verzeichnis zu erstellen:

```
new-item -path .\Test -itemtype "directory"
```

legt im aktuellen Verzeichnis ein Unterverzeichnis mit dem Namen *Test* an.

MOVE

`MOVE [`*`/y | /-y`*`]` *`Dateien Ziel`*

Verschiebt Verzeichnisse und Dateien in ein anderes Verzeichnis. Die Option `/y` unterdrückt Bestätigungsaufforderungen zum Überschreiben bestehender Zieldateien, `/-y` hebt diese Option auf, falls sie in der Umgebungsvariablen `COPYCMD` voreingestellt ist. Sind zu verschiebende Dateien oder Verzeichnisse in Benutzung, wird der Vorgang unvollendet abgebrochen.

PoSh: Mit dem Alias `move` oder dem PowerShell-Cmdlet `move-item` verschieben Sie ausgewählte Objekte, unter anderem auch Dateien und Verzeichnisse.

```
move-item test test2
```

verschiebt ein Objekt mit dem Namen *test* in das existierende Verzeichnis *test2*. Existiert das Verzeichnis *test2* nicht, entspricht der Vorgang einem Umbenennen des Originalobjekts.

---

movefile WWW

`movefile` *`Gesperrte-Datei Ersatz-Datei`*

Dieses Sysinternals-Tool weist das Betriebssystem an, eine gesperrte Datei beim nächsten Start zu ersetzen.

movefile *Gesperrte-Datei ""*

Löscht eine gesperrte Datei beim nächsten Start.

---

pendmoves WWW

pendmoves

Dieses Sysinternals-Tool zeigt an, welche Dateien beim nächsten Betriebssystemstart ersetzt oder gelöscht werden. Siehe auch `movefile`.

RD

RD [*Optionen***]** *Pfad*

Löscht ein Verzeichnis oder ganze Verzeichnisbäume. `RMDIR` ist ein Synonym für `RD`. Dateiattribute, NTFS-Berechtigungen des aufrufenden Benutzers und durch Benutzer oder Programme offen gehaltene Elemente können das vollständige Löschen einer Verzeichnisstruktur verhindern.

Optionen

`/s`

Löscht das Verzeichnis samt aller Unterverzeichnisse und Dateien.

`/q`

Unterdrückt die Bestätigungsaufforderung bei Verwendung von `/s`.

Warnung: Mit `RD` gelöschte Verzeichnisse lassen sich nicht über den Papierkorb wiederherstellen.

PoSh: Mit dem Cmdlet `remove-item` können nicht nur Dateien, sondern auch ganze Verzeichnisstrukturen gelöscht werden.

`remove-item -rec` *Pfad* `-force`

löscht ohne weitere Rückfrage das als *Pfad* angegebene Verzeichnis rekursiv, also inklusive aller Unterverzeichnisse und enthaltenen Dateien.

PowerShell kann jedoch auch selektiv vorgehen: Wollen Sie beispielsweise in einem Projektordner (der Pfad ist in der Variablen `$folder` hinterlegt) mit vielen Unterverzeichnissen alle Ordner mit dem Namen *debug* löschen, können Sie folgende Syntax anwenden (nach Weglassen von `-whatif`):

```
get-childitem$folder -filter 'debug' -recurse -force | ?
{$_.psIsContainer} | remove-item -force -recurse -whatif
```

Und hier die Kurzfassung:

```
gci $folder -fil 'debug' -r -fo | ? {$_.psIsContainer} |
ri -fo
-recurse -whatif
```

---

recover

`recover` *`Dateiname`*

Liest eine Datei Sektor für Sektor ein und stellt die lesbaren Sek-toren wieder her. Dieser Befehl kann verwendet werden, um noch lesbare Teile von Dateien eines defekten Datenträgers zu retten. Er sollte nur in äußersten Ausnahmesituationen zum Einsatz kommen, besser ist es, auf Datenrettung spezialisierte Software dafür einzusetzen.

REN

`REN` *`Alter-Name Neuer-Name`*

Benennt die angegebene(n) Datei(en) um. `RENAME` ist ein Synonym für `REN`. Der Speicherort der Zieldatei bleibt dabei unverändert. Wollen Sie ihn ändern, verwenden Sie den Befehl `move`.

PoSh: Mit `rename-item` (Alias `ren`) bringt PowerShell auch für diese Aufgabenstellung ein geeignetes Werkzeug mit. Die erweiterten Möglichkeiten von PowerShell geben dem Cmdlet `rename-item` erheblich mehr Flexibilität.

Folgender Zweizeiler benennt alle JPG-Dateien im angegebenen Ordner nach dem Schema *Foto202503* plus fortlaufende Nummer aufwärts von *001* an um:

```
$global:z=202503001
dir 'c:\fotos\test' -filter *.jpg | rename-item -newname
{"Foto{0}.jpg" -f $global:z++}
```

---

replace

replace *Quelldateien Ziel* **[*Optionen*]**

Ersetzt bzw. aktualisiert Dateien im Zielverzeichnis.

Optionen

/a

Fügt neue Dateien dem Zielverzeichnis hinzu (kann nicht mit /u oder /s kombiniert werden).

/u

Aktualisiert nur diejenigen Zieldateien, die älter als die entsprechenden Quelldateien sind.

/s

Führt den Befehl auch in allen Unterverzeichnissen aus.

/r

Ersetzt auch schreibgeschützte Dateien.

/p

Erwartet Bestätigung für jedes Ersetzen bzw. Hinzufügen einer Datei.

---

robocopy

robocopy *Quelle Ziel* **[***Dateien***] [*Optionen*]**

Äußerst leistungsfähiges und zuverlässiges Tool zum Kopieren und

Synchronisieren. *Quelle* und *Ziel* sind Verzeichnisse, die auch als UNC-Pfade angegeben werden können. Die zu kopierenden *Dateien* können mit Platzhaltern eingeschränkt werden (Standard: `*.*`). Hier werden nur die wichtigsten Optionen beschrieben.

Optionen

`/s | /e`

Kopiert alle Unterverzeichnisse (`/e`) bzw. alle nicht leeren Unterverzeichnisse (`/s`).

`/b`

Verwendet den Sicherungsmodus (wie ein Backup-Programm): Sofern der angemeldete Benutzer auf dem Quellserver über das Privileg *SeBackupPrivilege* (Sichern von Dateien und Verzeichnissen) verfügt, kann er auf alle Dateien zugreifen, unabhängig von den gesetzten Berechtigungen.

`/copyall`

Kopiert zusätzlich zu Daten, Attributen und Zeitstempeln auch die Sicherheitsbeschreibungen (Berechtigungen, Besitzer, Überwachungseinstellungen) der Dateien. Entspricht `/copy:DATSOU`.

`/efsraw`

EFS-verschlüsselte Dateien werden im EFS-RAW-Modus kopiert.

`/fft`

Geht von FAT-Uhrzeitangaben für Dateien aus, womit Abweichungen des Zeitstempels von bis zu zwei Sekunden toleriert werden.

`/j`

Kopiert ohne E/A-Pufferung. Empfohlen bei sehr großen Dateien, gerade auch beim Kopieren auf Netzlaufwerke, falls der Kopiervorgang andernfalls nach einer Zeit abbricht.

`/lev:`*`n`*

Kopiert nur die obersten *n* Ebenen der Quellverzeichnisstruktur.

`/log[+]:`*`Datei`*

Protokolliert alle Vorgänge in der angegebenen Datei. Bei Verwendung

des Pluszeichens werden neue Informationen an die Datei angehängt, anstatt sie zu überschreiben.

`/mir`

Spiegelt den Verzeichnisbaum. In *`Quelle`* nicht vorhandene Dateien werden in *`Ziel`* gelöscht!

`/move`

Verschiebt Dateien und Verzeichnisse, statt zu kopieren.

`/maxage:`*`n`*

Schließt Dateien vom Kopiervorgang aus, die älter als *`n`* Tage oder Datum *`n`* sind.

`/purge`

Löscht Dateien und Verzeichnisse im Zielpfad, die in der Quelle nicht mehr vorhanden sind.

`/r:`*`n`*

Anzahl der Wiederholungen (Versuche, eine Datei zu kopieren, die z.B. gerade gesperrt ist). Der Standardwert beträgt eine Million! Im Normalfall ist der Wert 0 für *`n`* am sinnvollsten.

`/sec`

Kopiert die Sicherheitsinformationen der Dateien. Sinnvoll insbesondere bei der Spiegelung mit `/mir`. Damit kann sich der Kopierende allerdings das Recht zum Überschreiben entziehen.

`/tee`

Sinnvoll in Kombination mit `/log`. Gibt alle Ausgaben zusätzlich über die Standardausgabe aus.

`/xf` *`Namen`***`| /xd`** *`Namen`*

Schließt die angegebene(n) Datei(en) (`/xf`) bzw. das/die angegebene(n) Verzeichnis(se) (`/xd`) vom Kopiervorgang aus.

Hinweis: `robocopy` wurde stetig weiterentwickelt. Daher unterscheiden sich Parameter und Leistungsfähigkeit teilweise erheblich von mit früheren Betriebssystemen gelieferten Versionen.

sfc

sfc [*Optionen*]

Überprüft geschützte Systemdateien und ersetzt gegebenenfalls falsche Versionen durch die jeweiligen Microsoft-Originalversionen.

Optionen

/scannow
: Überprüft und repariert gegebenenfalls alle Systemdateien.

/verifyonly
: Überprüft, ohne zu reparieren.

/scanfile=*Datei*
: Überprüft und repariert gegebenenfalls die angegebene Systemdatei.

/verifyfile=*Datei*
: Überprüft die angegebene Datei, ohne zu reparieren.

/offbootdir=*Pfad*
: Bei Offlinereparatur: Pfad zur zu reparierenden Bootpartition.

/offwindir=*Pfad*
: Pfad zum zu reparierenden Offline-Windows-Verzeichnis.

Beispiel

sfc /scannow /offbootdir=c:\ /offwindir=c:\windows
: Repariert die angegebene Windows-Installation im Offlinemodus. Das kann nützlich sein, wenn Windows aufgrund beschädigter Dateien nicht mehr startet: Dann bootet man vom Installationsdatenträger in den Reparaturmodus und führt in der Eingabeaufforderung, die sich gegebenenfalls mit der Tastenkombination *Umschalt+F10* öffnen lässt, obigen Befehl aus.

sigcheck WWW

sigcheck [*Optionen*] *Datei(en)*

Zeigt die Version und weitere Informationen der angegebenen *Datei(en)*.

Die Option –n reduziert die Ausgabe auf die Versionsnummer:

```
sigcheck -n "C:\Programme\Windows Defender\MsMpEng.exe"
```

PoSh: Der folgende Einzeiler zeigt, wie die Versionsnummer einer Datei alternativ mit der PowerShell ermittelt werden kann:

```
(get-command "C:\Programme\Windows
Defender\MsMpEng.exe").fileversioninfo | select
FileVersion
```

---

streams WWW

streams [-s] [-d] *Pfad*

Dieses Sysinternals-Tool zeigt *Alternate Data Streams* (ADS) der in *Pfad* angegebenen Dateien/Verzeichnisse an. Optional können alle Unterverzeichnisse nach ADS durchsucht werden (`-s`). Mit `-d` werden alle gefundenen ADS gelöscht.

---

tree

tree *Verzeichnis* **[*Optionen*]**

Zeigt eine Baumansicht der Unterverzeichnisse des angegebenen oder aktuellen Verzeichnisses.

Optionen

/a

Erzwingt die Verwendung von ASCII-Zeichen anstelle von erweiterten Grafikzeichen.

/f

Zeigt auch Dateinamen an.

PoSh: Der Befehl `tree` lässt sich mit einer in PowerShell erstellten Funktion unter Zuhilfenahme von `get-childitem` zwar emulieren, die Einbindung derart komplexer Skriptbeispiele würde aber den Umfang dieses Buchs sprengen. Einer der Fälle, in denen es sich empfiehlt, beim Original zu bleiben.

TYPE

`TYPE` *`Datei`*

Zeigt den Inhalt einer oder mehrerer Dateien an, wobei es sich wegen der Lesbarkeit um Textdateien handeln sollte. Komfortabler ist zu diesem Zweck die Verwendung des Befehls `more`.

PoSh: Mit dem Befehl `get-content` *`Datei`* oder dem Alias `type` wird Ihnen der Inhalt einer Datei in der PowerShell-Konsole angezeigt. Wie auch beim Befehl `more` kann die Befehlsausgabe in der einfachen Konsolenumgebung und in der *Terminal*-App (nicht aber in der ISE) für die seitenweise Anzeige an `out-host -p` weitergeleitet werden.

---

wbadmin

Verwaltet Datensicherungen und führt Backups durch. Als Speichermedien werden Festplatten, Freigaben im Netzwerk und optische Medien (z.B. DVDs) unterstützt, aber keine Bandlaufwerke.

`wbadmin get status`

Zeigt den Status des aktuellen Auftrags an.

`wbadmin get versions [-backuptarget:`*`Pfad`***` [-machine:`***`Computer`***`]]`**

Listet Sicherungen auf dem lokalen System oder am angegebenen (Netzwerk)Pfad auf. Optional kann auf Sicherungen eines bestimmten Computers gefiltert werden.

`wbadmin get items -version:`*`VersionsID`***` [-backuptarget:`***`Pfad`***` [-machine:`***`Computer`***`]]`**

Zeigt die Elemente einer durch ihre *VersionsID* spezifizierten Sicherung an (sonstige Optionen wie bei `get versions`). Verfügbare Versions-IDs können mit `get versions` aufgelistet werden. Sie werden im Format *MM/TT/JJJJ-hh:mm* angegeben.

wbadmin start backup-backuptarget:*Pfad* **-include:***Laufwerke* **[-allcritical] [-user:***Benutzername***] [-password:***Kennwort* **] [-noverify] [-noinheritacl] [-vssfull | -vsscopy] [-quiet]**

Führt eine Sicherung der angegebenen *Laufwerke* (kommaseparierte Liste) durch und speichert sie in (Netzwerk)*Pfad*. Bei Angabe von `allcritical` werden alle Laufwerke gesichert, die Betriebssystemkomponenten enthalten. Optional kann die Überprüfung von auf Wechseldatenträgern gespeicherten Sicherungen übergangen (`noverify`) werden, und der stille Modus ohne Benutzerinteraktion (`-quiet`) kann eingeschaltet werden. Durch (`-user`) und (`-password`) werden Anmeldedaten für den Zugriff auf die Freigabe eines anderen Computers angegeben, in der die Sicherung gespeichert werden soll. Durch (`noinheritacl`) wird kontrolliert, ob alle Benutzer mit Zugriff auf die Remotefreigabe oder nur der mit `-user` angegebene Benutzer sowie Administratoren und Sicherungsoperatoren der Remotemaschine auf die Sicherung zugreifen können.

`-vssfull` bewirkt eine vollständige Sicherung mithilfe des Volumenschattenkopiedienstes. Dabei wird der Verlauf jeder gesicherten Datei aktualisiert, um deren Sicherungsstatus anzugeben. Durch `-vsscopy` werden die Dateien gesichert, ihr Sicherungsstatus wird aber nicht angepasst. Das ist auch die Standardeinstellung, wenn der Parameter nicht angegeben wird.

wbadmin stop job [-quiet]

Bricht die aktuell durchgeführte Sicherung oder Wiederherstellung ab.

wbadmin enable backup
[-addtarget:*Sicherungsziel***]**
[-removetarget:*Sicherungsziel***]**
[-include:*einzuschließende Elemente***]**
[-nonrecurseinclude:*einzuschließende Elemente***]**
[-exclude:*auszuschließende Elemente***]**

[-nonrecurseexclude: *auszuschließende Elemente*]
[-hyperv:*einzuschließende Hyper-V-Komponenten***]**
[-schedule:*Zeitplan***] [-allcritical]**
[-systemstate] [-user:*Benutzername***]**
[-password:*Kennwort***]**
[-quiet] [-allowdeleteoldbackups]

Konfiguriert den Backup-Zeitplan, nach dem automatisch einmal oder mehrmals täglich bestimmte Laufwerke gesichert werden. Ohne Angabe von Optionen werden die aktuellen Einstellungen ausgegeben. Als Ziele können unter Windows 11/Server 2025 Datenträger, Volumen oder ein UNC-Pfad angegeben werden. Diese werden über ihre Disk-ID referenziert. `-addtarget` fügt ein Backup-Ziel hinzu, `removetarget` entfernt eines. Wird ein freigegebener Ordner hinzugefügt, können keine weiteren Backup-Ziele angegeben werden. Der ***Zeitplan*** wird als kommaseparierte Liste von Zeiten im Format *hh:mm* angegeben.

Warnung: Per `-addtarget` hinzugefügte Laufwerke werden vor der Verwendung automatisch formatiert!

Zu `-hyperv` werden die zu sichernden Komponenten von *Hyper-V* auf entsprechend konfigurierten Computern als kommaseparierte Liste angegeben.

Für die Erklärung der weiteren Parameter sei auf `wbadmin start backup` sowie auf die mit `wbadmin enable backup /?` aufzurufende ausführliche Hilfe verwiesen.

wbadmin disable backup [-quiet]

Löscht den konfigurierten Backup-Zeitplan.

wbadmin start recovery -version:*VersionsID* **-items:***Elemente* **-itemtype:***Elementtyp* **[-backuptarget:***Pfad***] [-machine:***QuellSystem* **] [recoverytarget:***Ziel***] [-*recursive***] [-overwrite:***Methode***] [-*notrestoreacl***] [-*skipbadclustercheck***] [-*norollforward***] [-*alternatelocation***] [-*recreatepath***] [-quiet]**

Stellt ***Elemente*** aus einer Sicherung wieder her. Sicherungen werden über Versions-IDs identifiziert, die mit `wbadmin get versions` ermittelt werden können. Bei den wiederherzustellenden Elementen

(`wbadmin get items` zeigt den Inhalt einer Sicherung an) kann es sich um Dateien/Ordner, Anwendungen (die Windows Backup unterstützen müssen) und ganze Volumen handeln. Als *`Elementtyp`* wird dazu *file*, *app* oder *volume* angegeben. Optional können unter anderem angegeben werden: der Pfad zur Sicherung (`-backuptarget`), der Computer, von dem die Sicherung stammt (`-machine`), sowie der Zielpfad bzw. das Ziellaufwerk (`-recoverytarget`).

`wbadmin start systemstatebackup`
`-backuptarget:`*`Laufwerkbuchstabe`* **`[quiet]`**

Sichert den Systemstatus (u.a. Systemdateien, Registrierung und Active Directory) auf das angegebene Laufwerk.

`wbadmin start systemstaterecovery -version:`*`VersionsID`* **`[-backuptarget:`***`Laufwerkbuchstabe`***`] [-machine:`***`QuellSystem`***`] [-recoverytarget:`***`Zielpfad`***`] [-quiet]`**

Stellt den Systemstatus aus einer Sicherung wieder her. Sicherungen werden über Versions-IDs identifiziert, die mit `wbadmin get versions` ermittelt werden können. Optional können unter anderem angegeben werden: das Laufwerk mit den Sicherungen (`-backuptarget`), der Computer, von dem die Sicherung stammt (`-machine`), sowie der Zielpfad (`-recoverytarget`).

`wbadmin delete systemstatebackup {-keepversions:`*`Anzahl`* **`| -version:`***`VersionsID`* **`| -deleteoldest} [-backuptarget:`***`Laufwerk`***`] [-machine:`***`QuellSystem`***`] [-quiet]`**

Löscht Sicherungen des Systemstatus. Es kann angegeben werden, wie viele Versionen aufbewahrt werden sollen (`-keepversions`, *`0`* löscht alle Sicherungen), welche Sicherung gelöscht werden soll (`-version`) oder dass die älteste Sicherung gelöscht wird. Einzelne Sicherungen werden über Versions-IDs identifiziert, die mit `wbadmin get versions` ermittelt werden können. Optional können unter anderem angegeben werden: das Laufwerk mit den Sicherungen (`-backuptarget`), der Computer, von dem die Sicherung stammt (`-machine`), sowie der Zielpfad (`-recoverytarget`).

Hinweis: Unter Windows Server ist dieser Befehl nur verfügbar, wenn das Feature *Windows Server-Sicherung* installiert ist.

PoSh: Die Installation des Features *Windows Server-Sicherung* fügt in Windows Server zahlreiche PowerShell-Cmdlets für die Steuerung der Datensicherung hinzu. Diese finden Sie in der ISE durch eine Suche nach `*wb*` oder in der Konsole mit dem Befehl `get-command *wb*`.

---

where

`where [/r` *`Verzeichnis`*`] [Optionen]`** *`Dateiname`*

Durchsucht das aktuelle Verzeichnis, den Suchpfad (`%path%`) oder ein *`Verzeichnis`* samt Unterverzeichnissen nach Dateien des angegebenen Namens. Zu jeder Fundstelle wird der volle Pfad ausgegeben. Bei der Angabe des zu suchenden Namens können die Platzhalter `*` und `?` verwendet werden. Standardmäßig hängt `where` nacheinander alle in der Umgebungsvariablen `PATHEXT` gespeicherten Endungen an Dateinamen ohne Endung, sodass bei der Suche nach EXE-Dateien die Endung weggelassen werden kann.

Optionen

`/q`
: Stiller Modus, gibt lediglich den Beendigungscode zurück.

`/f`
: Schließt den Namen der Trefferdatei in Anführungszeichen ein.

`/r`
: Durchsucht das angegebene Verzeichnis rekursiv.

`/t`
: Zeigt Größe und Änderungszeit aller gefundenen Dateien an.

PoSh: Mit dem Cmdlet `get-childitem` lässt sich das Programm `where` recht gut ersetzen. So sucht der folgende Befehl im aktuellen Verzeichnis und allen Unterverzeichnissen nach Dateien des Typs *.ps1* und zeigt diese mit vollständigem Pfad an:

```
gci *.ps1 -rec | select FullName
```

xcopy

xcopy *Quelle Ziel* **[***Optionen***]**

Kopiert Dateien und ganze Verzeichnisbäume. `xcopy` ist mächtiger als `copy`, muss sich jedoch `robocopy` geschlagen geben. *`Quelle`* und *`Ziel`* sind jeweils Pfade. Falls *`Quelle`* Wildcards enthält, kopiert `xcopy` alle darauf passenden Dateien bzw. Verzeichnisse. Wenn das *`Ziel`* ein (möglicherweise noch nicht vorhandenes) Verzeichnis ist, sollte ein Backslash an den Zielordner angehängt oder die Option `/i` verwendet werden.

Optionen

`/b`

Kopiert symbolische Links statt der Dateien, auf die sie verweisen.

`/c`

Setzt das Kopieren fort, wenn bei einzelnen Dateien Fehler auftreten. Ohne diese Option wird abgebrochen, wenn auch nur eine der zu kopierenden Dateien nicht gelesen oder geschrieben werden kann.

`/d`

Kopiert nur Dateien, die in der Quelle neuer als im Ziel sind.

`/d:`*`MM-TT-JJJJ`*

Kopiert nur Dateien, die nach dem angegebenen Datum geändert wurden.

`/e`

Kopiert auch alle Unterverzeichnisse samt Inhalt.

`/exclude:`*`Datei1`***`[`***`+Datei2`***`] [`***`+Datei3`***`]...`**

Der Inhalt jeder Zeile von *Datei* wird als Filter verwendet. Sollen also beispielsweise alle temporären Dateien vom Kopieren ausgeschlossen werden, reicht es, eine Datei mit Inhalt »temp« anzugeben, um alle Dateien (und Verzeichnisse) mit »temp« im Pfadnamen zu übergehen.

`/g`

Muss angegeben werden, wenn EFS-verschlüsselte Dateien an ein Ziel kopiert werden sollen, das keine Verschlüsselung unterstützt.

`/h /r`

Meist in Kombination sinnvoll: Es werden auch versteckte und Systemdateien kopiert (`/h`) und gegebenenfalls vorhandene schreibgeschützte Zieldateien überschrieben (`/r`).

`/j`

Kopiert ohne Pufferung (kann bei großen Dateien nützlich sein, wenn z. B. das Kopieren auf eine Netzwerkfreigabe während des Kopierens häufig abbricht).

`/k`

Kopiert alle Dateiattribute.

`/o`

Kopiert den Dateibesitzer und Zugriffsberechtigungen (DACL).

`/t`

Kopiert die Verzeichnisstruktur ohne Inhalte. Leere Unterverzeichnisse werden nur mit der zusätzlichen Option `/e` kopiert.

`/u`

Kopiert nur Dateien, die bereits im Zielverzeichnis vorhanden sind.

`/v`

Überprüft die Größe jeder neuen Datei.

`/x`

Kopiert auch die Überwachungseinstellungen (SACL).

`/y`

Automatische Bestätigung des Überschreibens von Zieldateien.

PoSh: Mit dem vielseitig einsetzbaren Cmdlet `copy-item` lassen sich Dateien und auch ganze Verzeichnisstrukturen ähnlich wie mit `xcopy` bei deutlich höherer Flexibilität in Auswahl und Weiterverarbeitung kopieren.

Dateisysteme, Volumen und Festplatten

---

convert

convert *x:* **/fs:ntfs [***/v***] [/CvtArea:***Dateiname***] [***/NoSecurity***] [***/x***]**

Konvertiert das durch den Laufwerkbuchstaben *x* angegebene FAT-Volumen in das NTFS-Dateisystem. Eine Konvertierung hin zu FAT oder zu ReFS ist nicht vorgesehen.

Optionen

/x

Hebt die Bereitstellung des Volumens auf. Alle offenen Handles auf dem Volumen werden ungültig. Damit kann die Konvertierung auch bei geöffneten Dateien erfolgen. Allerdings kann dieses Vorgehen Datenverlust zur Folge haben. Wird `/x` nicht angegeben, fragt der Befehl gegebenenfalls nach, ob das Volumen beim nächsten Systemstart konvertiert werden soll.

/v

Gibt ausführliche Meldungen aus.

/NoSecurity

Es werden keine Standardberechtigungen auf das Volumen gesetzt.

---

chkdsk

chkdsk *x:***[***Optionen***]**

Überprüft das Dateisystem auf Laufwerk *x:*.

Optionen

`/f [/x] [/r [/b]]`

Behebt gefundene Fehler. `/x` erzwingt das Aufheben der Bereitstellung des Datenträgers vor der Überprüfung. `/r` sucht fehlerhafte Sektoren und versucht, deren Daten wiederherzustellen. `/b` überprüft als fehlerhaft markierte Cluster erneut.

`/l:`*`n`*

Ändert die Größe der Protokolldatei auf *n* KByte (nur auf NTFS-Dateisystemen). Fehlt die Größenangabe, wird die aktuelle Größe angezeigt.

`/i`

Verringert den Zeitaufwand durch Auslassen der Prüfung von Indexeinträgen (nur NTFS).

`/c`

Verringert den Zeitaufwand durch Auslassen der Prüfung von Zyklen innerhalb der Ordnerstruktur (nur NTFS).

`/scan`

Führt einen proaktiven Scan von NTFS-Volumen aus.

`/forceofflinefix`

Im Zusammenwirken mit `/scan` werden erforderliche Reparaturen in die Warteschlange für die Offlinereparatur verschoben.

`/markclean`

Wurden keine Beschädigungen erkannt, werden Volumen mit einem der FAT-Dateisysteme auch ohne Angabe des Parameters `/f` als fehlerfrei markiert.

`/offlinescanandfix`

Führt eine Offlineüberprüfung und reparatur auf dem Volumen aus.

`/perf`

Es werden mehr Systemressourcen zur Beschleunigung des Durchlaufs bereitgestellt. Dadurch können andere laufende Anwendungen in ihrer

Leistung beeinträchtigt werden.

/spotfix

Repariert Beschädigungen auf NTFS-Volumen.

/sdcleanup

Führt auf NTFS-Volumen eine Bereinigung nicht benötigter Sicherheitsbeschreibungen aus.

/v

Aktiviert den ausführlichen Anzeigemodus.

PoSh: Anstelle des Befehls *chkdsk.exe* kann der PowerShell-Befehl `repair-volume` genutzt werden, um Volumen unter Windows zu prüfen und zu reparieren.

So prüft der folgende Befehl das Laufwerk *C:*, ohne eine Reparatur auszuführen:

```
Repair-Volume -DriveLetter C -Scan
```

---

chkntfs

Verwaltet den Autocheck-Vorgang beim Systemstart, der als fehlerhaft (dirty) markierte Volumen prüft. Fehlen die benötigten Administratorrechte, wird das nur durch eine kryptische Meldung bemängelt.

chkntfs *x*:

Zeigt den Dateisystemtyp und den Status des Dirty-Bits für das genannte Laufwerk an.

chkntfs /d

Stellt den Standard wieder her: Beim Systemstart werden als fehlerhaft markierte Volumen überprüft.

chkntfs /t:*Sekunden*

Zeigt bzw. setzt die Countdown-Zeit, die beim Systemstart vor Beginn einer Überprüfung gewartet wird.

chkntfs /x *y*: *z*:

Nimmt die angegebenen durch Leerzeichen getrennten Volumen von der

nächsten Überprüfung aus.

chkntfs /c *y*: *z*:

Plant die Prüfung der angegebenen durch Leerzeichen getrennten Volumen beim nächsten Systemstart durch Setzen des Dirty-Bits.

---

cleanmgr

cleanmgr [/d x /sageset:n | /sagerun:n | /tuneup:n | /lowdisk | /verylowdisk | /setup | /autoclean]

Dieses in der grafischen Benutzeroberfläche unter dem Namen *Datenträgerbereinigung* bekannte Werkzeug lässt sich durch Angabe von Parametern auch zur skriptgesteuerten Freigabe von Speicherplatz einsetzen. Das setzt allerdings vorherige Interaktionen mit der GUI voraus.

Optionen

/autoclean

Entfernt automatisch die Dateien der vorherigen Windows-Installation.

/d *x*

Öffnet den Dialog bezogen auf das Laufwerk mit dem Buchstaben *x* (ohne Doppelpunkt). Lässt sich nicht mit **/sagerun** kombinieren.

/sageset:*n*

Zeigt die grafische Schnittstelle und übernimmt die dort ausgewählten Einstellungen unter der Nummer *n* in die Registrierung. Die gespeicherten Einträge befinden sich unterhalb des Schlüssels *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\VolumeCaches*.

/sagerun:*n*

Führt die Bereinigung der unter *n* gespeicherten Elemente auf allen Volumen automatisch aus.

/tuneup:*n*

Kombiniert `/sageset` und `/sagerun`.

/lowdisk

Öffnet den Dialog mit Standardeinstellungen.

/verylowdisk

Führt den Vorgang auf allen Volumen aus.

---

defrag

defrag *x*: **[*Optionen*]**

Optimiert und defragmentiert das angegebene Laufwerk bzw. Volumen zur Verbesserung der Systemleistung.

defrag /c [*Optionen*]

Defragmentiert alle Volumen.

defrag /e *x*: *y*: **[*Optionen*]**

Defragmentiert alle Volumen außer den angegebenen (*x:* und *y:*).

Optionen

/a

Analysiert Volumen, ohne zu defragmentieren, und zeigt den Grad der Fragmentierung an.

/c

Führt den Vorgang auf allen Volumen aus.

/d

Führt eine herkömmliche Defragmentierung aus.

/e

Schließt angegebene Volumen aus.

/g

Optimiert Speicherebenen auf angegebenen Volumen.

/h

Führt den Vorgang mit normaler statt niedriger Priorität aus.

/k

Führt eine Bereichskonsolidierung auf angegebenen Volumen aus.

/l

Optimiert das Volumen erneut.

/m

Defragmentiert alle angegebenen Volumen parallel. Sinnvoll, wenn auf mehreren physischen Festplatten je ein Volumen liegt.

/o

Führt die für den Medientyp geeignete Optimierung aus.

/t

Zeigt Informationen zu einem gerade laufenden Defragmentierungsvorgang an. Ermöglicht es, per *Strg+C* den Vorgang zu beenden, selbst wenn er von einer anderen Konsole aus gestartet wurde.

/u

Gibt während der Laufzeit Statusinformationen aus.

/v

Ausführliche Ausgabe mit Fragmentierungsstatistiken.

/x

Konsolidiert den freien Speicherplatz auf angegebenen Volumen.

Warnung: Von einer Defragmentierung von SSDs wird ausdrücklich abgeraten, da sich hierdurch Performance und Lebensdauer dieser Datenträger reduzieren können.

---

disk2vhd WWW

disk2vhd <[*laufwerk:* [*laufwerk:*]...]|[*]> *vhddatei*

Dieses Sysinternals-Programm kann aus einer physischen Festplatte mit einem oder mehreren logischen Laufwerken eine virtuelle Festplatte in Form einer VHD-Datei erzeugen. Diese kann unter aktuellen Windows-Versionen gemountet werden. Abhängig von der Größe der Originalfestplatte kann auf diese Weise direkt eine physische in eine virtuelle Maschine konvertiert und in *Hyper-V* oder *Virtual PC* gestartet werden. Als Alternative für eine von den physischen Eigenschaften der Festplatte unabhängige Abbilderstellung

empfiehlt sich die Nutzung von `dism` bzw. `imagex`, sofern das zu klonende System vorher mit `sysprep` vorbereitet werden kann.

---

v2v_converterconsole WWW

`v2vconverterconsole command [`*`parameter1=value1 parameter2=value2`***`]...]`**

Das Befehlszeilenwerkzeug der kostenlosen Software *StarWind V2V Converter* kann virtuelle und physische Maschinen sowie deren Datenträger sehr flexibel in verschiedene Formate umwandeln. Für mehr Bedienkomfort sei hier für erste Schritte aber die grafische Benutzeroberfläche empfohlen.

---

diskpart

Leistungsfähiges Werkzeug zur Verwaltung von (auch virtuellen) Festplatten, Partitionen und Volumen. Kann interaktiv bedient oder durch ein Skript gesteuert werden. Kommandos beziehen sich immer auf das Objekt, das aktuell den Fokus hat. Zum Setzen des Fokus kann zunächst mit `list` die Nummer des jeweiligen Objekts ermittelt werden, anschließend wird es mit `select` ausgewählt. Nachdem ein Objekt den Fokus erhalten hat, können Befehle darauf angewendet werden:

`active`

Setzt die gewählte Partition als Bootpartition, von der aus das Betriebssystem gestartet wird.

`add disk=`*`n`*

Spiegelt das ausgewählte einfache Volumen auf dem angegebenen Datenträger mit der Nummer *n*.

`assign [letter=`*`L`*** `| mount=`***`Pfad`***`]`**

Weist dem aktiven Volumen einen Laufwerkbuchstaben oder einen Bereitstellungspunkt zu. Ohne weitere Optionen wird der nächste freie Buchstabe zugewiesen.

`attach vdisk [`*`readonly`*`]`

Mountet eine ausgewählte VHDX-Datei und bindet sie als virtuelle Festplatte ins System ein, ohne jedoch einen Laufwerkbuchstaben zuzuordnen (dazu muss sie zunächst ganz wie eine neue Festplatte partitioniert werden). Sind bereits Partitionen vorhanden, werden diesen automatisch Buchstaben zugewiesen, sobald der Datenträger online geschaltet ist.

attributes {*volume* | *disk*} {*set* | *clear*} {*hidden* | *readonly* | *nodefaultdriveletter* | *shadowcopy*}

Setzt oder löscht Attribute von Volumen und Festplatten. Letztere unterstützen nur den Schreibschutz (`readonly`), während Erstere versteckt (`hidden`) und als Laufwerk für Schattenkopien gekennzeichnet werden können (`shadowcopy`). Ferner kann verhindert werden, dass einem Volumen ein Laufwerkbuchstabe zugewiesen wird (`nodefaultdriveletter`).

automount {*enable* | *disable* | *scrub*}

Aktiviert bzw. deaktiviert das automatische Einbinden von Basisdatenträgern, die z.B. per USB neu erkannt werden. Die Option `scrub` entfernt Informationen (u.a. den zugeordneten Laufwerkbuchstaben) über früher eingebundene Medien.

break

Teilt eine Spiegelung von Datenträgern (RAID 1) auf.

clean [*all*]

Löschen der Konfigurationsinformationen des physischen Datenträgers. `clean all` überschreibt zusätzlich alle Cluster des Datenträgers mit null, was einem vollständigen Löschen ohne Wiederherstellungsmöglichkeit sehr nahekommt, je nach Größe und Geschwindigkeit des Datenträgers, aber geraume Zeit in Anspruch nehmen kann.

compact vdisk

Versucht, die physische Größe der VHD-Datei zu reduzieren.

convert {*basic* | *dynamic* | *gpt* | *mbr*}

Konvertiert zwischen den Typen »dynamisch« und »Basis« sowie zwischen »GPT« und »MBR«.

create partition {*primary* | *efi* | *extended* | *msr* | *logical* }

[*size=n*] [*offset=n*] [*id={<BYTE> | <GUID>}*] [*align=<N>*] [*noerr*]

Erzeugt eine primäre, eine EFI-System-, eine erweiterte oder eine MSR-Partition bzw. ein logisches Laufwerk innerhalb einer erweiterten Partition. `size` gibt die Größe in MByte an (Standard: maximale Größe), und `offset` legt fest, wie viel freier Platz in MByte vor der Partition gelassen werden soll (Standard: 0). Mit dem Wert `id` wird bestimmt, welcher Art die erstellte Partition sein soll, wobei eine Angabe im Byteformat für MBR-Datenträger gilt, während für GPT-Datenträger eine dem gewünschten Partitionstyp entsprechende GUID anzugeben ist. Details zu den möglichen Werten zeigt der Befehl **`help crea part prim`** (oder die voll ausgeschriebene Form des Befehls) innerhalb des `diskpart`-Prompts. Der Parameter `align` dient der Optimierung der Leistung auf Hardware-RAID-Arrays mit logischen Gerätenummern (LUN), `noerr` setzt die Verarbeitung im Fehlerfall fort, was bei skriptbasierten Vorgängen hilfreich sein kann.

Hinweis: Bei der Neuinstallation des Betriebssystems kann es hilfreich sein, ein bereits vorhandenes Laufwerk zu löschen und bei der Neuanlage eine EFI-Partition selbst zu definieren. Erfahrungen haben gezeigt, dass die von Windows selbst erzeugte EFI-Partition in einigen Fällen zu klein ist, um über Windows Update verteilte BIOS-Updates erfolgreich zu installieren. Hierzu sind folgende Schritte/ Befehle erforderlich:

- Öffnung einer Eingabeaufforderung mit der Tastenkombination *Umschalt+F10*
- `diskpart`
- `list disk`
- `select disk` *n* (*n* steht für das beim vorherigen Befehl aufgelistete Betriebssystemlaufwerk)
- `clean` (Achtung: Löscht alle Inhalte auf dem zuvor gewählten physischen Laufwerk ohne Rückfrage!)
- `convert gpt` (konvertiert das nun leere Laufwerk ins GPT-Format)

- `create partition EFI size=`*`512`* (erzeugt eine EFI-Partition in einer Größe von 512 MByte).

Wird jetzt das Windows-Setup fortgesetzt, nutzt es die angelegte EFI-Partition.

`create volume {`*`raid`*` | `*`simple`*` | `*`stripe`*` | `*`mirror`*`} [...]`

Erstellt ein Volumen des angegebenen Typs. Die verfügbaren Parameter variieren in Abhängigkeit vom Volumentyp.

`create vdisk file=`*`Datei`*** `maximum=`***`n`*** **`[`*`type=expandable`*`]`**

Erstellt eine VHD-Datei der angegebenen Maximalgröße in MByte, die optional nur auf die tatsächlich belegte Größe anwächst (`type=expandable`).

`delete {`*`disk`*` | `*`partition`*` | `*`volume`*`}`

Löscht das ausgewählte Objekt.

`detach vdisk`

Die selektierte VHD-Datei wird geschlossen.

`detail {`*`disk`*` | `*`partition`*` | `*`volume`*` | `*`vdisk`*` }`

Zeigt Detailinformationen an.

`exit`

Beendet `diskpart`.

`extend [`*`size`*`=`*`n`*`]`**

Erweitert das ausgewählte Volumen um *n* MByte oder um allen an das Volumen grenzenden freien Speicherplatz auf dem Datenträger.

`expand vdisk`

Erweitert die maximal verfügbare Größe eines VHD-Datenträgers.

`filesystems`

Zeigt Informationen zum Dateisystem des ausgewählten Volumens und zu unterstützten Dateisystemen zum Formatieren des Volumens.

`format [quick] [`*`fs`*`=`*`<FS>`***`] [`*`label`*`=`***`Name`***`] [`*`nowait`*`] [...]`**

Formatiert das selektierte Volumen (schnell) im gewählten Dateisystem (`fs`) und vergibt dabei optional eine Laufwerkbezeichnung. Bei Angabe von `nowait` wird der Befehl asynchron im Hintergrund ausgeführt. Weitere Parameter zur Steuerung des Verhaltens und der Auswirkungen

können der Hilfe entnommen werden.

gpt attributes=*n*

Weist auf GPT-Datenträgern der ausgewählten Partition Attribute zu.

help [*Befehl*]

Zeigt die Hilfe zu den aktuell verfügbaren Befehlen oder dem angegebenen Befehl an.

import

Importiert eine fremde Datenträgergruppe in die Onlinedatenträgergruppe des lokalen Computers.

inactive

Entfernt, sofern vorhanden, die Markierung der ausgewählten Partition als aktiv, wodurch von dieser das System nicht mehr starten kann.

list {*disk* | *partition* | *vdisk* | *volume*}

Listet Festplatten, Partitionen und Volumen auf. Zeigt unter anderem die Nummer einer jeden Festplatte an, nützlich für `select disk`.

merge vdisk [*depth*=*n***]**

Führt untergeordnete virtuelle Datenträger in der Tiefe *n* mit dem übergeordneten der differenzierenden Kette zusammen.

offline | online

Schaltet ein als online gekennzeichnetes Objekt offline und umgekehrt.

recover

Aktualisiert den Status aller Datenträger im ausgewählten Paket, versucht die Wiederherstellung von Datenträgern in einem ungültigen Paket und startet die erneute Synchronisation gespiegelter Volumen und RAID5-Datenträger mit veralteten Plex- oder Paritätsdaten.

remove

Entfernt einen Laufwerkbuchstaben oder die Zuweisung eines Bereitstellungspunkts.

repair

Repariert RAID5-Volumen mit einem fehlerhaften Mitglied.

rescan

Sucht nach neuen Datenträgern, die dem Computer möglicherweise hinzugefügt wurden.

retain

Bereitet das ausgewählte dynamische einfache Volumen für die Verwendung als Start- oder Systemvolumen vor.

san

Zeigt oder setzt die SAN-Richtlinie des aktiven Betriebssystems.

select {*disk* | *partition* | *volume*}=*n** [noerr]*

Selektiert ein Objekt anhand seiner Nummer, die mit `list` ermittelt werden kann. Für Volumen können Sie stattdessen auch den Laufwerkbuchstaben angeben, wenn vorhanden.

select vdisk file=Datei

Selektiert eine VHD-Datei.

shrink querymax

Gibt aus, wie viel Speicher vom ausgewählten Volumen maximal freigegeben werden kann.

shrink [*desired*=*m***] [*minimum*=***n***] [*nowait*] [*noerr*]**

Verkleinert das ausgewählte Volumen um den Maximalbetrag bzw. um die mit `desired` angegebenen *m* MByte. Falls `minimum` angegeben wird, schlägt der Befehl fehl, wenn das Volumen nicht um mindestens *n* MByte verkleinert werden kann. Durch `nowait` wird eine sofortige Rückgabe des Befehls erzwungen, während die Verkleinerung im Hintergrund noch ausgeführt wird.

uniqueid disk [*id*={*dword*** | ***GUID***}**

Ohne Parameter zeigt der Befehl die MBR-Signatur bzw. den Bezeichner der GUID-Partitionstabelle des ausgewählten Datenträgers an. Mit dem Parameter `id` können Sie eine neue Signatur oder eine neue GUID für den Bezeichner festlegen.

Option für alle Befehle

noerr

Bei einem Fehler wird nicht abgebrochen, sondern der nächste Skriptbefehl ausgeführt.

Zusätzliche Details können der Hilfe des Befehls entnommen werden.

Beispiele

Die folgende Befehlssequenz löscht einen USB-Datenträger und bereitet ihn als bootfähiges Medium vor, beispielsweise zur Erzeugung eines Windows-Installationsdatenträgers. Beachten Sie dabei, dass selbst erzeugte Bootdatenträger es teilweise erfordern, Secure Boot im BIOS des Computers zeitweilig zu deaktivieren, um von diesen starten zu können, und daher empfohlen wird, Windows-Installationsmedien mit dem Media Creation Tool von Microsoft zu erzeugen.

```
diskpart
list disk (ID des Datenträgers ermitteln)
select disk n (Auswahl der dem USB-Laufwerk zugeordneten Nummer)
clean (Löschen des Datenträgers)
create partition primary (Erzeugen einer neuen primären Partition)
active (Setzen der Partition als aktive Partition)
format fs=ntfs quick label=USBStick (Schnellformatierung und Benennung des Datenträgers)
assign (Laufwerkbuchstaben zuweisen, um die Partition im Explorer sichtbar zu machen)
exit (Beenden von diskpart)
```

Hinweis: Hier ist die `diskpart`-Version von Windows 11/Server 2025 beschrieben.

Tipp: Die internen Befehle von `diskpart` lassen sich oft bis auf drei Buchstaben verkürzt eingeben – so funktioniert anstelle von `select disk 0` auch `sel dis 0`.

PoSh: Mit PowerShell erhalten Sie eine umfassende Sammlung von Cmdlets zur Datenträgerverwaltung. Daher können an dieser Stelle nur kurze Beispiele für grundlegende Aufgaben dieses Bereichs präsentiert werden.

`get-disk` listet alle für PowerShell sichtbaren physischen Datenträger auf. Um einen neuen Datenträger, der als Nummer *n* im vorherigen Befehl angezeigt wird, zu initialisieren, geben Sie den Befehl `initialize-disk n`

`–partitionstyle mbr` ein. Lassen Sie die explizite Angabe von `partitionstyle` weg, wird standardmäßig ein GPT-Datenträger erzeugt.

Ein neues Volumen, das den gesamten Datenträger mit der Nummer *n* belegt, erzeugen Sie mit:

```
new-partition –disknumber n -usemaximumsize -
assigndriveletter
```

`clear-disk n -removedata` entfernt alle Daten und Konfigurationsinformationen vom Datenträger mit der Nummer *n*.

Warnung: Sowohl die PowerShell-Befehle zum Löschen als auch `diskpart` können bei leichtfertiger Anwendung einen vollständigen, gegebenenfalls endgültigen Datenverlust bewirken. Prüfen Sie deshalb besonders aufmerksam, auf welche Zieldatenträger Sie den jeweiligen Befehl anwenden, und erzeugen Sie möglichst regelmäßig Sicherungen Ihrer wichtigen Daten.

---

diskshadow WS2025

Leistungsfähiger Kommandozeileninterpreter zur Verwaltung von Schattenkopien. Die Bedienung ähnelt der von `diskpart`. Kann interaktiv bedient oder durch ein Skript gesteuert werden, das mit dem Parameter `-s` angegeben wird.

Im Folgenden die wichtigsten Befehle von `diskshadow`:

`help | *Befehl* -?`

Zeigt die Hilfe zu den aktuell verfügbaren Befehlen oder dem angegebenen Befehl an.

`exit`

Beendet `diskshadow`.

`list writers`

Zeigt Informationen zu allen Verfassern auf dem Server an.

`list shadows {*all* | *set* *SatzID* | *id* *SchattenkopieID*}`

Listet Schattenkopien auf, wahlweise alle (`all`), die eines Satzes (`set`)

oder eine bestimmte (`id`).

`list providers`

Zeigt Informationen zu den bei VSS registrierten Anbietern an.

`set`

Zeigt die aktuellen Optionen sowie die von `diskshadow` als Umgebungsvariablen gesetzten Aliase an.

`set context {`*`clientaccessible`*` | `*`persistent`*` | `*`volatile`*`}`

Setzt den Typ zu erzeugender Schattenkopien. Diese können Clients zugänglich gemacht werden (`clientaccessible`), die aktuelle Programmsitzung überdauern (`persistent`) oder danach gelöscht werden (`volatile`).

`set option transportable`

Die Schattenkopie wird nach der Erstellung nicht importiert. Vielmehr werden die Daten in eine CAB-Datei geschrieben, mit deren Hilfe die Schattenkopie später auf einem anderen System importiert werden kann.

`set verbose {`*`on`*` | `*`off`*`}`

Schaltet den ausführlichen Modus an bzw. aus.

`set metadata` *`Pfad`*

Legt den Pfad (inklusive Dateiname) zur zu erstellenden Metadatendatei fest.

`begin {`*`backup`*` | `*`restore`*`}`
`end {`*`backup`*` | `*`restore`*`}`

Startet bzw. beendet eine Sicherungs- bzw. Wiederherstellungsoperation.

`simulate restore`

Simuliert eine Wiederherstellungsoperation.

`add`

Zeigt die einer Schattenkopie hinzuzufügenden Volumen und die definierten Aliase.

`add volume `*`Laufwerk`*` [`*`alias`*` `*`Aliasname`*`]`

Fügt ein Laufwerk (Format: *C:*) zum Schattenkopiesatz hinzu und gibt ihm optional einen in einer Umgebungsvariablen gespeicherten Aliasnamen.

`break [`*`writeable`*`]` *`Set-ID`*

Entfernt die Bindung eines Schattenkopievolumens an VSS und macht es zu einem normalen Volumen. Durch Angabe von `writeable` wird das Laufwerk beschreibbar.

`create`

Erstellt die Schattenkopie. Vorher muss mit `add` mindestens ein Laufwerk hinzugefügt worden sein.

`expose` *`SchattenkopieID`* **`{`*`Laufwerk`*** **`|`** ***`Freigabe`*** **`|`** ***`Bereitstellungspunkt`*****`}`**

Macht eine Schattenkopie als Laufwerk, lokale Freigabe oder durch Mounten in ein Verzeichnis (den *`Bereitstellungspunkt`*) verfügbar. Die Schattenkopie muss entweder persistent sein, oder dieser Befehl muss mit `begin backup` und `end backup` verwendet werden. Als *`SchattenkopieID`* wird am einfachsten ein vorher mit `add` erstellter Alias verwendet, der in Prozentzeichen eingeschlossen wird.

`revert` *`Schattenkopie-ID`*

Setzt das Volumen auf den durch die *`Schattenkopie-ID`* beschriebenen Stand zurück.

`unexpose {`*`SchattenkopieID`* **`|`** ***`Laufwerk`*** **`|`** ***`Freigabe`*** **`|`** ***`MountPoint`*****`}`**

Hebt die Bereitstellung einer Schattenkopie wieder auf.

`exec` *`Skript`*

Führt ein Skript aus, das z.B. eine Datensicherung von einer bereitgestellten Schattenkopie durchführt.

`delete shadows {`*`all`* **`|`** ***`volume`*** *`Volume`* **`|`** ***`oldest`*** *`Volume`* **`|`** ***`set`*** *`SatzID`* **`|`** ***`id`*** *`SchattenkopieID`* **`|`** ***`exposed`*** *`Bereitstellungspunkt`***`}`**

Löscht Schattenkopien, und zwar alle (`all`), alle eines Volumens (`volume`), die älteste eines Volumens (`oldest`), alle eines Satzes (`set`) oder eine bestimmte, die durch ihre *`SchattenkopieID`* (`id`) oder den *`Bereitstellungspunkt`* bezeichnet wird.

`load metadata` *`Pfad`*

Lädt Metadaten aus einer Schattenkopie-CAB-Datei für den anschließenden Import der Daten aus der CAB-Datei.

`import`

Importiert eine Schattenkopie aus einer vorher erstellten CAB-Datei,

deren enthaltene Metadaten bereits mit `load metadata` geladen worden sein müssen.

`reset`

Setzt `diskshadow` auf den ursprünglichen Zustand zurück. Dabei gehen nicht persistente Schattenkopien verloren.

Weitere Kommandos können der Hilfe des Befehls entnommen werden.

---

dism

`dism [`*`DISM-Optionen`*`] {`*`WIM-Befehl`*`} [`*`WIM-Argumente`*`]`

`dism {/Image:`*`Pfad zum Offlineabbild`***` | `*`/Online`*`} [`*`DISM-Optionen`*`] {`*`Wartungsbefehl`*`} [`*`Wartungsargumente`*`]]`**

Dient dem Ansehen, Installieren, Deinstallieren, Konfigurieren und Aktualisieren von Funktionen und Paketen in Windows-Installationen und Abbilddateien (Typ *.wim*). Welche Befehle im Einzelnen verfügbar sind, hängt vom Betriebssystem in der Abbilddatei ab und davon, ob es sich um ein Online- oder Offlineabbild handelt.

Optionen

Die Optionen des Befehls `dism` setzen sich in der Regel aus dem auszuführenden Wartungsbefehl, dem Ziel (bei dem es sich um eine Abbilddatei vom Typ *.wim*, *.vhd* oder *.vhdx*, um eine Offlineinstallation von Windows oder das laufende Betriebssystem handeln kann) sowie den für die Steuerung von `dism` gegebenenfalls gewünschten oder erforderlichen Argumenten zusammen. Aufgrund des umfassenden Funktionsumfangs kann hier nur auf ausgewählte Optionen eingegangen werden. Weiterführende Informationen zum Befehlsumfang entnehmen Sie bitte der Hilfe zum Befehl oder den Webseiten von Microsoft zur Thematik »Abbildverwaltung für die Bereitstellung« (DISM, *Deployment Image Servicing and Management*). Indem Sie `/?` erst nach Eingabe einer Option an den Befehl anhängen, werden zusätzliche Informationen zur jeweiligen Option angezeigt.

Hinweis: Folgt ein Parameter auf einen Doppelpunkt, muss dieser dem Doppelpunkt unmittelbar – ohne Leerzeichen – folgen.

`/capture-image`

Zeichnet das Image eines Laufwerks in einer neuen WIM-Datei auf.

`/imagefile:`*`Pfad-zu-Dateiname.wim`*

Gibt den Pfad zur Abbilddatei an, die erzeugt werden soll.

`/capturedir:`*`Quellverzeichnis`*

Legt das Verzeichnis fest, dessen Inhalt der WIM-Datei hinzugefügt werden soll. Die Aufzeichnung enthält sämtliche nicht leeren Unterverzeichnisse und Dateien.

`/description:`*`Beschreibung`*

Fügt dem Abbild eine Beschreibung hinzu.

`/configfile:`*`wimscript.ini`*

Verweist auf eine Konfigurationsdatei, die beispielsweise Ausnahmen für die Aufzeichnung und Informationen zur Komprimierung beinhaltet.

`/compress:{`*`fast`* **`|`** ***`max`*** **`|`** ***`none`***`}`

Legt die Kompressionsrate fest. Je höher diese ist, desto länger dauert die Erzeugung der Imagedatei.

`/bootable`

Kennzeichnet ein Windows-PE-Image eines Volumens als startfähig.

`/checkintegrity`

Ermittelt Beschädigungen der WIM-Datei.

`/verify`

Sucht nach Fehlern und doppelten Dateien.

`/norpfix`

Deaktiviert die Reparatur des Analysepunktkennzeichens.

`/cleanup-mountpoints | cleanup-wim`

Löscht Ressourcen, die mit nicht korrekt abgemeldeten bereitgestellten Images bzw. WIM-Abbildern verbunden sind.

`/commit-image | commit-wim`

Speichert in einem bereitgestellten WIM- oder VHD-Abbild vorgenommene Änderungen.

/get-mountedimageinfo | get-mountedwiminfo

Zeigt Informationen zum Zustand aktuell bereitgestellter WIM- oder VHD-Images an.

/get-imageinfo /wimfile:*Pfad-zu-Dateiname.wim*

Zeigt Informationen zu in der angegebenen WIM- oder VHD-Abbilddatei enthaltenen Abbildern an.

/mount-image

Stellt ein Abbild aus einer WIM-Datei heraus bereit. Folgende Argumente werden für die Bereitstellung unterstützt:

/imagefile:*Pfad-zu-Dateiname.wim* | *.vhd* | *.vhdx*

Gibt den Pfad zur Abbilddatei an, die bereitgestellt werden soll.

/name:*Abbildname*

Der Name des Abbilds kann anstelle der Indexnummer verwendet werden.

/mountdir:*Verzeichnispfad*

Gibt den Pfad zu einem existierenden leeren Verzeichnis an, in dem das Abbild für die Bearbeitung bereitgestellt wird.

/mount-wim

Stellt ein Abbild aus einer WIM-Datei heraus bereit. Folgende Argumente werden für die Bereitstellung unterstützt:

/wimfile:*Pfad-zu-Dateiname.wim*

Gibt den Pfad zur Abbilddatei an, die gemountet werden soll.

/index:*Nummer*

Gibt die Position des zu mountenden Abbilds aus der Abbilddatei an, da eine Abbilddatei mehrere verschiedene Abbilder enthalten kann.

/remount-image | /remount-wim /mountdir:*Verzeichnis*

Stellt ein verwaistes Bereitstellungsverzeichnis wieder her.

/unmount-image | /unmount-wim ***/commit | /discard***

Hebt die Bereitstellung eines Abbilds auf. `commit` integriert die im Bereitstellungsverzeichnis vorgenommenen Änderungen in die Abbilddatei, `discard` verwirft sie und lässt die Abbilddatei unverändert.

/online

Verwendet das derzeit laufende Betriebssystem als Ziel.

`/image`

Gibt den Pfad zum Stammverzeichnis des Offlineabbilds einer Windows-Installation an, das sich auf einem anderen logischen Laufwerk befinden kann.

`/get-drivers`

Dieser Wartungsbefehl zeigt Informationen zu den im angesprochenen Abbild oder der laufenden Windows-Installation enthaltenen Treibern an. Standardmäßig werden nur Treiber von Drittanbietern angezeigt. Um im Betriebssystem enthaltene Treiber mit aufzulisten, geben Sie den Parameter `/all` an.

`/get-driverinfo /driver:`*`Pfad-zur-Treiber.inf`*

Zeigt detaillierte Informationen zu einem angegebenen Treiber an. Der angegebene Pfad kann sich auf den Namen einer installierten INF-Datei beschränken oder den Pfad zu einem nicht ins Abbild integrierten Treiber angeben.

`/add-driver /driver:`*`Pfad`* ***`/recurse /forceunsigned`***

Fügt den oder die Treiber dem Abbild hinzu, die sich im angegebenen Pfad befinden. Als Pfad kann ein Ordner oder auch der komplette Pfad zu einer INF-Datei angegeben werden. `recurse` sucht nach Treibern auch in den Unterverzeichnissen des angegebenen Ordners, `forceunsigned` erzwingt das Installieren unsignierter Treiber in 64-Bit-Versionen von Windows.

`/remove-driver /driver:`*`Pfad-zur-Treiber.inf`*

Entfernt den angegebenen Drittanbietertreiber aus dem Offlineabbild.

Warnung: Wenn der entfernte Treiber für den Startvorgang des Betriebssystems kritisch ist, kann das Betriebssystem im bearbeiteten Abbild möglicherweise nicht mehr starten. Das gilt auch für unsignierte Treiber, wobei mittlerweile an für Treiber verwendete Signaturen erhöhte Anforderungen gestellt werden. Alle treiberbezogenen Wartungsbefehle funktionieren nur mit entpackten Treibern, die über eine gültige INF-Datei verfügen.

`/get-provisionedappxpackages`

Zeigt Informationen zu App-Paketen, die für jeden neuen Benutzer installiert werden sollen.

/add-provisionedappxpackage /FolderPath:*AppX_Ordnerpfad* **[/SkipLicense] [/CustomDataPath:***Pfad zu benutzerdefinierten Daten***]**

/add-provisionedappxpackage /PackagePath:*Pfad_zum_Hauptpaket* **[/DependencyPackagePath:***Pfad_zum_Abhängigkeitspaket***] {[/LicensePath:***Pfad_zur_Lizenzdatei***] | [/SkipLicense]} [/CustomDataPath:***Pfad zu benutzerdefinierten Daten***]**

Fügt dem Image mindestens ein Paket hinzu. Dieses kann entweder samt Hauptpaket, Abhängigkeitspaketen und Lizenzdatei im unter `FolderPath` angegebenen Ordner hinterlegt oder mit `PackagePath` direkt über eine APPX-Datei ausgewählt werden. Mit `SkipLicense` sollten Sie nur Apps installieren, für die keine separaten Lizenzen erforderlich sind.

/remove-provisionedappxpackage /packagename:*Paketname*

Entfernt das angegebene Anwendungspaket.

/get-packages

Zeigt Basisinformationen zu allen im ausgewählten Abbild installierten Paketen an.

/get-packageinfo {/packagename:*Name-im-Abbild* **| /packagepath:***Pfad-zur-CAB-Datei***}**

Zeigt detaillierte Informationen zum per Name oder Pfad angegebenen Paket an.

/add-package /packagepath:*Pfad-zur-CAB-Datei* **[/ignorecheck]**

Fügt dem Abbild das mit dem Parameter /`packagepath` angegebene Paket hinzu, das entweder als CAB- oder als MSI-Datei vorliegen muss oder den Pfad zu einem Ordner mit einer einzelnen entpackten CAB-Datei, einer einzelnen MSU-Datei oder mehreren CAB- oder MSU-Dateien beinhaltet. Mit dem Parameter `/ignorecheck` wird die Prüfung der Verfügbarkeit der einzelnen Pakete übergangen.

/remove-package {/packagename:*Name-im-Abbild* **| /packagepath:***Pfad-zur-CAB-Datei***}**

Entfernt das per Name oder Pfad der Originalquelle angegebene Paket. Dieses muss als CAB-Datei vorliegen.

/get-features [{*/packagename:Name-im-Abbild* | */packagepath:Pfad-zur-CAB-Datei*}]

Listet grundlegende Informationen zu allen oder zu den durch angegebene Parameter ausgewählten Betriebssystemfeatures auf. Bei Angabe eines Featurenamens ist die Groß- und Kleinschreibung zu berücksichtigen, das gilt auch für alle folgenden featurebezogenen Optionen. Als packagepath kann entweder eine CAB-Datei oder ein Ordner angegeben werden.

/get-featureinfo /featurename:*Name-im-Abbild* [*/packagename:Name-im-Abbild*] [*/packagepath:Pfad-zur-CAB-Datei*]

Zeigt detaillierte Informationen zum mit dem Parameter /featurename angegebenen Feature im Abbild an. Die Parameter packagename und packagepath können verwendet werden, um ein bestimmtes Feature im angegebenen Paket zu finden.

/enable-feature /featurename:*Name-im-Abbild* [*/packagename:Name-im-Abbild*] [*/packagepath:Pfad-zum-Paket*]

Aktiviert oder aktualisiert das angegebene Feature im Abbild. Für im Betriebssystem enthaltene Features muss die Option /packagename nicht angegeben werden. Mit der Option /packagepath können Sie auf die Originalquelle des Pakets verweisen. Falls das Paket noch nicht installiert ist, wird es installiert und aktiviert.

/disable-feature /featurename:*Name-im-Abbild* [*/packagename:Name-im-Abbild*]

Deaktiviert das angegebene Feature im Abbild.

/cleanup-image [*/revertpendingactions*]

Führt Bereinigungs- oder Wiederherstellungsvorgänge im Abbild durch. Die Option /revertpendingactions entspricht einer Systemwiederherstellung für ein Abbild und sollte nur eingesetzt werden, wenn das betreffende Windows-Abbild nicht mehr gestartet werden kann. Sie ist nicht auf das laufende Betriebssystem oder auf Abbilder von Windows PE oder Windows RE (*Windows Recovery Environment*) anwendbar.

/cleanup-image {/analyzecomponentstore | /checkhealth |

/scanhealth | /restorehealth [/source:<*Pfad*>]}

Mittels /analyzecomponentstore lässt sich ein Bericht zum WinSxS-Komponentenspeicher erstellen. Durch /checkhealth wird das ausgewählte Image auf Beschädigungen und Reparierbarkeit geprüft, /scanhealth prüft auf Fehler des Komponentenspeichers im gewählten Image, und /restorehealth prüft auf derartige Fehler und versucht deren automatische Reparatur. Mit dem Parameter /source können Sie für die Reparatur auf einen Speicherort mit als funktionsfähig bekannten Dateiversionen verweisen.

/cleanup-image /spsuperseded [*/hidesp*]

Entfernt während der Installation eines Servicepacks erstellte Sicherungsdateien; /hidesp verbirgt den Eintrag zur Deinstallation des Servicepacks.

/cleanup-image /startcomponentcleanup [/ResetBase [/Defer]]

Bereinigt den Komponentenspeicher von abgelösten Komponenten zur Reduzierung der Größe. Mit /ResetBase setzen Sie die Basis abgelöster Komponenten zurück und reduzieren dadurch die Größe des Komponentenspeichers weiter. Durch /Defer in Kombination mit /ResetBase verlagern Sie länger anhaltende Bereinigungsvorgänge bis zur nächsten automatischen Wartung.

Warnung: Nach Anwendung von /ResetBase lassen sich zuvor installierte Windows-Updates nicht mehr entfernen.

/english

Gibt alle angezeigten Informationen in englischer Sprache aus.

/format:*table*|*list*

Bestimmt das Ausgabeformat eines Berichts. table bewirkt die Anzeige als Tabelle, list als Liste von Namen und zugehörigen Werten.

/norestart

Unterdrückt die Aufforderung zum Neustart nach Änderungen.

/quiet

Unterdrückt alle Ausgaben mit Ausnahme von Fehlermeldungen.

/scratchdir:*Pfad-zum-Extraktionsverzeichnis*

Gibt den Pfad zum Extrahieren zu wartender Dateien an. Ohne

Pfadangabe wird das temporäre Verzeichnis verwendet.

/windir:*Pfad-zum-Windows-Verzeichnis*

Gibt den Pfad des Windows-Ordners relativ zum Pfad des Abbilds an. Standardmäßig wird der Windows-Ordner im Basisverzeichnis des Abbilds angesprochen.

```
dism /online /cleanup-image /spsuperseded
```

Mit diesem im Administratorkontext ausgeführten Befehl löschen Sie die während der Installation eines Servicepacks erzeugte Kopie alter Dateiversionen im laufenden Betriebssystem. Sie sollten diesen Befehl nur ausführen, wenn Sie sicher sind, dass Sie das Servicepack nicht mehr entfernen wollen.

Die in den Installationsmedien inzwischen oft anzutreffenden *.esd*-Dateien werden durch `dism` nicht direkt unterstützt, sondern müssen bei Bedarf zunächst in eine *.wim*-Datei umgewandelt werden. Das lässt sich mit einem unter dem Laufwerkbuchstaben *D:* gemounteten Installationsdatenträger mit den folgenden Schritten realisieren:

```
DISM /Get-WimInfo /WimFile:D:\Sources\install.esd
```

listet die in der *.esd*-Datei enthaltenen Installationsimages auf, für Windows 11 Pro sieht der Eintrag beispielsweise wie folgt aus:

```
Index: "5"
Name: "Windows 11 Pro"
Beschreibung: "Windows 11 Pro"
Größe: 20.663.779.355 Bytes
```

Um dieses Image mit der Indexnummer 5 zu exportieren, verwenden Sie folgenden Befehl:

```
dism /export-image
/SourceImageFile:d:\sources\install.esd
/SourceIndex:5
/DestinationImageFile:c:\Setup\win11pro24h2.wim
/Compress:max /CheckIntegrity
```

Die auf diese Art erzeugte Datei können Sie mit den weiteren `dism`-Befehlen referenzieren.

PoSh: PowerShell beinhaltet Cmdlets, die anstelle von DISM-Befehlen zum Einsatz kommen können. So können Sie mit `mount-windowsimage` eine WIM-Datei in einem Ordner öffnen, mit `add-windowsdriver` einem Offlineimage Treiber hinzufügen und mit `save-windowsimage` gemachte Änderungen in WIM- und VHD-Dateien abspeichern.

Das folgende Beispiel stellt den Inhalt des Abbilds in der mit obigem Befehl erzeugten *win11pro24h2.wim* im Ordner *c:\mount* bereit:

mount-windowsimage -imagepath *c:\Setup\win11pro24h2.wim* **-path**
c:\mount **-index** *1*

---

format

format *x*: **[*Optionen*]**

Formatiert das durch den Laufwerkbuchstaben *x* gekennzeichnete Volumen.

Optionen

/fs:*Typ*

Gibt den Typ des Dateisystems an (NTFS, UDF, exFAT, FAT32, FAT, REFS).

/v:Datenträgerbezeichnung

Gibt die Bezeichnung des Datenträgers an.

/a:n

Setzt die Clustergröße auf `n` Bytes (ergänzen Sie `KB` nach `n`, wenn `n` in KByte angegeben ist), wobei `n` eine Potenz von 2 sein muss. Gültige Werte unterscheiden sich abhängig vom Dateisystem.

/c

Komprimiert standardmäßig alle Dateien auf dem NTFS-Dateisystem.

/i:{*enable* | *disable*} *Volumen*

Aktiviert bzw. deaktiviert die automatische Zuweisung des Integritätsprüfungsattributs für alle Dateien und Verzeichnisse des

Volumens bei REFS-formatierten Laufwerken.

/l:{*enable* | *disable*} *Volumen*

Formatiert NTFS-Volumen mit großen Datensätzen.

/p:Anzahl

Überschreibt jeden Sektor des Volumens zunächst mit 0, danach der angegebenen Anzahl der Wiederholungen entsprechend mit einer Zufallszahl.

/r:Revision

Erzwingt ein bestimmtes Revisionsformat für UDF.

/s:{enable | disable}

Aktiviert bzw. deaktiviert Unterstützung für kurze Dateinamen.

/txf:Status

Gibt an, ob TXF (transaktionsbasiertes NTFS) aktiviert (Standard) oder deaktiviert ist.

/x

Erzwingt vor der Formatierung das Aufheben der Bereitstellung des Volumens (Unmount vor der Formatierung).

/dax[:enable|disable]

Aktiviert den Direktzugriffsspeichermodus (DAX) für NTFS-Volumen auf geeigneter Hardware.

PoSh: PowerShell stellt das Cmdlet `format-volume` zum Formatieren existierender Volumen zur Verfügung. Für die Erzeugung von Volumen auf unpartitionierten Laufwerken können Sie die Cmdlets `initialize-disk` und `new-partition` verwenden. Das folgende Beispiel formatiert das Laufwerk mit dem Laufwerkbuchstaben *E:* im FAT32-Format. Ohne Angabe von `fullformat` erfolgt eine Schnellformatierung.

```
format-volume -driveletter E -filesystem FAT32 -
fullformat –force
```

---

fsutil

Verwaltet interne und komplexe Eigenschaften von Dateisystemen. Details dieses umfangreichen Befehls können der Windows-Hilfe entnommen werden. Im Folgenden werden wichtige Funktionen aufgelistet.

`8dot3name`

Verwaltet die kurzen Dateinamen (8.3-Namen).

`behavior`

Fragt folgende Eigenschaften ab bzw. setzt sie: Deaktivieren von 8.3-Namen, Zeichen aus erweiterten Zeichensätzen in 8.3-Namen, Festhalten der Zeit des letzten Zugriffs, Frequenz der Eventlog-Einträge bei Quota-Überschreitung, Speichernutzung des (Non)Paged Pool, Größe der MFT, Kompression ausschalten, Verschlüsselung ausschalten, Auslagerungsdatei verschlüsseln, Auflösung symbolischer Links, Bug Check (»Blue Screen«) bei Schäden.

`bypassIo`

BypassIO-Verwaltung (bei BypassIO handelt es sich um einen optimierten E/A-Pfad zum Lesen von Dateien).

`dax`

DAX-Volumenverwaltung.

`devdrv`

Volumenverwaltung für Entwickler.

`dirty {`*`query`*` | `*`set`*`}` *`Laufwerk`*

Fragt das Dirty-Bit ab bzw. setzt das Dirty-Bit, das festhält, ob ein Laufwerk in einem inkonsistenten Zustand ist und beim nächsten Systemstart durch *autochk.exe* überprüft wird.

`file createnew` *`Dateiname Länge`*

Erstellt eine leere Datei mit der angegebenen Länge in Byte.

`fsinfo`

Zeigt detaillierte Informationen zu Laufwerken und Volumen an.

`hardlink create` *`NeuerName ExistierendeDatei`*

Erzeugt einen festen Link mit dem Dateinamen *NeuerName*, der auf die Datei *ExistierendeDatei* auf dem gleichen Volumen verweist.

`objectid {`*`create`*` | `*`delete`*` | `*`query`*` | `*`set`*`}` *`ExistierendeDatei`*

Erstellt, löscht, liest oder ändert Objektbezeichner, bestehend aus 32-stelligen Hexadezimalwerten für Objektkennung, Anfangsvolumenkennung, Anfangsobjektkennung, Domänenkennung. Bei Verwendung des Parameters `set` sind die Kennungen mit anzugeben.

`quota` *`Option`*

Aktiviert weiche und harte (`enforce`) bzw. nur weiche (`track`) Kontingente, deaktiviert (`disable`), listet (`query`) oder bearbeitet (`modify`) Kontingente eines Volumens. Schließlich können Kontingentüberschreitungen aus der Ereignisanzeige ausgelesen und aufgelistet werden (`violations`).

`repair` *`Option`*

Listet die Einträge des Beschädigungsprotokolls eines Volumens auf (`enumerate`). Erfragt (`query`) oder setzt (`set`) den Status der Selbstreparatur des Volumens. Fragt den Beschädigungsstatus eines Volumens ab (`state`). Wartet auf Beendigung der aktuellen bzw. aller Reparaturen (`wait`). Initiiert die Reparatur einer Datei (`initiate`).

`reparsepoint {`*`query`*` | `*`delete`*`}` *`Pfad`*

Zeigt Informationen zu einem auf *`Pfad`* gesetzten Analysepunkt an oder löscht ihn.

`resource`

Verwaltet den Transaktionsressourcenmanager.

`sparse`

Erzeugung von und Informationen über Dateien mit geringer Datendichte (Sparse-Files).

`storageReserve`

Speicherreserveverwaltung (verwaltet den durch Windows 11/Server 2025 beispielsweise für temporäre und Cache-Dateien reservierten Speicherplatz).

`tiering`

Verwaltet die Speicherstaffelungseigenschaften.

`trace`

Verwaltet die Dateisystemablaufverfolgung.

`transaction`

Verwaltet Dateisystemtransaktionen (u.a. *Commit* und *Rollback*).

`usn`

Verwaltet USN-Journale (u.a. erstellen, löschen, Daten ausgeben).

`volume {`*`diskfree`*` | `*`dismount`*` | `*`querycluster`*`}` *`Laufwerk`*

Zeigt den freien Platz eines Volumens an, hebt dessen Bereitstellung auf oder fragt ab, von welcher Datei ein bestimmter Cluster verwendet wird.

`wim`

Transparente WIM-Hostverwaltung.

---

label

`label [/mp] [x:] [`*`Datenträgerbezeichnung`***`]`**

Vergibt für das Laufwerk *x* eine Datenträgerbezeichnung (ohne Laufwerkangabe wird die Bezeichnung des aktuellen Laufwerks geändert). Wenn keine Datenträgerbezeichnung angegeben wurde, wird sie abgefragt. Mit dem Parameter `/mp` wird angegeben, dass das Volumen als Bereitstellungspunkt oder Volumen behandelt werden soll.

PoSh: Mit dem Cmdlet `set-volume` kann eine neue Datenträgerbezeichnung folgendermaßen zugewiesen werden:

`set-volume -driveletter` *`C`* `-newfilesystemlabel` *`W11System`*

---

manage-bde

Dient der Verwaltung der Bitlocker-Laufwerkverschlüsselung auf Datenträgervolumen in Betriebssystemversionen, die Bitlocker unterstützen. Volumen können durch den Laufwerkbuchstaben mit Doppelpunkt, den Volumen-GUID-Pfad oder den Pfad eines eingebundenen Volumens angesprochen werden. Da dieser Befehl sehr umfangreich ist, entnehmen Sie die Parameter der einzelnen Optionen der Windows-Hilfe, die Sie über `manage-bde -Option -?` erreichen.

manage-bde -Option [*Argumente*]

Optionen

-status

Stellt Informationen zu Bitlocker-fähigen Laufwerken bereit.

-on *Volume* **[*Parameter*]**

Verschlüsselt das angegebene Volumen und aktiviert den Bitlocker-Schutz auf dem lokalen oder einem angegebenen Computer. Durch weitere Parameter können Sie beispielsweise ein Wiederherstellungskennwort festlegen, Verschlüsselungsdetails einstellen oder die Hardwareüberprüfung überspringen.

-off *Volume* **[-{*Computername*|*cn*}** *Computername*]

Entschlüsselt das angegebene Volumen auf dem lokalen oder dem angegebenen Computer und deaktiviert den Bitlocker-Schutz.

-{pause | resume} *Volume* **[*Parameter*]**

Hält die Verschlüsselung, die Entschlüsselung oder das sichere Löschen des freien Speicherplatzes an (`-pause`) bzw. nimmt den angehaltenen Vorgang wieder auf (`-resume`).

-{lock | unlock} *Volume* **[*Parameter*]**

Blockiert den Zugriff auf durch Bitlocker geschützte Daten (`-lock`) oder gibt diesen wieder frei (`-unlock`).

-autounlock {-*enable* | -*disable* | -*clearallkeys*} *Volume*

Verwaltet die automatische Entsperrung von Datenvolumen. Durch `enable` wird diese aktiviert, `disable` deaktiviert die Entsperrung, und `clearallkeys` entfernt alle gespeicherten externen Schlüssel vom Betriebssystemvolumen.

-protectors {-*adbackup* | -*aadbackup* | -*add* | -*delete* | -*disable* | -*enable* | -*get*} *Volume*

Verwaltet die Schutzmethoden für den Verschlüsselungsschlüssel. Dabei zeigt `get` verwendete Schlüsselschutzmethoden an, `add` fügt welche hinzu, und `delete` löscht sie. Mit `enable` wird der Schutz durch Entfernen nicht genutzter Schlüssel vom Datenträger aktiviert, `disable` hält den Schutz für alle Benutzer durch Bereitstellung des

Verschlüsselungsschlüssels auf einem Datenträger an, `adbackup` sichert Wiederherstellungsinformationen für das Laufwerk im Active Directory und `aadbackup` ebenso, allerdings im Azure Active Directory.

-{setidentifier | si} *Volume*

Setzt das Feld mit der Volumen-ID auf den in der Gruppenrichtlinie angegebenen Wert.

-{forcerecovery | fr}

Erzwingt den Wechsel eines Bitlocker-geschützten Betriebssystems in den Wiederherstellungsmodus beim Neustart. TPM-bezogene Schlüsselschutzvorrichtungen auf dem Betriebssystemvolumen werden dabei gelöscht. Zum Entsperren wird dann ein Wiederherstellungskennwort oder ein Wiederherstellungsschlüssel benötigt.

-{changepassword | changepin} *Volume*

Ändert das Kennwort bzw. die PIN für den Zugriff auf ein verschlüsseltes Volumen.

-changekey *Volume*

Ändert den Systemstartschlüssel für ein auf Basis von TPM geschütztes System.

-{keypackage | kp} *Volume*
-id *ID der Schlüsselschutzvorrichtung*
-path *Pfad zum Schlüsselpaketverzeichnis*

Generiert ein neues Schlüsselpaket für das angegebene Volumen. Dabei gibt `id` den Bezeichner der Schlüsselschutzvorrichtung an und *`Pfad`* den Ordner, in dem das neue Schlüsselpaket gespeichert wird.

-upgrade *Volume*

Aktualisiert die Bitlocker-Version.

-{wipefreespace | w}

Führt das sichere Löschen des freien Speicherplatzes durch Entfernen aller in diesem vorhandenen Datenfragmente aus.

PoSh: Das PowerShell-Bitlocker-Modul enthält alle erforderlichen Cmdlets, um Bitlocker unter Windows 11 und Server 2025 zu verwalten. Die Liste der verfügbaren Cmdlets wird mit `get-command *Bitlocker*` oder in der ISE

durch den Suchbegriff *Bitlocker* angezeigt.

Mit dem Befehl

```
suspend-bitlocker -mountpoint C:
-rebootcount 0
```

unterbrechen Sie die Bitlocker-Verschlüsselung bis Widerruf, um beispielsweise ohne Gefahr ein BIOS-Update einzuspielen.

---

mbr2gpt

`mbr2gpt {/validate | /convert}`
`[/disk:`*`Disk-ID`***`] [/logs:`***`Zielverzeichnis`***`]`**
`[/map:`*`Quelle=Ziel`***`] [/allow-FullOS]`**

Konvertiert einen physischen oder virtuellen Datenträger mit installiertem Betriebssystem ohne Datenverlust vom MBR-Format in das GPT-Format. Das kann auch heute weiterhin aktuell sein, wenn ein historisch gewachsenes System, das noch mit klassischem BIOS gestartet wird, auf die neuen Betriebssystemversionen aktualisiert werden soll, denn diese unterstützen nur noch UEFI-basiertes Starten und erfordern hierfür eine GPT-basierte Festplatte. Der Befehl `mbr2gpt` kann oft im laufenden Betrieb ausgeführt werden. In einigen Konfigurationen wird es nötig sein, die Eingabeaufforderung in den erweiterten Startoptionen von Windows zu nutzen oder ein virtuelles Laufwerk temporär in das Hostbetriebssystem einzubinden. Die Disk-ID ermitteln Sie mittels `diskpart`, indem Sie von dort aus den Befehl `list disk` ausführen. In der Regel wird das Systemlaufwerk die Datenträger-ID *0* aufweisen, in Systemen mit mehreren physischen Datenträgern kann diese aber abweichen, ebenso bei einem eingebundenen virtuellen Datenträger. Bei Letzterem gilt noch die Besonderheit, dass der virtuelle Datenträger beim Einsatz von *Hyper-V* in das VHDX-Format konvertiert werden muss, was sich bei ausreichend freiem Speicherplatz beispielsweise mit dem *Star-Wind V2V Converter* vor einer Umstellung auf GPT erledigen lässt – mit der dadurch erzeugten Kopie können Sie gefahrlos experimentieren.

Optionen:

`/convert [/allowfullos]`

Das Systemvolumen wird nach erfolgreicher Prüfung bei Bedarf verkleinert, und eine EFI-Systempartition wird erzeugt. Anschließend werden die UEFI-Bootdateien und GPT-Komponenten in der neuen Partition installiert. Die Bootkonfiguration im BCD wird aktualisiert. Nach einer Erfolgsmeldung trennen Sie das von Ihnen im Host eingebundene virtuelle Laufwerk, sofern Sie diesen Weg gegangen sind.

`/logs:`*`Zielverzeichnis`*

Gibt das Verzeichnis an, in dem die Protokolle von `mbr2gpt` abgelegt werden. Das Verzeichnis muss bereits vorhanden sein.

`/map:`*`Quelle=Ziel`*

Gibt andere Partitionstypzuordnungen zwischen MBR und GPT an. Dabei wird die MBR-Partitionsnummer links vom Gleichheitszeichen in dezimaler statt in hexadezimaler Notation angegeben, und die GPT-GUID kann geschweifte Klammern enthalten. Sollten mehrere Zuordnungen erforderlich sein, können weitere `/map`-Optionen angegeben werden.

`/validate [/allowfullos]`

Prüft, ob der Datenträger die Anforderungen zur Konvertierung erfüllt. Bei Fehlermeldungen hilft oft eine Internetrecherche. Mittels `/allowfullos` teilen Sie dem Befehl mit, dass auf dem angesprochenen Datenträger das aktive Betriebssystem läuft.

Beim Neustart eines physischen Rechners nach abgeschlossener Konvertierung müssen Sie die Startart im BIOS auf *UEFI-Boot* umstellen, andernfalls startet Windows nicht mehr. Eine virtuelle Maschine, die auf *Hyper-V* basiert, wird eine VM der Generation 1 sein. Da diese nicht umgestellt werden kann, muss eine neue VM der Generation 2 erzeugt werden, an die die virtuelle Festplatte (im Format VHDX) als SCSI-Laufwerk angehängt wird.

Hinweis: Erzeugen Sie ein Backup außerhalb der zu konvertierenden Festplatte, auch wenn von einer datenverlustfreien Konvertierung ausgegangen wird. Falls Ihr System aus irgendeinem Grund nicht mehr

startet, fällt es dann weniger schwer, einen Schnitt zu machen und neu zu beginnen oder den ursprünglichen Zustand wiederherzustellen.

Hinweis: Ist Ihr Betriebssystemlaufwerk mit Bitlocker verschlüsselt, lassen Sie es pausieren, oder heben Sie die Verschlüsselung zeitweilig ganz auf.

Warnung: Die Neuanlage einer virtuellen Maschine der Generation 2 in *Hyper-V* entspricht einem Hardwarewechsel und triggert die Produktaktivierung des Betriebssystems. Falls Sie die ursprüngliche Installation mit einem Windows-7- oder Windows-8.x-Schlüssel aktiviert haben, funktioniert dieser für die Neuaktivierung nicht mehr.

MKLINK

`MKLINK [/d | /h | /j]` *`Verknüpfung Ziel`*

Erstellt eine symbolische Verknüpfung (*Symbolic Link*) auf eine Datei bzw. ein Verzeichnis (`/d`). Anstelle einer symbolischen Verknüpfung kann eine feste Verknüpfung (`/h`, *Hard Link*) oder eine Verzeichnisverbindung (`/j`, *Junction Point*) erzeugt werden.

PoSh: PowerShell ermöglicht ebenfalls das Erstellen symbolischer oder fester Verknüpfungen. Hierzu findet der Befehl `new-item` mit dem Parameter `-type` und einem der Werte `SymbolicLink`, `HardLink` oder `Junction` Anwendung.

---

mountvol

`mountvol [`*`Verzeichnis`***`] [`***`Volumen`*** **`| /d | /l | /p | /d]`**

Ermöglicht das Erzeugen, Anzeigen und Löschen von Bereitstellungspunkten für Volumen in einem leeren Verzeichnis einer NTFS-Partition. *`Volumen`* muss als GUID in der Form `\\?\Volume{GUID}\` angegeben werden. Die GUIDs der existenten Volumen liefert ein Aufruf von `mountvol` ohne Parameter. `/d` hebt die Bereitstellung eines Volumens auf. `/l` zeigt den

Namen des gegebenenfalls in *Verzeichnis* bereitgestellten Volumens an. /p entfernt den Volumenbereitstellungspunkt vom angegebenen Verzeichnis, entfernt die Volumenbereitstellung und setzt das Volumen auf nicht mehr bereitstellbar.

mountvol /r

Bereinigung: Löscht Bereitstellungsverzeichnisse und Registrierungseinstellungen für im System nicht mehr vorhandene Volumen.

mountvol {*/n* | */e*}

Deaktiviert (/n) bzw. aktiviert (/e) das automatische Bereitstellen neuer Volumen.

mountvol [*Laufwerk*] /s

Sorgt für die Bereitstellung der EFI-Systempartition auf dem angegebenen Laufwerk.

PoSh: PowerShell unterstützt ebenfalls den direkten Zugriff auf Bereitstellungspunkte und Laufwerkbuchstaben.

So zeigt der Befehl

```
(get-partition -DiskNumber 0).AccessPaths
```

die Bereitstellungspunkte des Datenträgers 0 an.

Um die Partition *n* im leeren Ordner *E:\mount* bereitzustellen, verwenden Sie:

```
(get-partition -PartitionNumber n) | Add-
PartitionAccessPath -AccessPath "E:\mount"
```

Um den Bereitstellungspunkt aufzuheben, nutzen Sie anstelle von Add-PartitionAccessPath den Befehl Remove-PartitionAccessPath.

---

repair-bde

repair-bde *Quellvolumen* **[*Optionen*]**

Versucht die Reparatur oder Entschlüsselung von mit Bitlocker-

Laufwerkverschlüsselung geschützten Daten im Fall einer schwerwiegenden Beschädigung der Festplatte. Die Verwendung des Befehls setzt die Verfügbarkeit eines gültigen Wiederherstellungskennworts oder schlüssels voraus. Eine weitere Voraussetzung für die Nutzung des Befehls ist die vollständig abgeschlossene Verschlüsselung des komplett verschlüsselten Datenträgers.

Optionen

`Quellvolumen`

Das zu reparierende mit Bitlocker verschlüsselte Volumen.

`Ausgabevolumen oder Image`

Volumen oder Dateispeicherort einer Abbilddatei zum Speichern der entschlüsselten Daten.

`-rk | -recoverykey`

Externer Schlüssel zum Entsperren des Volumens.

`-rp | -recoverypassword`

Numerisches Kennwort zum Entsperren des Volumens.

`-pw | -password`

Kennwort zum Entsperren des Volumens.

`-kp | -keypackage`

Pfad zum Schlüsselpaket für das Entsperren des Volumens.

`-f | -force`

Erzwingt die Aufhebung der Bereitstellung des Quellvolumens.

`-lf | -logfile`

Gibt den Pfad für die Aufzeichnung von Statusinformationen an.

---

subst

`subst` *`Laufwerkbuchstabe`*`:` **`{`***`Pfad`* `|` **`/d}`**

Weist einem Pfad im Dateisystem einen Laufwerkbuchstaben zu bzw. hebt die Zuordnung wieder auf. Der Befehl kann unter anderem dazu verwendet

werden, sehr lange Pfade (länger als 260 Zeichen) zu erzeugen oder wieder zu löschen, da die Pfadlängenbegrenzung auf 260 Zeichen immer nur für den Pfad selbst gilt, unabhängig von der Datenstruktur auf der Festplatte.

Hinweis: Die Begrenzung der Pfadlänge auf 260 Zeichen gilt nicht grundsätzlich, sondern nur für Programme, die die erweiterten Möglichkeiten von NTFS mit Pfaden bis 32.767 Zeichen Länge nicht unterstützen. Leider gehört der Windows-Explorer auch in Windows 11/Server 2025 noch dazu.

PoSh: PowerShell kann zwar eigenständig Laufwerke auf lokale Pfade, Netzwerkpfade und sogar Registrierungsschlüssel legen, die Wirksamkeit ist mit Ausnahme von Netzlaufwerken aber auf die PowerShell-Umgebung beschränkt.

Mit `new-psdrive -name x -root C:\Datenbank -psprovider filesystem` erzeugen Sie ein PowerShell-Laufwerk *x:*, das auf das Verzeichnis *C:\ Datenbank* verweist. In dieses können Sie mit `cd x:` wechseln. Die Bereitstellung des Laufwerks heben Sie durch Schließen der PowerShell-Sitzung oder mit dem Befehl `remove-psdrive x` auf. Sollten Sie das Laufwerk regelmäßig unter der PowerShell benutzen, integrieren Sie den Befehl in Ihr PowerShell-Profil.

VOL

`VOL [`*`x`*`:]`

Zeigt die Datenträgerbezeichnung und die Seriennummer des aktuellen oder des angegebenen Datenträgers an.

PoSh: Mit dem Befehl

```
get-ciminstance -classname Win32_LogicalDisk | select deviceid,volumename,volumeserialnumber
```

zeigen Sie Laufwerkbuchstaben, Datenträgerbezeichnung und Seriennummer der logischen Laufwerke an.

vssadmin

Der Befehl `vssadmin` verwaltet den Volumenschattenkopiedienst.

Hinweis: In Windows 11 ist nur ein geringer Teil der hier beschriebenen Optionen enthalten, den vollen Funktionsumfang bietet Server 2025.

`vssadmin add shadowstorage /for=`*`x`*`: /on=`*`y`*`: [/maxsize=`*`Größe`*`]`

Fügt eine Schattenkopie-Speicherassoziation zwischen den Laufwerken *x:* und *y:* hinzu und begrenzt optional den maximal zu verwendenden Speicher. Die maximale Größe muss mindestens 320 MByte betragen und kann unter anderem mit folgenden SuffiXEn angegeben werden: `KB`, `MB`, `GB`, `TB`, `PB` und `EB`. Bei Angabe von `UNBOUNDED` ist die Größe unbegrenzt.

`vssadmin resize shadowstorage /for=`*`x`*`: /on=`*`y`*`: [/maxsize=`*`Größe`*`]`

Ändert den maximal von Schattenkopien zu belegenden Speicherplatz. Bei einer Verkleinerung können Schattenkopien verloren gehen.

`vssadmin delete shadowstorage /for=`*`x`*`: [/on=`*`y`*`]`

Löscht Schattenkopie-Speicherassoziationen für das angegebene Volumen auf allen oder auf dem mit `/on` spezifizierten Volumen.

`vssadmin list shadowstorage {/for=`*`x`*`: | /on=`*`y`*`:}`

Zeigt Schattenkopie-Speicherassoziationen für das angegebene Quell- (`/for`) oder Zielvolumen (`/on`) an.

`vssadmin create shadow /for=`*`x`*`: [/autoretry=`*`TimeoutMinuten`*`]`

Erstellt eine Schattenkopie des angegebenen Volumens. Falls angegeben und falls gerade ein anderer Prozess eine Schattenkopie erstellt, wird *`TimeoutMinuten`* lang versucht, die Kopie zu erstellen, bevor abgebrochen wird.

`vssadmin delete shadows /for=`*`x`*`: [/oldest]`

Löscht alle Schattenkopien des angegebenen Volumens.

`vssadmin delete shadows /shadow=`*`SchattenkopieID`*

Löscht die Schattenkopien mit der angegebenen ID.

`vssadmin delete shadows /all`

Löscht alle Schattenkopien aller Volumen.

vssadmin list shadows [/for=*x*:] [/shadow=*SchattenkopieID*]

Zeigt Informationen zu allen oder der angegebenen Schattenkopie an.

vssadmin list volumes

Zeigt jene Volumen an, von denen Schattenkopien erstellt werden können.

vssadmin revert shadow /shadow=*SchattenkopieID* [/ForceDismount] [/Quiet]

Stellt den Zustand auf einem Volumen wieder her, der zum Zeitpunkt der Erstellung einer Schattenkopie herrschte. Alle späteren Änderungen und auch Schattenkopien gehen dabei verloren! Optional können offene Handles gewaltsam geschlossen werden (`/ForceDismount`); der Vorgang würde ansonsten fehlschlagen.

Drucker und Warteschlangen

---

lpq

`lpq -S` *`Server`* **`-P`** *`Drucker`* **`[-I]`**

Zeigt Status und Inhalt der angegebenen LPD-Druckerwarteschlange an. Einen ausführlichen Statusbericht erhalten Sie mit der Option `-l`.

Hinweis: `lpq` ist nicht im Standardinstallationsumfang enthalten, stattdessen muss das Feature *LPR-Portmonitor* über *Windows-Features hinzufügen* installiert werden.

---

lpr

`lpr -S` *`Server`* **`-P`** *`Drucker`* **`[`***`Optionen`***`]`** *`Datei`*

Druckt *`Datei`* auf dem angegebenen Netzwerkdrucker auf einem *`Server`*, der den LPD-Dienst anbietet.

Optionen

`-C` *`Klasse`*
: Wählt basierend auf der Auftragsklasse eine Trennseite aus (es werden nicht alle Drucker unterstützt).

`-J` *`Auftragsname`*
: Definiert einen Auftragsnamen für das Drucken auf der Trennseite.

`-o l`
: Definiert den Auftragstyp als binär; der Standardwert ist *Text*. Dieser Befehl ist manchmal hilfreich, um PostScript-Dateien zu drucken.

`-d`

Sendet zuerst die Datendatei.

Hinweis: Der Befehl `lpr` muss ebenso wie `lpq` über *Windows-Features hinzufügen* eingebunden werden.

---

print

`print /d: [`*`Drucker`*`] `*`Dateien`**

Druckt Textdateien auf dem angegebenen lokalen oder einem entfernten Drucker.

PoSh: Um Textdateien aus Windows PowerShell heraus zu drucken, können Sie den Befehl `out-printer` verwenden. Das folgende Beispiel gibt den Inhalt der Datei *info.txt* im aktuellen Verzeichnis an den PDF-Drucker weiter. Leider lässt sich kein Zieldateiname an dessen Treiber übergeben.

```
get-content .\info.txt | out-printer "Microsoft Print to PDF"
```

Die Namen der installierten Drucker erhalten Sie mit `get-printer`.

---

printbrm

`printbrm -b|r|q [-s `*`Server`*`] -f `*`Datei`*` [-d `*`Verzeichnis`*`] [-o `*`force`*`] [-p `*`all`*`|`*`orig`*`] [-nobin] [-lpr2tcp] [-c `*`Konfigurationsdatei`*`] [-noacl]`

Ermöglicht die Druckermigration von einem Druckserver auf einen anderen. Dieses Programm befindet sich im Verzeichnis *%windir%\system32\spool\tools*.

`-b -f `*`Dateiname`**

Sicherung der Druckserverkonfiguration in der angegebenen Datei.

`-r -f `*`Dateiname`**

Importiert die Druckserverkonfiguration aus der angegebenen Datei.

`-q [-f `*`Dateiname`**`]`**

Fragt die Informationen des Druckservers ab oder die aus der angegebenen Sicherungsdatei.

Optionen

`-o`

Erzwingt das Überschreiben vorhandener Objekte.

`-p all | orig`

Veröffentlicht alle Drucker im Verzeichnis oder nur die ursprünglich veröffentlichten Drucker.

`-nobin`

Unterdrückt die Sicherung der Binärdateien (Treiber).

`-lpr2tcp`

Konvertiert bei Wiederherstellung/Import LPR-Ports zu Standard-TCP/IP-Ports.

`-c` *`Dateiname`*

Verwendet die angegebene Konfigurationsdatei.

`-noacl`

Entfernt bei einer Wiederherstellung die ACLs der Druckerwarteschlange.

Tipp: Um auch angepasste Druckerformulare von einem Server zu einem anderen zu übertragen, exportieren Sie auf dem Quellserver den Registrierungsschlüssel *HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Print\Forms* und importieren ihn anschließend auf dem Zielsystem.

---

prncnfg.vbs

Konfiguriert Drucker auf dem angegebenen oder einem lokalen Computer.

`cscript prncnfg [-gtx?] [-s` *`Server`*`] [-p`** *`Druckername`*`]`

`-x`

Änderung des Druckernamens.

-?

Zeigt die Hilfe zum Skript sowie Anwendungsbeispiele an.

prncnfg.vbs -g [-s *Computer*`]` **[-u** *Benutzer*`]` **[-w** *Passwort*`]` **-p** *DruckerName*

Zeigt die Konfiguration des angegebenen Druckers an.

prncnfg.vbs -x [-s *Computer*`]` **[-u** *Benutzer*`]` **[-w** *Passwort*`]` **-p** *DruckerName***-z** *NeuerDruckerName*

Benennt einen Drucker um.

prncnfg.vbs -t [-s *Computer*`]` **[-u** *Benutzer*`]` **[-w** *Passwort*`]` **-p** *DruckerName* **[-r** *PortName*`]` **[-l** *Standort*`]` **[-m** *Kommentar*`]` **[-h** *Freigabename*`]` **[{+|-}shared] [{+|-}published]**

Konfiguriert einen Drucker. Es können Port, Standort, Kommentar und Freigabename zugewiesen werden. `shared` legt fest, ob der Drucker freigegeben wird, und `published` steuert, ob der Drucker im Active Directory veröffentlicht wird.

Hinweis: Dieses und die nachfolgenden Druckerverwaltungsskripte befinden sich nicht im Suchpfad, sondern sprachabhängig in Unterverzeichnissen von *Windows\system32\Printing_Admin_Scripts*. Wenn Sie `cscript` nicht schon als Standard für die Ausführung von VBS-Skriptdateien festgelegt haben, müssen Sie dem Aufruf *cscript.exe* voranstellen.

PoSh: PowerShell bietet neben weiteren druckerbezogenen Cmdlets die Befehle `set-printer`, `set-printerproperties` und `set-printconfiguration`. Mit diesen können Eigenschaften eines Druckers umfassend konfiguriert werden.

Folgendes Beispiel setzt die Papiergrößeneinstellung des angegebenen Druckers auf *A4*:

`set-printconfiguration -printername "HP Laserjet" -papersize` *A4*

---

prndrvr.vbs

Verwaltet Druckertreiber auf dem angegebenen oder dem lokalen Computer.

prndrvr [-*adlx?*] [-m *Treibermodellname*]
[-v *Version*] **[-e** *Umgebung*] **[-s** *Servername*]
[-u *Benutzername*] **[-w** *Kennwort*]
[-h *Treiberdateipfad*] **[-i** *Inf-Datei*]

Optionen

-a

Fügt den angegebenen Drucker hinzu.

-d

Löscht den angegebenen Drucker.

-l

Zeigt alle Treiber an.

-x

Löscht alle nicht verwendeten Treiber.

-?

Zeigt Hilfe und Beispiele an.

-e "Windows {*NT x86* | *X64* | *IA64*}"

Legt fest, für welche Umgebung der Treiber installiert wird.

-h *Treiberdateipfad* **[-i** *INF-Datei*]

Legt den Pfad zu den Dateien mit den Druckertreibern fest. Wird keine INF-Datei mit dem Parameter `-i` angegeben, sucht das Skript nach einer der *printer.inf*-Dateien aus dem *inf*-Unterverzeichnis des Treiberordners.

-v

Definiert die Treiberversion. Für die im Buch aufgeführten Windows-Versionen ist die Versionsnummer 3.

`prndrvr -a -m "Treiber" -v 3 -e "Windows x64" -i c:\temp\treiber\ druckername.inf -h c:\temp\treiber`

Fügt einen Druckertreiber für die 64-Bit-Version von Windows hinzu, deren Treiberdateien sich im Ordner *c:\temp\treiber* befinden.

PoSh: Die Cmdlets `get-printerdriver` und `add-printerdriver` dienen der Abfrage bzw. dem Hinzufügen von Druckertreibern mit der PowerShell.

prnjobs.vbs

Verwaltet Druckaufträge auf dem angegebenen oder lokalen Computer.

prnjobs [-*lmxz?*] [-s *Server*] **[-p** *Drucker*] **[-j** *Auftragskennung*] **[-u** *Benutzername*] **[-w** *Kennwort*]

Optionen

-l

Zeigt alle Aufträge an.

-m

Setzt den Auftrag fort.

-x

Bricht den Auftrag ab.

-z

Pausiert den Auftrag.

prnmngr.vbs

Verwaltet Drucker(-Verbindungen) und setzt den Standarddrucker auf dem angegebenen oder lokalen Computer.

prnmngr [-*adxgtl?*] [*c*] [-*s* *Server*] [-*p* *Drucker*] [-*m* *Treibermodell*] [-*r* *Anschluss*] [-*u* *Benutzername*] [-*w* *Kennwort*]

Optionen

-a

Fügt einen lokalen Drucker hinzu.

-ac

Fügt eine Verbindung zu einem Netzwerkdrucker hinzu.

-d

Löscht den Drucker.

-g

Ermittelt den Standarddrucker.

-l

Zeigt alle Drucker an.

-m

Benennt den zu verwendenden Treiber.

-p

Legt den Namen des Druckers fest.

-r

Weist den zu verwendenden Anschlussnamen zu.

-t

Legt den Standarddrucker fest.

-x

Löscht alle Drucker.

-xc

Löscht alle Druckerverbindungen.

-xo

Löscht alle lokalen Drucker.

prnmngr.vbs -a [-s *Computer*] **[-u** *Benutzer*] **[-w** *Passwort*]
-p *DruckerName* **-m** *TreiberName* **-r** *PortName*

Fügt einen lokalen Drucker hinzu, der den angegebenen Treiber und Port verwendet.

prnmngr.vbs -ac -p *DruckerName*

Installiert den angegebenen Netzwerkdrucker.

prnmngr.vbs -d [-s *Computer*] **[-u** *Benutzer*]
[-w *Passwort*] **-p** *DruckerName*

Löscht den angegebenen Drucker.

prnmngr.vbs -x [-s *Computer*] **[-u** *Benutzer*]
[-w *Passwort*]

Löscht sämtliche Drucker vom lokalen oder angegebenen Computer.

prnmngr.vbs -g

Gibt den Namen des Standarddruckers aus.

prnmngr.vbs -t -p *DruckerName*

Setzt den angegebenen Drucker als Standarddrucker.

prnmngr.vbs -l [-s *Computer***] [-u** *Benutzer***] [-w** *Passwort***]**

Gibt Informationen zu allen installierten Druckern aus.

PoSh: Mit dem Cmdlet `add-printer` fügen Sie lokale und Netzwerkdrucker hinzu, mit `remove-printer` entfernen Sie den angegebenen Drucker.

`add-printer -connectionname \\`*`Server`*`\`*`Freigabe`*

---

prnport.vbs

Verwaltet Druckerports vom Typ *Standard TCP/IP-Port* auf dem angegebenen oder dem lokalen Computer.

prnport [-*adlgt?*] [-*r* *Anschluss*] [-*s* *Server*] [-*u* *Benutzername*] [-*w* *Kennwort*] [-*o* *raw*|*lpr*] [-*h* *Hostadresse*] [-*q* *Warteschlange*] [-*n* *Nummer*] [-*me*|-*md*] [-*i* *SNMP-Index*] [-*y* *Community*] [-*2e*|-*2d*]

Optionen

-a

Fügt einen Anschluss hinzu.

-d

Löscht den angegebenen Anschluss.

-g

Ermittelt die Konfiguration für einen TCP-Port.

-h

Die IP-Adresse des Geräts.

-i

Bei konfiguriertem SNMP der SNMP-Index des Geräts.

-l

Listet alle TCP-Ports auf.

-m[e|d]

SNMP-Typ aktivieren (e) oder deaktivieren (d).

-n

Portnummer bei TCP-RAW-Anschlüssen.

-o [raw | lpr]

Legt fest, ob der Port vom Typ RAW oder LPR ist.

-q

Warteschlangenname (nur für TCP-Ports).

-r

Anschlussname.

-t

Setzt die Konfiguration für einen TCP-Port.

-y

SNMP-Community-Name (nur wenn SNMP aktiviert ist).

-2[e | d]

Aktivieren (e) oder Deaktivieren (d) des doppelten Spool-Vorgangs für TCP-LPR-Ports.

Beispiele

cscript prnport.vbs -l -s *Server*

Zeigt alle TCP-Ports des angegebenen Servers an.

prnport -d -s *Server* **-r** *IP_1.2.3.4*

Löscht den Port mit dem Namen *IP_1.2.3.4* vom angegebenen Server.

prnport -a -s *Server* **-r** *IP_1.2.3.4* **-h** *1.2.3.4* **-o raw -n 9100**

Fügt dem angegebenen Server einen neuen Anschluss vom Typ RAW mit dem Namen *IP_1.2.3.4* und der IP-Adresse *1.2.3.4* sowie Port *9100* hinzu.

prnport -t -s *Server* **-r** *IP_1.2.3.4* **-me -y** *public* **-i 1 -n 9100**

Aktiviert und konfiguriert SNMP für den genannten Port des Servers.

prnport -g -s *Server* **-r** *IP_1.2.3.4*

Fragt die Konfiguration des genannten Ports auf dem Server ab.

prnport -a -r *IP_1.2.3.4* **-h** *1.2.3.4*

Fügt dem lokalen System einen Druckerport mit dem Namen *IP_1.2.3.4* hinzu, der auf das Gerät mit der IP-Adresse *1.2.3.4* verweist.

PoSh: PowerShell-Cmdlets zur Verwaltung der Druckerschnittstellen sind `add-printerport` zum Erstellen, `get-printerport` zum Abrufen von Informationen und `remove-printerport` zum Entfernen.

```
add-printerport -name "Laserdrucker01:" -PrinterHostAddress "192.168.0.123"
```

Erzeugt einen neuen TCP-Druckerport mit dem Namen *Laserdrucker01*, der auf die IP-Adresse *192.168.0.123* verweist.

---

prnqctl.vbs

Verwaltet Druckerwarteschlangen auf dem angegebenen oder lokalen Computer.

prnqctl [-*emxz?*] [-*s* *Server*] [-*p* *Drucker*]
[-*u* *Benutzername*] [-*w* *Kennwort*]

-e

Druckt die Testseite.

-m

Reaktiviert den (zuvor mit `-z` angehaltenen) Drucker.

-x

Löscht alle Druckaufträge vom Drucker.

-z

Pausiert den Drucker.

Beispiele

prnqctl.vbs -e [-s *Computer*] **[-u** *Benutzer*]
[-w *Kennwort*] **-p** *DruckerName*

Druckt eine Testseite auf dem angegebenen Drucker.

prnqctl.vbs -x [-s *Computer*`]` **[-u** *Benutzer*`]`
[-w *Kennwort*`]` **-p** *DruckerName*

Bricht alle Druckaufträge des angegebenen Druckers ab.

PoSh: PowerShell-Cmdlets zur Verwaltung von Druckaufträgen beinhalten `printjob` als Namensbestandteil. Dazu gehören unter anderem `get-printjob` zum Abrufen von Informationen, `restart-printjob` zum Neustarten eines hängenden Druckauftrags und `remove-printjob` zum Löschen.

`remove-printjob -printername "`*`Laserjet`*`" -ID `*`1`*

Entfernt den Druckjob mit der ID *`1`* vom Drucker *`Laserjet`*.

---

pubprn.vbs

Stellt einen einzelnen oder alle freigegebenen Drucker eines Servers in einem Active-Directory-Container bereit.

[cscript] pubprn.vbs [`Server | Server\Freigabename`**]**
"LDAP:`//OU=..,DC=...`**"**

Beispiele

pubprn.vbs *Server* **"LDAP://CN=**`Drucker`**,DC=**`Test`**,DC=**`Firma`**,DC=**`de`**"**

Stellt alle von *`Server`* freigegebenen Drucker im Container *Drucker* auf Domänenebene bereit.

pubprn.vbs `\\Server\Drucker` **"LDAP:**
`//`**CN=**`Drucker`**,DC=**`Test`**,DC=**`Firma`**,DC=**`de`**"**

Stellt den als *`Drucker`* freigegebenen Drucker des Servers *`Server`* im Container *Drucker* bereit.

---

rundll32 printui.dll

Ermöglicht die skriptbasierte Installation, das Verbinden, die Konfiguration und das Löschen von lokalen und Netzwerkdruckern. Es handelt sich hier um keine reine Konsolenanwendung, daher wird an dieser Stelle auf die Windows-Hilfe verwiesen. Groß- und Kleinschreibung ist bei diesem Befehl zu berücksichtigen. Die Syntax sieht wie folgt aus:

```
rundll32 printui.dll,PrintUIEntry [Optionen]
[@Befehlsdatei]
```

Die Hilfe zu den verfügbaren Optionen erhalten Sie durch diesen Aufruf:

```
rundll32 printui.dll,PrintUIEntry /?
```

Registrierung

lodctr

lodctr *Argumente*

Aktualisiert die Registrierungswerte, die sich auf die Leistungsindikatoreneinstellungen (Performance-Counter) für Dienste oder Treiber beziehen, sichert die Leistungsindikatoreneinstellungen und die Erklärungen und stellt sie wieder her.

Argumente

INI-Dateiname

Bei der INI-Datei handelt es sich um eine Initialisierungsdatei, die die Definitionen der Leistungsindikatorennamen und erklärungen für eine erweiterbare DLL beinhaltet.

/s:*Sicherungsdateiname*

Speichert die aktuellen Zeichenfolgen der Leistungsindikatorenregistrierung und der Erklärungen in der genannten Sicherungsdatei im Textformat.

/r

Ohne weitere Argumente werden aus aktuellen Registrierungseinträgen und Sicherungsdateien die Zeichenfolgen der Leistungsindikatorenregistrierung und die zugehörigen Erklärungen neu aufgebaut.

/r:*Sicherungsdateiname*

Überschreibt die Zeichenfolgen der Leistungsindikatorenregistrierung und die Erklärungen unter Verwendung der in der Sicherungsdatei enthaltenen Informationen.

/t:*Dienstname*

Legt den Leistungsindikatorendienst als vertrauenswürdig fest.

/e:*Dienstname*

Aktiviert den angegebenen Leistungsindikatorendienst.

/d:*Dienstname*

Deaktiviert den angegebenen Leistungsindikatorendienst.

/q

Fragt Informationen zum Leistungsindikatorendienst ab.

/q:*Dienstname*

Fragt Informationen zu einem bestimmten Dienst ab.

/m:*Leistungsindikatormanifest*

Installiert die XML-Datei mit der Leistungsindikatorenanbieter-Definition in das System-Repository.

Zum Löschen von Namen und Erklärungen für erweiterbare Leistungsindikatoren können Sie den Befehl `unlodctr` verwenden.

---

reg

reg *Operation Argumente* **[***Optionen***]**

Sehr mächtiger Befehl zur Bearbeitung der Registrierung, der oftmals auch dann noch funktioniert, wenn z.B. durch Malware GUI-basierte Methoden für Registrierungsanpassungen deaktiviert wurden. Die Syntax ändert sich je nach der angegebenen Operation. Im Folgenden sind alle Operationen beschrieben, jedoch aus Platzgründen nur mit den gebräuchlichsten Optionen. Beachten Sie, dass alle Registrierungsschlüssel als komplette Pfade angegeben werden, beginnend mit einer der Standardabkürzungen für die Rootschlüssel (*HKLM*, *HKCU*, *HKCR* oder *HKCC*).

Häufige Optionen

/v *Name-des-Werts* **| /ve**

Gibt den zu bearbeitenden Registrierungswert an. `/ve` bezeichnet den Standardwert (oberste Zeile des Schlüssels in Regedit). Wird die Option

nicht angegeben, erfolgt die Abfrage aller Werte unter dem Schlüssel.

/s

Der Befehl bezieht sich rekursiv auf alle dem angegebenen Schlüssel untergeordneten Strukturen.

/se

Gibt das Trennzeichen für den Typ *REG_MULTI_SZ* an. Der Standardwert ist `\0`.

/f

Gibt die zu suchenden Daten bzw. Muster an. Standard ist * als Platzhalter für beliebige Daten.

/k | /d

Gibt an, dass nur in Schlüsselnamen (`/k`) oder in Daten (`/d`) gesucht werden soll.

/c

Gibt an, dass Groß- und Kleinschreibung bei der Suche berücksichtigt werden sollen.

/e

Bewirkt, dass nur exakte Übereinstimmungen zurückgegeben werden.

/t *REG_SZ* | *REG_MULTI_SZ* | *REG_EXPAND_SZ* | *REG_DWORD* | *REG_QWORD* | *REG_BINARY* | *REG_NONE*

Gibt den Datentyp für den Registrierungswert an. Standardmäßig werden alle Typen verwendet.

/reg:32 | 64

Gibt an, dass der Zugriff auf den Schlüssel über die 32-Bit- oder die 64-Bit-Registrierungsansicht erfolgen soll.

Verfügbare Operationen

reg query *[**Rechner*\]**Rootschlüssel**Unterschlüssel* [*Optionen*]**

Gibt untergeordnete Schlüssel, Werte und Daten des angegebenen Schlüssels aus.

reg add [*Rechner*\]**Schlüssel* [*Optionen*]**

Fügt den angegebenen Schlüssel oder Wert hinzu. Akzeptiert `/v`, `/ve` und

`/f`. Des Weiteren werden unterstützt: `/t`*Typ* (gibt den Datentyp an), `/d`*Daten* (setzt den Wert) und `/s`*Zeichen* (gibt ein Trennzeichen für Werte des Typs *REG_EXPAND_SZ* an; das Standardzeichen ist `\0`).

reg copy [*Rechner***]*Quellschlüssel* [***Rechner***]*Zielschlüssel* [*Optionen*]**

Kopiert einen Schlüssel. Unterstützt `/s` und `/f`.

reg delete [*Rechner***]*Schlüssel* [*Optionen*]**

Löscht einen Schlüssel oder Wert. Unterstützt `/f`, `/v`, `/ve` und `/va` (löscht alle Werte).

reg compare [*Rechner1***]*Schlüssel1* [***Rechner2***]*Schlüssel2* [*Optionen*] [*Ausgabeoption*]**

Vergleicht zwei Schlüssel. Mit `/s` werden zwei Unterbäume verglichen. Die Optionen `/v` und `/ve` werden unterstützt. *Ausgabeoption* definiert, wie Unterschiede ausgegeben werden. Folgende Optionen sind möglich: `/oa` (alle Informationen), `/od` (nur Unterschiede), `/os` (nur Übereinstimmungen) oder `/on` (keine Ausgabe, sondern Ergebnisse als Rückgabecode [Errorlevel]: 0 = gleich, 2 = unterschiedlich, 1 = Fehler).

reg export *Schlüssel* *Datei* **[*/y*]**

Exportiert einen lokalen Registrierungsschlüssel in eine Datei (mit `/y` wird *Datei* ohne Rückfrage überschrieben, falls sie schon existiert). Diese Operation arbeitet rekursiv.

reg import *Datei*

Importiert mit `reg export` gesicherte Registrierungsdaten aus *Datei* in die lokale Registrierung.

reg save [*Rechner***]*Schlüssel*** *Datei*

Speichert den angegebenen Schlüssel in *Datei* ab. Diese Operation arbeitet rekursiv.

reg restore [*Rechner***]*Schlüssel*** *Datei*

Stellt die Registrierungsdaten aus *Datei*, die mit `reg save` gespeichert wurden, an der ursprünglichen (angegebenen) Stelle wieder her.

reg load [*Rechner***]*Schlüssel*** *Datei*

Stellt die Registrierungsdaten aus *Datei*, die mit `reg save` gespeichert wurden, an einer anderen Stelle temporär wieder her. Die geladenen

Daten bleiben nach einem Neustart nicht erhalten.

`reg unload [\\`*`Rechner`*`\]`*`Schlüssel`*

Entlädt einen vorher temporär geladenen Registrierungsschlüssel.

`reg flags` ***`Schlüssel`*** **`[set [`*`DONT_VIRTUALIZE`*`] [`*`DONT_SILENT_FAIL`*`] [`*`RECURSE_FLAG`*`]]`**

Erfragt (ohne `set`) oder setzt Virtualisierungsflags für einen Schlüssel unterhalb von *HKLM*: nicht virtualisieren (`DONT_VIRTUALIZE`), bei abgeschalteter Virtualisierung alle Zugriffsrechte des Benutzers gewähren (`DONT_SILENT_FAIL`) oder Virtualisierungseinstellungen auf neue Unterschlüssel vererben (`RECURSE_FLAG`).

PoSh: PowerShell ermöglicht das Editieren der Registrierung auf vielfältige Art und Weise – Sie können jegliches Registry-Tool der Eingabeaufforderung verwenden oder auch manuell die Registrierung bearbeiten. Die Vielfalt der Möglichkeiten kann in diesem Buch nicht abgehandelt werden, daher folgt an dieser Stelle nur ein kurzer Exkurs.

Warnung: Jeder unsachgemäße Eingriff in die Registrierung kann das System startunfähig machen oder anderweitig Datenverlust herbeiführen. Deshalb sollten Sie diesbezüglich besondere Sorgfalt walten lassen und für den Fall der Fälle eine aktuelle Datensicherung besitzen.

PowerShell behandelt die Registrierung als eigenständiges Laufwerk und kann auf die Schlüssel *HKLM* und *HKCU* direkt zugreifen:

Mit `CD HKLM`: wechseln Sie auf das PowerShell-Laufwerk. Alternativ können Sie auch ohne Wechsel auf das Laufwerk direkt mit `getchilditem` auf die Schlüssel zugreifen:

`get-childitem -path hkcu:\Software`

Listet die unmittelbaren Unterschlüssel des Schlüssels *HKCU\Software* und deren Werte auf. Um auch sämtliche Unterschlüssel des angegebenen Schlüssels angezeigt zu bekommen, hängen Sie `-recurse` an den Befehl.

`new-item -path hkcu:\software\`*`EinTest`*

Erzeugt den neuen Schlüssel *`EinTest`* unter *HKCU\Software*.

`remove-item -path 'hkcu:\software\`*`EinTest'`*

Löscht den Schlüssel. Die Hochkommata sind nur erforderlich, wenn der

Schlüsselname Leerzeichen oder durch PowerShell anderweitig interpretierte Sonderzeichen enthält. Beinhaltet der Schlüssel weitere Elemente, werden Sie gefragt, ob Sie diese ebenfalls löschen wollen. Diese Abfrage umgehen Sie durch ein angehängtes `-recurse`.

Um gezielt Werte abzufragen, können Sie folgenden Befehl anpassen:

```
get-item -path
HKLM:\software\Microsoft\Windows\CurrentVersion\Run
```

Dieser Befehl zeigt die Werte an, die im *Run*-Schlüssel in *HKLM* eingetragen sind, also beim Starten des Betriebssystems ausgeführt werden.

Um dort einen Wert gezielt zu entfernen, den z.B. Adware hinterlegt hat, können Sie das Cmdlet `remove-itemproperty` verwenden:

```
remove-itemproperty -path
HKLM:\software\Microsoft\Windows\CurrentVersion\Run -name
xxyyzz
```

Dieser Befehl entfernt den Wert mit dem Namen *xxyyzz* aus dem *Run*-Schlüssel, sofern zum einen die PowerShell als Administrator gestartet wurde und zum anderen laufende Malware nicht aktiv das Löschen verhindert (was sich oft durch den Start von Windows im abgesicherten Modus umgehen lässt).

---

regini

regini *Skriptdatei*

Importiert Schlüssel und Werte in die Registrierung von lokalen und Remotesystemen anhand der Informationen aus der angegebenen Skriptdatei. Als Besonderheit können Berechtigungen gesetzt werden.

Syntax der Skriptdateien

```
Registry\Hive\Schlüssel\Unterschlüssel1 [ACL]
Unterschlüssel2a [ACL]
Wert = Datentyp Wertdaten [ACL]
```

Unterschlüssel2b [*ACL*]

Die Hierarchie von Schlüsseln, Unterschlüsseln und Werten kann entweder durch Verwendung von Backslashs zwischen einem Schlüssel und seinem Unterschlüssel ausgedrückt werden oder durch eine entsprechende Einrückung unter Verwendung von Leer- oder Tabulatorzeichen. Zu setzende Berechtigungen (ACLs) werden als durch Leerzeichen getrennte Zahlen angegeben (1 bis 23 sind definiert), die jeweils für eine bestimmte Berechtigung stehen. Einige gängige Berechtigungen sind folgende:

1

Administratoren: Vollzugriff

7

Jeder: Vollzugriff

8

Jeder: Lesen

17

System: Vollzugriff

Die gängigsten Datentypen werden wie folgt verwendet, um Werte zu setzen oder zu löschen:

Datentyp

Wert

REG_SZ

Eine Zeichenkette ohne Anführungszeichen. Das ist zugleich Standard, wenn kein Wert festgelegt wurde.

REG_EXPAND_SZ

Eine Zeichenkette ohne Anführungszeichen (darin enthaltene Umgebungsvariablen werden beim Auslesen expandiert).

REG_MULTI_SZ

Mehrere Zeichenketten jeweils in Anführungszeichen, getrennt durch Leerzeichen. Ein Backslash am Zeilenende gibt an, dass die Liste in der nächsten Zeile fortgesetzt wird.

REG_DWORD

Eine Zahl. Hexwerte werden durch das Präfix *0x* gekennzeichnet.

REG_BINARY

Die erste Zahl spezifiziert die Anzahl der folgenden Bytes. Die Datenbytes werden jeweils als 4-Byte-Zahlen angegeben, getrennt durch Leerzeichen. Ein Backslash am Zeilenende gibt an, dass die Liste in der nächsten Zeile fortgesetzt wird. Hexwerte werden durch das Präfix *0x* gekennzeichnet.

DELETE

Löscht den zugehörigen Wert.

PoSh: PowerShell verfügt über die Cmdlets `get-acl` zum Auslesen von Berechtigungen und `set-acl`, um Berechtigungen zu setzen. Das folgende Skriptbeispiel setzt unter Verwendung der auf den MSDN-Webseiten von Microsoft dokumentierten .NET-Klasse `Registry-AccessRule` für den angegebenen Benutzer Berechtigungen auf den der Variablen `$regkey` zugeordneten Schlüssel.

```
$regkey = 'HKCU:\Software\Test'
$acl = get-acl $RegKey
$rule = new-object
System.Security.AccessControl.RegistryAccessRule
("Computer\Benutzer","SetValue, CreateSubKey,
ReadKey","ContainerInherit","None","Allow")
$acl.SetAccessRule($rule)
$acl | set-acl -path $regkey
```

Prozesse

handle WWW

handle [-a**] [**-u**] [**-c *Handle* **[**-y**]]** | **[**-s**]] [**-p *Prozessname*| PID **[***Name***]**

Dieses Sysinternals-Tool zeigt offene Handles aller Prozesse (-a) oder nach Benutzernamen des Besitzers (-u) an, um zum Beispiel Performanceengpässen oder nicht löschbaren Dateien auf die Spur zu kommen. Optional werden nur die Handles eines per Name oder PID angegebenen Prozesses oder eines bestimmten Objekts aufgelistet. Mit -c und als Hexadezimalzahl angegebenem Handle kann dieses geschlossen werden. Dabei ist das Risiko einer Destabilisierung des laufenden Betriebssystems zu beachten.

Beispiel

handle -p *expl temp*

Listet alle Objekte (Dateien, Registry-Einträge) auf, die gerade von einem Prozess mit *expl* im Namen geöffnet sind und in einem Pfad liegen, der die Zeichenkette *temp* enthält – kurz, alle vom Explorer geöffneten temporären Dateien.

psexec WWW

psexec [*Optionen*] *Befehl*

Dieser leistungsfähige Befehl aus der zu Microsoft gehörenden Toolschmiede Sysinternals startet Kommandozeilenprogramme auf entfernten Systemen, wobei Ein- und Ausgabe auf den lokalen Computer umgeleitet werden. Auf den entfernten Systemen erzeugt psexec über das Netzwerk kurzerhand

einen Dienst, der zum Starten des angegebenen Befehls dient und nach dessen Beendigung wieder gelöscht wird.

An dieser Stelle werden nur die wichtigsten Optionen beschrieben. Ein Aufruf des Befehls ohne Parameter liefert Erklärungen zu den umfangreichen Optionen.

Optionen

*****Computer1***[,***Computer2***[,...]] | @***Datei*

Führt den Befehl auf den angegebenen Computern oder den in *Datei* angegebenen Computern aus.

-c

Kopiert die angegebene Befehlsdatei auf den entfernten Computer. Falls diese Option nicht angegeben wird, muss der verwendete Befehl im Pfad des entfernten Systems liegen. Alternativ kann der volle Pfad zum Befehl aus Sicht des entfernten Systems angegeben werden.

-w *Arbeitsverzeichnis*

Setzt das *Arbeitsverzeichnis* für den Befehl relativ zum entfernten System.

[-u *Benutzer* **[-p** *Kennwort***]]**

Verwendet nicht den aktuellen, sondern den angegebenen Benutzer. Wird kein *Kennwort* angegeben, wird es an der Eingabeaufforderung abgefragt.

Tipp: Interessant kann die Ausführung eines Befehls im Kontext des lokalen Systemkontos sein, um Zugriff auf Bereiche zu erlangen, die selbst Konten mit Administratorrechten verwehrt sind:

```
psexec -s cmd
```

startet auf dem eigenen Computer eine Instanz der Eingabeaufforderung im Systemkontext. Ersetzen Sie *cmd* bei Bedarf durch ein anderes Programm, beispielsweise PowerShell.

PoSh: PowerShell ermöglicht den Remotezugriff auf andere Rechner. Das gilt zum einen für Cmdlets, die sich über Computernamen auf andere

Systeme beziehen, und zum anderen für komplette PowerShell-Sitzungen.

Ist der Zielrechner korrekt konfiguriert und Mitglied derselben Domäne und hat der Benutzer das Recht zur lokalen Anmeldung auf dem Zielrechner, können Sie mit dem folgenden Befehl eine Konsolensitzung direkt auf dem Remoterechner eröffnen:

`new-pssession -computername` *`Zielrechner`* `-credential` *`Domänenname`*`\` *`Benutzername`*

Sie beenden die Sitzung mit `exit-pssession`.

---

runas

`runas [`*`Optionen`*`] /user:`*`Benutzername Befehl`***

Führt den *`Befehl`* im Kontext des angegebenen Benutzers aus.

`runas [ [`*`/noprofile`*` | `*`/profile`*`] [`*`/env`*`] [`*`/savecred`*` | `*`/netonly`*`] ] /user:`*`Benutzername`*** **`Programm`**

`runas [ [`*`/noprofile`*` | `*`/profile`*`] [`*`/env`*`] [`*`/savecred`*`] ] `*`/smartcard`*` [`*`/user`*`:`*`Benutzername`***`] `*`Programm`*

`runas [ [/machine:<`*`machinetype`***`>] ] `*`/trustlevel`*`:`***`Vertrauensstufe`*** **`Programm`**

Optionen

`/env`

Verwendet die aktuelle Umgebung statt der des angegebenen Benutzers. Dadurch enthält z.B. die Umgebungsvariable `%userprofile%` den Pfad nicht zum Profil des angegebenen, sondern des aufrufenden Benutzers.

`/machine`

Gibt die Computerarchitektur des Prozesses an. Dabei sollte *`<machinetype>`* einer der Typen `x86|amd64|arm|arm64` *`sein.`*

`/netonly`

Der angegebene Benutzer muss nur über das Recht zur Anmeldung über das Netzwerk verfügen, nicht über das Recht zur lokalen Anmeldung.

`/noprofile`

Lädt das Benutzerprofil des angegebenen Benutzers nicht, was bei einigen Anwendungen zu Fehlern führen kann. Ohne Angabe dieser Option wird das Profil geladen.

`/savecred`

Speichert das Kennwort bei der Eingabe bzw. fragt nicht nach dem Kennwort des angegebenen Benutzers, falls es zuvor gespeichert wurde. Diese Option ist unter Sicherheitsaspekten bedenklich.

`/smartcard`

Verwendet auf einer Smartcard gespeicherte Anmeldeinformationen.

`/showtrustlevels`

Zeigt die definierten Vertrauensstufen, die für die Option `/trustLevel` verwendbar sind.

`/trustLevel:`*`Stufe`*

Führt einen Befehl in der angegebenen Vertrauensstufe aus, um beispielsweise als Administrator Programme mit eingeschränkten Rechten zu starten.

Falls `/netonly` nicht angegeben wird, muss der angegebene Benutzer über die Berechtigung zur lokalen Anmeldung verfügen. `/netonly funktioniert nicht mit /profile.`

Das Kennwort des angegebenen Benutzers wird an der Eingabeaufforderung abgefragt. Es ist keine Möglichkeit bekannt, es automatisiert zu übergeben (mit Ausnahme der Verwendung von `/savecred`), was den Nutzen von `runas` in Skripten stark einschränkt.

Sollten einzelne Befehle nicht wie gewünscht arbeiten, kann es hilfreich sein, zunächst eine Instanz von *cmd.exe* im gewünschten Kontext aufzurufen und von hier aus das gewünschte Programm zu starten.

PoSh: PowerShell ermöglicht die Weitergabe oder Abfrage von Anmeldeinformationen mit dem Parameter `–credential 'Domäne\Benutzername'`. Dieser fragt entweder per Pop-up-Dialog nach dem Kennwort des angegebenen Benutzerkontos, oder es wird von vornherein ein Credential-Objekt hinterlegt und dem Befehl mitgegeben. Die

Hochkommata sind dringend empfohlen, damit bestimmte Sonderzeichen im Kennwort nicht zu Irritationen der PowerShell-Befehle führen.

```
$benutzer = 'Domäne\Admin'
$kennwort = 'Start@25$'
$securePassword = convertto-securestring $kennwort
-asplaintext -force
$credential = new-object
System.Management.Automation.PSCredential
$benutzer, $securePassword
Start-Process notepad.exe -credential $credential
```

Warnung: Sie können das Credential-Objekt ebenfalls in Dateiform abspeichern, aber auch wenn der Inhalt von `$securePassword` nicht länger das Kennwort im Klartext enthält, kann es in falsche Hände gelangen und missbraucht werden. Sichern Sie zumindest den Speicherort gegen unbefugten Zugriff auf das Dateisystem.

START

START ["*Fenstertitel*"] [***Optionen***] *Befehl Parameter*

Führt einen Befehl aus. Auf Beendigung der Ausführung kann gewartet werden, Prozesspriorität und Prozessoraffinität können festgelegt werden.

Optionen

/d *Verzeichnispfad*

Setzt das Arbeitsverzeichnis für den Befehl.

/b

Startet den Befehl als Hintergrundprozess, erstellt also kein neues Fenster.

/i

Übergibt die aktuelle Umgebung an den gestarteten Prozess.

/low* | */belownormal* | */normal* | */abovenormal* | */high* | */realtime

Legt die Priorität des neuen Prozesses fest.

/affinity *Prozessor*

Setzt die Nummer des Prozessors, auf dem der neue Prozess ausgeführt werden soll (als Hexadezimalzahl).

/machine [*x86* | *amd64* | *arm* | *arm64*]

Gibt die Computerarchitektur des Anwendungsprozesses an.

/node

Legt den Knoten der bevorzugten NUMA (*Non-Uniform Memory Architecture*) als ganzzahligen Dezimalwert fest. Bei Kombination mit der Affinitätsmaske des Prozessors wird diese abweichend interpretiert.

/wait

Startet den Befehl und wartet auf seine Beendigung.

/min* | */max

Startet den Befehl in einem minimierten oder maximierten Fenster.

---

tasklist

tasklist [/s *System* **[/u** *Benutzername* **[/p [***Kennwort***]]]] [/m [***Modul***]** | **/svc** | **/apps** | **/v] [/fi** *Filter***] [/fo** *Format***] [/nh]**

Zeigt ähnlich wie der Task-Manager laufende Prozesse und deren Speichernutzung auf dem angegebenen oder lokalen System an (Benutzer und Kennwort können zur Verbindung angegeben werden). Zusätzlich können mit `/m` die von den einzelnen Prozessen geladenen DLLs aufgelistet bzw. nach Prozessen gefiltert werden, die ein bestimmtes Modul geladen haben. In einem Prozess enthaltene Dienste (`/svc`) oder Store-Apps und deren verknüpfte Prozesse (`/apps`) können angezeigt oder detailliertere Informationen (`/v`) ausgegeben werden. Mit der Option `/fo` wird das Ausgabeformat festgelegt: tabellarisch (`table`), Listenform (`list`) oder CSV-Format (`csv`). `/nh` unterdrückt die Ausgabe von Spaltenüberschriften.

Zur Definition von Filtern sei auf die Hilfe des Befehls verwiesen.

PoSh: Mit dem PowerShell-Cmdlet `get-process` erhalten Sie eine leistungsstarke Methode zum Auflisten und Filtern laufender Prozesse. So listet der folgende Befehl die mit *c* beginnenden Prozessnamen mit Prozess-ID und Anzahl der durch diesen Prozess verwendeten Handles auf:

```
get-process c* | format-table Name, ID, handles -autosize
```

---

taskkill

taskkill [/fi *Filter***] [/pid** *Prozess-ID* | **/im** *EXE-Datei***] [/s** *System* **[/u** *Benutzer* **[/p [**Passwort**]]]] [/f] [/t]**

Beendet einen oder mehrere Prozesse lokal oder auf einem Remotesystem, zu dem mit dem angegebenen Benutzernamen eine Verbindung hergestellt werden kann. Die zu beendenden Tasks werden durch ihre PID (*Prozess-ID*) oder den Namen der EXE-Datei ausgewählt, alternativ oder zusätzlich kann ein Filter angegeben werden, dessen Syntax der des `tasklist`-Befehls entspricht. `/f` erzwingt das Beenden. Mit `/t` wird der ganze Prozessbaum beendet, also auch die vom angegebenen Prozess gestarteten Kindprozesse.

Zur Definition von Filtern sei auf die Hilfe des Befehls verwiesen.

PoSh: Prozesse lassen sich mit Windows PowerShell auf verschiedene Weise beenden. Stellvertretend wird an dieser Stelle das Cmdlet `stop-process` aufgeführt.

```
get-process msedge | stop-process
```

Beendet alle laufenden Instanzen von Microsoft Edge. Anstelle des Prozessnamens können Sie auch die Prozess-ID angeben. Mit `-force` als zusätzlichem Parameter wird das Schließen von Prozessen erzwungen.

---

unlodctr

unlodctr Argumente

Deinstalliert den angegebenen Leistungsindikatoranbieter.

Argumente

`Dienstname`

Deinstalliert den Leistungsindikatoranbieter der Version 1.0, der dem angegebenen Dienst zugeordnet ist.

`/m:`*`Manifest`*

Deinstalliert einen Leistungsindikatoranbieter der Version 2.0 mittels der Anbieter-GUID aus dem angegebenen XML-Manifest.

`/g:`*`{Anbieter-GUID}`* **`| /p:`***`Anbietername`*

Durch die Anbieter-GUID oder den Anbieternamen wird der zu deinstallierende Leistungsindikatorenanbieter angegeben.

Dienste

net continue

`net continue` *`Dienst`*

Setzt die Ausführung eines angehaltenen Diensts fort.

net pause

`net pause` *`Dienst`*

Hält einen laufenden Dienst vorübergehend an.

net start

`net start [`*`Dienst`***`]`**

Startet einen Dienst. Falls der angegebene Dienstname Leerzeichen enthält, muss er in Anführungszeichen gesetzt werden. Ohne Parameter gibt dieser Befehl die aktiven Dienste mit ihren internen Namen aus.

net stop

`net stop` *`Dienst`*

Beendet den angegebenen Dienst.

pssuspend WWW

pssuspend [-*r*] [*computer* **[-*u* *Benutzer*]**
[-*p* *Kennwort*]] [*Prozessname*|*Prozess-ID*]

Dieses Sysinternals-Tool ermöglicht es, laufende Prozesse pausieren zu lassen oder aus irgendwelchen Gründen pausierte Prozesse fortzusetzen.

Argumente

/r

Setzt einen pausierten Prozess fort.

Computer

Gibt den Namen des Windows-Rechners an, auf dem der Prozess pausiert oder wiederaufgenommen werden soll. Das System muss über die Netzwerkumgebung erreichbar sein.

/u *Benutzer* **| /p [***Kennwort***]**

Falls Sie auf ein Remotesystem zugreifen wollen und Ihr lokales Anmeldekonto auf diesem keine Administratorrechte hat, geben Sie hiermit den Benutzernamen und das Kennwort für ein berechtigtes Konto ein. Wenn Sie den Parameter /p weglassen, können Sie das Kennwort versteckt eingeben.

Prozessname|*Prozess-ID*

Definiert den Namen oder die Task-ID des anzusteuernden Prozesses.

SC

sc [*Server***]*** *Befehl Dienst [Optionen]*

Leistungsstarker Befehl zur Verwaltung von Diensten. Beachten Sie bei der Eingabe des Befehls, dass nach Gleichheitszeichen zur Angabe von Optionen ein Leerzeichen vor dem Wert erforderlich sein kann.

Befehle

query | queryex [Dienstname] [type= {*service* | *driver* | *all*}]
[state= [*all* | *inactive*] [bufsize= *Puffergröße***] [ri=***Index***]**

[group= *Dienstgruppe*]

Zeigt ohne Angabe von Dienstnamen den Status (mit `queryex` den erweiterten Status) aller aktiven Dienste und Treiber an. Durch Angabe eines Typbezeichners (`service`, `driver`, `all`, Standardwert ist `service`) mit `type` kann die Liste der angezeigten Dienste gefiltert werden. Die Angabe des Kriteriums `state` (Status) ermöglicht das Filtern nach aktiven (ist Standard ohne Angabe eines Parameters), inaktiven oder allen Zuständen. `bufsize` ändert die Größe des Auflistungspuffers (Standard sind 4.096 Byte), `ri` legt die Indexnummer zum Fortsetzen fest, bei der die Auflistung beginnen soll, und mit `group` ist die Filterung nach Dienstgruppen möglich.

start *Dienstname* [*Argumente*]

Startet den angegebenen Dienst.

stop *Dienstname* [*Grund***] [***Kommentar***]**

Beendet einen laufenden Dienst und hält optional einen angegebenen Grund sowie einen Kommentar in der Ereignisanzeige fest. Der Grund wird als Aneinanderreihung numerischer Werte aus einer in der Hilfe verfügbaren Liste angegeben, Kennung, Hauptgrund und weiterer Grund werden per Doppelpunkt miteinander verbunden. So bedeutet *1:4:6 Ungeplant, Software, Aufgehängt.* Der Kommentar darf maximal 127 Zeichen lang sein.

pause | continue *Dienstname*

Der `pause`-Befehl hält einen Dienst an, `continue` setzt ihn wieder fort.

config *Dienstname* [type= {*own* | *share* | *interact* | *kernel* | *filesys* | *rec* | *adapt* | *userown* | *usershare*}] [start= {*boot* | *system* | *auto* | *demand* | *disabled* | *delayed-auto*}] [error= {*normal* | *severe* | *critical* | *ignore*}] [binpath= *Binärpfadname zur EXE-Datei*] **[group= *Dienstgruppe*] [tag= {*yes* | *no*}] [depend=** *Abhängigkeiten* **(Trennung durch Schrägstrich) [obj= {***Kontenname* **|** *Objektname*}] **[displayname=** *Anzeigename*] **[password=** *Kennwort*]**

Mit der Operation `config` ändert `sc` Einträge zu einem Dienst in der Registrierung und der Dienstdatenbank. Die Optionen entsprechen den wesentlichen Angaben zu Diensten in Windows, beispielsweise können der Typ des Diensts (`type`), sein Startverhalten (`start`), die Relevanz eines Fehlers (`error`), der Pfad der ausführbaren Datei (`binpath`) und

der Anzeigename (displayname) festgelegt werden.

description [*Dienstname*] [*Beschreibung*]

Ändert die Beschreibung eines Diensts.

failure [*Dienstname*] [reset= (*Zeitraum ohne Fehler in Sekunden, nach dem der Fehlerzähler auf 0 gesetzt wird*)] [reboot= (*vor dem Neustart übertragene Nachricht*)] [command= (*bei Fehler auszuführende Befehlszeile*)] [actions= {*run* | *restart* | *reboot*}/(*Verzögerungszeit in Millisekunden*)]

Legt fest, was bei einem Fehler des Diensts passiert: den Zeitraum, nach dem der Fehlerzähler auf 0 gesetzt wird (reset), im Fehlerfall auszuführende Befehle (command) und im Zusammenhang mit der reset-Option, welche Aktionen ausgeführt werden sollen (actions). Bei letzterer Operation werden sowohl die einem Fehler folgende Aktion und Verzögerungszeit (in Millisekunden) bis zu ihrer Ausführung als auch Folgeaktionen durch Schrägstrich getrennt, z.B. run/5000/reboot/1000.

failureflag [*Dienstname*] [*Flag*]

Ändert das Flag, das das Verhalten eines Diensts bei einem auftretenden Fehler bestimmt. Bei der Standardeinstellung 0 werden konfigurierte Vorgänge zur Fehlerbehandlung vom Dienststeuerungsmanager nur aktiviert, wenn der Dienst nicht den Status Stopped aufweist. Wird die Einstellung 1 gesetzt, erfolgt die Aktivierung konfigurierter Vorgänge nur, wenn zusätzlich zum Status Stopped der Win32-Beendigungscode nicht auf 1 gesetzt ist.

sidtype [*Dienstname*] [type= {*none* | *unrestricted* | *restricted*}

Verändert den SID-Typ des angegebenen Diensts und fügt die SID dem Dienstprozesstoken hinzu. Die Einstellung unrestricted (unbeschränkt) gilt nur für Dienste, die im Win32-Benutzermodus laufen. Mit der Einstellung restricted (eingeschränkt), die ebenfalls nur für Dienste im Win32-Benutzermodus gilt, wird die SID des Diensts in die Liste der eingeschränkten SIDs im Prozesstoken eingetragen. Die Angabe none fügt den Dienst nicht dem Prozesstoken für Dienste hinzu. Diese Einstellungen treten erst in Kraft, wenn der betreffende Dienst neu gestartet wird.

privs [*Dienstname*] [*Berechtigungen*]

Ändert die erforderlichen Berechtigungen für einen Dienst. Diese können abweichend von den Berechtigungen für das für den Start des Diensts verwendete Konto konfiguriert werden. Mehrere Berechtigungen werden durch Schrägstrich voneinander getrennt angegeben.

managedaccount [*Dienstname*] [type= {*true* | *false*}]

Konfiguriert, ob das für den Dienststart verwendete Konto ein verwaltetes Kennwort verwendet. In der Einstellung *true* wird das Kennwort beim Start des Diensts von Netlogon angefordert, bei *false* wird das konfigurierte Kennwort verwendet.

qc [*Dienstname*] *Puffergröße*

Fragt die Konfiguration eines Diensts ab.

qdescription [*Dienstname*]

Fragt die Beschreibung eines Diensts ab.

qfailure [*Dienstname*] *Puffergröße*

Fragt die Aktionen ab, die bei Fehlschlagen eines Diensts erfolgen.

qfailureflag [*Dienstname*]

Fragt das Fehleraktionskennzeichen eines Diensts ab.

qsidtype [*Dienstname*]

Fragt den SID-Typ eines Diensts ab.

qprivs [*Dienstname*]

Fragt die erforderlichen Rechte eines Diensts ab.

qprotection [*Dienstname*]

Fragt die Schutzebene eines Diensts ab.

qtriggerinfo [*Dienstname*]

Fragt die Auslöseparameter eines Diensts ab.

qrunlevel [*Dienstname*]

Fragt die niedrigste Ausführungsebene für einen Dienst ab.

qmanagedaccount [*Dienstname*]

Fragt ab, ob der Dienst ein Konto verwendet, das vom LSA verwaltet wird.

qpreferrednode [*Dienstname*]

Fragt den bevorzugten NUMA-Knoten für einen Win32-Dienst ab.

`qusersservice [`*`Dienstvorlagenname`*`]`

Fragt die von der angegebenen Benutzerdienstvorlage erstellte Benutzerdienstinstanz des aktuellen Benutzers in der aktuellen Sitzung ab.

`delete [`*`Dienstname`*`]`

Löscht einen Diensteintrag aus der Registrierung und damit auch aus der Liste der Dienste. Das ist insbesondere nützlich, wenn eine Deinstallationsroutine zur deinstallierten Anwendung gehörige Dienste schlicht »vergisst«. Läuft der Dienst oder wird er durch einen anderen Prozess offen gehalten, wird er zum Löschen markiert.

`create [`*`Dienstname`*`] [binpath= ] [type= {`*`own`*` | `*`share`*` | `*`interact`*` | `*`kernel`*` | `*`filesys`*` | `*`rec`*`}] [start= {`*`boot`*` | `*`system`*` | `*`auto`*` | `*`demand`*` | `*`disabled`*` | `*`delayed-auto`*`}] [error= {`*`normal`*` | `*`severe`*` | `*`critical`*` | `*`ignore`*`}] [group=` *`Dienstgruppe`* **`] [tag= {`*`yes`*` | `*`no`*`}] [depend=`** *`Abhängigkeit1/Abhängigkeit2/...`* **`] [obj= {`***`Kontoname`*** **`|`** ***`Objektname`*** **`}] [displayname=`** *`Anzeigename`* **`] [password=`** *`Kennwort`* **`]`**

Erstellt einen neuen Dienst durch Anlegen der zugehörigen Einträge in der Registrierung und der Dienstdatenbank. Neben dem Namen des Diensts ist der Pfad zur zugehörigen ausführbaren Datei in der Option `binpath` anzugeben. Mit `type` legen Sie die Art fest, auf die der Dienst mit dem Betriebssystem zusammenwirkt, `start` bestimmt durch Einstellen des Starttyps, ob und wann der Dienst gestartet wird, und `error` legt die Bedeutung eines Fehlers für das Betriebssystem fest.

`control [`*`Dienstname`*`]` *`Wert`*

Sendet als *`Wert`* einen dienstspezifischen Steuerungscode oder einen der Befehle `paramchange`, `netbindadd`, `netbindremove`, `netbindenable`, `netbinddisable` an einen Dienst.

`sdshow [`*`Dienstname`*`]` *`Rechteanzeigen`*

Zeigt die Sicherheitsbeschreibung eines Diensts im SDDL-Format an.

`sdset [`*`Dienstname`*`]` *`Sicherheitsbeschreibung`*

Setzt die Sicherheitsbeschreibung eines Diensts im SDDL-Format.

`showsid [`*`Dienstname`*`]`

Zeigt die SID an, die einem beliebigen Dienstnamen zugeordnet ist.

triggerinfo [*Dienstname*] *Option1* **[***Option2 ...***]**

Dient der Änderung der Auslöseparameter eines Diensts, die bestimmen, unter welchen Umständen ein Dienst gestartet oder beendet wird, und ermöglicht die Löschung dieser Parameter.

preferrednode [*Dienstname*] *Knotennummer*

Setzt den bevorzugten NUMA-Knoten eines Win32-Diensts. Es muss sich um einen eigenen Verarbeitungsdienst handeln, und die angegebene Knotennummer muss gültig sein. Zum Löschen eines zugewiesenen NUMA-Knotens verwenden Sie den Wert `-1`.

runlevel [*Dienstname*] *Nummer der Ausführungsebene*

Ändert die niedrigste Ausführungsebene für einen Dienst. Dieser muss eine der Startarten `Automatisch`, `Manuell` oder `Deaktiviert` nutzen oder ein Treiberdienst ohne PnP sein. Die Ausführungsebene eines Diensts kann nicht niedriger sein als die Ausführungsebene eines Diensts, von dem er abhängt. Wird `0` als Ausführungsebene angegeben, wird die eingestellte Ausführungsebene gelöscht.

getdisplayname [*Dienstname*] [*Puffergröße***]**

Ermittelt den Anzeigenamen eines Diensts.

getkeyname [*Anzeigename des Diensts*] [*Puffergröße***]**

Ermittelt aus dem gegebenen Anzeigenamen des Diensts den Dienstnamen. Schließen Sie Anzeigenamen mit Leerzeichen in Anführungszeichen ein.

enumdepend [*Dienstname*] [*Puffergröße***]**

Zeigt von diesem Dienst abhängige Dienste an. Die Puffergröße muss gegebenenfalls erhöht werden, um alle Informationen anzuzeigen.

boot {*ok* | *bad*}

Legt fest, ob die beim aktuellen Systemstart des lokalen Computers verwendete Konfiguration als letzte als funktionierend bekannte Konfiguration (»last known good configuration«) gespeichert werden soll.

lock

Sperrt die Dienstdatenbank des lokalen Systems.

querylock

Fragt den Sperrstatus der Dienstdatenbank ab.

PoSh: Für die allgemeine Diensteverwaltung auf lokalen und entfernten Rechnern existieren in Windows PowerShell folgende Cmdlets:

- `get-service` zum Abrufen der Informationen zu installierten Diensten.
- `start-service`, `restart-service` und `stop-service` zum Starten, Neustarten bzw. Beenden von Diensten.
- `suspend-service` zum Pausieren laufender Dienste und `resume-service` zum Fortsetzen pausierter Dienste.
- `new-service` zum Erstellen eines neuen Diensts.
- `set-service` zum Ändern der Dienstekonfiguration.
- Weitere Cmdlets beziehen sich auf spezielle, anwendungsbezogene Dienste.

Um einen Dienst zu löschen, können Sie in PowerShell 7.5 das Cmdlet `remove-service` nutzen oder in älteren Versionen die folgende Syntax verwenden:

```
(gwmi win32_service -filter "name='Dienstname'").delete()
```

Berechtigungen und Rechte

auditpol

auditpol */Befehl* **[*Optionen*]**

Verwaltet die Richtlinien zur Sicherheitsüberwachung. Im Folgenden werden die wichtigsten Einsatzmöglichkeiten dieses Befehls gezeigt. Für eine vollständige Übersicht sei auf die Windows-Hilfe verwiesen.

Befehle

/get *Optionen*

Zeigt die aktuelle Überwachungsrichtlinie an, zum Beispiel:

auditpol /get /category:*

Listet die aktuellen Überwachungseinstellungen für alle Kategorien und Unterkategorien auf.

/set *Optionen*

Legt die aktuelle Überwachungsrichtlinie fest, zum Beispiel:

auditpol /set /subcategory:*Unterkategorie* [/success: {*enable* | *disable*}] [/failure:{*enable* | *disable*}]

Konfiguriert die detaillierte Sicherheitsüberwachung für eine Unterkategorie. Erfolgreiche und fehlgeschlagene Objektzugriffsversuche können unabhängig voneinander aktiviert oder deaktiviert werden. Details zu weiteren Optionen entnehmen Sie bitte mit `auditpol /set /?` der Hilfe.

/list {/user | /category | /subcategory: "*Unterkategorie1*","*Unterkategorie2*",...} [/v]

Zeigt die auswählbaren Elemente der Überwachungsrichtlinie an. Die Verwendung der Option `/v` bewirkt die Anzeige der GUID der Unterkategorie, zum Beispiel:

auditpol /list /subcategory:* [/v]

Gibt die Namen aller verfügbaren (Unter)Kategorien aus.

/backup /file:*Pfad zur Sicherungsdatei*

Sichert Überwachungsrichtlinien in einer Datei.

/restore /file:*Pfad zur Sicherungsdatei*

Stellt Überwachungsrichtlinien aus einer zuvor mit /backup gesicherten Datei wieder her.

/clear [/y]

Löscht alle konfigurierten Sicherheitsüberwachungseinstellungen, bei Angabe von /y ohne Rückfrage.

/remove [/user[:*Benutzername***|{***SID***}]] [/allusers]**

Entfernt Benutzerüberwachungsrichtlinien für alle Benutzer oder für ein Benutzerkonto, das durch Benutzernamen oder SID angegeben wird.

---

dsacls

dsacls *Objekt-DN* **[***Optionen***] [/user:***Benutzer* **/passwd:***Passwort***]**

Zeigt bzw. verändert die ACL von Active-Directory-Objekten. Das Objekt wird mit seinem vollqualifizierten Namen (*Distinguished Name*) angegeben und kann optional einen Servernamen enthalten, der auf einen bestimmten Domänencontroller verweist. Ohne Angabe von Optionen wird die ACL des angegebenen Objekts ausgegeben. Die Verbindung zum Server kann optional mit einem anderen Benutzerkonto hergestellt werden.

Aufgrund des Befehlsumfangs können an dieser Stelle nur ausgewählte Optionen vorgestellt werden.

Optionen

/a

Gibt zusätzlich zu den Berechtigungen Besitzer und Überwachungseinstellungen aus.

/d | /g *Benutzer-oder-Gruppenliste*:***Berechtigungen***

Vergibt (/g) oder verweigert (/d) den angegebenen Benutzern bzw. Gruppen die angegebenen Berechtigungen. Die Namen werden in folgender Form angeführt: *Name@Domäne* oder *Domäne\Name*.

Berechtigungen werden in einer komplexen Syntax angegeben; weitere Informationen erhalten Sie in der Hilfe. In der einfachsten Form werden Berechtigungen mit zwei hintereinanderstehenden Buchstaben dargestellt: GR (Lesen), GE (Ausführen), GW (sSchreiben), GA (alles), SD (Löschen), DT (lLöschen des Objekts inklusive der untergeordneten Objekte), RC (lLesen der Berechtigungen), WD (Ändern der Berechtigungen), WO (Besitzer ändern), LC (untergeordnete Objekte auflisten), CC (Erstellen von untergeordneten Objekten), DC (Löschen von untergeordneten Objekten), WS (auf sich selbst schreiben), RP (Eigenschaften lesen), WP (Eigenschaften schreiben), CA (Zugriffsrecht) und LO (Zugriff anzeigen).

/r *Benutzer-oder-Gruppenliste*

Entfernt alle Einträge der angegebenen Benutzer oder Gruppen.

/n

Ersetzt die aktuelle ACL des Objekts, anstatt sie zu bearbeiten.

/p:{*y* | *n*}

Protection: Schützt (y) das Objekt vor Vererbung vom Elternobjekt bzw. aktiviert (n) die Vererbung. Ohne Angabe von /p werden keine Veränderungen an dieser Einstellung vorgenommen.

/i:{*t* | *s* | *p*}

Definiert die Vererbung der angegebenen Berechtigungen. Zur Auswahl stehen t (dieses Objekt und untergeordnete Objekte), p (nur Objekte eine Ebene darunter) und s (nur untergeordnete Objekte).

/s [/t]

Setzt die Sicherheitseinstellung des Objekts wieder auf den im Schema definierten Ursprungszustand der Objektklasse zurück. /t gibt an, dass sich der Befehl rekursiv auf alle untergeordneten Objekte im Baum auswirkt.

/resetDefaultDACL

Setzt die Berechtigungen (DACL) des Objekts wieder auf den im Schema

definierten Ursprungszustand der Objektklasse zurück.

`/resetDefaultSACL`

Setzt die Überwachungseinstellungen (SACL) des Objekts wieder auf den im Schema definierten Ursprungszustand der Objektklasse zurück.

`/takeOwnership`

Übernimmt den Besitz an dem Objekt.

---

icacls

Ersetzt die veralteten Befehle zur Verwaltung von Berechtigungen `cacls` und `xcacls`. `icacls` kann Dateisystemberechtigungen sichern und wiederherstellen sowie Integritätsstufen setzen. Die allgemeine Syntax sieht wie folgt aus:

`icacls` *`Pfad`* **`/grant[:r]`** *`SID1:Berechtigung1`* **`[`***`SID2:Berechtigung2`***`]`**

Gewährt die angegebenen Berechtigungen (zum Format siehe den Abschnitt *Optionen*). `:r` ersetzt alle nicht vererbten Berechtigungen für SID durch die angegebenen Berechtigungen. Anstelle der SID kann in den meisten Aufrufen auch der Benutzername eingesetzt werden.

`icacls` *`Pfad`* **`/deny`** *`SID1:Berechtigung1`* **`[`***`SID2:Berechtigung2`***`]`**

Verweigert die angegebenen Berechtigungen (zum Format siehe den Abschnitt *Optionen*).

`icacls` *`Pfad`* **`/remove[:g|:d]`** *`SID1`* **`[`***`SID2`***`]`**

Entfernt alle nicht vererbten gewährten (`:g`), verweigerten (`:d`) oder beide Typen von Berechtigungen für die angegebenen SIDs.

`icacls` *`Pfad`* **`/setintegritylevel`** *`Integritätsebene`*

Setzt die Integritätsebene (zum Format siehe den Abschnitt *Optionen*).

`icacls` *`Pfad`* ***`Befehl`*** *`Parameter`* **`[`***`Optionen`***`]`**

Wendet einen Befehl (mit Parametern) auf die im Pfad angegebenen Dateien oder Verzeichnisse an. Im Pfad können Platzhalter (`*` und `?`) genutzt werden.

`icacls` *`Pfad`* **`/save`** *`Datei`* **`[`***`Optionen`***`]`**

Sichert die ACLs der angegebenen Dateien und Verzeichnisse in ***`Datei`***.

icacls *Pfad* **/restore:***Datei* **[/substitute** *SIDalt SIDneu***]** **[***Optionen***]**

Wendet die in *Datei* mit `save` gesicherten ACLs auf die angegebenen Dateien und Verzeichnisse an. Dabei können mehrere Paare von SIDs angegeben werden, von denen jeweils *SIDalt* durch *SIDneu* ersetzt wird.

icacls *Pfad* **/setowner** *SID* **[***Optionen***]**

Setzt den Besitzer auf *SID*.

icacls *Pfad* **/findsid** *SID* **[***Optionen***]**

Gibt diejenigen Dateien und Verzeichnisse in *Pfad* aus, deren ACL einen Eintrag für *SID* enthält.

icacls *Pfad* **/verify [***Optionen***]**

Überprüft die ACLs und gibt Dateien/Verzeichnisse mit ungültigen Einträgen aus.

icacls *Pfad* **/reset [***Optionen***]**

Aktiviert die Vererbung von übergeordneten Objekten und entfernt Berechtigungen, die direkt auf die in *Pfad* angegebenen Objekte gesetzt sind.

Optionen

/c

Der Vorgang wird auch bei Fehlern fortgesetzt.

/l

Der Vorgang wird sowohl für einen symbolischen Link als auch für dessen Ziel durchgeführt.

/q

Unterdrückt Erfolgsmeldungen und beschränkt die Ausgabe auf Fehler.

/t

Der Vorgang wird rekursiv für alle unterhalb des angegebenen Pfads liegenden Dateien und Verzeichnisse ausgeführt, deren Namen auf das im Pfad angegebene Suchmuster passen.

[(*Vererbung1*)][(*Vererbung2*)](*Berechtigung1*,*Berechtigung2*,...)

Berechtigungen können als einfache oder spezifische Rechte angegeben

werden: `f` (voll), `m` (ändern), `rx` (lesen und ausführen), `r` (lesen), `w` (schreiben) sowie `d` (löschen), `rc` (Lesesteuerung), `wdac` (DAC schreiben), `wo` (Besitzer schreiben), `s` (synchronisieren), `as` (Systemsicherheitszugriff), `ma` (maximal zulässig), `gr` (allgemeiner Lesezugriff), `gw` (allgemeiner Schreibzugriff), `ge` (allgemeiner Ausführungszugriff), `ga` (allgemeiner Vollzugriff), `rd` (Daten lesen/Verzeichnis auflisten), `wd` (Daten schreiben/Datei hinzufügen), `ad` (Daten anfügen/Unterverzeichnis hinzufügen), `rea` (erweiterte Attribute lesen), `wea` (erweiterte Attribute schreiben), `x` (ausführen/durchsuchen), `dc` (untergeordnetes Element löschen), `ra` (Attribute lesen), `wa` (Attribute schreiben).

Vererbungsoptionen legen fest, wie die Berechtigungen eines Verzeichnisses an darin enthaltene Dateien und Unterverzeichnisse weitergegeben werden sollen: `oi` (an Objekte/Dateien), `ci` (an Container/Verzeichnisse), `io` (nur vererben, nicht auf das Verzeichnis selbst anwenden), `np` (nicht vererben, nur auf das Verzeichnis selbst anwenden).

`SID`

Gibt einen Benutzer oder eine Gruppe an, entweder durch den Namen oder durch die SID.

`[(`*`Vererbung1`*`) ] [(`*`Vererbung2`*`) ]` ***`Integritätsebene`***

Spezifiziert einen Integritäts-ACE, dessen Vererbung wie beim Setzen von Berechtigungen angegeben werden kann. Mögliche Integritätsebenen: niedrig (`l[ow]`), mittel (`m[edium]`), hoch (`h[igh]`).

Beispiele

`icacls d:\home\Benutzername /GRANT:`*`R Domäne\Benutzername: (CI) (OI)M Administratoren:(CI)(OI)F`* `/inheritance:R /T`

Konfiguriert für den angegebenen Domänenbenutzer *`Domäne\Benutzername`* den Zugriff auf sein Home-Verzeichnis mit Ändern-Berechtigung für Dateien und Unterverzeichnisse und räumt den Mitgliedern der Gruppe der *`Administratoren`* Vollzugriff ein. Der Vorgang erfolgt bei bereits vorhandenen Daten rekursiv, nicht durch

Vererbung übertragene bisherige Berechtigungen werden ersetzt.

`icacls` *`d:\home\Benutzername`* `/setowner` *`Domäne\Benutzername`*

Überträgt den Besitz des Ordners *`d:\home\Benutzername`* an das Benutzerkonto *`Domäne\Benutzername`*.

PoSh: Zur Abfrage und zum Setzen von Berechtigungen für den Zugriff auf Dateien und Registrierungsschlüssel stellt PowerShell die Cmdlets `get-acl` und `set-acl` zur Verfügung. Für die Verwendung von `set-acl` müssen Sie auf ein Muster zurückgreifen, das die entsprechenden Berechtigungen bereits aufweist. Beispielsweise überträgt der folgende Befehl die Berechtigungen von *datei.txt* auf alle Dateien in *d:\daten*:

```
get-acl d:\temp\datei.txt | set-acl d:\daten\*.*
```

---

setacl WWW

`setacl-on` *`Objektname`* **`-ot`** *`Objekttyp`* **`[-actn`** *`Aktion Parameter`*`]` **`[`*`Optionen`*`]`**

Leistungsfähiges Tool zum Setzen und Verwalten von Berechtigungen im Dateisystem, der Registrierung, auf Dienste, Freigaben und Drucker (*http://setacl.sourceforge.net*).

Eine oder mehrere Aktionen legen fest, was das Programm tut. Zu jeder Aktion gehören ein oder mehrere Parameter zum näheren Spezifizieren der Aktion. Zusätzlich können allgemeine Optionen angegeben werden.

Der Pfad bzw. Name des zu bearbeitenden Objekts wird als Objektname angegeben. Je nach Typ des Objekts kann das ein Dateisystempfad (*C:\Verzeichnis* oder *\\Server\Freigabe\Verzeichnis*), ein Registrierungsschlüssel (*[\\Server\]HKLM\Schlüssel*), ein Dienstname (*[\\Server\]Dienst*), ein Drucker (*[\\Server\]Drucker*) oder eine Freigabe (*\\Server\Freigabe*) sein. *`Objekttyp`* kann einen der folgenden Werte annehmen: `file` (Datei/Verzeichnis), `reg` (Registrierungsschlüssel), `srv` (Dienst), `prn` (Drucker), `shr` (Freigabe).

Folgende Aktionen werden unterstützt:

`ace`

Verarbeitet die mit dem Parameter `-ace` angegebenen ACEs (*Access Control Entries*). Dient dem Hinzufügen bzw. Entfernen von Berechtigungs- oder Überwachungseinträgen.

`trustee`

Kopiert, ersetzt oder löscht alle ACEs der angegebenen Trustees (Benutzer/Gruppe). Ein oder mehrere Trustees werden mit dem Parameter `-trst` angegeben.

`domain`

Ähnlich wie `trustee`, verarbeitet jedoch alle ACEs einer mit `-dom` angegebenen Domäne.

`list`

Anzeige gesetzter Berechtigungen. Das Verhalten wird durch den Parameter `-lst` gesteuert. Mit `-bckp` kann eine Backup-Datei angegeben werden, die das Listing aufnimmt. Sofern es im SDDL-Format erzeugt wurde, können die Berechtigungen mit der Aktion `restore` wiederhergestellt werden.

`restore`

Stellt vollständige SDs (*Security Descriptors*) aus einer mit `-bckp` angegebenen Datei wieder her, die im SDDL-Format vorliegen muss.

`setowner`

Setzt den Eigentümer des Objekts. Das kann ein beliebiger Trustee (Benutzer/Gruppe) sein, nicht nur Administratoren.

`clear`

Löscht alle nicht vererbten (direkt auf das Objekt gesetzten) ACEs. Wird durch den Parameter `-clr` gesteuert.

`setprot`

Setzt die Protection-Eigenschaft des Objekts, die festlegt, ob vererbbare Berechtigungen übergeordneter Objekte auf dieses Objekt angewendet werden. Steuerung durch den Parameter `-op`.

`rstchldrn`

Setzt die Berechtigungen aller Kindobjekte zurück und deaktiviert deren Protection-Eigenschaft, um die Vererbung im ganzen Baum zu aktivieren.

Steuerung durch den Parameter `-rst`.

Optionen und Parameter

`-ace "n:`*`Trustee`*`;p:`*`Berechtigung`*`;s:`*`IsSID`*`;i:`*`Vererbung`*`;m:`*`Modus`*`;w:`*`Wo`*`"`

Setzt *`Berechtigung`* für *`Trustee`*. *`IsSID`* (`y | n`) gibt an, ob der Trustee durch die SID statt durch den Namen spezifiziert wurde. Optional kann die Art der *`Vererbung`* als Kombination (durch Komma getrennt) der folgenden Werte angegeben werden: `so` (Unterobjekte), `sc` (Untercontainer), `np` (keine Vererbung), `io` (nur vererben, keine Auswirkung auf das angegebene Objekt). *`Modus`* gibt an, ob die Berechtigung gesetzt (`set`), verweigert (`deny`) oder entzogen (`revoke`) werden soll. Handelt es sich um einen Eintrag in der SACL (Überwachungs-ACL), erzeugt `aud_succ` einen Audit-Success-ACE, `aud_fail` dagegen einen Audit-Failure-ACE. *`Wo`* gibt schließlich an, ob die DACL (`dacl`), die SACL (`sacl`) oder beide (`dacl,sacl`) bearbeitet werden sollen.

`-trst "n1:`*`Trustee`*`;n2:`*`Trustee`*`;s1:`*`IsSID`*`;s2:`*`IsSID`*`;ta:`*`TrusteeAction`*`;w:`*`Wo`*`"`

Je nach *`TrusteeAction`* werden alle ACEs eines bei `n1` angegebenen Trustee gelöscht (`remtrst`), durch den bei `n2` angegebenen Trustee ersetzt (`repltrst`) oder kopiert (`cpytrst`).

`-dom "n1:`*`Domain`*`;n2:`*`Domain`*`;da:`*`DomainAction`*`;w:`*`Wo`*`"`

Entspricht `-trst`, bearbeitet jedoch die ACEs aller Trustees einer Domäne. *`DomainAction`* kann die Werte `remdom` (löschen), `repldom` (ersetzen) und `cpydom` (kopieren) annehmen. Beim Ersetzen und Kopieren wird die im ACE enthaltene SID zunächst zum Namen des Trustee aufgelöst. Dieser Name wird dann in der bei `n2` angegebenen Domäne gesucht, und der gefundene Trustee dieses Namens wird in den ACE eingetragen.

`-rec `*`Rekursion`*

Gibt an, dass das Dateisystem bzw. die Registrierung rekursiv durchlaufen werden soll. Durch die Vererbungsmechanismen wird das selten benötigt, meist für die List-Funktion. Bei der *`Rekursion`* können

Container (`cont`), Objekte (`obj`) oder beides (`cont_obj`) verarbeitet werden.

`-op "dacl:`*`Protection`*`;sacl:`*`Protection`*`"`

Setzt die Protection-Eigenschaft des Objekts getrennt nach DACL und SACL auf einen der folgenden Werte: `nc` (keine Änderung), `np` (nicht geschützt, Vererbung aktiviert), `p_c` (geschützt, vererbbare ACEs des Elternobjekts werden kopiert), `p_nc` (geschützt, Objekt erhält eine leere ACL).

`-lst "f:`*`Format`*`;w:`*`Was`*`;i:`*`ListInherited`*`;s:`*`DisplaySID`*`"`

Legt das Verhalten des List-Befehls fest. Das Listing kann in einem der folgenden Formate erzeugt werden: `csv` (kommasepariert), `tab` (gut lesbar), `SDDL` (standardisiert, enthält den vollständigen SD). *`Was`* ist eine durch Kommata getrennte Liste der Elemente des SD, die angezeigt werden sollen: `d` (DACL), `s` (SACL), `o` (Besitzer), `g` (primäre Gruppe). *`ListInherited`* (`y | n`) gibt an, ob auch vererbte ACEs angezeigt werden sollen. *`DisplaySID`* legt fest, ob Trustees als Namen (`n`), SIDs (`y`) oder beides (`b`) angezeigt werden.

Weitere Optionen sind in der Kommandozeilenhilfe des Programms beschrieben.

---

takeown

`takeown [/s` *`Computer`* **`[/u`** *`Benutzer`* **`[/p [`***`Kennwort`***`]]]] /f`** *`DateiMuster`* **`[/a] [/r [/d {`***`j`*** **`|`** ***`n`***`}] [/SKIPSL]]`**

Übernimmt den Besitz von Dateien auf *`Computer`*. Falls angegeben, wird der Befehl unter dem angegebenen Benutzerkonto ausgeführt. *`DateiMuster`* kann ein Datei- oder Verzeichnisname inklusive eines Sterns als Wildcard sein. `/a` weist den Besitz der Gruppe der Administratoren statt dem angemeldeten Benutzer zu. `/r` wendet den Befehl rekursiv auch auf Inhalte aller untergeordneten Verzeichnisse an, wobei **`/SKIPSL`** anweist, symbolischen Verknüpfungen nicht zu folgen. Falls der Benutzer nicht die Berechtigung »Ordner auflisten« zum Anzeigen eines Verzeichnisses hat,

kann mit `/d j` immer der Besitz übernommen und mit `/d n` die Ausführung abgebrochen werden.

---

tpmvscmgr

Das Programm hilft bei der Verwaltung virtueller Smartcards.

tpmvscmgr [/quiet] create *Name* **/adminkey** ***prompt*** | ***default*** | ***random*** **[/puk** ***prompt*** | ***default*****] /pin** ***prompt*****|** ***default*** **[/generate] [/machine** *Computername***] [/pinpolicy [*** ***Richtlinienoptionen]*** **[/attestation** ***AIK_AND_CERT*** **|** ***AIK_ONLY]***

Erzeugt eine neue virtuelle TPM-Smartcard. Dabei wird durch *prompt* zur Parametereingabe aufgefordert, *default* verwendet den jeweiligen Standardwert des Parameters, und *random* erzeugt einen Zufallswert. Der Parameter `/generate` bewirkt eine Formatierung der virtuellen Smartcard, um sie für das Registrieren von Zertifikaten vorzubereiten. Voraussetzung für die Erzeugung ist die Aktivierung von TPM im BIOS des Computers. Eine Richtlinie für die PIN mittels `/generate` kann nur im Zusammenspiel mit dem Parameter `/pin` zugewiesen werden.

tpmvscmgr destroy [/quiet] /instance *Geräteinstanz-ID* **[/machine** *Computername***]**

Löscht eine virtuelle Smartcard unter Verwendung der *Geräteinstanz-ID*, die bei der Erzeugung dieser Karte zurückgegeben wurde.

Systemdiagnose und -information

---

diskperf

diskperf [-y[d|v] | [-n[d|v]] [*Computername*]

Startet oder deaktiviert die Leistungsindikatoren für physische und logische Laufwerke auf dem lokalen oder dem angegebenen Computer.

Optionen

-y[d|v]

Durch -y werden alle datenträgerbezogenen Leistungsindikatoren beim Rechnerneustart geladen, -yd aktiviert die Leistungsindikatoren physischer Laufwerke und -yv diejenigen logischer Laufwerke und Volumen.

-n[d|v]

Deaktiviert alle datenträgerbezogenen Leistungsindikatoren, -nd deaktiviert sie für die physischen Laufwerke, -nv für logische.

---

dispdiag

dispdiag [-testacpi] [-d] [-delay *Sekunden*] [-brightnesslogging] [-out *Pfad**Dateiname*]

Schreibt Informationen zum Bildschirm in eine Datei im aktuellen Verzeichnis oder im angegebenen Pfad.

Optionen

-testacpi

Führt eine Diagnose der Hotkeys durch.

-d

Erzeugt eine Datei mit zusätzlichen Daten.

-delay *n*

Verzögert die Datensammlung um *n* Sekunden.

-brightnesslogging

Fügt Informationen zur Helligkeit hinzu.

-out *Pfad\Dateiname*

Erzeugt die Ausgabedatei im angegebenen Pfad.

---

driverquery

driverquery [/s *System* **[/u** *Benutzer* **[/p [***Kennwort***]]]] [/fo** *Format***] [/nh] [/si] [/v]**

Zeigt Informationen über installierte Gerätetreiber lokal oder auf einem *System* an, zu dem mit dem angegebenen *Benutzer*/*Kennwort* eine Verbindung hergestellt wird.

Optionen

/fo {*table* | *list* | *csv*}

Legt das Ausgabeformat fest.

/nh

Lässt die Spaltenüberschriften weg.

/si

Zeigt Informationen über die Signatur von Treibern an.

/v

Detaillierte Ausgabe, nicht gültig in Verbindung mit `/si`.

PoSh: Zum Anzeigen von Informationen zu allen installierten Treibern im laufenden Betriebssystem können Sie folgenden Befehl in der Windows PowerShell verwenden:

```
get-windowsdriver -online
```

---

dtrace

dtrace [-BCeFhlqSvVwYZ] [-b *bufsz***] [-c** *cmd***] [-D** *name***[=def]] [-I** *path***] [-L** *path***] [-o** *output***] [-p** *pid***] [-s** *script***] [-U** *name***] [-x opt[=val]] [-X a|c|s|t] [-y symbol path]**

[-P provider [[predicate] action]] [-m [provider:]

module [[predicate] action]] [-f [[provider:]
module:]

func [[predicate] action]] [-n [[[provider:] module:] func:]

name [[predicate] action]] [-i probe-id [[predicate] action]] [args ...]

Dieses in Windows Server 2025 neu integrierte Befehlszeilentool ermöglicht die Überwachung und das Debugging der Systemleistung in Echtzeit.

Vor der Nutzung muss das Tool zunächst aktiviert werden, was der folgende Befehl und ein anschließender Neustart des Servers erwirkt:

bcdedit /set dtrace *on*

Die Parameter des Befehls weisen die Besonderheit auf, dass Groß-/Kleinschreibung beachtet werden muss.

Aufgrund der sehr speziellen Einsatzszenarien dieses Befehls und der Vielzahl und Komplexität der möglichen Parameter wird für weiterführende Informationen auf die Onlinedokumentation von Microsoft verwiesen.

---

eventcreate

eventcreate [/s *Computer* **[/u** *Benutzer* **[/p** *Kennwort***]]] /id** *EreignisID* **/l** *Protokollname [***/so** *Quelle]* **/t** *Typ* **/d** *Text*

Erlaubt Administratoren, einen selbst definierten Eintrag mit ID und

Meldungstext im Ereignisprotokoll eines Computers zu erzeugen.

Optionen

`/id` *`Nummer`*

Legt eine ID im Bereich zwischen 1 und 1000 fest.

`/so` *`Quelle`*

Gibt die Quelle in Form einer Zeichenkette an. Sollte dieser Parameter nicht spezifiziert werden, wird EventCreate als Quelle eingesetzt.

`/l` *`Eventlog`*

Spezifiziert, in welchem Protokoll das Ereignis erstellt wird: *application* oder *system*.

`/t {`*`success`* **`|`** ***`error`*** **`|`** ***`warning`*** **`|`** ***`information`*****`}`**

Legt die Ebene des Ereigniseintrags fest: `success` und `information` erzeugen einen Eintrag in der Ebene *Informationen*, `warning` generiert einen Eintrag in der Ebene *Warnung*, `error` in der Ebene *Fehler*.

`/d` *`Text`*

Legt den Meldungstext fest, der beim Eintrag hinterlegt wird.

PoSh: Mit dem Cmdlet `write-eventlog` können Sie Ereignisse in die Windows-Ereignisanzeige aufnehmen.

```
write-eventlog -logname "Application" -source "Mein
Skript" -eventid 999 -message "Ein Test"
```

Dieser Befehl schreibt im Anwendungslog des Rechners ein Ereignis mit der ID 999 und dem Text "*Ein Test*". Existiert die Quelle (Parameter `-source`) noch nicht, muss sie vorher einmalig angelegt werden:

```
new-eventlog -source "Mein Skript" -logname "Application"
```

---

getmac

`getmac [/s` *`Computer`* **`[/u`** *`Benutzer`* **`[/p [`***`Passwort`***`]]]]`**
`[`*`Optionen`***`]`**

Zeigt die MAC-Adresse der Netzwerkkarten des angegebenen Rechners an.

Optionen

`/fo {`*`table`*` | `*`list`*` | `*`csv`*`}`

Legt das Ausgabeformat fest.

`/nh`

Lässt die Spaltenüberschriften in den Formaten *`table`* und *`csv`* weg.

`/v`

Liefert ausführlichere Informationen, z.B. den Namen der einzelnen Karten.

PoSh: Die MAC-Adresse und die IP-Adresse(n) für Netzwerkadapter mit aktiviertem TCP-Protokoll lassen sich in PowerShell mit folgendem Befehl ermitteln:

```
gcim win32_networkadapterconfiguration -filter "IpEnabled
= TRUE" | select description, macaddress, ipaddress
```

---

ktmutil

Verwaltet Kernel-Transaktionen (typischerweise im Bereich der Registry oder des Dateisystems).

`ktmutil tm {`*`list`*` | `*`info`*`} [`*`GUID`*`]`
`ktmutil tx {`*`list`*` | `*`info`*` | `*`force`*` | `*`forget`*`} [`*`GUID`*`]`

Dient der Kontrolle der Transaktionsmanager (`tm`) oder laufender Transaktionen (`tx`).

Weitere Optionen zur Beeinflussung von Transaktionen entnehmen Sie bei Bedarf bitte der Hilfe des Befehls.

---

logman

`logman [`*`create`*` | `*`query`*` | `*`start`*` | `*`stop`*` | `*`delete`*` | `*`update`*` | `*`import`*` | `*`export`*`] [`*`Optionen`*`]`

Verwaltet Datensammler.

logman create *Typ* **-n** *Name* **[*Optionen*]**

Erstellt einen neuen Datensammler des angegebenen Namens vom Typ `counter` (Indikatorendatensammler), `trace` (Ablaufverfolgungsdatensammler), `alert` (Warnungsdatensammler) oder `cfg` (Konfigurationsdatensammler).

logman query [*providers* | -n *Name***] [*Optionen*]**

Fragt Eigenschaften von Datensammlern ab. Falls `providers` angegeben werden, erfolgt die Ausgabe der registrierten Anbieter. Falls *Name* nicht angegeben ist, werden alle Datensammler aufgeführt.

logman start -n *Name* **[-as] [*Optionen*]**

Startet einen bestehenden Datensammler und setzt die Startzeit auf »manuell«. `as` legt fest, dass der Vorgang asynchron ausgeführt wird.

logman stop -n *Name* **[-as] [*Optionen*]**

Beendet einen bestehenden Datensammler und setzt die Endzeit auf »manuell«. `as` legt fest, dass der Vorgang asynchron ausgeführt wird.

logman delete -n *Name* **[*Optionen*]**

Löscht einen bestehenden Datensammler.

logman update *[counter | trace | alert | cfg | providers]* **-n** *Name* **[*Optionen*]**

Ändert die Eigenschaften des vorhandenen Datensammlers *Name* vom Typ `counter` (Leistungsindikator), `trace` (Ablaufverfolgung), `alert` (Warnung) oder `cfg` (Konfiguration) oder zeigt über `providers` registrierte Anbieter an.

logman import -n *Name* **-xml** *Dateiname* **[*Optionen*]**

Importiert einen Datensammlersatz unter dem angegebenen Namen aus der angegebenen Datei.

logman export -n *Name* **-xml** *Dateiname* **[*Optionen*]**

Exportiert den angegebenen Datensammlersatz in die angegebene XML-Datei.

Optionen für alle Befehlsvarianten

-s *Computer*

Statt des lokalen Systems wird der angegebene Computer angesprochen.

-config *Datei*

Kommandozeilenparameter werden der angegebenen Datei entnommen.

-ets

Sendet die Befehle direkt an die Ablaufverfolgungssitzung, ohne die Sammlung zu speichern oder zu planen.

Optionen zu logman create und logman update

-b *tt.MM.jjjj hh:mm:ss*

Startet den Datensammler zur angegebenen Zeit.

-e *tt.MM.jjjj hh:mm:ss*

Beendet den Datensammler zur angegebenen Zeit.

-rf [[*hh:***]***mm:***]***ss*

Dauer der Ausführung des Datensammlers. Kann anstelle von `-e` verwendet werden.

-m {[*start*] | [*stop*]}

Der Start bzw. Stopp des Datensammlers erfolgt manuell, nicht zeitgesteuert. Die Parameter `-b` und `-e` dürfen nicht gleichzeitig angegeben werden.

-[-]r

Wiederholt den Datensammler täglich zur gleichen Anfangsund Endzeit. Ein doppeltes Minuszeichen deaktiviert die Wiederholung.

-o *Verzeichnis* **|** *DSN!Log-Name*

Gesammelte Daten werden in das angegebene Verzeichnis (Dateiname: *Name.blg*) oder die angegebene SQL-Datenbank geschrieben.

-[-]a

Hängt die Ausgabe an die Protokolldatei an, anstatt sie zu überschreiben. Ein zweites Minuszeichen deaktiviert diese Option.

-[-]ow

Überschreibt eine existierende Protokolldatei. Ein zweites Minuszeichen deaktiviert diese Option.

-[-]v {*nnnnnn* **|** *mmddhhmm***}**

Hängt die angegebene Versionsnummer oder das angegebene Datum an

den Dateinamen der Ausgabedatei an. Ein zweites Minuszeichen deaktiviert diese Option.

-[-]rc *Befehl*

Führt den angegebenen Befehl nach Beendigung der Protokollierung aus.

-max *Wert*

Maximalgröße der Ausgabedatei in MByte oder maximale Anzahl an Datensätzen in der SQL-Datenbank. `--max` hebt die Beschränkung auf.

-cnf [[*hh:***]***mm:***]***ss*

Erstellt eine neue Ausgabedatei, wenn die Maximalgröße erreicht wurde oder die angegebene Zeitspanne verstrichen ist. Muss zusammen mit `-v` verwendet werden. `--cnf` deaktiviert diese Option.

-y

Beantwortet alle Fragen mit *Ja*.

Optionen zu logman update

-si [[*hh:***]***mm:***]***ss*

Gibt das Datensammlungsintervall an.

-f {*bin* | *bincirc* | *csv* | *tsv* | *sql*}

Protokollformat des Datensammlers: binär, zirkulär binär, komma- oder tabulatorsepariert, SQL-Datenbank.

-sc *Wert*

Maximale Anzahl an Datensätzen, die mit einem Leistungsindikatorendatensammler gesammelt werden sollen.

-bs *Wert*

Puffergröße in KByte für eine Ereignisablaufverfolgungssitzung.

-nb *min max*

Anzahl der Puffer für Ablaufverfolgungssitzungen.

-ct {*perf* | *system* | *cycle*}

Für das Protokollieren verwendete Zeitauflösung: Abfrageleistungsindikator, Systemzeit oder CPU-Zyklus.

-ln *Logger-Name*

Name des Protokollerstellers.

-ft [[*hh:***]***mm:***]***ss***

Leerungszeitgeber für Ablaufverfolgungssitzungen.

-[-]p *Provider* [*Schlüsselwörter*** [*Ebene*]]**

Ein einzelner zu aktivierender Ablaufverfolgungsanbieter.

-pf *Datei*

Datei mit einer Liste der zu aktivierenden Ablaufverfolgungsanbieter.

-c *Leistungsindikator1 Leistungsindikator2* **[...] | -cf** *Datei*

Spezifiziert die Leistungsindikatoren, deren Daten gesammelt werden. Diese können alternativ aus einer Datei gelesen werden, die einen Eintrag pro Zeile enthält.

-[-]rt

Verwendet den Echtzeitmodus für die Ereignisablaufverfolgung.

-[-]ul

Führt die Ablaufverfolgung im Benutzermodus aus.

-mode *Modus*

Protokollierungsmodus bei der Ablaufverfolgung.

-[-]el

Aktiviert oder deaktiviert die Ereignisprotokollierung.

-[-]ni

Aktiviert oder deaktiviert die Abfrage der Netzwerkschnittstelle.

-th *Grenzwert1 Grenzwert2* **[...]**

Leistungsindikatorenschwellenwerte, ab denen eine Warnung erzeugt wird.

-[-]rdcs *Name*

Sammlung, die gestartet wird, wenn die Warnung ausgelöst wird.

-[-]tn *Name*

Task, der ausgeführt wird, wenn die Warnung ausgelöst wird.

-[-]targ *Name*

Argumente für den mit `-tn` festgelegten geplanten Task.

-reg *Pfad1 Pfad2* **[...]**

Zu sammelnde Registrierungswerte.

-mgt *Abfrage1 Abfrage2* **[...]**

Zu sammelnde WMI-Objekte, angegeben als WMI-Abfragen.

-ftc *Pfad1 Pfad2* **[***...***]**

Zu sammelnde Dateien.

PoSh: Unter anderem mit dem Cmdlet `get-counter` unterstützt PowerShell das Sammeln der Daten und die Auswertung von Leistungsindikatoren.

---

relog

relog *Datei1 Datei2* **[*Optionen*]**

Konvertiert Daten aus Leistungsindikatoren-Protokolldateien in andere Formate.

Optionen

-a

Hängt an die Ausgabedatei an, anstatt sie zu überschreiben.

-c *Leistungsindikator1 Leistungsindikator2* **| -cf** *Datei*

Spezifiziert die Leistungsindikatoren, deren Daten konvertiert werden. Diese können alternativ aus einer Datei gelesen werden, die einen Eintrag pro Zeile enthält.

-f {*bin* | *csv* | *tsv* | *sql*}

Ausgabeformat: binär, komma- oder tabulatorsepariert, SQL-Datenbank.

-t *n*

Jeder *n*-te Messpunkt wird konvertiert (Standard: `jeder`).

-o *Ausgabedatei* **|** *DSN!Log-Name*

Konvertierte Daten werden in die angegebene Datei oder die angegebene SQL-Datenbank geschrieben.

-b *dd.MM.jjjj hh:mm:ss*

Daten werden ab diesem Sammelzeitpunkt konvertiert.

-e *tt.mm.yyyy hh:mm:ss*

Daten werden bis zu diesem Sammelzeitpunkt konvertiert.

-config *Datei*

Befehlszeilenparameter werden der angegebenen Datei entnommen.

-q

Listet die Leistungsindikatoren und Zeitintervalle der Sammlung aus den Protokolldateien auf.

-y

Beantwortet alle Fragen mit *Ja*.

---

systeminfo

systeminfo [/s *System* **[/u** *Benutzer* **[/p [***Kennwort***]]]] [/fo {*table* | *list* | *csv*}] [/nh]**

Ermittelt detaillierte Informationen zur Systemkonfiguration (inklusive installierter Patches) für ein entferntes System oder den lokalen Computer und schreibt diese in die Standardausgabe. Zum Zugriff auf *System* können ein *Benutzerkonto* und ein *Kennwort* angegeben werden. Wird Letzteres weggelassen, werden diese Informationen abgefragt.

Optional kann das Ausgabeformat auf tabellarisch (`table`), Liste (`list`, Standard) oder `csv` eingestellt werden. Bei Angabe von `/nh` werden die Spaltenüberschriften im tabellarischen und im CSV-Format weggelassen.

PoSh: Die Ermittlung von Systeminformationen mit Windows PowerShell ist eine der Stärken im Vergleich zum `systeminfo`-Befehl der Eingabeaufforderung, weil Abfragen einzelner Systemkomponenten und eigenschaften sehr granular erfolgen können.

Folgendes Beispiel gibt einen Einblick in die Inventarisierung von Hardware mit einem PowerShell-Skript, das mit leichten Abwandlungen bei jedem Herunterfahren ausgeführt werden könnte, um jederzeit Zugriff auf aktuelle Inventarinformationen zu haben:

```
$computer = gcim win32_computersystem

$bios = gcim win32_bios
```

```
$os = (gcim win32_operatingsystem).caption

$serial = $bios.serialnumber

$colCPUs = gcim win32_processor

$n = 0

foreach ($cpu in $colCPUs){

$n = $n + 1

$processor = (($cpu.name -replace ' {2,}',' ')

-replace "\(TM\)") -replace '\(R\)')

}

$cpuspeed = "{0:0.00 GHz}" -f($cpu.maxclockspeed/1000)

$physmem = ([int] ($computer.TotalPhysicalMemory /1MB))

$disk = gcim win32_diskdrive -filter "InterfaceType <> 'USB'"

$disksize=""

foreach ($objitem in $disk) {$disksize=$disksize+";"+[Convert]::

ToString(([int]($objitem.size/1000000000)))}

$disksize = $disksize.substring(1)

$lan = gcim win32_networkadapterconfiguration -filter

"ipenabled=true"

$lancount = gcim win32_networkadapterconfiguration -filter

"ipenabled=true"

$ip=""
```

```
foreach($lan in $lancount){$ip = $ip + ";" +

[string]($lan.ipaddress[0])}

$ip = $ip.substring(1)

write-host $computer.name $computer.model $serial $processor

$cpuspeed $physmem $disksize $os $ip -separator ";"
```

Das Skript fragt folgende Computerinformationen ab und gibt sie als Zeile aus: Rechnername, Rechnermodell laut BIOS, Seriennummer laut BIOS, Prozessor, Prozessorgeschwindigkeit, RAM, Datenträgergröße, Betriebssystem und primäre IP-Adressen der Netzwerkadapter.

Falls Sie diese Informationen in eine Datei umlenken wollen, können Sie folgende Zeilen hinzufügen:

```
$strCompName=($env:Computername)

$logfile="\\Server\Freigabe\Rechner\$strCompName.csv"

$allinfo=[string]$computer.name+"`t"+
[string]$computer.model+"`t"+[string]$serial+"`t"+
[string]$processor+"`t"+[string]$cpuspeed+"`t"+
[string]$physmem+"`t"+[string]$disksize+"`t"+[string]$os+"`t"+
[string]$ip

write-output $allinfo | format-table | out-file $logfile
```

In diesem Beispiel wird die Variable `$allinfo` erzeugt und in eine Datei mit dem Namen des Computers im Unterverzeichnis *Rechner* in der Freigabe übertragen. Dabei werden die Werte durch Tabulatoren getrennt und die ursprünglichen Objekte als Zeichenkette integriert. Mit einem weiteren Skript lassen sich mehrere dieser Dateien in einer einzelnen für weitere Auswertungen zusammenfassen, um eine weiter verarbeitbare Bestandstabelle zu erhalten.

typeperf

typeperf {*Leistungsindikator1 Leistungsindikator2* **[...] | -cf** *Dateiname* | **-q [***object***] | -qx** *object***} [Optionen]**

Zeigt den aktuellen Wert der angegebenen Leistungsindikatoren in der Befehlszeile an oder schreibt sie in eine Protokolldatei und aktualisiert diese bis zum Abbruch mit *Strg+C*. Die Leistungsindikatoren werden in diesem Format angegeben: *[\\Server]\Objekt\Leistungsindikator*.

typeperf -qx [*Objekt***]**

Zeigt die installierten Leistungsindikatoren an, gegebenenfalls beschränkt auf diejenigen des angegebenen Objekts.

Informationen zu den weiteren Optionen dieses Befehls erhalten Sie mit `typeperf /?`.

tzutil

tzutil `/g | /s` ***Zeitzonen-ID*****[***_dstoff***]** `| /l`

Hilfsprogramm zur Anzeige und Anpassung der Zeitzone in Windows. Dabei zeigt `/g` die aktuelle Zeitzone an, `/s` setzt die aktuelle Zeitzone auf den angegebenen Wert. Gültige Zeitzonen-IDs zeigen Sie mit dem Parameter `/l` an. Das an die Zeitzone angehängte Suffix `_dstoff` bewirkt die Deaktivierung der automatischen Sommerzeitanpassungen.

PoSh: Die Cmdlets `get-timezone` und `set-timezone` dienen der Abfrage und Einstellung der Zeitzone mittels PowerShell. Um herauszufinden, ob gerade die Sommerzeit aktiv ist, können Sie das mit `get-date` erhaltenen Objekt und eine Methode von .NET zu Hilfe nehmen:

```
(get-date).IsDaylightSavingTime()
```

verifier

`verifier ist ein Werkzeug zur Treiberüberprüfung.`

`verifier /standard /all`

Aktiviert die Standardoptionen für die Treiberüberprüfung und wendet sie auf alle Treiber im System an.

`verifier /driver` *`Name`* **`[`***`Name ...`***`]`**

Gibt den Namen einer oder mehrerer Treiberdateien in der *Form Name.Erweiterung* an.

`verifier /log` *`Name der Protokolldatei`* **`[/interval`** *`Sekunden`***`]`**

Erstellt eine Logdatei und schreibt im angegebenen Abstand (Standardwert 30 Sekunden) Einträge. Zum Schließen der Logdatei und zur Rückkehr in die Eingabeaufforderung muss die Tastenkombination *Strg+C* verwendet werden.

`verifier /query`

Zeigt eine Zusammenfassung der aktuellen Aktivitäten der Treiberüberprüfung an.

`verifier /reset`

Löscht alle Einstellungen für die Treiberüberprüfung nach dem Neustart des Systems.

`verifier /volatile`

Ändert die Einstellungen der Treiberüberprüfung dynamisch ohne Neustart des Computers.

Weitere Parameter entnehmen Sie bitte der Hilfe des Programms.

---

w32tm

`w32tm [Befehl] [`*`Optionen`*`]`

Dient der Konfiguration, Diagnose und Überwachung zahlreicher Aspekte des Windows-Zeitdiensts. Im Folgenden werden lediglich die wichtigsten Möglichkeiten von `w32tm` beschrieben, da auch dieser Befehl sehr komplex ist.

`w32tm [/register | unregister]`

Registriert `w32tm` als Dienst und fügt der Registrierung die Standardkonfiguration hinzu (`register`) bzw. hebt die Registrierung als Dienst auf und entfernt alle relevanten Einträge aus der Registrierung (`unregister`).

`w32tm /monitor [/domain:`*`DomänenName`*`]`
`[/computers:`*`ComputerName1,ComputerName2,...`*`]`
`[/threads:`*`n`*`] [/ipprotocol:{`*`4`*`|`*`6`*`}] [/nowarn]`

Überwacht die lokale oder angegebene Domäne bzw. die angegebenen Computer. Die Anzahl der Threads (`/threads`) bestimmt die Zahl der gleichzeitig zu analysierenden Computer und kann zwischen `1` und `50` liegen (Standard: 3). Mit `/ipprotocol` grenzen Sie ein, welche IP-Version für die Überwachung verwendet wird (Standard: alle verfügbaren). Durch `/nowarn` unterbinden Sie Warnmeldungen des Befehls.

`w32tm /ntte` *`Systemzeit`*

Wandelt eine Windows-Systemzeit (Format: 100-ns-Intervalle seit dem 01.01.1601) in ein lesbares Format um.

`w32tm /ntpte` *`Systemzeit`*

Wandelt eine Windows-Systemzeit (Format: 2^-32s-Intervalle seit dem 01.01.1900) in ein lesbares Format um.

`w32tm /resync [/computer:`*`ComputerName`*`] [/nowait] [/rediscover]`
`[soft]`

Veranlasst eine baldige Zeitsynchronisation des lokalen oder angegebenen Computers. `/nowait` wartet nicht auf die Beendigung des Vorgangs. Die Fehlerstatistiken werden verworfen (das kann durch den zu Kompatibilitätszwecken vorhandenen Parameter `/soft` verhindert werden). `/rediscover` führt vor der Synchronisation eine erneute Ermittlung der Netzwerkkonfiguration durch und sucht nach Zeitquellen im Netzwerk.

`w32tm /stripchart /computer:`*`ComputerName`*
`[/period:`*`Intervall`***`] [/dataonly]`**
`[/samples:`*`AnzahlMesspunkte`***`]`**
`[/packetinfo] [/ipprotocol:{4|6}]`

Vergleicht die Zeitdifferenz zwischen dem lokalen und dem angegebenen

Computer alle *Intervall* Sekunden, bis mit *Strg+C* abgebrochen wird oder *AnzahlMesspunkte* gesammelt wurden. /dataonly unterdrückt die Ausgabe einer »grafischen« Darstellung der Zeitdifferenz. /packetinfo gibt die NTP-Paketantwortmeldung aus, /ipprotocol legt das zu verwendende IP-Protokoll fest.

w32tm /config [/computer:*ComputerName***] [/update]**
[/manualpeerlist: *ComputerName1 ComputerName2***]**
[/syncfromflags:*Quelle***] [/localclockdispersion:***Sekunden***]**
[/reliable:{*yes*|*no*}] [/largephaseoffset:*Millisekunden***]**

Ändert die Konfiguration des Zeitdiensts des lokalen oder angegebenen Computers. /update benachrichtigt den Zeitdienst über die geänderte Konfiguration, sodass diese wirksam wird. /manualpeerlist erlaubt die Angabe mehrerer Computer über DNS-Namen oder IP-Adressen, mit denen synchronisiert werden soll. Die bei /syncfromflags angegebene Quelle legt fest, nach welchem System synchronisiert wird: manual zur Verwendung der Peer-Liste, domhier zur Synchronisation gemäß der Domänenhierarchie, no synchronisiert ohne Angabe von Peers und Domänencontrollern, all verwendet sowohl manuelle als auch Domänenquellen. Durch die Angabe /localclockdispersion wird die Genauigkeit der internen Uhr konfiguriert, sodass bekannt ist, wenn keine Zeit von den konfigurierten Quellen erfasst werden kann. Der Parameter /reliable bestimmt im Fall der Ausführung auf einem Domänencontroller, ob dieser eine Zeitquelle ist. Mit /largepaseoffset bestimmen Sie, welche Zeitdifferenz zwischen lokaler und Netzwerkzeit als Spike bezeichnet wird.

w32tm /tz

Zeigt die Zeitzoneneinstellungen an.

w32tm /dumpreg [/subkey:*Schlüssel*] [/computer:*Name***]**

Gibt die Registrierungswerte des Zeitdiensts des lokalen oder angegebenen Computers aus, Standardschlüssel ist *HKLM\-System\CurrentControlSet\Services\W32Time*. Die Angabe eines Unterschlüssels grenzt die Anzeige auf diesen Unterschlüssel des obigen Standardschlüssels ein.

w32tm /query [/computer:*ComputerName***] {/*source* | /*configuration* |**

***/peers* | */status*} [/verbose]**

Zeigt Informationen des Zeitdiensts auf dem lokalen oder angegebenen Computer an. `/source` gibt die Synchronisationsquelle aus, `/configuration` die Konfiguration des Diensts, `/peers` eine Liste der Peers und `/status` den aktuellen Status. `/verbose` aktiviert die ausführliche Ausgabe.

w32tm /debug {/disable | {/enable /file:*Name* **/size:***Bytes* **/entries:***Wert* **[/truncate]}}**

Aktiviert (`/enable`) oder deaktiviert (`disable`) das private Protokoll des Windows-Zeitdiensts auf dem lokalen Computer. Dabei gibt `/file` den absoluten Dateinamen der Logdatei an, `/size` die maximale Größe der Datei. Durch `/entries` können Sie eine Liste von Flags für die zu protokollierenden Informationen festlegen, die durch kommaseparierte Zahlen zwischen 0 und 300 angegeben werden. Dabei kann ein Bereich durch Bindestrich verbunden eingeschlossen werden, `0-300` protokolliert alle Informationen. Mit `/truncate` bewirken Sie die Kürzung einer vorhandenen Protokolldatei.

Hinweis: Bei virtuellen Maschinen wirken häufig unerwartete Faktoren auf die Erzeugung der Systemzeit ein. So ist häufig der Hyper-V-Host als Zeitgeber tätig und überschreibt alle sorgsam festgelegten Zeitermittlungsroutinen. Gerade in Domänenumgebungen sollte diese Funktion deaktiviert werden.

---

wevtutil

Mit `wevtutil` werden Ereignisprotokolle ausgewertet und konfiguriert. Die Befehle können als Kurzwort oder alternativ in voller Länge angegeben werden. Aus Platzgründen sind hier nur die wichtigsten Optionen aufgeführt.

wevtutil {el | enum-logs}

Zeigt die Namen aller Ereignisprotokolle an. Aufgrund der Länge dieser Liste ist es angeraten, die Ausgabe mit `more`, durch Umleitung in eine Textdatei oder durch Filterung mit `findstr` abzufangen.

wevtutil {*gl* | *get-log*} *Protokoll* **[/format:{*text* | *xml*}]**

Zeigt Konfigurationsinformationen wie die maximale Größe oder den Speicherort eines Protokolls im Text- (Standard) oder XML-Format an.

wevtutil {*gli* | *get-loginfo*} *Protokoll* **[/logfile:{*true* | *false*}]**

Zeigt Informationen wie Zeitstempel, Anzahl der Einträge und Dateigröße auf der Festplatte zum angegebenen Protokoll an. Dabei kann es sich auch um den Pfad zu einem gesicherten Protokoll handeln. Das wird mit `/logfile:true` angegeben.

wevtutil {*qe* | *query-events*} *Protokoll* **[/lf:{*true* | *false*}] [/count:***Anzahl***] [/format:{*text* | *xml*}]**

Gibt (alle oder maximal *Anzahl*) Ereignisse des Protokolls (Protokollname oder mit `/lf` oder `/logfile:true` Pfad zum Protokoll) im XML-Format (Standard) oder als Text aus.

wevtutil {*epl* | *export-log*} *Protokoll Datei* **[/overwrite:{*true* | *false*}]**

Exportiert ein Protokoll in eine Datei, die überschrieben wird, falls sie schon existiert.

wevtutil {*al* | *archive-log*} *Protokolldatei* **[/locale:***Gebietsschema***]**

Archiviert alle gebietsschemaspezifischen Informationen aus einer Protokolldatei in einem Unterverzeichnis. Es wird entweder das aktuelle oder das angegebene Gebietsschema verwendet. *Gebietsschema* wird als zusammengezogenes Kürzel in der Form *Sprache-Land* angegeben, z.B. `en-US` (Englisch, USA) oder `de-DE` (Deutsch, Deutschland).

wevtutil {*cl* | *clear-log*} *Protokoll* **[/backup:***Datei***]**

Löscht ein Protokoll, nachdem es optional in eine Datei (Endung *.evtx*) gesichert wurde.

wevtutil {*ep* | *enum-publishers*}

Listet die auf dem System registrierten Herausgeber auf.

wevtutil {*gp* | *get-publisher*} *Herausgeber* **[/getevents:{*true* | *false*}] [/getmessage:{*true* | *false*}] [/format:{*text* | *xml*}]**

Zeigt Konfigurationsinformationen über einen Herausgeber an. Optional werden alle von diesem Herausgeber registrierten Ereignisse (`/getevents:true`) und Meldungstexte (`/getmessage:true`) aufgelistet.

wevtutil {*sl* | *set-log*} *Protokoll* **[***set-log-Optionen***] [*AllgemeineOptionen*]**

Ändert die Konfiguration eines Protokolls.

Wichtige Optionen für set-log

/{*e* | *enabled*}:[*true* | *false*]

Aktiviert (`true`) oder deaktiviert (`false`) das Protokoll.

/{*i* | *isolation*}:*Isolationsmodus* [/ca:*SDDL***]**

Legt den Protokoll-Isolationsmodus auf einen der gültigen Werte `system`, `application` oder `custom` fest. Im Fall von `custom` muss eine Sicherheitsbeschreibung im SDDL-Format angegeben werden.

/{*lfn* | *logfilename*}:*Protokolldatei***

Vollständiger Pfad zur Protokolldatei, in der die Ereignisse des Protokolls gespeichert werden.

/{*q* | *quiet*}:{*true* | *false*}

Option zur stillen Ausgabe ohne Benutzerinteraktion. Ohne Angabe eines Werts wird `true` angenommen.

/{*rt* | *retention*}:[*true* | *false*]

Bei vollem Protokoll werden neue Ereignisse verworfen (`true`), bzw. sie überschreiben die ältesten Ereignisse im Protokoll (`false`).

/{*ab* | *autobackup*}:[*true* | *false*]

Falls `true`, wird das Protokoll automatisch gesichert. In einem solchen Fall muss `/retention` ebenfalls auf *`true`* gesetzt werden.

/{*ms* | *maxsize*}:*Byte***

Maximale Größe des Protokolls in Byte.

Weitere Optionen regeln den Ebenenfilter des Protokolls (`/l`), den Schlüsselwortfilter (`/k`), ermöglichen das Setzen der Zugriffsberechtigung (`/ca`) oder das Einlesen der Einstellungen aus einer Konfigurationsdatei im XML-Format (`/c`). Weiterführende Details dazu entnehmen Sie bitte der Windows-Hilfe zum Befehl.

Allgemeine Optionen

/{r | remote}:*Computer* **[/{u | username}:***Benutzer*
[/{p | password}:{*Kennwort* **| *}]]**

Führt den Befehl auf ***Computer*** statt auf dem lokalen System aus. Wird von den Befehlen `install-manifest` und `uninstall-manifest` nicht unterstützt. Zur Verbindung können ein ***Benutzer*** sowie ein ***Kennwort*** angegeben werden. Wird statt Letzterem ein Sternchen verwendet, wird zur Eingabe des Kennworts aufgefordert.

/{a | *authentication*}:*Authentifizierungstyp*

Legt die Art der Authentifizierung am entfernten Computer fest. Mögliche Werte sind `Default`, `Negotiate` (Standard), `Kerberos`, `NTLM`.

/{*uni* | *unicode*}:{*true* | *false*}

Unicode-Ausgabe: ja oder nein.

PoSh: PowerShell bietet umfangreiche Möglichkeiten, die Ereignisanzeige auszulesen oder Einträge darin zu erfassen. Diesbezügliche Cmdlets beinhalten das Wort `eventlog` als Bestandteil des Namens, beispielsweise `get-eventlog` gefolgt vom Namen der gewünschten Logdatei. Die jüngsten zehn Einträge im Systemlog zeigt der folgende Befehl:

```
get-eventlog –logname system –newest 10
```

---

wmic

Die WMI-Konsole gilt als veraltet, stattdessen soll laut Microsoft PowerShell genutzt werden.

Die WMI-Konsole ermöglicht die direkte Interaktion mit dem WMI-Dienst des lokalen oder eines entfernten Systems. Mit `wmic` können äußerst detaillierte Informationen abgefragt oder gesetzt werden. Der Befehl bietet eine interaktive Kommandozeile (Start durch Aufruf ohne Parameter, Beenden mit `exit`), kann Kommandos aber auch im Batchmodus ausführen (Aufruf mit Parametern). Ein angehängtes `/?` gibt auf jeder Ebene einen passenden Hilfetext aus.

wmic path *WMI-Klasse* **[get** *Attributliste* **[*Get-Optionen*]]**

[/format:*Ausgabeformat***] [/output:***Ausgabedatei***]**

Gibt Informationen über Objektinstanzen einer WMI-Klasse aus. Mit `/get` kann die Ausgabe auf einzelne Attribute der einzelnen Objekte beschränkt werden (Standard: `get /all`). Als *Ausgabeformat* können unter anderem folgende gewählt werden: `table` (Standard), `list`, `csv` (kommasepariert), `htable` (tabellarisch im HTML-Format) und `hform` (Liste im HTML-Format). Die Ausgabe kann mit `/output` in eine Datei umgeleitet werden.

Anstelle der etwas umständlichen Angabe von `path` *WMI-Klasse* kann einer der zahlreichen vordefinierten Aliase verwendet werden. Eine Liste mit diesen kann mit `wmic alias get friendlyname,target` erstellt werden.

wmic path *WMI-Klasse* **set** *Attribut=Wert*

Setzt ein Attribut auf den angegebenen Wert. Eine Liste der für eine Klasse verfügbaren Attribute erhält man mit `wmic path WMI-Klasse set /?`.

wmic path *WMI-Klasse* **call** *Methode Parameterliste*

Ruft die angegebene *Methode* für eine Objektinstanz der *WMI-Klasse* auf und übergibt dabei die *Parameterliste*. Auch hier können mit `call /?` die für eine Klasse gültigen Methoden ermittelt werden. Viele Methoden können nur auf konkrete Instanzen angewendet werden, sodass z.B. `wmic os call reboot` fehlschlägt, während `wmic os where "csname='`*Name*`'" call reboot` nach einer Rückfrage die durch *Name* angegebene Betriebssysteminstanz neu startet.

Optionen

/namespace:"*Namespace***"**

Legt den Namespace für die Ausführung des Aliasvorgangs fest. Wird dieser nicht explizit mit `"\\"` begonnen, wird er als relativ zum aktuellen Namespace behandelt.

/role:"Namespace"

Pfad für die Funktion, die die Aliasdefinitionen enthält.

/node:{*@Datei1* **|** *Computername1***} [***@Datei2* **|** *Computername2***][...]**

Gibt an, auf welchen Systemen der Befehl ausgeführt wird. Anstatt die Computernamen in der Befehlszeile anzugeben, kann die Liste der Systeme aus *Datei* gelesen werden. Der WMI-Dienst muss auf den Zielsystemen gestartet werden, und die erforderliche Freigabe muss in der Firewall konfiguriert sein.

/implevel:*Identitätswechselebene* **[/authority:***Autoritätstyp***]**

Bestimmt die Identitätswechselebene (*Impersonation Level*; anonymous, identify, impersonate oder delegate), auf der der Befehl ausgeführt wird. Der Schalter /authority erlaubt die Angabe eines Autoritätstyps.

/authlevel:*Authentifizierungsebene*

Bestimmt die Authentifizierungsebene der Befehlszeile (default, none, connect, call, pkt, pktintegrity, pktprivacy). Standard ist pktprivacy.

/locale:*Gebietsschemabezeichnung*

Bestimmt die Sprachkennung für die Befehlszeile. Die Gebietsschemabezeichnung muss im Format ms_xxx angegeben werden, wobei xxx ein sprachabhängiger numerischer Wert ist. Beispiel für Englisch: ms_409.

/privileges:{*enable* | *disable*}

Aktiviert bzw. deaktiviert alle Berechtigungen.

/trace:{*on* | *off*}

Schaltet die Ausgabe von Debuginformationen nach stderr ein oder aus.

/record:*Dateipfad*

Protokolliert alle Benutzereingaben und Bildschirmausgaben von WMIC in der als *Dateipfad* angegebenen XML-Datei.

/interactive:{*on* | *off*]

Setzt den interaktiven Modus bzw. setzt ihn zurück.

/user:*Domäne\Benutzer*

Legt das Benutzerkonto fest, unter dem der Befehl ausgeführt wird.

/password:*Kennwort*

Legt das Kennwort für Benutzer fest.

/output:{*stdout* | *clipboard* | *Dateiname*}

Bestimmt das Ziel der Ausgabeumleitung: Standardausgabe (`stdout`), Zwischenablage (`clipboard`) oder die angegebene Datei.

/append: {*stdout* | *clipboard* | *Dateiname*}

Bestimmt den Modus der Ausgabeumleitung.

/aggregate:{*on* | *off*

Bestimmt den Ergebnisanzeigemodus.

where WQL-Filter

Begrenzt die Menge der Objekte, auf die sich der Befehl bezieht, durch einen in der SQL-ähnlichen Abfragesprache WMI Query Language verfassten *`WQL-Filter`*.

Beispiele

wmic {*os* | *computersystem* | *bios* | *cpu*} list full

Gibt detaillierte Informationen über das (Betriebs)System bzw. das BIOS oder den Prozessor aus.

wmic process where "name='wmic.exe'" list full

Zeigt alle verfügbaren Informationen über den Prozess *wmic.exe* an.

PoSh: Mit PowerShell sind umfassende Möglichkeiten vorhanden, per WMI oder besser CIM auf die Systeminformationen zuzugreifen. Dazu dienen vorrangig die Cmdlets `get-wmiobject` (Alias `gwmi`) und `get-ciminstance` (Alias `gcim`), gefolgt von der gewünschten Klasse, beispielsweise in dieser Form:

```
gwmi win32_bios
gwmi win32_computersystem
gcim win32_physicalmemory
```

Durch Anhängen von `| select *` werden auch standardmäßig nicht angezeigte Informationen mit ausgegeben. Die auf dem jeweiligen System verfügbaren Klassen können Sie entweder mit

```
gwmi –namespace "root\cimv2" –list
```

oder mit dem Befehl `get-cimclass` ermitteln. Ein konkretes Skriptbeispiel

finden Sie im PowerShell-Abschnitt zum Befehl `sysinfo`.

Systemkonfiguration

at

Konfiguriert geplante Aufgaben. Der Befehl ist veraltet. Wenngleich er in Windows 11/Server 2025 noch aufgerufen werden kann, funktioniert er nicht mehr. Stattdessen wird die Verwendung von `schtasks` empfohlen.

bcdedit

Verwaltet den Startkonfigurationsdatenspeicher (*BCD Store*). Hier werden aus Platzgründen nur die wichtigsten Optionen vorgestellt. Die generelle Syntax sieht so aus:

`bcdedit [/store Dateiname]`*`Befehl`* **`[{`***`ID1`***`} [{`***`ID2`***`} [...]]] [`***`Optionen`***`]`**

Wird der Parameter `/store` nicht verwendet, bezieht sich der Befehl auf den Startkonfigurationsdatenspeicher des laufenden Systems. Die meisten Befehle erwarten Bezeichner (GUIDs, *Globally Unique Identifier*, im Format *{xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}*, wobei jedes *x* für eine Hexadezimalziffer steht) von Einträgen im Datenspeicher als Parameter. Geschweifte Klammern und Bindestriche sind Bestandteile des Bezeichners. Wollen Sie eine vorhandene GUID verwenden, markieren Sie sie am besten in der Eingabeaufforderung, kopieren sie in die Zwischenablage und fügen sie an der gewünschten Stelle ein.

`bcdedit /enum all` listet alle Einträge im Speicher samt ihren IDs (*Bezeichner* genannt) auf. Einträge, für die keine bekannte ID existiert, werden anhand ihrer GUID angesprochen. Wird der Parameter `/v` angegeben, werden auch für bekannte Bezeichner vollständige IDs anstelle der

Kurzschreibweise angezeigt. Der Aufruf von `bcdedit` ohne Parameter entspricht dem Aufruf `bcdedit /enum active`, der alle Einträge in der Anzeigereihenfolge des Startmanagers anzeigt.

`bcdedit {/export | /import}` *`Datei`*

Exportiert den Inhalt des Datenspeichers in die angegebene Datei bzw. stellt den Datenspeicher daraus wieder her.

Tipp: Die Exportfunktion sollten Sie unbedingt nutzen, bevor Sie den Befehl `bcdedit` zu Änderungen einsetzen, um im Fall eines Fehlers einfach zur ursprünglichen Konfiguration zurückkehren zu können.

`bcdedit /copy {`*`ID`***`} /d`** *`Beschreibung`*

Erstellt eine Kopie des durch *`ID`* bezeichneten Eintrags und setzt die *`Beschreibung`* für den neuen Eintrag.

`bcdedit /create [{`*`ID`***`} | /application {`***`Anwendungstyp`***`}] /d`** *`Beschreibung`*

Erstellt einen neuen Eintrag, dessen Typ entweder durch Angabe einer bekannten ID oder durch Spezifikation des Anwendungstyps (`bootsector`, `osloader`, `resume` oder `startup`) gesetzt wird.

`bcdedit /delete {`*`ID`***`} [/f] [`*****`cleanup | nocleanup`*****`]`**

Löscht einen Eintrag und entfernt ihn aus der Anzeigereihenfolge. Zum Löschen bekannter IDs muss `/f` angegeben werden. `nocleanup` löscht den Eintrag, ohne ihn aus der Anzeigereihenfolge zu entfernen.

`bcdedit /deletevalue [{`*`ID`***`}]`** *`Datentyp`*

Löscht einen Wert eines Eintrags im Datenspeicher. Ohne Angabe von *`ID`* wird `{current}` verwendet. Eine Liste der Datentypen erhalten Sie mit `bcdedit /?types`.

`bcdedit /set [{`*`ID`***`}]`** *`Datentyp Wert`*

Setzt einen Wert eines Eintrags im Datenspeicher. Ohne Angabe von *`ID`* wird `{current}` verwendet.

`bcdedit /bootsequence {`*`ID1`***`} [{`***`ID2`***`} [...]] [/addfirst | /addlast | /remove]`**

Legt die Startreihenfolge für den nächsten Neustart auf die durch die IDs angegebene Reihenfolge fest. Wenn nur eine einzige ID angegeben wird, kann diese an den Anfang (`/addfirst`) oder an das Ende (`/addlast`)

gesetzt oder ganz entfernt (`/remove`) werden.

`bcdedit /displayorder {`*`ID1`*`} [{`*`ID2`*`} [...]] [/addfirst | /addlast | /remove]`

Legt die Anzeigereihenfolge des Startmanagers fest. Die Optionen entsprechen denen des Befehls `/bootsequence`.

`bcdedit /default {`*`ID`*`}`

Legt den Standardeintrag fest, der zum Starten verwendet wird, wenn das Timeout abläuft.

`bcdedit /timeout` *`Sekunden`*

Legt die Zeit in Sekunden fest, die in Multibootkonfigurationen gewartet wird, bevor der Standardeintrag verwendet wird.

`bcdedit /hypervisorsettings`

Zeigt die Hypervisor-Debuggereinstellungen an oder setzt sie (mit weiteren Parametern).

PoSh: In Windows 11 und Server 2025 sind zahlreiche PowerShell-Cmdlets zur Verwaltung des Startkonfigurationsspeichers enthalten. Mit dem Befehl

```
get-command -module Microsoft.Windows.Bcd.Cmdlets
```

können Sie sich diese Cmdlets anzeigen lassen. Die darin enthaltenen Befehle decken prinzipiell alle mit `bcdedit` zu erfüllenden Aufgaben ab, stehen aber nicht in der Windows-PE-Umgebung zur Verfügung.

---

bcdboot

`bcdboot` *`Quelle`* **`[/l`** *`Gebietsschema`***`] [/s`** *`Laufwerkbuchstabe`***`] [/v] [/vbcd] [/m] [{`***`ID des Betriebssystem-Ladeprogramms`***`}] [/addlast] [/p] [/c]`**

Dient zum Kopieren wichtiger Startdateien in die Systempartition und zum Erzeugen eines neuen System-BCD-Speichers.

Optionen

`Quelle`

Gibt den Speicherort des Windows-Systemstamms an.

/l

Gibt einen optionalen Gebietsschemaparameter an, der anstelle von US-Englisch für die Initialisierung des BCD-Speichers verwendet wird.

/s [/f {*uefi* | *bios* | *all*}]

Gibt ein optionales Ziellaufwerk an, auf das die Startumgebungsdateien kopiert werden. Standardmäßig wird als Ziel die durch die Firmware identifizierte Systempartition verwendet. Mit `/f` können Sie den Firmwaretyp der Zielsystempartition angeben.

/m *{ID des Betriebssystem-Ladeprogramms}*

Bei Angabe einer GUID wird das angegebene Ladeprogramm-objekt mit der Systemvorlage zusammengeführt, um einen startfähigen Eintrag zu erzeugen. Ohne Angabe einer GUID werden nur globale Objekte zusammengeführt.

/v

Aktiviert den ausführlichen Modus.

/vbcd

Aktiviert die BCD-Protokollierung.

/addlast

Legt fest, dass der Eintrag des Windows-Startmanagers am Schluss angefügt werden soll (Standard: am Anfang).

/bcdclean [*full*]

Bereinigt den BCD-Speicher, wobei standardmäßig nur Duplikate entfernt werden. Mit dem Parameter ***full*** bewirken Sie eine Überprüfung jedes Eintrags. Ist das im Eintrag referenzierte Gerät nicht vorhanden, wird dieser gelöscht.

/d

Sorgt für den Erhalt des vorhandenen Windows-Standardstarteintrags.

/p

Sorgt für die Beibehaltung der Position des Windows-Startmanager-Firmwareeintrags. Ohne diesen Parameter wird an der ersten Position ein neuer Eintrag erstellt.

/c

Unterbindet das Migrieren aller vorhandenen Objekte, die von der Vorlage beschrieben werden.

---

cmdkey

Verwaltet gespeicherte Anmeldeinformationen für Server oder Domänen.

cmdkey /list[*:Ziel***]**

Zeigt gespeicherte Anmeldeinformationen für ***Ziel*** an. Falls dieses weggelassen wird, werden alle gespeicherten Informationen ausgegeben.

cmdkey /add:***Ziel*** **[**/smartcard | **/user:***Domäne\Benutzer***]** **[/pass:***Kennwort***]**

Fügt einen Eintrag für ***Ziel*** hinzu. Sollte das ***Kennwort*** nicht angegeben werden, wird es beim Herstellen der Verbindung abgefragt. Für als Server fungierende Einzelplatzrechner im lokalen Netz geben Sie anstelle einer Domäne den Rechnernamen an. Das angegebene Benutzerkonto muss auf dem Zielsystem existieren und gegebenenfalls das Recht zur Anmeldung über das Netzwerk haben. Für die Erstellung generischer Anmeldeinformationen können Sie den Schalter /add durch /generic ersetzen.

cmdkey /delete:***Ziel***

Löscht gespeicherte Anmeldeinformationen für ***Ziel***.

cmdkey /delete /ras

Löscht RAS-Anmeldeinformationen.

Tipp: Eine praktische Anwendung des Befehls ist das Speichern von Zugangsdaten für Remotedesktopverbindungen, die mit normalen Mitteln gespeicherte Kennwörter abweisen. Hierzu verwenden Sie folgende Syntax in Ihrem normalen Benutzerkontext:

cmdkey /generic:TERMSRV/***Rechnername*** /user: ***"Domäne\Benutzername" /pass***

Geben Sie das Kennwort nicht gleich im Klartext hinter dem Parameter /***pass*** ein, wird es abgefragt.

Um den Eintrag wieder aus der Anmeldeinformationsverwaltung zu entfernen, nutzen Sie folgenden Befehl:

```
cmdkey /delete:TERMSRV/Rechnername
```

---

fondue

Dient der Aktivierung optionaler Windows-Features durch Herunterladen der Dateien von Microsoft Update oder aus per Gruppenrichtlinie zugewiesenen Quellen.

`fondue /enable-feature`*`:Wert`* **`[/caller-name:`***`aufrufendes Programm`***`] [hide-ux:{all|rebootRequest}`**

Mit `enable-feature` wird ein Windows-Feature festgelegt, das auf dem lokalen System aktiviert werden soll. Zusätzlich kann für die Berichterstattung an Microsoft mit `caller-name` der Name des aufrufenden Prozesses oder Skripts angegeben werden und mit `hide-ux` für die Nutzung von Skripten das Anzeigen aller Meldungen (`all`) ausgeblendet werden oder nur solcher Meldungen, die einen Neustart anfordern (`rebootRequest`). Das Ausblenden von Status- und Berechtigungsanforderungen für Windows Update bewirkt ein Fehlschlagen der Installation, falls die Berechtigung erforderlich ist.

PoSh: In PowerShell erhalten Sie mit dem Befehl

```
get-windowsoptionalfeature –online | ft
```

eine Auflistung aller optionalen Windows-Features der laufenden Windows-Installation. Um nur nicht installierte tabellarisch anzuzeigen, hängen Sie `| where {$_.State –eq "Disabled"} | ft` an den obigen Befehl an.

Mit folgendem Befehl aktivieren Sie das Feature *TIFFIFilter*:

```
enable-windowsoptionalfeature -featurename TIFFIFilter -online
```

Um ein Feature zu deaktivieren, verwenden Sie das Cmdlet `disable-windowsoptionalfeature`.

In Windows Server 2025 stehen Ihnen zudem die Cmdlets `get-windowsfeature` sowie `install-windowsfeature` zur Verfügung, um verfügbare und installierte Serverfeatures aufzulisten und zu installieren.

---

licensingdiag

Das Programm `licensingdiag` dient der Auswertung von Informationen zum Lizenzierungsstatus von Windows.

`licensingdiag -report` *`Dateiname.xml`* **`-log`** *`Dateiname.cab`*

Erstellt einen Diagnosebericht im XML-Format sowie eine CAB-Datei mit diesem Bericht und weiteren relevanten Dateien für die genauere Fehleranalyse im Fall von Aktivierungsproblemen.

---

powercfg

Die Kommandozeilenversion des Systemsteuerungs-Applets *Energieoptionen* verwaltet Energieschemata. Der Befehl verfügt über zahlreiche Optionen; die hier nicht aufgelisteten entnehmen Sie bitte der Hilfe.

Optionen

`powercfg /? [`*`Option`***`]`**

Zeigt die allgemeine Hilfe zu `powercfg` bzw. zur angegebenen Befehlsoption an.

`powercfg /{l | list}`

Zeigt die vorhandenen Energieschemata an und gibt auch deren GUID aus, die für etliche der weiteren Optionen benötigt wird.

`powercfg /{q | query} [`*`Energieschema`***`]`**

Zeigt die Einstellungen des aktuellen oder des angegebenen Energieschemas an.

`powercfg /duplicatescheme` *`Schema-GUID Ziel-GUID`*

Dupliziert ein Energieschema unter Angabe der GUID des existierenden Schemas. Wird keine Ziel-GUID angegeben, erstellt der Befehl eine.

powercfg /changename *Schema-GUID Name* **[***Beschreibung***]**

Ändert den Namen und optional die Beschreibung des durch die GUID gewählten Energieschemas. Da ein Energieschema nicht mehr direkt erstellt werden kann, ist die Kombination aus Duplizieren und Umbenennen eine Alternative für ein eigenes Schema.

powercfg /{d | delete} *Schema-GUID*

Löscht das angegebene Energieschema.

powercfg /{s| setactive} *Energieschema*

Aktiviert das angegebene Energieschema.

powercfg /{x | change} *Einstellung Wert*

Ändert eine Einstellung im aktuellen Energieschema.

powercfg {/export | /import} *Datei* **[***Schema-GUID***]**

Exportiert das durch *Schema-GUID* angegebene Energieschema in eine Datei bzw. liest eine Schemadefinition aus einer Datei wieder ein. Wenn beim Import keine GUID angegeben wird, wird diese neu erstellt.

powercfg /aliases

Zeigt alle Aliasnamen und deren GUIDs an. Die Namen können in `powercfg`-Kommandos anstelle der GUIDs angegeben werden.

powercfg /{h| hibernate} {on | off} [/size xx]

Schaltet die Unterstützung für den Ruhezustand ein bzw. aus. Der optionale Parameter `size` gibt die gewünschte Größe der Ruhezustandsdatei (*hiberfil.sys*) prozentual zur Größe des Arbeitsspeichers an und muss mindestens 50 sein.

powercfg {*deviceenablewake* | *devicedisablewake*} *Gerätename*

Aktiviert oder deaktiviert die Fähigkeit eines Geräts, das System aus dem Stand-by-Modus zu reaktivieren. Die Gerätenamen können Sie mit dem Befehl `powercfg /devicequery wake_programmable` abfragen.

powercfg /waketimers

Zeigt die im System aktiven Aktivierungszeitgeber an.

powercfg /lastwake

Gibt an, wodurch das System aus dem letzten Stand-by-Modus reaktiviert wurde.

`powercfg /energy`

Analysiert das System auf allgemeine Probleme hinsichtlich Energieeffizienz und Akkulaufzeit. Das Ergebnis wird in einer HTML-Datei gespeichert.

`powercfg /batteryreport`

Erzeugt einen Bericht zur Akkuverwendung in Mobilgeräten.

`powercfg /systempowerreport`

Erzeugt einen Bericht zu den Leistungsübergängen des Systems (ersetzt `/systemsleepdiagnostics`).

PoSh: PowerShell weist nur rudimentäre Möglichkeiten zur direkten Steuerung der Energieverwaltung auf. Der aktuelle Befehlsumfang beschränkt sich auf die Cmdlets
`Get-NetAdapterPowerManagement`,
`Enable-NetAdapterPowerManagement`,
`Set-NetAdapterPowerManagement` und
`Disable-NetAdapterPowerManagement`
zur Verwaltung der Energiespareinstellungen von Netzwerkadaptern.

---

pwlauncher

`pwlauncher [/enable] [/disable]`

Fragt ohne Parameter die aktuelle Einstellung der *Windows To Go*-Startoption ab, schaltet sie ein (`enable`) oder deaktiviert sie (`disable`). Dabei sei angemerkt, dass *Windows To Go* selbst bereits in Windows 10 durch Microsoft entfernt wurde.

---

reagentc

`reagentc Befehl [`Argumente`]`

Konfiguriert die Windows-Wiederherstellungsumgebung (WinRE) und die Systemzurücksetzung.

Befehle

`/info`

Zeigt Informationen zur Konfiguration der Windows-Wiederherstellungsumgebung und der Systemzurücksetzung an.

`/setreimage /path` *`Verzeichnispfad`*
`[/target` *`Pfad zur Windows-Installation`***`]`**

Gibt das Verzeichnis mit dem benutzerdefinierten WinRE-Image (*winre.wim*) an und – sofern nicht das laufende Betriebssystem adressiert wird – den Pfad zum Verzeichnis der Windows-Installation.

`/enable | /disable`

Aktiviert bzw. deaktiviert die lokale Kopie der Wiederherstellungsumgebung.

`/boottore`

Konfiguriert das System so, dass beim nächsten Neustart die Wiederherstellungsumgebung gestartet wird.

`/setosimage /path` *`Pfad`* **`/index`** *`Imageindex`*
`[/target` *`Windows-Pfad`***`]`**

Gibt den Verzeichnispfad des für die Systemwiederherstellung verwendeten Wiederherstellungsimage an, den Index der Betriebssysteminstallation innerhalb der Imagedatei und den Pfad zur Windows-Installation, sofern nicht das laufende Betriebssystem angesprochen wird.

`/setbootshelllink /configfile` *`XML-Datei`* **`[/target`** *`Verzeichnispfad`***`]`**

Fügt der Seite für das Zurücksetzen und Wiederherstellen im Startmenü einen per XML-Konfigurationsdatei beschriebenen Link hinzu. Sollte sich der Eintrag nicht auf das laufende Betriebssystem beziehen, kann der Pfad zur Windows-Installation im Parameter `/target` angegeben werden.

register-cimprovider

register-cimprovider -namespace *Namespace-Name* **-providername** *Anbietername* **-path** *Pfad zur Anbieter-DLL* **[-Optionen]** *-help*

Registriert einen CIM-Anbieter im System. Dabei sind Ziel-Namespace, Anbietername und Pfad zur Anbieterbinärdatei immer anzugeben. Es stehen weitere umfangreiche Optionen zur Verfügung, die Sie bei Bedarf der Windows-Hilfe entnehmen können. Dafür nutzen Sie anstelle von `/?` den Parameter `-help`.

regsvr32

regsvr32 [/u] [/s] [/n] [/i[Befehlszeile]] *DLL*

Registriert eine DLL (typischerweise eine COM-Komponente) auf dem System.

Optionen

/u

Hebt die Registrierung der Komponente auf.

/s

Stiller Modus: Es werden keine Meldungen ausgegeben.

/i [*Befehlszeile***]**

Ruft `DLLInstall` mit optionaler Befehlszeile auf, `/u` ruft die Deinstallation auf.

/n

Ruft `DLLRegisterServer` nicht auf. Dieser Parameter muss zusammen mit `/i` verwendet werden.

schtasks

schtasks */Befehl* **[**Optionen**] [/s** *System* **[/u** *Benutzer* **[/p [***Kennwort***]]]]**

Verwaltet geplante Aufgaben auf dem lokalen oder einem Remotecomputer. `/query` zeigt die geplanten Aufgaben eines Computers an. `/delete` löscht eine mit Namen angegebene oder alle geplanten Aufgaben. `/run` bzw. `/end` starten bzw. beenden eine geplante Aufgabe. `/create` und `/change` erzeugen bzw. modifizieren eine Aufgabe. Dabei können alle auch in der grafischen Oberfläche vorhandenen Einstellungen vorgenommen werden. `/showsid` zeigt die Sicherheits-ID an, die dem Namen einer geplanten Aufgabe entspricht.

Aus Platzgründen muss leider für eine Beschreibung der äußerst umfangreichen Optionen auch hier wieder auf die Hilfe des Befehls verwiesen werden, die Sie über `schtasks /?` bzw. zu den einzelnen Befehlen von `schtasks` mit `schtasks /Befehl /?` aufrufen können.

PoSh: Die Windows-Aufgabenplanung lässt sich sehr gut zur geplanten oder durch Systemereignisse getriggerten Ausführung von PowerShell-Skripten verwenden. Zugleich beinhaltet PowerShell Cmdlets, die das Erstellen (`new-scheduledtask`), Bearbeiten (`set-scheduledtask`), Verwalten (`enable-scheduledtask`, `disable-scheduledtask`) und Löschen geplanter Aufgaben zum Inhalt haben.

---

sconfig WS2025

Das Programm `sconfig` ist nur in Windows-Server-Versionen verfügbar. Insbesondere unter Server-Core-Installationen erleichtert es die Konfiguration eines Servers aus der Eingabeaufforderung heraus. Folgende Konfigurationsschritte werden unter anderem unterstützt:

- Beitritt zur Domäne.
- Umbenennung des Servers.
- Konfiguration der Remoteverwaltung durch Servermanager und PowerShell einschließlich der Firewall-Konfiguration.

- Konfiguration von Windows Update, insbesondere auch die Möglichkeit, einzustellen, Updates nur herunterzuladen oder nie nach Updates zu suchen, die aus der normalen GUI entfernt wurden.
- Aktivierung von Remote Desktop.
- Netzwerkkonfiguration.

Die einzelnen Einträge sind selbsterklärend und werden daher nicht weiter ausgeführt.

Für die meisten Belange der Serverkonfiguration wird unter Windows Server 2025 die Verwendung von PowerShell empfohlen.

---

scregedit.wsf WS2025

Dieser Befehl vereinfacht gängige Konfigurationstätigkeiten insbesondere in einer *Windows Server Core*-Installation, die ansonsten ein direktes Bearbeiten der Registrierung erfordern würde. Zudem enthält er einen nützlichen Hilfebildschirm mit weiteren oft benötigten Kommandos.

Optionen

`scregedit.wsf /cli`

Zeigt einen Hilfebildschirm mit gängigen Administrationsaufgaben und den entsprechenden Kommandozeilenbefehlen an.

`scregedit.wsf /au {/v | 4 | 3 | 1}`

Automatische Updates: aktuelle Einstellungen anzeigen (`/v`), aktivieren (`4`), nur Downloads durchführen (`3`) oder deaktivieren (`1`).

`scregedit.wsf /ar {/v | 0 | 1}`

Fernzugriff über Remotedesktopdienste: aktuelle Einstellungen anzeigen (`/v`), aktivieren (`0`) oder deaktivieren (`1`).

`scregedit.wsf /cs {/v | 0 | 1}`

Remotedesktopdienste-Zugriff von älteren Clients ohne Cred-SSP: aktuelle Einstellungen anzeigen (`/v`), zulassen (`0`) oder verweigern (`1`).

scregedit.wsf /im {/v | 0 | 1}

Remoteverwaltung von IPSEC mit IP-Sicherheitsmonitor zulassen: aktuelle Einstellungen anzeigen (`/v`), Remoteverwaltung verweigern (`0`) oder zulassen (`1`).

scregedit.wsf /dp {/v} [*Wert*]

Änderung der Priorität von DNS-SRV-Datensätzen (Stichwort: *LdapSrvPriority*) auf Domänencontrollern: aktuelle Einstellungen anzeigen (`/v`), *Wert* von 0 bis 65535, empfohlener Wert 200.

scregedit.wsf /dw {/v} [*Wert*]

Änderung der Gewichtung von DNS-SRV-Datensätzen (Stichwort: *LdapSrvWeight*) auf DCs: aktuelle Einstellungen anzeigen (`/v`), *Wert* von 0 bis 65535, empfohlener Wert 50.

---

slmgr.vbs

Verwaltet die Windows-Lizenz(en) und kann die Installation aktivieren. *slmgr.vbs* ist unter Server Core besonders nützlich, da dort die grafische Oberfläche zur Aktivierung von Windows fehlt. Auch für zahlreiche Optionen zur Lizenzverwaltung im Enterprise-Bereich gibt es keine korrespondierenden GUI-Werkzeuge. An dieser Stelle werden die wichtigsten Kommandos aufgeführt, weitere entnehmen Sie bitte der Hilfe:

slmgr.vbs {*/dli* | */dlv*} *all*

Zeigt (detaillierte) Informationen über die installierten Lizenzen an.

slmgr.vbs /xpr

Zeigt das Ablaufdatum der aktuellen Lizenz an bzw. ob die Installation permanent aktiviert ist.

slmgr.vbs /ipk *Lizenzschlüssel*

Installiert den angegebenen Lizenzschlüssel.

slmgr.vbs /ato

Aktiviert die Windows-Installation.

slmgr.vbs /rearm

Setzt den Lizenzierungsstatus des Computers zurück. Dies geschieht ohne

weitere Rückfrage und sollte auf produktiv genutzten Systemen nur in begründeten Ausnahmefällen durchgeführt werden.

`slmgr.vbs /rearm-app` *`<Anwendungs-ID>`*

Setzt den Lizenzierungsstatus der angegebenen App zurück.

`slmgr.vbs /skms {Name[:Port] | : Port] [Aktivierungs-ID]`

Legt den Namen und/oder Port fest, den der Clientrechner für die Aktivierung einer Volumenlizenz per Schlüsselverwaltungsdienst (*Key Management Service*, KMS) verwenden soll. Anstelle eines Namens kann auch die IP-Adresse angegeben werden.

Hinweis: Wird der Befehl nicht mit `cscript` und dem vollen Pfad zum Skript (meist *%windir%\system32*) aufgerufen, erfolgt die Ausgabe in einem GUI-Fenster, was beispielsweise den Abschluss aufrufender Skripte blockiert.

---

winmgmt

`winmgmt [/backup` *`Dateiname`***`]`**
`[/restore` *`Dateiname Kennzeichen`***`]`**
`[/resyncperf] [/standalonehost [`*`Ebene`***`]]`**
`[/sharedhost] [/verifyrepository [`*`Pfad`***`]]`**
`[/salvagerepository] [/resetrepository]`

Ermöglicht die Sicherung, Wartung und Wiederherstellung des WMI-Repository.

`winmgmt /backup` *`Dateiname`*

Veranlasst *Windows Management Instrumentation* (WMI) zur Sicherung des Repository mit dem angegebenen Dateinamen.

`winmgmt /resetrepository`

Setzt das Repository auf den Zustand unmittelbar nach der Installation des Betriebssystems zurück.

`winmgmt /restore` *`Dateiname Kennzeichen`*

Stellt das WMI-Repository aus der angegebenen Sicherungsdatei manuell wieder her. Das *`Kennzeichen`* bestimmt den Wiederherstellungsmodus:

`1` bedeutet Erzwingen (Trennung der Benutzer und Ausführen der Wiederherstellung), `0` bewirkt die Ausführung, wenn keine Benutzer verbunden sind.

`winmgmt /resyncperf`

Registriert die Leistungsbibliotheken bei WMI.

`winmgmt /salvagerepository`

Führt eine Konsistenzprüfung für das Repository aus und erstellt es neu, falls eine Inkonsistenz erkannt wird.

`winmgmt /verifyrepository [`*`Pfad`*`]`

Prüft die Konsistenz für das Live-Repository oder ein im Pfad gespeichertes vollständiges Repository.

Informationen zu weiteren Parametern des Befehls entnehmen Sie bitte seiner Hilfe.

---

winrm

Konfiguriert die Windows-Remoteverwaltung. Dieser Befehl bietet zahlreiche Optionen, von denen hier aus Platzgründen nur diejenigen zur Aktivierung und Deaktivierung des Fernzugriffs aufgeführt sind.

`winrm quickconfig`

Konfiguriert den Windows-Remoteverwaltungsdienst für den Fernzugriff durch `winrs` oder PowerShell.

`winrm set winrm/config/client '@{TrustedHosts="PC1,PC2"}'`

Fügt die Computer *`PC1`* und *`PC2`* der Liste der vertrauten Hosts für den Fernzugriff mittels `winrs` oder PowerShell hinzu, in diesem Fall nach den Schrägstrichen ohne Leerzeichen, da an dieser Stelle ein Pfad angegeben wird und keine Parameter.

`winrm delete winrm/config/listener?Address=*+Transport=HTTP`

Deaktiviert den Fernzugriff durch Löschen des Listeners. Gegebenenfalls muss HTTP durch HTTPS ersetzt werden.

PoSh: Um mit PowerShell Remotesitzungen durchführen zu können, muss

die Windows-Remoteverwaltung auf dem Zielrechner der Sitzung ebenfalls konfiguriert worden sein. Das lässt sich auch direkt aus PowerShell heraus erledigen. Führen Sie dazu folgende Befehle in einer administrativen Instanz aus:

```
set-wsmanquickconfig
enable-psremoting
```

Trusted Hosts lassen sich mit folgenden Befehlen ermitteln bzw. hinzufügen:

```
get-item WSMan:\localhost\Client\Trustedhosts
get-item WSMan:\localhost\Client\Trustedhosts –value 'PC1,PC2'
```

Um eine Remotesitzung mit der PowerShell-Konsole zu starten, genügt dann folgender Befehl, gegebenenfalls um den Parameter `–credential 'Domäne\Benutzer'` erweitert:

```
enter-pssession –computername Server
```

---

winrs

`winrs [`*`Optionen`***`]`** ***`Befehl`***

Führt ***`Befehl`*** in der Shell *cmd.exe* auf einem entfernten System aus. Voraussetzung dafür ist die korrekte Konfiguration des Diensts *Windows-Remoteverwaltung* auf dem anderen Computer, z.B. mit `winrm`.

Anstelle der *Enter*-Taste verwenden Sie die Tastenkombination *Strg+C* oder *Strg+Pause*. Ein nochmaliges Drücken der Tastenkombination beendet die `winrs`-Sitzung.

Optionen

`-[r | remote]:`*`Endpunkt`*

Spezifiziert den Computer, auf dem der Befehl ausgeführt werden soll (Standard: `localhost`). Der Endpunkt kann durch seinen NetBIOS-Namen oder eine URL angegeben werden.

-[u | username]:*Benutzer [***-[p | password]:***Kennwort]*

Gibt den zur Verbindung mit dem entfernten System zu verwendenden Benutzernamen und optional das dazugehörige Kennwort an, das ansonsten abgefragt wird.

-[t | timeout]:*Sekunden*

Falls ein Timeout für den auszuführenden Befehl verwendet werden soll, kann er hiermit angegeben werden. Die Option gilt als veraltet.

-[d | directory]:*Pfad*

Gibt das Startverzeichnis für den entfernten Befehl an. Falls nicht spezifiziert, wird *%HOMEDRIVE%%HOMEPATH%* verwendet.

-[env | environment]:*Variable=Wert*

Setzt eine Umgebungsvariable im entfernten Kommandozeileninterpreter. Mehrere dieser Parameter können verwendet werden.

-[noe | noecho]

Deaktiviert das Echo (Eingaben werden nicht lokal angezeigt).

-[nop | noprofile]

Verhindert das Laden des Benutzerprofils. Diese Option ist erforderlich, wenn das verwendete Benutzerkonto auf dem Remotecomputer nicht Mitglied der Gruppe der Administratoren ist.

-[ad | allowdelegate]

Erlaubt die Verwendung der Benutzeranmeldeinformationen für den Zugriff auf eine Remotefreigabe, die sich nicht auf dem angesteuerten Endpunkt befindet.

-[comp | compression]

Aktiviert die (von älteren Installationen auf Remotesystemen möglicherweise nicht unterstützte) Komprimierung.

-[use | usessl]

Verwendet bei Ansteuerung eines Remoteendpunkts SSL.

Beispiel

Das folgende Beispiel zeigt, wie einfach auf einem entfernten Computer

Befehle abgesetzt werden können. Die Zirkumflexe (^) sind notwendig, da die Variable `%computername%` ansonsten schon auf dem lokalen System durch den Computernamen ersetzt würde.

```
winrs -r:AndererComputer echo ^%computername^%
```

Netzwerk

Grundlegende Netzwerkbefehle

---

arp

Zeigt die Zuordnung von MAC-Adressen zu IP-Adressen an und erlaubt die Manipulation dieser Einträge.

`arp -a [`*`IP-Adresse`*`] [-N `*`Schnittstelle`*`] [-v]`

Zeigt die aktuellen Zuordnungen an, gegebenenfalls begrenzt auf die angegebene IP-Adresse oder Schnittstelle, `-v` aktiviert den ausführlichen Modus.

`arp -s `*`IP-Adresse MAC-Adresse`*** **`[`***`Schnittstelle`***`]`**

Legt eine Adresszuordnung fest, gegebenenfalls begrenzt auf die angegebene Netzwerkschnittstelle.

`arp -d `*`IP-Adresse`*** **`[`***`Schnittstelle`***`]`**

Löscht eine Adresszuordnung, gegebenenfalls begrenzt auf die angegebene Netzwerkschnittstelle.

PoSh: Den ARP-Cache lesen Sie mit dem PowerShell-Cmdlet `getnetneighbor` aus und modifizieren ihn mit `set-netneighbor`.

---

bitsadmin

Verwaltet Downloadjobs des BITS-Diensts (*Background Intelligent Transfer Service*), der auch vom Dienst *Windows Update* verwendet wird. An dieser Stelle werden aus Platzgründen nur die wichtigsten Optionen dieses mächtigen Befehls beschrieben. Eine vollständige Beschreibung aller

Optionen finden Sie in der Hilfe des Befehls. Microsoft empfiehlt die Verwendung der in die PowerShell integrierten Cmdlets zur Verwaltung von BITS.

Optionen

`bitsadmin /transfer *Job-Name* /download /priority normal *URL* *LokalerPfad*`

Erzeugt den Job *`Job-Name`*, fügt die in *`URL`* (wird inklusive Protokoll bezeichnet, z.B. *https://*) angegebene Datei dem Job hinzu und aktiviert und komplettiert (nach erfolgreichem Download) den Job. Die in *`URL`* angegebene Datei wird durch den Job im Hintergrund auf den lokalen Rechner übertragen und im angegebenen lokalen Pfad gespeichert.

`bitsadmin /list`

Listet alle Jobs mit GUID, Name, Status und Anzahl der übertragenen Bytes auf.

`bitsadmin /info *Job-Name* /verbose`

Zeigt sehr detaillierte Informationen zu einem Job an. Anstelle des Jobnamens kann auch dessen GUID (zu ermitteln mit dem Parameter `/list` oder `/info`) angegeben werden, da die Namen nicht eindeutig sein müssen.

`bitsadmin /cancel *Job-Name*`

Bricht einen Job ab und löscht alle vom Job erzeugten temporären Dateien. Statt des Jobnamens kann dessen GUID angegeben werden.

`bitsadmin /reset`

Löscht alle Jobs aus der Warteschlange.

PoSh: In PowerShell existieren mehrere Cmdlets rund um den BITS-Dienst. Diese sind am Namensbestandteil `Bits` zu erkennen.

---

checknetisolation

`checknetisolation [*Name des Moduls] [Argumente]`

Dies ist ein Hilfsprogramm für Entwickler zum Testen und zur Fehlersuche für die Netzwerkfähigkeiten von Windows-Store-Apps.

Verfügbare Module sind `loopbackexempt` zur Steuerung der Ausnahme für App-Container und Paketfamilien während der Anwendungsentwicklung und `debug` zum Starten einer Problemdiagnosesitzung für den Netzwerkverkehr eines App-Containers oder einer Paketfamilie.

checknetisolation loopbackexempt -{a|d|c|s} [-n=*Name*|-p=*SID*]

Durch `-a` wird der App-Container oder die Paketfamilie der Loop-back-Ausnahmeliste hinzugefügt, `-d` löscht sie von dieser. Die gesamte Ausnahmeliste wird durch `-s` angezeigt und mit `-c` gelöscht. App-Container oder Paketfamilie kann durch Angabe des Namens (`-n=`) oder der SID (`-p=`) ausgewählt werden.

checknetisolation debug [-n=*Name*|-p=*SID*]

Legt den App-Container oder die Paketfamilie durch Angabe des Namens oder der SID für die Fehleranalyse fest.

---

ftp

ftp [-v] [-d] [-i] [-n] [-g]
[-s:*Dateiname*] [-a] [-A] [-x:*Sendepuffer*]
[-r:*Empfangspuffer*] [-b:*Asynchronpuffer*]
[-w:*Fenstergröße*] [*Rechnername* | *IP-Adresse*]

Startet eine Sitzung zur Dateiübertragung mit einem anderen Rechner, der als FTP-Server konfiguriert sein muss.

Optionen

-A

Anmeldung als *Anonymous*.

-n

Unterdrückt das automatische Anmelden bei der ersten Verbindung mit dem entfernten Computer.

-w:*Größe*

Setzt die Größe des Übertragungspuffers in Byte (Standardwert: 65535).

-i

Schaltet Bestätigungsaufforderungen beim Übertragen mehrerer Dateien aus.

-s:*Dateiname*

Gibt eine Textdatei an, die `ftp`-Befehle enthält. Diese Befehle werden nach dem Starten von `ftp` automatisch ausgeführt.

-v

Unterdrückt Antwortmeldungen vom entfernten Server.

-a

Verwendet eine beliebige lokale Schnittstelle zur Herstellung der Datenverbindung.

-g

Deaktiviert die Interpretation von Wildcards in Dateinamen, auch als *Globbing* bezeichnet.

-d

Aktiviert Debugnachrichten.

-x:*Sendepuffer*

Überschreibt die Standardgröße des Sendepuffers (8192).

-r:*Empfangpuffer*

Überschreibt die Standardgröße des Empfangspuffers (8192).

-b:*Asynchronpuffer*

Überschreibt den Standard-Asynchronpuffer (3).

PoSh: PowerShell verfügt bislang über keine eigenständige FTP-Clientfunktionalität, kann aber auf die .NET-Klasse `FtpWebRequest` zugreifen und darüber beispielsweise Dateien auf FTP-Server hochladen oder von diesen herunterladen.

hostname

`hostname`

Zeigt den DNS-Hostnamen des lokalen Systems an.

ipconfig

`ipconfig [/all]`

Zeigt die IP-Konfiguration des lokalen Systems an. `/all` erstellt eine detaillierte Ausgabe.

`ipconfig /allcompartments [/all]`

Zeigt die IP-Konfiguration des lokalen Systems getrennt nach Depots (Routing-Compartments) an. `/all` erstellt eine detaillierte Ausgabe.

`ipconfig {/renew | /release [`*`Verbindungsname`*`]}`

Erneuert die per DHCP bezogene IPv4-Adresse der angegebenen Netzwerkverbindung bzw. gibt sie frei (Wildcards werden unterstützt).

`ipconfig {/renew6 | /release6 [`*`Verbindungsname`*`]}`

Erneuert die per DHCP bezogene IPv6-Adresse der angegebenen Netzwerkverbindung bzw. gibt sie frei (Wildcards werden unterstützt).

`ipconfig {/displaydns | /flushdns}`

Zeigt den Inhalt des DNS-Auflösungscaches an (`/displaydns`) bzw. löscht ihn (`/flushdns`).

`ipconfig{/showclassid | /setclassid [`*`Verbindungsname`*`]}`

Zeigt alle für den angegebenen Verbindungsnamen zugelassenen DHCP-Klassen-IDs an bzw. ändert sie.

`ipconfig {/showclassid6 | /setclassid6 [`*`Verbindungsname`*`]}`

Zeigt alle für den angegebenen Verbindungsnamen zugelassenen IPv6-DHCP-Klassen-IDs an bzw. ändert diese.

PoSh: PowerShell bringt mehrere Cmdlets für die Verwaltung der IP-Konfiguration mit, beispielhaft zu nennen wären `get-netipaddress`, `get-netipconfiguration`, `get-netipinterface`, `set-netipaddress`,

`new-netipaddress` und `remove-netipaddress`. Zudem können Sie über WMI, CIM oder direkt mit `ipconfig` arbeiten.

---

irftp

`irftp` *`Pfad`* **`[/h] | /s`**

Sendet Dateien über eine meist nur bei älteren Geräten noch vorhandene Infrarotverbindung. `/s` öffnet das Eigenschaftsfenster der drahtlosen Verbindung, und `/h` unterdrückt die normalerweise erscheinende Dialognachricht der drahtlosen Verbindung, wenn Dateien gesendet werden.

---

netstat

`netstat [`*`Optionen`***`] [`***`n`***`]`**

Zeigt TCP/IP-Statistiken an. Falls *n* angegeben wurde, wird die Anzeige alle *n* Sekunden aktualisiert. Standardmäßig werden die aktuell offenen Verbindungen angezeigt.

Optionen

`-a`

Zeigt auch serverseitige Verbindungen an.

`-b`

Zeigt die am Erstellen der Verbindung beteiligten ausführbaren Dateien an.

`-c`

Zeigt eine nach Anzahl der aktuell genutzten TCP- und UDP-Ports sortierte Liste von Prozessen an.

`-d`

Zeigt den jeder Verbindung zugeordneten DSCP-Wert an (DSCP, *Differentiated Services Code Point*, für die Prioritätsklassifizierung in

QoS verwendet).

`-e`

Zeigt Ethernet-Statistiken an.

`-f`

Zeigt vollqualifizierte Domänennamen für Remoteadressen an.

`-i`

Zeigt die Zeitdauer einer TCP-Verbindung im aktuellen Zustand an.

`-n`

Zeigt numerische IP-Adressen und Portnummern an.

`-o`

Zeigt die mit jeder Verbindung verknüpfte übergeordnete Prozesskennung an.

`-p` *`Protokoll`*

Erlaubt die Auswahl des gewünschten Netzwerkprotokolls: `TCP`, `UDP`, `TCPv6`, `UDPv6`. In Verbindung mit `-s` werden zusätzlich unterstützt: `IP`, `IPv6`, `ICMP`, `ICMPv6`.

`-r`

Zeigt die Routingtabelle an.

`-s`

Zeigt Statistiken protokollweise an. Einzelne Protokolle können mit `-p` ausgewählt werden.

`-t`

Zeigt den aktuellen Abladungsstatus (»Offload-Status«) der Verbindung an.

`-x`

Zeigt Verbindungen, Listener und freigegebene Endpunkte für *NetworkDirect* an.

`-y`

Zeigt die TCP-Verbindungsvorlage für alle Verbindungen an. Kann nicht mit anderen Optionen kombiniert werden.

PoSh: Mit dem Cmdlet `get-nettcpconnection` können Sie den Status von TCP/IP-Netzwerkverbindungen unter PowerShell abfragen.

nslookup

nslookup [*Name* | *IP-Adresse* | **-] [***DNS-Server***]**

Löst den teilweise oder vollqualifizierten Namen bzw. die IP-Adresse auf dem primären oder angegebenen DNS-Server auf. `nslookup` startet im interaktiven Modus, wenn keine Argumente angegeben wurden. In diesem Modus erhalten Sie durch Eingabe von `help` Informationen zu weiteren Optionen. Die Eingabe von `exit` beendet den interaktiven Modus und kehrt zurück zur aufrufenden Instanz.

Hinweis: Eine Auflösung von IP-Adressen zu Namen erfolgt nur, wenn auf dem zuständigen DNS-Server eine Reverse-Lookup-Zone eingerichtet ist.

PoSh: Das PowerShell-Äquivalent zu `nslookup` ist das Cmdlet `resolvednsname`.

pathping

pathping [*Optionen***]** ***Ziel***

Zeigt die Route zum Ziel und berechnet Statistiken für jeden Abschnitt.

Optionen

-g *Rechnerliste*

Loose Source Routing gemäß Rechnerliste.

-h *Abschnitte*

Setzt einen Maximalwert für *Abschnitte* (Hops). Der Standardwert ist 30.

-i *Adresse*

Verwendet die angegebene Quelladresse.

-n

Löst die IP-Adressen nicht in Rechnernamen auf.

-p *Millisekunden*

Setzt die Wartezeit zwischen zwei aufeinanderfolgenden Pings. Der Standardwert ist 250.

-q *Abfrageanzahl*

Setzt die Anzahl der Abfragen zu jedem Hop. Der Standardwert ist 100.

-w *Millisekunden*

Setzt das Zeitlimit für jede Antwort. Der Standardwert ist 3.000.

-4

Erzwingt die Verwendung von IPv4.

-6

Erzwingt die Verwendung von IPv6.

---

ping

ping [*Optionen***]** *Ziel*

Es wird ein Ping auf das angegebene System ausgeführt, um die Erreichbarkeit über das Netzwerk zu prüfen. Das Ziel kann dabei per Rechnername oder IP-Adresse angesprochen werden.

Optionen

-t | -n *Zähler*

Der Ping läuft so lange, bis er manuell mit der Tastenkombination *Strg+C* abgebrochen wird (`-t`), bzw. so oft, wie unter *`Zähler`* angegeben (`-n`). Der Standardwert ist `4`.

-l *Größe*

Setzt die Größe des Sendepuffers in Byte. Der Standardwert ist 32 Byte, der Maximalwert beträgt 65.527.

-a

Löst eine IP-Adresse in einen Rechnernamen auf.

-c *Bezeichner*

Gibt den Routingdepotbezeichner an.

-f

Legt das Kennzeichen »Nicht fragmentieren« im Paket für IPv4 fest.

-i *TTL*

Setzt die Gültigkeitsdauer (TTL, *Time To Live*).

-w

Setzt das Zeitlimit für die Rückmeldung in Millisekunden.

-r *n* **[-s** *Zeitstempel***]**

Zeichnet die Route für bis zu *n* Hops auf. *n* kann zwischen 1 und 9 liegen. Des Weiteren kann ein optionaler Zeitstempel angegeben werden.

-j | -k *Hostliste*

Routet die IPv4-Pakete über die in der Liste angegebenen Hosts. Dabei kann angegeben werden, ob es erlaubt oder verboten sein soll, die Pakete über unterschiedliche Gateways zu leiten (*Loose Source Route* mit `-j` und *Strict Source Route* mit `-k`). Bis zu neun Hosts können in dieser Liste aufgeführt werden.

-4

Erzwingt die Verwendung von IPv4.

-6

Erzwingt die Verwendung von IPv6.

-S *Quelladresse*

Verwendet bei Computern mit mehreren Netzwerkadaptern oder mehreren IP-Adressen die angegebene Quelladresse.

PoSh: Mit dem Cmdlet `test-connection` können Sie Netzwerkverbindungen prüfen und dessen erweiterte Möglichkeiten für skriptbasierte Auswertungen zum Einsatz bringen.

---

route

route [*Optionen***] [*****Befehl*** **[***Ziel***] [mask** *Subnetzmaske***] [***Gateway***] ^[metric** *Hops***] if** *Schnittstelle***]]**

Zeigt die TCP/IP-Routingtabelle an und erlaubt Veränderungen. ***Befehl*** kann einen der folgenden Werte annehmen:

`print`
: Zeigt alle oder durch Optionen angegebene Routen an.

`add`
: Ergänzt die angegebene Route.

`delete`
: Entfernt die angegebene Route.

`change`
: Verändert das Gateway und/oder die Metrik der angegebenen Route.

Der Standardwert für *`Subnetzmaske`* ist 255.255.255.255 und für *`Hops`* 1.

Optionen

`-f`
: Entfernt alle Gateway-Eintragungen aus der Routingtabelle.

`-p`
: Kann mit `add` verwendet werden, um dauerhafte Routen zu definieren. Diese Routen stehen auch nach einem Neustart noch zur Verfügung.

`-4`
: Erzwingt die Verwendung von IPv4.

`-6`
: Erzwingt die Verwendung von IPv6.

PoSh: PowerShell bietet mehrere Cmdlets, um Routingtabellen anzuzeigen und zu bearbeiten. Diese beinhalten den Namensbestandteil `netroute`. Die IP4-Routingtabelle lesen Sie beispielsweise mit folgendem Befehl aus:

```
get-netroute -AddressFamily IPv4
```

---

telnet

`telnet [-a][-e` *`Escapezeichen`*`] [-f` *`Protokolldatei`*`][-l` *`Benutzer`*`] [-t` *`Begriff`*`] [`*`Rechner`* | *`IP-Adresse`* `[`*`Port`*`]]`

Startet eine interaktive Sitzung mit einem entfernten Rechner. Der entfernte

Rechner muss den Telnet-Dienst anbieten.

-a

Automatische Anmeldung mit dem Namen des angemeldeten Benutzers.

-e *Escapezeichen*

Escapezeichen zur Eingabe von Telnet-Clientbefehlen.

-f *Protokolldatei*

Dateiname zur Protokollierung des Clients.

-l *Benutzername*

Legt den Anmeldebenutzernamen für das Remotesystem fest, das dafür *telnet environ* unterstützen muss.

-t *Terminaltyp*

Legt die Verwendung eines der Terminaltypen *vt100, vt52, ansi* oder *vtnt* fest.

Rechner | *IP-Adresse*

Gibt an, mit welchem Telnet-Server sich der Client verbinden soll.

Port

Legt die Portnummer oder den Dienstnamen fest.

Hinweis: Der Telnet-Client ist standardmäßig nicht installiert, sondern muss über *Einstellungen* → *System* → *Optionale Features* → *Mehr Windows-Funktionen* und Auswahl des Features *Telnet Client* installiert werden.

---

tftp

tftp [-i] *Rechner* **{get | put}** *Datei* **[***Ziel***]**

Führt einen Dateitransfer auf der Basis von Trivial-FTP aus. Die Option `-i` gibt an, dass es sich um eine binäre Datei handelt; der Transfer von ASCII-Dateien ist die Standardeinstellung. Die Schlüsselwörter `put` und `get` definieren entweder einen Transfer von lokal nach entfernt oder umgekehrt. Das Argument *Ziel* ist optional und gibt das Ziel der zu übertragenden Datei an. Wird anstelle von *Datei* ein Bindestrich angegeben, wird die Eingabe

bzw. Ausgabe von der Standardeingabe bzw. Standardausgabe gelesen.

Hinweis: Der TFTP-Client ist standardmäßig nicht installiert, sondern muss über *Einstellungen* → *System* → *Optionale Features* → *Mehr Windows-Funktionen* und Auswahl des Features *TFTP-Client* installiert werden.

---

tracert

`tracert` *`Rechner | IP-Adresse`*

Zeigt die Route zum angegebenen Ziel auf.

Optionen

`-d`
: Die IP-Adressen werden nicht in Rechnernamen aufgelöst.

`-h` *`n`*
: Setzt die maximale Anzahl von Hops, die zur Zielsuche verwendet werden dürfen.

`-j` *`Hostliste`*
: *Loose Source Route* gemäß Hostliste bei IPv4.

`-w` *`n`*
: Setzt das Zeitlimit für eine Antwort in Millisekunden.

`-R`
: Verfolgt bei IPv6 auch den Pfad zurück.

`-S` *`Quelladresse`*
: Die bei IPv6 zu verwendende Quelladresse.

`-4`
: Erzwingt die Verwendung von IPv4.

`-6`
: Erzwingt die Verwendung von IPv6.

PoSh: Um die verwendete Route in PowerShell anzuzeigen, können Sie das

Cmdlet `test-netconnection` verwenden. Ein kurzes Beispiel:

```
tnc www.microsoft.de -traceroute
```

Microsoft-Netzwerk

---

nbtstat

nbtstat [*Optionen*] [*n*]

Zeigt Namenstabellen und aktuelle Verbindungen von NBT (NetBIOS über TCP/IP) an. Falls angegeben, wird die Anzeige alle *n* Sekunden aktualisiert.

Optionen

-a *Computername* **| -A** *IP-Adresse*
: Zeigt die Namenstabelle des angegebenen Computers an.

-c
: Zeigt den Inhalt des Remotenamenscaches mit IP-Adressen an.

-n
: Zeigt lokale NetBIOS-Namenszuordnungen an.

-r
: Zeigt Namen an, die über WINS oder Broadcasts aufgelöst wurden.

-S | -s
: Zeigt tabellarisch die offenen Sitzungen an. Entfernte Rechner werden entweder durch ihre IP-Adresse (`-S`) oder ihren Rechnernamen (`-s`) kenntlich gemacht.

-R
: Löscht die Remotecache-Namenstabelle und legt sie neu an.

-RR
: Gibt registrierte Namen im WINS frei und registriert sie anschließend wieder.

n

Das Intervall in Sekunden, nach dem die ausgewählte Statistik erneut angezeigt wird. Mit *Strg+C* wird die Intervallanzeige beendet.

---

net computer

`net computer \\`*`Computername`* **`{/add | /del}`

Fügt den angegebenen Computer der Domäne hinzu (`/add`) oder entfernt ihn (`/del`) aus der Domäne.

PoSh: Mit dem Befehl

`Add-Computer -DomainName` *`domaene.loc`* `-Restart`

fügen Sie ein System der Domäne *`domaene.loc`* hinzu und starten ihn nachfolgend neu, um die Änderung umzusetzen. Die Domäne muss bereits existieren, und die Domänencontroller müssen über DNS aufgelöst werden. Zum Entfernen eines Rechners aus der Domäne bietet PowerShell das Cmdlet `remove-computer`.

---

net config

`net config {server | workstation} [`*`Optionen`***`]`**

Zeigt Informationen über die Dienstkonfiguration des Serverdiensts (`server`) oder des Arbeitsplatzdiensts (`workstation`) an bzw. ändert diese. In den aktuellen Windows-Versionen erlaubt `net config workstation` keine Änderungen mehr.

Optionen von net config server:

`/autodisconnect:`*`Minuten`*

Trennt inaktive Serververbindungen nach *`Minuten`*. Der Standardwert beträgt 15 Minuten, gültige Werte liegen zwischen –1 (abschalten) und 65.535.

`/srvcomment:`*`Text`*

Fügt einen Kommentar dem Server hinzu. Die maximale Länge beträgt 48 Zeichen.

`/hidden:{`*`yes`*` | `*`no`*`}`

`yes` entfernt den Eintrag aus der Serverliste. Der Standardwert ist `no`.

PoSh: Der Funktionsumfang der Cmdlets *`get-smbserverconfiguration`* und `set-smbserverconfiguration` deckt eine Vielzahl der Einstellungen eines SMB-Servers ab.

---

net file

`net file [`*`id`***` [/close]]`**

Ohne Optionen werden alle Dateien mit ID-Nummern angezeigt, die von entfernten Computern aus geöffnet sind. Wenn `id` angegeben wurde, werden nur die Informationen zur entsprechenden Datei ausgegeben. Wurde zusätzlich `/close` angegeben, wird die Datei mit der angegebenen ID geschlossen. Alternativ kann der Befehl `openfiles` eingesetzt werden.

PoSh: Mit den Cmdlets `get-smbopenfile` und `close-smbopenfile` lassen sich über Netzwerkfreigaben geöffnete Dateien anzeigen und schließen.

---

net session

`net session [\\`*`Rechner`***`] [/delete] [/list]`**

`/list` zeigt alle eingehenden Verbindungen zum lokalen System oder zu *`Rechner`* an. Gibt man keine Parameter an, werden die eingehenden Verbindungen des lokalen Systems ebenfalls angezeigt.

`/delete` schließt die angegebene Sitzung, falls ein Rechner ausgewählt wurde. Andernfalls werden alle Sitzungen geschlossen, inklusive aller offenen Dateien.

net share

net share *Freigabe*[=*Pfad*] [*Optionen*]

Gibt ein Verzeichnis für den Netzzugriff frei oder löscht eine Freigabe. *Pfad* wird nur bei Erstellung einer neuen Freigabe angegeben.

Ohne Angabe von Optionen werden alle vorhandenen Freigaben angezeigt. Wenn eine Freigabe ohne weitere Optionen angegeben wurde, werden Informationen über diese Freigabe ausgegeben.

Optionen

/grant:*Benutzer*,*Berechtigung*

Standardmäßig wird auf neue Freigaben die Berechtigung *Jeder/Lesen* gesetzt. Mit dieser Option können statt dieses Standards andere Berechtigungen vergeben werden: `read`, `change` oder `full`.

/users:*n*

Stellt die Anzahl der Benutzer ein, die gleichzeitig auf die Freigabe zugreifen dürfen.

/unlimited

Die Freigabe kann von beliebig vielen Anwendern gleichzeitig genutzt werden.

/remark:"*Text***"**

Fügt der Freigabe eine Beschreibung hinzu.

/cache:*Typ*

Konfiguriert den Typ des clientseitigen Caches: deaktiviert (`none`), manuell (`manual`), nur Dokumente automatisch (`documents`), Programme und Dokumente automatisch (`programs`) oder Cache für eine Niederlassung (`branchcache`).

/delete

Löscht die angegebene Freigabe. Name, Pfad oder Druckername der zu löschenden Freigabe kann angegeben werden.

PoSh: Freigaben lassen sich durch PowerShell-Cmdlets mit dem

Namensbestandteil `SMBShare` verwalten.

So erstellen Sie mit dem Befehl

```
new-smbshare -name Test$ -Path "C:\Test" -fullaccess
Jeder
```

eine versteckte Freigabe *`Test$`* des mittels `–path` angegebenen Ordners und erteilen der Gruppe *Jeder* Vollzugriff auf die Freigabe.

---

net statistics

`net statistics workstation`

Zeigt Statistikinformationen des Arbeitsstationsdiensts an.

Hinweis: Der Aufruf `net statistics server` ist in den aktuellen Betriebssystemversionen nicht mehr enthalten.

---

net time

`net time` *`Quelle`* **`[/set]`**

Zeigt die Zeit eines bestimmten Computers an. Der Parameter `/set` synchronisiert die lokale Uhrzeit mit der von *`Quelle`*. Das Argument *`Quelle`* wird als *\\Computername, /domain:Name* oder */rtsdomain: Name* angegeben. Letzteres spezifiziert einen vertrauenswürdigen Zeitserver in der Domäne *Name*. Dieser Befehl wurde durch `w32tm` abgelöst.

PoSh: Hier gibt es lediglich die Möglichkeit, die entsprechenden Registrierungseinträge direkt zu setzen. Beispielsweise legen Sie den Zeitserver fest mittels:

```
Set-ItemProperty -Path
HKLM:\SYSTEM\CurrentControlSet\Services\W32Time\Parameters
-Name NtpServer -Value "de.pool.ntp.org,0x8"
```

Tipp: Vergessen Sie nicht, nach entsprechenden Änderungen den Dienst `w32time` neu zu starten.

net use

net use [*Gerät***: | *] [***Computer******Freigabe***] [***Kennwort* | *] [/user:***Benutzer***] [***Optionen***]**

Ordnet eine Netzwerkressource einem lokalen Gerät zu. Dabei kann es sich um einen Laufwerkbuchstaben oder einen Druckeranschluss (LPT*n*) handeln. Wird anstelle von ***Gerät*** ein Sternchen (*) angegeben, wird das nächste verfügbare Gerät verwendet (i.d.R. der nächste freie Laufwerkbuchstabe).

Computer und ***Freigabe*** werden nur bei der Definition von neuen Zuordnungen angegeben.

Ein benötigtes Kennwort kann in der Befehlszeile angegeben werden. Wenn stattdessen ein Sternchen (*) eingegeben wurde, wird es an der Eingabeaufforderung abgefragt. Ohne Angabe von ***Benutzer*** und ***Kennwort*** werden die Anmeldeinformationen des aktuell angemeldeten Benutzers verwendet. Der Benutzername kann klassisch in der Syntax *{Domäne | Rechnername}\Benutzername* oder in der modernen Form als *Benutzername@Domänenname.xyz* angegeben werden.

Optionen

/persistent:{*yes* | *no*}

Erstellt eine ständige Verbindung, die nach jeder Anmeldung wiederhergestellt wird. Der Standardwert ist die letzte verwendete Einstellung. Wird diese Option ohne weitere Argumente angegeben, wird der aktuelle Standardwert verändert.

/home

Ordnet den angegebenen Laufwerkbuchstaben dem Basisverzeichnis des Benutzers zu. Die Angabe einer Ressource ist nicht notwendig.

/blockntlm

Die Verwendung der NTLM-Authentifizierung beim Aufbau einer Netzwerkverbindung wird unterbunden.

/delete

Die angegebene Gerätezuordnung wird dauerhaft gelöscht.

/global

Die Verbindung wird für alle Benutzer des lokalen Systems auf- oder abgebaut.

/netonly

Die Anmeldeinformationen werden nur für Netzwerkverbindungen verwendet.

/savecred

Speichert die angegebenen Anmeldeinformationen. Bei der nächsten Herstellung der Verbindung zur selben Freigabe auf demselben Server müssen sie nicht erneut eingegeben werden.

/smartcard

Gibt an, dass zur Herstellung der Verbindung eine Smartcard verwendet wird.

/requestcompression:{*yes* | *no*}

Fordert die automatische Komprimierung von über das verbundene Laufwerk übertragenen Daten an.

/requireintegrity {*0* | *1*}

Zwischen Client und Server werden bei Verwendung des Werts *1* Prüfungen erzwungen, die eine Manipulation von Daten während der Übertragung verhindern sollen.

/requireprivacy {*0* | *1*}

Die Übertragung von Daten ist bei Verwendung des Werts *1* in einer Form zu verschlüsseln, die es einem lauschenden Drittsystem unmöglich macht, Daten abzugreifen.

/transport:{*tcp* | *quic* } [/skipcertcheck]

Ermöglicht, vom klassischen Verbindungsverfahren (*tcp*) zu einer sicheren Verbindung (*quic*) über einen TLS-1.3-verschlüsselten Tunnel über UDP-Port 443 zu wechseln, sofern die entsprechenden Voraussetzungen geschaffen wurden. Serverseitig ist Windows Server 2025 in allen Editionen dafür vorbereitet, bei Windows Server 2022 beschränkte sich die Unterstützung noch auf die Azure Edition.

/writethrough

Die Datenübertragung erfolgt unter Umgehung von Cachemechanismen

der beteiligten Betriebssysteme.

Mit den Parametern */tcpport*, */quicport* und */rdmaport* legen Sie einen abweichenden Port für die Kommunikation fest.

PoSh: Um ein Netzlaufwerk mittels PowerShell zu verbinden, können Sie das Cmdlet `new-smbmapping` verwenden. Beachten Sie, dass ein solcher Aufruf mit den normalen Rechten des Benutzers ausgeführt werden muss, um das Laufwerk in Windows-Explorer und Eingabeaufforderung des Benutzers verfügbar zu machen. Und auch dann müssen Sie im Gegensatz zur Verwendung von `net use` nach dem ersten Aufruf den Dateiexplorer in der Regel neu starten, damit das verbundene Laufwerk dort sichtbar wird. Das folgende Beispiel verbindet *Freigabe* auf *Server* mit dem Laufwerkbuchstaben *V:*

```
new-smbmapping –localpath V: -remotepath
"\\server\freigabe" –persistent $true
```

---

net view

net view [*Computername* **[***/cache***] |[***/all***] |**
*/domain***[***:Domäne***]]**

Zeigt die Namen von Computern in einer Domäne oder einem Netzwerk an. Es können auch die Namen freigegebener Ressourcen auf einem entfernten Computer angezeigt werden. Folgende Befehlsoptionen sind möglich:

Computername

Der Name eines Computers, dessen Freigaben angezeigt werden sollen.

/domain:*Domäne*

Eine Domäne, deren Mitglieder angezeigt werden sollen. Ohne Angabe einer Domäne werden alle Domänen des lokalen Netzes angezeigt.

Ohne Optionen werden alle Computer in der lokalen Domäne angezeigt.

PoSh: Vergleichbare Funktionen zur Anzeige der Freigaben eines Computers bietet das Cmdlet `get-smbshare`.

netcfg

netcfg [Optionen**]**

Installiert Netzwerkkomponenten (Protokolle, Dienste und Clients) unter Verwendung der zugehörigen INF-Dateien.

Optionen

-c *{p | s | c}*

Gibt an, ob es sich bei der Komponente um ein Netzwerkprotokoll (p), einen Dienst (s) oder einen Client (c) handelt.

-l "*Pfad zur INF-Datei*"

Gibt den vollständigen Pfad zur INF-Datei mit den Informationen zur zu installierenden Komponente an.

-{b | i | q | u} *Komponenten-ID*

Installiert eine Netzwerkkomponente (i), deinstalliert sie (u) oder zeigt ihre Bindungen (b) oder ihren Installationsstatus (q) an.

-winpe

Installiert TCP, NetBIOS und Microsoft-Client unter Windows PE.

-v

Zeigt ausführliche Informationen an.

PoSh: PowerShell verfügt über zahlreiche spezialisierte Cmdlets zur Konfiguration von Netzwerkkomponenten. Vielen davon ist der Namensbestandteil `Net` gemeinsam.

openfiles

Zeigt (lokal oder über das Netzwerk) geöffnete Dateien an oder trennt sie.

openfiles /disconnect [/s *Computer* **[/u** *Benutzer* **[/p [***Kennwort***]]]] [/op** *Dateiname***] /{id** *DateiID* **| /a** *Benutzer* **| /o** *Modus***}**

Trennt geöffnete Dateien auf *Computer* anhand ihrer ID, des Benutzers, des Öffnungsmodus (`read`, `write`, `read/write`) oder des Dateinamens.

Für jeden Parameter kann ein Stern (*) als Wildcard angegeben werden.

openfiles /query [/s *Computer* **[/u** *Benutzer* **[/p [***Kennwort***]]]] [/fo {table | list | csv}] [/nh] [/v]**

Zeigt geöffnete Dateien an. Das Ausgabeformat kann tabellarisch, in Listenform oder im CSV-Format sein. `/nh` unterdrückt die Ausgabe von Spaltenüberschriften, `/v` zeigt ausführliche Informationen an.

openfiles /local [on | off]

Aktiviert bzw. deaktiviert das globale Systemflag *Maintain Objects List*, das zur Anzeige lokal geöffneter Dateien aktiviert sein muss.

PoSh: Mit den Cmdlets `get-smbopenfile` und `close-smbopenfile` werden die Kernfunktionalitäten von `openfiles` bezogen auf über das Netzwerk geöffnete Dateien in PowerShell umgesetzt.

---

rasdial

rasdial *Eintrag***[***Benutzer* **{***Kennwort* **| *} [/domain:***Domäne***]] [***Optionen***]**
rasdial [*Eintrag***] /Disconnect**

In der Syntax der ersten Zeile wählt der Befehl über den RAS-Dienst einen Eintrag aus dem Telefonbuch an, und in der zweiten Zeile beendet er eine bestehende Verbindung. Wird anstelle eines Kennworts ein Stern (*) angegeben, wird bei Ausführung danach gefragt.

Optionen

/Phone:*Nummer*

Wählt die Telefonnummer.

/PhoneBook:*Datei*

Verwendet das angegebene Telefonbuch, dann das Standardtelefonbuch (*%SystemRoot%\System32\RAS\Benutzername.pbk*).

/CallBack:*Nummer*

Definiert eine Rückrufnummer.

/PrefixSuffix

Verwendet die TAPI-Wählregeln.

Administration von Microsoft-Netzwerkdiensten

---

dcpromo

dcpromo [/answer[:*Dateiname*] | /unattend[:*Dateiname*] | /unattend | /adv] /uninstallBinaries [/CreateDCAccount | /UseExistingAccount:Attach] | /ForceRemoval | [/?[: {Promotion | CreateDcAccount | UseExistingAccount | Demotion | ForceRemoval}]]

`dcpromo` war einst bekannt dafür, die GUI-basierte Einrichtung oder Herabstufung von Active-Directory-Domänencontrollern zu starten. Diese Funktionalität wurde eingestellt, jedoch funktioniert die Ausführung in der Eingabeaufforderung mit den entsprechenden Parametern weiterhin. Da seit mehreren Serverversionen Microsoft PowerShell den Vorzug gibt, werden an dieser Stelle nur die grundlegenden Parameter aufgeführt.

/answer[:Dateiname]

Gibt eine Antwortdatei mit den zu nutzenden Installationsparametern und -werten an.

/unattend[:Dateiname]

Arbeitet analog dem Parameter `/answer`.

/adv

Aktiviert erweiterte Benutzeroptionen.

/UninstallBinaries

Deinstalliert die Dateien der AD-Domänendienste.

/CreateDCAccount |/UseExistingAccount:Attach

Erstellt ein Konto für einen schreibgeschützten Domänencontroller bzw. verbindet einen Server mit einem vorhandenen RODC-Konto.

/ForceRemoval

Erzwingt die Deinstallation der Active-Directory-Domänendienste auf diesem Domänencontroller. Das Konto für den Domänencontroller wird nicht im Verzeichnis gelöscht. Änderungen, die auf dem Domänencontroller seit der letzten Replikation mit einem Partner vorgenommen wurden, gehen verloren.

/?[:{Promotion | CreateDcAccount | UseExistingAccount | Demotion | ForceRemoval}]

Zeigt die bei Aufruf von `/Promotion`, `/CreateDCAccount`, `/UseExistingAccount`, `/Demotion` oder `/ForceRemoval` nutzbaren aufgabenspezifischen Parameter an.

Eine Domänencontrollerinstallation lässt sich auch direkt ohne Konfigurationsdatei ausführen, wie im folgenden Beispiel für einen neuen Domänencontroller in der neuen Domäne *testdomaene.de* innerhalb einer neuen Gesamtstruktur demonstriert (der folgende Befehl ist in einer einzelnen Zeile einzugeben, anstelle des Zeilenumbruchs nutzen Sie bitte ein Leerzeichen):

```
dcpromo /unattend /InstallDns:yes
/replicaOrNewDomain:domain /newdomain:forest
/newDomainDnsName:testdomaene.de
/DomainNetbiosName:testdomaene
/databasePath:"c:\ntds" /logPath:"c:\ntdslogs"
/sysvolpath:"c:\sysvol"
/safeModeAdminPassword:TeSt25MPasswort!
/forestLevel:10 /domainLevel:10
/rebootOnCompletion:yes
```

PoSh: Zur Installation der Active-Directory-Domänendienste und der Einrichtung eines Domänencontrollers existieren verschiedene Cmdlets, unter anderem `Install-ADDSDomain`, `Install-ADDSDomainController` und `Install-ADDSForest`.

---

dnscmd

dnscmd [*Server***]** */Hauptoption Weitere_Argumente*

Verwaltet einen DNS-Server. Das Format des Befehls variiert je nach Funktion des Servers. *Server* gibt den anzusprechenden DNS-Server an; standardmäßig wird der aktuelle DNS-Server verwendet. Ein Punkt anstelle des Servernamens verweist auf das lokale System. *Hauptoption* gibt die durchzuführende Aktion an. Der Befehl ist nur verfügbar, wenn das optionale Feature *RSAT: DNS-Servertools* hinzugefügt wurde. Im Folgenden sehen Sie die wichtigsten Befehle im Überblick.

Optionen

/info *[Eigenschaft]*

Zeigt grundsätzliche Informationen über den angegebenen DNS-Server an, gegebenenfalls begrenzt auf die angegebenen Eigenschaften. Die Liste der Eigenschaften finden Sie in der Hilfe des Befehls.

/statistics *[Filtermaske]* **[/clear]**

Zeigt die Statistiken des DNS-Servers an. Die Ausgabe kann über eine Maske gefiltert werden (die Komponenten der Maske sind in der Hilfe des Befehls definiert). /Clear setzt alle Zähler auf null, die diese Option unterstützen.

/enumzones [/primary | /secondary | /forwarder | /stub | /cache | /auto-created] [/forward | /reverse | /ds | /file] [/domaindirectorypartition | /forestdirectorypartition | /customdirectorypartition | /legacydirectorypartition | /directorypartition *partitionFQDN***]**

Zählt die Zonen eines DNS-Servers auf. Die weiteren Parameter beschränken die Liste der DNS-Server auf die im Filter angegebenen Zonen.

/enumrecords *Zone Knoten [Weitere-Optionen]*

Listet alle Einträge der angegebenen DNS-Subdomäne auf. *Zone* gibt die gewünschte Zone an, und *Knoten* definiert innerhalb der Zone den Startpunkt der Ausgabe. *Knoten* muss entweder @ (Root der Zone), ein FQDN (*Fully Qualified Domain Name*) eines Knotens in der Zone oder ein einfacher Name sein, der relativ zur Zone interpretiert wird. Weitere Optionen erlauben die Beschränkung der auszugebenden Einträge. Nähere Informationen sowie die komplette Liste erhalten Sie in der Hilfe

des Befehls.

/config [*Zone***]** *Eigenschaft hex-wert*

Setzt verschiedene Parameter des DNS-Servers, entweder für die angegebene Zone oder für alle Zonen. Sie finden alle Eigenschaften in der Hilfe des Befehls.

/clearcache

Löscht den Cache des angegebenen DNS-Servers.

/ipvalidate context [*Zone***] [***IP-Adresse***]**

Führt verschiedene Tests der bekannten DNS-Server durch: andere DNS-Server (/dnsservers), Forwarder (/forwarders), Root Hints (/roothints) und primäre DNS-Server einer Zone (/zone-masters *Zone*).

/resetlistenaddresses *[IP-Adresse]*

Definiert oder löscht die Liste der Rechner, die DNS-Anfragen bedienen.

/resetforwarders *[IP-Adresse]* **[/slave] [/timeout** *Sek***]**

Setzt oder löscht die Liste der DNS-Forwarder. /slave definiert *Server* als Slave-DNS-Server, und *Sek* gibt die Zeitüberschreitungsdauer in Sekunden an (Standardwert ist 5).

/restart

Startet den angegebenen DNS-Server neu.

/startscavenging

Aktiviert den Aufräumprozess (*Scavenging*). Dieser Prozess sucht nach veralteten DNS-Einträgen und löscht sie. Der Aufräumprozess ist standardmäßig nicht aktiviert.

/writebackfiles [*Zone***]**

Schreibt alle Hinweise auf das Stammverzeichnis oder die Zonendatendateien eines DNS-Servers.

/ageallrecords *Zone Knoten* **[/tree] [/f]**

Aktiviert das Altern der Zoneneinträge ab dem angegebenen Knoten. Zum Aktivieren des Alterns für die komplette Subdomäne verwenden Sie /tree. /f unterdrückt Bestätigungsmeldungen.

/recordadd *Zone Name* **[/aging] [ttl]** *Typ* **Daten**

Fügt einen Ressourceneintrag zur angegebenen Zone des angegebenen DNS-Servers hinzu. *Typ* ist der DNS-Datensatztyp, *Name* sind die primären Daten des Typs, und unter *Daten* sind verschiedene weitere Informationen definiert, die für diesen Typ von Eintrag benötigt werden. Ein Beispiel anhand eines Eintrags: *Name* ist der Rechnername, und *Daten* wäre die IP-Adresse des Rechners. /aging aktiviert das Altern des Eintrags, ist aber standardmäßig deaktiviert. ttl ist die Lebenszeit (*time to live*) des Eintrags. Standardmäßig entspricht dieser Wert dem Wert im SOA-Eintrag.

/recorddelete *Zone Typ Daten* **[/f]**

Löscht den angegebenen DNS-Eintrag. /f unterdrückt die Bestätigungsmeldung.

/nodedelete *Zone Knoten* **[tree] [/f]**

Löscht alle DNS-Einträge auf dem angesprochenen Knoten. /f unterdrückt die Bestätigungsmeldungen, und /tree löscht die komplette Subdomäne, beginnend ab diesem Knoten.

/zoneinfo *Zone [Eigenschaften]*

Zeigt Informationen über die angegebene DNS-Zone an. Die Ausgabe kann auf *Eigenschaften* beschränkt werden. Eine vollständige Liste der Eigenschaften finden Sie in der Hilfe des Befehls.

/zoneexport *Zone Datei*

Schreibt alle Records einer Zone in eine Datei.

/zoneprint *Zone* **[/detail]**

Listet alle Records einer Zone (gegebenenfalls im Detail) auf.

/zoneadd *Zone* **/primary /file** *Datei* **[/load] [/a Admin]**
/zoneadd *Zone* **/secondary Primäre-IPs [/File** *Datei* **[/Load]]**
/zoneadd *Zone* **/dsprimary**

Erstellt eine neue DNS-Zone vom angegebenen Typ. /File definiert die Zonendatei, und /Load gibt an, dass bestehende Zonendaten daraus gelesen werden sollen. Folgende Zonentypen werden unterstützt: /dsprimary für eine integrierte Active-Directory-Zone, /primary für eine primäre Standardzone, /secondary für eine sekundäre Standardzone, /sub für eine dateigestützte Stubzone, /dsstab für eine

integrierte Active-Directory-Stubzone, `/forwarder` zur Weiterleitung nicht aufgelöster Abfragen an einen anderen DNS-Server und `/dsforwarder` für die Weiterleitung von durch die erstellte integrierte Active-Directory-Zone nicht auflösbaren Abfragen an einen weiteren DNS-Server.

`/zonedelete` *`Zone`***`[/dsdel] [/f]`**

Löscht die angegebene Zone. Geben Sie `/dsdel` an, falls die Zone eine Active-Directory-integrierte Zone ist. `/f` unterdrückt die Bestätigungsmeldung.

`/zonerefresh` *`Zone`*

Erzwingt eine sofortige Aktualisierung der angegebenen Zone vom primären DNS-Server.

`/zonereload` *`Zone`*

Lädt auf dem DNS-Server die angegebene Zone neu, entweder aus der Datei oder aus dem Active Directory.

`/zoneupdatefromds` *`Zone`*

Aktualisiert auf dem DNS-Server eine Active-Directory-integrierte Zone.

`/zonepause` *`Zone`*

Hält die Zone auf dem DNS-Server an.

`/zoneresume` *`Zone`*

Reaktiviert die angehaltene Zone auf dem DNS-Server.

`/zoneresetscavengeservers` *`Zone [IP-Adresse]`*

Definiert bzw. setzt eine Liste der DNS-Server zurück, die den Aufräumprozess durchführen.

`/zoneresetsecondaries` *`Zone`* **`[/secure]`** *`[IP-Adressen]`*

Definiert oder setzt die Benachrichtigungsliste der sekundären DNS-Server der angegebenen Zone zurück. `/secure` beschränkt den Zugriff auf die aufgeführten sekundären DNS-Server.

`/zoneresettype` *`Zone`* **`/primary /file`** *`Datei`* **`[/a Admin]`** *`[Optionen]`*
`/zoneresettype` *`Zone`* **`/secondary`** *`Primäre-IPs [/file Datei]`*
`/zoneresettype` *`Zone`* **`/dsprimary`** *`[Optionen]`*

Ändert den Typ des DNS-Servers der angegebenen Zone. Dabei werden die Optionen verwendet, die bereits unter `/ZoneAdd` erklärt wurden.

Weitere Optionen sind `/Overwrite_Mem`, um die Zonendaten im Speicher des DNS-Servers mit denen aus dem Active Directory zu überschreiben, und `/Overwrite_DS`, um die Zonendaten im Active Directory mit den Zonendaten aus dem Speicher des DNS-Servers zu überschreiben. Weitere Zonentypen sind beim Parameter `/zoneadd` benannt.

`/zonewriteback` *`Zone`*

Schreibt auf dem DNS-Server die Zonendaten zurück in die Datei.

`/enumdirectorypartitions`

Zeigt die Partitionen des DNS-Applikationsverzeichnisses an.

`/directorypartitionInfo` *`FQDN`* **`[/Detail]`**

Zeigt (gegebenenfalls detaillierte) Informationen über eine DNS-Verzeichnispartition an, die als *Fully Qualified Domain Name* angegeben wird.

`/createdirectorypartition` *`FQDN`*

Erzeugt eine neue DNS-Verzeichnispartition.

`/createbuiltindirectorypartition [/forest][/alldomains]`

Erzeugt die Standard-DNS-Verzeichnispartitionen für die Domäne (Name: *DomainDnsZones*), die Gesamtstruktur (`/forest`, Name: *ForestDnsZones*) oder alle Domänen der Gesamtstruktur (`/alldomains`).

`/deletedirectorypartition` *`FQDN`*

Löscht eine DNS-Verzeichnispartition.

`/enlistdirectorypartition` *`FQDN`*

Fügt den Server zum Replikationssatz der angegebenen DNS-Verzeichnispartition hinzu.

`/unenlistdirectorypartition` *`FQDN`*

Entfernt den Server vom Replikationssatz der angegebenen DNS-Verzeichnispartition.

PoSh: PowerShell enthält zahlreiche Cmdlets zur DNS-Server-Administration. Um diese aufzulisten, suchen Sie nach Cmdlets mit dem Namensbestandteil `dnsserver`.

dfscmd

dfscmd [*Optionen***]**

Konfiguriert einen DFS-Baum (*Distributed File System*).

Optionen

/view ***DFS-Stamm****Freigabe** [/Partial | /Full | /Batch | /BatchRestore]**

Zeigt die Volumen der angegebenen Freigabe an. /Partial fügt der Ausgabe Freigabekommentare hinzu, und /Full zeigt alle Server eines jeden Volumens an. /Batch erzeugt eine Batchdatei, um das DFS wiederherzustellen.

/map ***DFS-Stamm****Freigabe******Pfad** ***Server******Freigabe******Pfad** [***Kommentar***] [/Restore]**

Fügt dem DFS-Baum an der angegebenen Stelle ein freigegebenes Verzeichnis hinzu. /Restore unterdrückt alle Überprüfungen des Zielservers und erzwingt die Zuordnung.

/unmap ***DFS-Stamm****Freigabe******Pfad**

Entfernt ein Volumen aus dem DFS-Stamm.

/add ***DFS-Stamm****Freigabe******Pfad** ***Server******Freigabe******Pfad** [/Restore]**

Fügt dem angegebenen DFS-Volumen eine Replik hinzu. /Restore unterdrückt alle Überprüfungen des Zielservers und erzwingt die Zuordnung.

/remove ***DFS-Stamm****Freigabe******Pfad** ***Server******Freigabe******Pfad**

Entfernt eine Replik eines DFS-Volumens.

/move ***DFS-Stamm****Freigabe******Pfad1** ***DFS-Stamm******Freigabe******Pfad2** [/Force]**

Verschiebt einen Ordner im DFS-Namespace an einen anderen logischen Pfad. Optional werden bestehende Links ersetzt (/Force).

PoSh: Um die DFS-Administrationswerkzeuge in PowerShell aufzulisten, suchen Sie nach dem Namensbestandteil DFSN.

dfsdiag

dfsdiag *Hauptoption Weitere_Argumente*

dfsdiag unterstützt den Administrator bei der Suche nach den Ursachen von Problemen mit der DFS-Konfiguration.

/testdcs

Prüft die Konfiguration der Domänencontroller.

/testsites

Prüft die Standortzuordnungen im Active Directory.

/testdfsconfig

Prüft die DFS-Namespacekonfiguration.

/testdfsintegrity

Prüft die Integrität des DFS-Namespace.

/testreferral

Prüft Weiterleitungsantworten.

dfsutil

dfsutil *Hauptoption Weitere_Argumente*

dfsutil ist ein mächtiges Tool zur Administration des verteilten Dateisystems (DFS).

Als Besonderheit kann mit der Option /oldcli auf die alte Syntax umgeschaltet werden, die weiter unten beschrieben wird.

dfsutil Root

Verwaltet DFS-Stämme.

DFS-Stamm

Zeigt Informationen zum als UNC-Pfad angegebenen DFS-Stamm an.

AddDom *DFS-Stamm* **[***Version***] [***Kommentar***]**

Erstellt einen domänenbasierten DFS-Stamm, optional mit Angabe einer

Version (`V1`, kompatibel mit Windows 2000, oder `V2`, kompatibel mit Windows Server 2008) und eines Kommentars.

AddStd *DFS-Stamm* **[***Kommentar***]**

Erstellt einen eigenständigen DFS-Stamm, optional mit Angabe eines Kommentars.

Remove *DFS-Stamm*

Löscht einen DFS-Stamm.

Export *DFS-Stamm Datei*

Schreibt die Konfiguration eines DFS-Stamms in *`Datei`*.

Import Set *Quelle Ziel-DFS-Stamm*

Importiert die Konfiguration eines DFS-Stamms von einem anderen Stamm (in diesem Fall ist *`Quelle`* ein UNC-Pfad) oder aus einer Datei, die mit `dfsutil Root Export` erzeugt wurde (dann ist *`Quelle`* der Dateiname), in einen Ziel-DFS-Stamm. Dieser wird dabei überschrieben!

Import Merge *Datei Ziel-DFS-Stamm*

Importiert die Konfiguration eines DFS-Stamms aus einer Datei, die mit `dfsutil Root Export` erzeugt wurde, in einen Ziel-DFS-Stamm. Die Informationen aus der Datei werden mit den im Stamm vorhandenen Daten zusammengeführt.

Import Compare *Quelle Ziel-DFS-Stamm*

Vergleicht die Konfiguration eines DFS-Stamms (*`Quelle`* kann ein UNC-Pfad zu einem DFS-Stamm oder eine mit `dfsutil Root Export` erzeugte Datei sein) mit einem Ziel-DFS-Stamm.

dfsutil Link

Verwaltet Verknüpfungen im DFS.

Add *DFS-Pfad Freigabepfad*

Bindet eine Freigabe in den DFS-Namespace ein.

Remove *DFS-Pfad*

Entfernt eine Freigabe aus dem DFS-Namespace.

Move *DFS-Pfad-Alt DFS-Pfad-Neu* **[Replace]**

Verschiebt eine Verknüpfung innerhalb eines DFS-Namespace. Optional wird ein bestehender Zielordner ersetzt (`Replace`).

dfsutil Target

Verwaltet Verknüpfungsziele im DFS.

`Add [`*`DFS-Verknüpfung`***`]`** ***`Freigabepfad`***

Ohne Angabe einer DFS-Verknüpfung wird ein neuer DFS-Stammserver zum DFS-Stamm hinzugefügt. Ansonsten wird ***`Freigabepfad`*** als weiteres Ziel zur vorhandenen DFS-Verknüpfung hinzugefügt.

`Remove [`*`DFS-Verknüpfung`***`]`** ***`Freigabepfad`***

Ohne Angabe einer DFS-Verknüpfung wird ein DFS-Stammserver vom DFS-Stamm entfernt. Ansonsten wird ***`Freigabepfad`*** als Ziel von einer DFS-Verknüpfung entfernt.

dfsutil Property

Zeigt oder ändert Eigenschaften des DFS-Namespaces, die hier aus Platzgründen nur stichwortartig aufgeführt sind: Berücksichtigung von Standortkosten (`Sitecosting`), Root-Scalability-Modus (`RootScalability`), Access Based Enumeration (`ABDE`), Clients greifen nur auf Ziele im eigenen Standort zu (`Insite`), Failback-Verhalten von Clients nach Ausfall eines DFS-Ziels (`TargetFailback`), Berechtigungen einer Verknüpfung (`Acl`), Onlinestatus eines Stammservers oder einer Verknüpfung (`State`), Timeout-Intervall einer Verknüpfung (`TTL`), Priorität einer Verknüpfung (`PriorityRank`), Prioritätsklasse einer Verknüpfung (`PriorityClass`), Beschreibung/Kommentar eines Namespaces oder einer Verknüpfung (`Comment`).

dfsutil Client

Verwaltet Daten von DFS-Clients. Hier ist nur die wichtigste Option aufgeführt.

`SiteInfo` ***`DFS-Client`***

Zeigt den Standort eines über den Computernamen oder die IP-Adresse spezifizierten DFS-Clients an.

dfsutil Server

Zeigt ohne weitere Parameter an, welche DFS-Stämme ein Server beherbergt.

`Registry` *`Optionen`*

Zeigt oder ändert Registry-Werte, die folgende Eigenschaften eines DFS-Servers beeinflussen: Verwenden von FQDNs statt NetBIOS-Namen in Verweisen (`DfsDnsConfig`), Timeout-Intervall bei LDAP-Abfragen (`LdapTimeoutValue`), Häufigkeit der Abfrage des PDC-Emulators durch Stammserver (`SyncInterval`), Sortierreihenfolge von Verweisen außerhalb des Clientstandorts (`Site-CostedReferrals`), nur Verweise im Clientstandort zurückgeben (`InsiteReferrals`), Setzen des Anmelde-DC an die Spitze der Verweisliste (`PreferLogonDC`).

dfsutil Diag

Bietet verschiedene Diagnosemöglichkeiten.

`UnMapDomRoot | Clean` *`Optionen`*

Spezielle Befehle zum Löschen von Verweisen auf nicht mehr vorhandene DFS-Stämme.

`ViewDfsDirs` *`Laufwerk`* **`[RemoveReparse]`**

Zeigt alle Verzeichnisse des angegebenen Laufwerks (Format: Buchstabe mit Doppelpunkt dahinter) an, die einen DFS-Reparse-Point enthalten. Der optionale Parameter `RemoveReparse` löscht alle gefundenen Reparse Points.

`ViewDfsPath` *`DFS-Pfad`*

Löst einen DFS-Pfad auf und zeigt an, auf welchen UNC-Pfad er verweist.

dfsutil Domain Domänenname

Zeigt alle domänenbasierten DFS-Namespaces in der angegebenen Domäne an.

dfsutil Cache

Zeigt den Inhalt von bzw. löscht DFS-Caches.

`Domain [Flush]`

Anzeige oder Löschen (`Flush`) des Domänencaches.

`Referral [Flush]`

Anzeige oder Löschen (`Flush`) des Verweiscaches.

`Provider [Flush]`

Anzeige oder Löschen (`Flush`) des Providercaches.

Alte Syntax: Optionen für Server und Client

`/view:\\DFS-Stamm\`*`DFS-Verknüpfung`* **`[/dcname:`***`Name`***`] [/LEVEL:1]`**

Zeigt die Konfigurationsinformationen für die angegebene DFS-Verknüpfung an. `/LEVEL:1` erzeugt eine detaillierte Ausgabe.

`/addroot:` *`DFS-Stamm`* **`/server:`***`Server`* **`/share:`***`Freigabe`*

Erzeugt einen neuen DFS-Stamm auf *`Server`* in *`Freigabe`*.

`/unmap:``\\`*`DFS-Stamm`*`\`*`DFS-Verknüpfung`* **`/root:`**`\\`*`Server`*`\`*`Freigabe`*

Entfernt die in `/root` angegebene Freigabe vom angegebenen DFS-Stamm.

Alte Syntax: Clientoptionen

`/pktinfo`

Zeigt Informationen zum DFS-Partitionscache an. Dieser Befehl ist hilfreich, wenn ermittelt werden soll, von welchem Server der Client seine DFS-Informationen bezieht.

`/pktflush`

Löscht den *Partition Knowledge Table Cache* (den Cache, in dem der Client Informationen über die DFS-Infrastruktur speichert).

Hinweis: Um auf diesen Befehl zugreifen zu können, muss unter Server 2025 der Rollendienst *DFS-Namespaces* installiert sein. Unter Windows 11 muss das optionale Windows-Feature *RSAT: Tools für Dateidienste* installiert sein.

---

netsh

`netsh [-a` *`Aliasdatei`*`**] [-c** *`Kontext`*`**] [-r** *`Computer`*`**] [-u** *`Benutzer`*`**] [-p** *`Kennwort`* `|` `*] [`*`Befehl`* `|` **`-f`** *`Skriptdatei`*`]`

`netsh` ist das Administrationswerkzeug des Netzwerksubsystems. Dieses Tool kann entweder interaktiv oder per Skriptdatei gesteuert werden. Der Aufruf ohne Parameter startet den Befehl im interaktiven Modus, in dem er an der eigenen Kommandozeile Eingaben entgegennimmt (`help` oder ? zeigen jederzeit eine kontextbezogene Hilfe an). Aufgrund des begrenzten Platzangebots und der Mächtigkeit des Befehls sei für eine vollständige Beschreibung auf die Hilfe verwiesen.

Verfügbare Kontexte

Abhängig von der Betriebssystemversion (Server oder Client) sind nicht immer alle Kontexte verfügbar. Mit der Eingabe von `..` wechseln Sie zurück in den übergeordneten Kontext.

`advfirewall und firewall`
: Verwaltet Firewall-Richtlinien und Konfiguration.

`branchcache`
: Verwaltet die Einstellungen zum Branchcache.

`bridge`
: Verwaltet die Netzwerkbrücke.

`dhcp`
: Konfiguriert den DHCP-Serverdienst.

`dhcpclient`
: Verwaltet den DHCP-Client.

`dnsclient`

Verwaltet die DNS-Clienteinstellungen.

`http`

Verwaltet den HTTP-Servertreiber *http.sys*.

`interface`

Verwaltet die IP-Konfiguration (v4 und v6).

`ipsec`

Verwaltet IPSec-Richtlinien.

`ipsecdosprotection`

Verwaltet den Schutz vor IPSEC-DoS-(Denial-of-Service) Angriffen.

`lan und wlan`

Verwaltet drahtgebundene und drahtlose Netzwerkschnittstellen.

`mbn`

Verwaltet mobile Breitbandnetzwerke.

`namespace`

Verwaltet DNS-Clientrichtlinien.

`nap`

Verwaltet Network Access Protection.

`netio`

Verwaltet Bindungsfilter.

`p2p`

Verwaltet die Peer-to-Peer-Dienste von Windows.

`ras`

Verwaltet Remote-Access-Server.

`routing`

Konfiguriert den Routing- und RAS-Dienst.

`rpc`

Konfiguriert die Bindungen des RPC-Diensts.

`wcn`

Verwaltet drahtlose Netzwerkverbindungen.

`wfp`

Verwaltet Windows Filtering Platform.

`winhttp`

Konfiguriert Proxy- und Tracing-Einstellungen des HTTP-Clients von Windows.

`wins`

Konfiguriert den WINS-Serverdienst.

`winsock`

Setzt unter anderem die Winsock-Konfiguration zurück.

Beispiele

`netsh interface ip set dns "`*`LAN-Verbindung`*`" static` *`IP-Adresse`*

Setzt den primären DNS-Server für das lokale System.

`netsh dhcp server scope` *`Name`* **`add excluderange`** *`IP-Adresse1 IP-Adresse2`*

Fügt einem bereits existierenden DHCP-Adresspool einen ausgeschlossenen Bereich hinzu.

`netsh advfirewall firewall add rule name="`*`ICMPv4-Ping zulassen`***`" protocol=icmpv4:8,any dir=in action=allow`**

Erlaubt das zuvor durch die Windows Firewall blockierte Beantworten eingehender Ping-Anforderungen auf die eigenen IPv4-Adressen aus allen Zonen.

`netsh advfirewall firewall set rule name="`*`Datei- und Druckerfreigabe (Echoanforderung - ICMPv4 eingehend`*`)" new enable=`*`yes`*

Aktiviert die bereits vorkonfigurierte, aber standardmäßig bei Neuinstallationen von Windows 11 inaktive Richtlinie *`Datei- und Druckerfreigabe (Echoanforderung - ICMPv4 eingehend)`*.

PoSh: In PowerShell gibt es zahlreiche Cmdlets, die den Möglichkeiten der einzelnen `netsh`-Befehle entsprechen. Diesen ist der Namensbestandteil `Net` gemeinsam.

Internet Information Server

appcmd

Dieser in *%SystemRoot%\System32\InetSrv* befindliche Befehl ermöglicht das Konfigurieren von IIS in der Eingabeaufforderung. Die generelle Syntax sieht so aus:

appcmd *Befehl Objekttyp* **[***ID***] [***/Parameter1*:*Wert1 ...***]** **[***Optionen***]**

Folgende Objekttypen des IIS werden unterstützt: Anwendung (`app`), Anwendungspools (`apppool`), Sicherungen (`backup`), SSL-Bindungen (`binding`), Konfigurationsabschnitte (`config`), Servermodule (`module`), HTTP-Anforderungen (`request`), Sites (`site`), Ablaufverfolgungsprotokolle (`trace`), virtuelle Verzeichnisse (`vdir`) und Arbeitsprozesse (`wp`).

Je nach *Objekttyp* werden unterschiedliche Befehle unterstützt. Folgende vier Kommandos gelten für die meisten Objekte:

add
: Erstellt ein neues Objekt des angegebenen Typs und Namens/Pfads (*ID*) mit den angegebenen Parametern.

delete
: Löscht das Objekt des angegebenen Namens/Pfads (*ID*).

list
: Gibt die Objekte des angegebenen Typs aus. Optional kann mit *ID* ein bestimmtes Objekt ausgewählt und/oder nach Objekten gefiltert werden, bei denen einzelne Parameter geforderte Werte haben.

set
: Setzt einen oder mehrere Parameter beim durch *ID* spezifizierten Objekt.

Einzelne Objekttypen verfügen über weitere Befehle. Bitte entnehmen Sie

diese der Hilfe zum Befehl.

Optionen

`/commit:`*`Pfad`*

Schreibt Konfigurationsänderungen in den angegebenen Pfad, an dessen Stelle auch folgende Angaben gültig sind: `apphost` (auf Serverebene in *applicationHost.config*), `app` (Applikationsstamm in *web.config*), `webroot`, `machine`, `site` (Site-Stamm in *web.config*), `url` (Standard, schreibt auf der Ebene, für die die Konfiguration gesetzt wird).

`/config`*`[:*]`*

Ausgabe der unverarbeiteten XML-Konfigurationsdaten (mit Sternchen auch der vererbten Daten).

`/in`

Liest die Eingabe im XML-Format von der Standardeingabe.

`/text`*`[:Attribut`* **`|`** ***`*]`***

Ausgabe des angegebenen oder aller Attribute in Textform.

`/xml`

Ausgabe in XML statt als Text. Kann als Eingabe für `/in` verwendet werden.

Beispiele

`appcmd list sites /state:`*`stopped`*

Zeigt alle gestoppten Sites.

`appcmd add site /name:`*`NeueSite`* **`/id:2 /bindings:"`*`https/*:444:`*`" /physicalPath:"`***`C:\WWWRoot\NeueSite`***`"`**

Erstellt eine neue Site mit einigen Eigenschaften.

`appcmd delete site "`*`NeueSite`***`"`**

Löscht eine Site.

`appcmd add backup "`*`Sicherung`***`"`**

Erstellt ein Backup eines bestimmten Namens.

`appcmd restore backup "`*`Sicherung`***`"`**

Spielt ein Backup eines bestimmten Namens zurück.

PoSh: PowerShell stellt zahlreiche Cmdlets für die Administration von IIS über das Modul *WebAdministration* bereit. Diese finden Sie mit dem Befehl `get-command *web*` oder durch direkte Abfrage des Moduls mittels `get-command -module webadministration`.

---

iisreset

`iisreset [`*`Server`*`] [`*`Optionen`*`]`

Verwaltet den Internet-Information-Services-(IIS)Dienst auf *`Server`*.

Optionen

`/restart | /start | /stop | /status`
: Startet den IIS neu (`/restart` ist die Standardeinstellung), startet den IIS (`start`), stoppt den IIS (`/stop`) oder zeigt den Status des IIS-Diensts an (`/status`).

`/enable | /disable`
: Aktiviert bzw. deaktiviert das Neustarten des IIS-Diensts.

`/reboot`
: Startet den Computer neu.

`/rebootonerror [/timeout:`*`s`*`]`
: Falls eine Start-, Stopp- oder Neustartoperation fehlschlägt, wird der Server neu gestartet (dabei wird gegebenenfalls bis zu *`s`* Sekunden auf die erfolgreiche Durchführung der Operation gewartet).

`/noforce`
: War das Stoppen des IIS-Diensts nicht erfolgreich, den Prozess nicht zum Beenden zwingen, sondern auf eine erfolgreiche Stoppoperation warten. Das Erzwingen eines gestoppten IIS-Prozesses ist die Standardeinstellung.

Benutzer und Gruppen

net group

Zeigt globale Gruppen einer Domäne an oder ändert sie. `/domain` bedeutet, dass der Befehl auf einem Domänencontroller ausgeführt wird statt auf dem lokalen System.

`net group`

Zeigt die Namen globaler Gruppen in der aktuellen Domäne an.

`net group` *`Name`* **`[`***`Benutzer`***`] [/add] [/domain]`**

Legt eine Gruppe an oder fügt *`Benutzer`* einer existierenden Gruppe hinzu.

`net group` *`Name`* **`[`***`Benutzer`***`] /delete [/domain]`**

Löscht eine Gruppe oder entfernt *`Benutzer`* aus einer Gruppe.

`net group` *`Name`* **`[/add] /comment:`***`Beschreibung`* **`[/domain]`**

Fügt eine Beschreibung einer existierenden oder einer neuen Gruppe hinzu.

PoSh: Cmdlets zur Verwaltung von Gruppen im Active Directory beinhalten den Namensbestandteil `adgroup`.

```
get-adgroupmember "Domänen-Admins"
```

Listet Benutzernamen, GUID und SID der Mitglieder der Gruppe *Domänen-Admins* auf.

net localgroup

Zeigt eine lokale Gruppe an oder ändert sie. `/domain` bedeutet, dass der Befehl auf einem Domänencontroller ausgeführt wird und nicht auf dem lokalen System. Die Benutzerlisten werden mit Leerzeichen voneinander

getrennt.

net localgroup

Zeigt die Namen lokaler Gruppen in der aktuellen Domäne an.

net localgroup *Name* **[***Benutzer***] [/add] [/domain]**

Legt eine Gruppe an oder fügt *Benutzer* einer bestehenden Gruppe hinzu.

net localgroup *Name* **[***Benutzer***] /delete [/domain]**

Löscht eine Gruppe oder entfernt *Benutzer* aus einer Gruppe.

net localgroup *Name* **[/add] /comment:***Beschreibung* **[/domain]**

Fügt eine Beschreibung einer existierenden oder einer neuen Gruppe hinzu.

PoSh: Cmdlets zur Verwaltung lokaler Gruppen und Benutzerkonten beinhalten den Namensbestandteil `localgroup`. Das folgende Beispiel fragt die Mitglieder der Gruppe der Administratoren auf dem aktuellen Rechner ab:

```
get-localgroupmember Administratoren
```

---

net user

net user *Benutzername* **[***Passwort***| *] [/add [***Optionen***] | /delete] [/domain]**

Erlaubt das Anlegen und Verändern von Benutzerkonten. Auf den Benutzernamen kann ein Passwort oder ein Sternchen (*) folgen, wenn das Passwort bei der Eingabeaufforderung eingegeben werden soll. Ohne Optionen zeigt der Befehl alle Benutzerkonten der Domäne oder der lokalen Arbeitsstation an.

Optionen

/add | /delete

Erstellt (`/add`) oder löscht (`/delete`) das angegebene Benutzerkonto. Standardmäßig wird ein existierendes Konto verändert.

/domain

Führt den Befehl auf dem primären Domänencontroller aus.

/active:{*yes* | *no*}

Aktiviert oder deaktiviert das Konto.

/fullname:*Text*

Der Name des Benutzers.

/expires:{*Datum* **| *never*}**

Das Ablaufdatum des Kontos, falls gewünscht.

/homedir:*Pfad*

Das Stammverzeichnis.

/logonpasswordchg:{*yes* | *no*}

Definiert, ob das Passwort bei der nächsten Anmeldung geändert werden muss.

/passwordchg:{*yes* | *no*}

Definiert, ob das Passwort vom Benutzer geändert werden darf.

/passwordreq:{*yes* | *no*}

Definiert, ob für dieses Konto ein Passwort vorhanden sein muss.

/profilepath:*Pfad*

Pfad zum Benutzerprofil dieses Kontos.

/scriptpath:*Pfad*

Pfad zum Anmeldeskript dieses Benutzers.

/times:{*all* |*Zeiten*}**

Erlaubte Anmeldezeiträume.

/workstations:*Liste*

Beschränkt die Anmeldung auf die angegebenen Rechner (maximal acht Einträge).

/comment:*Zeichenfolge*, **/usercomment:***Zeichenfolge*

Beschreibende Kommentare zu diesem Konto.

/countrycode:*n*

Betriebssystem-Ländercode (0 bedeutet, dass der Standardwert des Rechners verwendet wird).

PoSh: Für die Verwaltung von Benutzerkonten in der Domäne finden Sie in

PowerShell die Cmdlets `get-aduser`, `set-aduser`, `new-aduser` und `remove-aduser`, für lokale Benutzerkonten verwenden Sie anstelle von `aduser localuser`.

Mit dem folgenden Befehl ändern Sie die Beschreibung für das existierende lokale Benutzerkonto *`Sabine Eichner`*:

```
set-localuser "Sabine Eichner" description "Stammkraft"
```

---

whoami

Zeigt Informationen aus dem Access Token des angemeldeten Benutzers an (u.a. Gruppenmitgliedschaften und Privilegien/Rechte).

`whoami [/upn | /fqdn | /logonid]`

Zeigt den Anmeldenamen im NTLM-Format (*Domäne\Benutzer*), den *User Principal Name* (`/upn`), den Benutzernamen samt vollqualifiziertem Domänennamen (`/fqdn`) oder die Anmeldekennung bzw. Logon-SID (`/logonid`) an.

`whoami [[/upn] [/fqdn] [/logonid] [/user] [/groups] [/claims] [/priv]] | [/all] [/fo {table | list | csv}] [/nh]`

Zeigt alle (`/all`) oder Teile der Informationen des Access Token an: `/all` listet den User Principal Name von Domänenbenutzern auf, `/fqdn` den Fully Qualified Distinguished Name, `/logonid` die Logon-ID. `/user` zeigt NTLM-Anmeldename und SID, `/groups` alle Gruppenmitgliedschaften (auch verschachtelte und »spezielle« Gruppen), `/claims` Benutzeransprüche und `/priv` die dem Konto zugeordneten Privilegien/Rechte samt Status (aktiviert oder deaktiviert). Das Ausgabeformat kann mit `/fo` gewählt werden. `/nh` unterdrückt die Ausgabe von Spaltenüberschriften.

Active-Directory-Verzeichnisdienst

---

adprep

`adprep /forestprep`
`adprep /domainprep [/gpprep]`
`adprep /rodcprep`

Bereitet eine unter einer älteren Betriebssystemversion von Windows Server laufende Gesamtstruktur bzw. eine Domäne auf das Upgrade existierender Domänencontroller (DCs) vor, indem unter anderem das Schema erweitert und die Standard-SDs bestimmter Objekte aktualisiert werden.

Zunächst muss die Option `/forestprep` ausgeführt werden, wofür eine Verbindung mit dem Schema-Master hergestellt werden muss. Nachdem die Änderungen auf alle DCs repliziert wurden, wird `/domainprep` auf den Infrastruktur-Mastern jeder Domäne der Gesamtstruktur ausgeführt. Die Option `/gpprep` ist eine Voraussetzung für den RSOP-Planungsmodus, ändert jedoch die Berechtigungen aller Gruppenrichtlinienobjekte, was deren vollständige Replikation bewirkt. Daher sollte diese Option in Netzen mit mehreren Standorten zu Zeiten geringer WAN-Auslastung durchgeführt werden. `/rodcprep` schließlich bereitet die Gesamtstruktur auf die Installation von Read-only-DCs vor, indem Berechtigungen auf Anwendungsverzeichnispartitionen angepasst werden. Dieser letzte Befehl kann von einem entfernten System aus abgesetzt werden und ist einmal pro Gesamtstruktur erforderlich.

Damit ist die Gesamtstruktur für die Installation von DCs mit neueren Betriebssystemversionen bereit, kann jedoch problemlos unbegrenzte Zeit weiter unter der älteren Betriebssystemversion laufen.

Bei Installation der Active-Directory-Domänendienste auf einem neuen Windows-Server-2025-System erfolgen diese Schritte automatisch.

Hinweis: Das Tool befindet sich auf der Betriebssystem-DVD im Verzeichnis *\support\adprep*.

PoSh: Da die Installation neuer Domänencontroller in bestehenden Domänen mittels PowerShell umgesetzt wird, werden dabei die erforderlichen Schritte automatisch ausgeführt. Hierzu dient das Cmdlet `install-addsdomaincontroller`.

---

csvde, ldifde

`csvde [`*`Optionen`***`]`**
`ldifde [`*`Optionen`***`]`**

Importiert bzw. exportiert Daten mithilfe einer LDIF- (`ldifde`) oder CSV-Datei (`csvde`) in das bzw. aus dem Active Directory.

Optionen für Im- und Export

`-a` *`BenutzerDN Kennwort`*
: Gibt den Benutzerkontext für den Befehl an (einfache Authentifikation).

`-b` *`Benutzer Domäne Kennwort`*
: Gibt den Benutzerkontext für den Befehl an (SSPI-Bindung).

`-i`
: Führt eine Importoperation durch (Standard: Export).

`-f` *`Datei`*
: Gibt den Namen der Import- bzw. Exportdatei an.

`-j` *`Pfad`*
: Gibt den Pfad der Protokolldatei an.

`-s` *`Server`*
: Gibt den zu verwendenden Domänencontroller an (Standard: Anmelde-DC).

-t *Port*

Gibt den Port an (Standard: 389).

-c *Alt Neu*

Wandelt beim Import Daten um. Alle gefundenen Einträge zu *Alt* werden durch *Neu* ersetzt. Zum Beispiel können der Domänenname oder andere globale Daten in allen Datensätzen geändert werden.

-h

Verwendet SASL zur Verschlüsselung.

-u

Verwendet UNICODE statt ANSI.

-v

Generiert eine ausführliche Befehlsausgabe.

-w *Sekunden*

Legt das Timeout-Intervall für die Kommunikation mit dem LDAP-Server fest (Standard: kein Timeout).

Optionen für den Export

-d *Stamm-DN*

Gibt den *Distinguished Name* (DN) des Ausgangspunkts der Exportoperation an.

-p *Bereich*

Definiert den Bereich der Exportoperation: `onelevel`, `base` oder `subtree` (Standardwert).

-r *Filter*

Setzt zur Auswahl der Exportdaten einen Filter im LDAP-Format (siehe »LDAP-Suchfilter«).

-l *Attributliste*

Gibt eine durch Kommata getrennte Liste von Attributen an, die exportiert werden sollen.

-o *Attributliste*

Gibt eine durch Kommata getrennte Liste von Attributen an, die nicht exportiert werden sollen.

-m

Schließt Active-Directory-spezifische Attribute aus.

-n

Schließt binäre Werte aus.

-x

Exportiert auch gelöschte Objekte, die als »tombstoned« markiert sind.

-g

Deaktiviert die seitenweise Suche.

Optionen für den Import

-k | -z

Ignoriert manche (`-k`) bzw. alle (`-z`) Fehler beim Import.

-y

Verwendet *Lazy Commit* beim Import zur Erhöhung der Leistung (Standard).

-e

Deaktiviert *Lazy Commit* beim Import.

-q *Threads*

Legt die beim Import zu verwendende Anzahl von Threads fest (Standard: einer).

PoSh: PowerShell ermöglicht sowohl Importe in den Verzeichnisdienst als auch Exporte aus diesem. Vorteile dieser Methode sind beispielsweise die bessere Erstellbarkeit und Lesbarkeit von CSV-Dateien, die hier als Quelle dienen können, sowie die Möglichkeit, beim Importieren von Benutzerkonten direkt ein Kennwort zu setzen. Vorausgesetzt, in der CSV-Datei befinden sich alle erforderlichen Informationen zur Neuanlage eines Benutzerkontos und der aufrufende Anwender verfügt über Domänenadministratorrechte, würden im folgenden Beispiel die in der Datei *import.csv* hinterlegten Benutzerkonten im AD angelegt und mit dem Kennwort *K3nnw0rt!* versehen werden, und die Benutzer würden bei der ersten Anmeldung zu dessen Wechsel aufgefordert werden:

```
import-csv C:\Admin\import.csv | new-aduser -
```

```
accountpassword (convertto-securestring –asplaintext
'K3nnw0rt!' -force) -enabled:$true -
changepasswordatlogon:$true
```

---

dcdiag

dcdiag /s:*Domänencontroller* **[***Optionen***]**

Testet wesentliche Aspekte der Funktionalität eines Domänencontrollers. Zu den über 30 Tests gehören unter anderem Netzwerkkonnektivität, DNS-Registrierung, Replikationstopologie und FSMO-Rollen.

Optionen

/u:*Domäne**Benutzer* **/p:***Kennwort***| ***

Definiert einen Benutzernamen mit Kennwort zur Authentifizierung. Anstelle des Kennworts kann ein Sternchen (*) angegeben werden, damit es an der Eingabeaufforderung abgefragt wird.

/test:*Liste* **| /skip:***Liste***[/***c***]**

Gibt an, welche Tests durchgeführt werden sollen. /test gibt eine Liste von durchzuführenden Tests an, und /skip gibt eine Liste von zu überspringenden Tests an (alle anderen werden durchgeführt). In beiden Fällen gibt /c an, dass der Test um weitere optionale Tests erweitert werden soll. In der Hilfe des Befehls finden Sie eine komplette Übersicht der Tests.

/c

Umfangreicher Modus: Führt alle Tests außer DcPromo und RegisterInDNS durch.

/fix

Führt als »sicher« angesehene Reparaturen durch.

/a | /e

Testet alle Server des Standorts (/a) bzw. alle Server im gesamten Netzwerk (/e).

/q | /v [/*i***]**

Der Befehl wird im stillen (`/q`) bzw. ausführlichen Ausgabemodus (`/v`) ausgeführt. `/i` ignoriert unnötige Fehlermeldungen.

`/f:`*`Datei1`* **`/ferr:`***`Datei2`*

Gibt eine Ausgabedatei (`/f`) und eine Datei für die Fehlerausgabe (`/ferr`) an. Falls beide Optionen verwendet werden, müssen die Dateinamen unterschiedlich sein.

---

dcgpofix

`dcgpofix [/ignoreschema] [/target: domain | dc | both]`

Stellt die Standarddomänen-Gruppenrichtlinienobjekte (Domänen- und/oder Domänencontroller-GPO) wieder her.

Optionen

`/ignoreschema`

Prüft nicht, ob die Schemaversion zur verwendeten Programmversion passt.

`/target: {domain | dc | both}`

Wiederherzustellendes Gruppenrichtlinienobjekt: Domänen-GPO, Domänencontroller-GPO oder beide.

---

djoin

`djoin [`*`/Optionen`***`]`**

Erstellt ein Computerkonto in der Domäne und ermöglicht den Offlinebeitritt zur Domäne. Eine ausführliche Beschreibung der Optionen finden Sie in der Hilfe zum Befehl.

Optionen

`/provision /domain` *`Domänenname`* **`/machine`** *`Computername`* **`[/reuse]`**

[/savefile *Dateiname*] [...]

Stellt ein Computerkonto in der Domäne bereit. Dabei sind der Domänenname und der Computername des zu erstellenden Kontos anzugeben. Mit `reuse` setzen Sie das Kennwort eines bereits bestehenden Computerkontos zurück. Mit `savefile` erstellen Sie eine Datei, die einem Computer den Domänenbeitritt ohne unmittelbare Verbindung zu einem Domänencontroller ermöglicht.

/requestodj /loadfile *Dateipfad* **/windowspath** *Pfad* **/localos**

Fordert beim nächsten Computerstart den Offlinebeitritt zur Domäne unter Verwendung einer zuvor mit `savefile` erstellten Datei an. Dabei gibt `windowspath` den Pfad zum Windows-Verzeichnis in einem Offlineimage an, in Kombination mit `localos` kann das gerade laufende Betriebssystem als Ziel des Befehls festgelegt werden.

PoSh: Einen Offline-Domänenbeitritt können Sie zum aktuellen Zeitpunkt mit PowerShell lediglich unter Zuhilfenahme des beschriebenen Befehls `djoin` realisieren.

Um einer Domäne mit PowerShell online beizutreten, können Sie das Cmdlet `add-computer` verwenden. Folgendes Beispiel fügt den Computer, an dem Sie aktuell angemeldet sind, der Domäne *domaene.de* in der im Parameter `–oupath` angegebenen Organisationseinheit unter Verwendung des Domänenbenutzerkontos *Admin1* hinzu:

```
add-computer -domainname domaene.de -oupath
"OU=Computer,OU=Firma, DC=domaene,DC=de" -cred
domaene\Admin1 -passthru –verbose
```

---

dsadd

dsadd *Optionen* **[-s** *DC* | **-d** *Domäne*] **[-u** *Benutzer*] **[-p** *Passwort* | ***] [-q] [-uc** | **-uco** | **-uci]**

Fügt dem Active Directory einen der folgenden Objekttypen hinzu: `Benutzer`, `Computer`, `Gruppe`, `Kontakt`, `OU`, `Verzeichnis-Quota`. Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter DC bzw. eine

bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe. Für Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

Optionen

computer *ComputerDN* **[-desc** *Beschreibung***]**
[-memberof *Gruppe1DN Gruppe2DN* **...]**

Erstellt das angegebene Computerobjekt und fügt es einer oder mehreren Gruppen hinzu.

contact *KontaktDN* **[-fn** *Vorname***] [-ln** *Nachname***]**
[-display *Anzeigename***] [-desc** *Beschreibung***]**

Erstellt den angegebenen Kontakt.

group *GruppenDN* **[-scope {***l* **|** *g* **|** *u***}]**
[-desc *Beschreibung***] [-memberof** *Gruppe1DN Gruppe2DN* **...]**
[-members *Gruppe1DN Gruppe2DN* **...]**

Erstellt das angegebene Gruppenobjekt samt Mitgliedschaften. Der Parameter `-scope` legt fest, ob eine lokale, globale oder universelle Gruppe angelegt wird.

ou *OUDN* **[-desc** *Beschreibung***]**

Erstellt eine Organisationseinheit.

user *BenutzerDN* **[-fn** *Vorname***] [-ln** *Nachname***]**
[-display *Anzeigename***] [-desc** *Beschreibung***]**
[-pwd *Kennwort* **| ******] [-memberof** *Gruppe1DN Gruppe2DN* **...]**

Erstellt den angegebenen Benutzer samt Gruppenmitgliedschaften. Sollte ein Stern als Passwort angegeben werden, wird dieses abgefragt.

quota -part *PartitionsDN* **-acct** *Konto* **-qlimit** *Limit* **[-rdn** *Name***]**

Erstellt ein Verzeichnis-Quota-Objekt, das die Anzahl der Objekte in einer Verzeichnispartition begrenzt, die ein bestimmtes Konto als Besitzer haben. *`Konto`* kann ein Benutzer, eine Gruppe oder ein Computer sein und als `DN` oder in der Form *Domäne\Anmeldename* angegeben werden. *`Limit`* legt die Anzahl an Objekten fest, die das angegebene Konto besitzen darf (`-1` steht für unbegrenzt). Der *`Name`* des zu erzeugenden Quota-Objekts wird, sofern nicht angegeben, auf

Domäne_KontoName gesetzt.

Für detailliertere Informationen zu den äußerst umfangreichen Optionen dieses Befehls sei zudem auf dessen Hilfe verwiesen.

---

dsamain

dsamain /dbpath *NTDS.dit* **/ldapport** *Port* **[/logpath** *Pfad***] [/adlds]**
[/sslport *Nummer***] [/gcort** *Nummer***] [/gcsslport** *Nummer***]**
[/allowupgrade] [/allownonadminaccess]

Stellt die gesamten Daten eines Active Directory (AD DS oder AD LDS) aus einer Sicherung, einem Schnappschuss oder einer Netzwerkfreigabe per LDAP bereit. Dieser Befehl ist nur verfügbar, wenn eine der Rollen *Active-Directory-Domänendienste* oder *Active-Directory-Lightweight-Domänendienste* installiert ist. Die in dem bereitgestellten Verzeichnis gesetzten Berechtigungen werden beim Zugriff darauf berücksichtigt.

Optionen

/dbpath

Spezifiziert den Pfad zur Datenbankdatei *NTDS.DIT*.

/ldapport

Spezifiziert den Port, unter dem der LDAP-Server erreichbar ist.

/sslport

Spezifiziert den SSL-Port, unter dem der LDAP-Server erreichbar ist. Standard: ldapPort + 1.

/gcport

Spezifiziert den Port, unter dem der globale Katalog erreichbar ist. Standard: ldapPort + 2.

/gcsslport

Spezifiziert den SSL-Port, unter dem der globale Katalog erreichbar ist. Standard: ldapPort + 3.

`/logpath`

Pfad zu den Logdateien, die beim Zugriff auf das bereitgestellte Verzeichnis erzeugt werden. Wenn kein solcher Pfad angegeben wird, werden die Logdateien im Temp-Verzeichnis angelegt.

`/adlds`

Gibt an, dass es sich bei der zu öffnenden Verzeichnisdienstdatenbank um eine AD-LDS-Datenbank handelt.

`/allowupgrade`

Wird benötigt, um Verzeichnisdienstdatenbanken älterer Windows-Versionen aktualisieren zu können. In einem solchen Fall muss die Datei sich auf einem beschreibbaren Datenträger befinden (also nicht z.B. in einer Schattenkopie).

`/allownonadminaccess`

Erlaubt den Zugriff beliebiger Benutzer. Standardmäßig können nur Domänen- oder Unternehmens-Admins der zu öffnenden Domäne zugreifen.

---

dsdbutil

Interaktives Tool zur Verwaltung der Datenbanken von AD DS und AD LDS.

`dsdbutil` wird ähnlich wie das ältere `ntdsutil` bedient und deckt zusammen mit `dsmgmt` dessen Funktionalität ab. Die verschiedenen Kommandos wirken sich auf eine bestimmte Instanz des Verzeichnisdiensts aus, die mit `Activate Instance` *Name* auf »NTDS« oder den Namen einer AD-LDS-Instanz festgelegt wird.

`dsdbutil` bietet unter anderem Kommandos zum Durchführen autoritativer Wiederherstellungen, zum Ändern des von den Domänendiensten verwendeten Benutzerkontos, zur Erstellung von Installationsmedien für die DC-Installation (IFM) sowie zur Verwaltung von Schnappschüssen.

dsget

dsget *Objekttyp* **[-s** *DC* | **-d** *Domäne***] [-u** *Benutzer***] [-p** *Kennwort* | ***] [-q] [-l] [-uc** | **-uco** | **-uci]**

Fragt Objekteigenschaften im Active Directory ab. Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter DC bzw. eine bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe, `-l` erstellt die Ausgabe in Listen- statt in Tabellenform. Für Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

Objekttypen

computer *ComputerDN* **[-dn] [-samid] [-sid] [-desc] [-memberof [expand]] [-part** *PartitionsDN* **[-qlimit] [-qused]]**

Fragt Eigenschaften eines Computerkontos ab. `-expand` zeigt die Gruppenmitgliedschaften rekursiv inklusive aller Verschachtelungen an. `-qlimit` und `-qused` dienen zur Abfrage der Verzeichnis-Quota des Computers.

contact *KontaktDN* **[-fn] [-ln] [-display] [-desc]**

Fragt Eigenschaften eines Kontakts ab.

group *GruppenDN* **[-dn] [-samid] [-sid] [-scope] [-desc] [-memberof [-expand]] [-members [-expand]] [-part** *PartitionsDN* **[-qlimit] [-qused]]**

Fragt Eigenschaften einer Gruppe ab.

ou *OUDN* **[-desc]**

Zeigt die Beschreibung einer Organisationseinheit an.

server *DCDN* **[-desc] [-dnsname] [-site] [-isgc]**

Fragt Eigenschaften eines Domänencontrollers ab: Beschreibung, DNS-Name, Standort, globaler Katalog.

server *DCDN* **-topobjowner** *Anzahl*

Zeigt eine Liste der *Anzahl* Konten im Verzeichnis, die Besitzer der meisten Objekte in allen Partitionen des DC sind. Eine *Anzahl* von 0 zeigt alle Objektbesitzer an.

server *DCDN* **-part**

Zeigt die DNs der Partitionen des DC an.

user *BenutzerDN* **[-fn] [-ln] [-display] [-desc] [-memberof [-expand]] [-part** *PartitionsDN* **[-qlimit] [-qused]]**

Fragt Eigenschaften eines Benutzerkontos ab.

subnet *SubnetDN* **[-desc] [-site] [-loc]**

Zeigt Eigenschaften eines Subnetzes an.

site *SiteDN* **[-desc] [-autotopology] [-cachegroups] [-prefGCsite]**

Zeigt Eigenschaften eines Standorts an: Automatic Intersite Topology Generator eingeschaltet (`-autotopology`), Caching der Mitglieder universeller Gruppen aktiviert (`-cachegroups`), bevorzugter GC (`-prefGCsite`).

quota [-acct] [-qlimit]

Zeigt Eigenschaften eines Quota-Objekts an.

partition *PartitionsDN* **[-qdefault] [-qtmbstnwt] [-topobjowner** *Anzahl***]**

Zeigt Eigenschaften einer Verzeichnispartition an: Standard-Verzeichnis-Quota (`-qdefault`), prozentuale Gewichtung von »Tombstoned«-Objekten (`-qtmbstnwt`, Objekte, die als »tombstoned« markiert sind, werden nur zu diesem Anteil bei der Berechnung der Quota berücksichtigt), Besitzer der meisten Objekte (`-topobjowner`).

Für detailliertere Informationen zu den äußerst umfangreichen Optionen dieses Befehls sei auf dessen Hilfe verwiesen.

---

dsmgmt

Interaktives Tool zur Verwaltung von Anwendungspartitionen und FSMO-Rollen und zum Löschen von im AD verbliebenen Daten nicht mehr existenter DCs.

`dsmgmt` wird ähnlich wie das ältere `ntdsutil` bedient und deckt zusammen mit `dsdbutil` dessen Funktionalität ab. Die Kommandos des Befehls sind auf unterschiedliche Ebenen verteilt. Mit `quit` kehrt man auf die jeweils

darüberliegende Ebene zurück. `help` gibt auf jeder Ebene die verfügbaren Kommandos aus. Bevor auf einen Domänencontroller zugegriffen werden kann, muss zunächst mit `connections` (auf den meisten Ebenen verfügbar) eine Verbindung zu ihm hergestellt werden.

Von den verfügbaren Kommandos sind insbesondere folgende praxisrelevant:

`Metadata Cleanup`
: Löscht Informationen aus dem Verzeichnis, die von nicht mehr vorhandenen Domänen, DCs oder Naming Contexts (Partitionen) stammen.

`Roles`
: Überträgt FSMO-Rollen zwischen DCs. Bietet auch einen Modus für den Fall, dass der aktuelle Inhaber einer Rolle nicht verfügbar ist (`Seize` *Rolle*).

`Security Account Management`
: Wartung der Sicherheitskonten, ermöglicht die Suche (und optional Korrektur) doppelter SIDs.

`Set DSRM Password`
: Setzt bzw. ändert das Kennwort für den Verzeichnisdienst-Wiederherstellungsmodus.

---

dsmod

`dsmod` *`Objekttyp`* **`[-s`** *`DC`* **`|`** **`-d`** *`Domäne`***`] [-u`** *`Benutzer`***`] [-p`** *`Kennwort`* **`|`** **`*] [-q] [-uc`** **`|`** **`-uco`** **`|`** **`-uci]`**

Modifiziert einen der folgenden Objekttypen im Active Directory: `Benutzer`, `Computer`, `DC`, `Gruppe`, `Kontakt`, `OU`, `Partition`, `Verzeichnis-Quota`. Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter Domänencontroller (*`DC`*) bzw. eine bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe. Für Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

Objekttypen

computer *ComputerDN* **[-desc** *Beschreibung***] [-disabled {*yes* | *no*}] [-reset]**

Setzt die Beschreibung des Computerkontos, (de)aktiviert es oder setzt es zurück.

contact *KontaktDN* **[-fn** *Vorname***] [-ln** *Nachname***] [-display** *Anzeigename***] [-desc** *Beschreibung***]**

Modifiziert die angegebenen Eigenschaften des Kontakts.

group *GruppenDN* **[-scope {*l* | *g* | u*}] [-addmbr | -rmmbr | -chmbr** *Mitglied1DN Mitglied2DN* **...]**

Ermöglicht das Ändern des Gruppentyps (lokal nach global bzw. umgekehrt ist prinzipiell nicht möglich bzw. nur per Umweg über eine universelle Gruppe). Ferner können angegebene Mitglieder hinzugefügt (`-addmbr`), entfernt (`-rmmbr`) oder die existierenden Mitglieder können vollständig durch die angegebenen neuen Mitglieder ersetzt werden (`-chmbr`).

ou *OUDN* **[-desc** *Beschreibung***]**

Lediglich die Beschreibung einer OU kann geändert werden.

server *DCDN* **[-desc** *Beschreibung***] [-isgc {*yes* | *no*}]**

Legt fest, ob ein DC globaler Katalog ist oder nicht, und setzt die Beschreibung.

user *BenutzerDN* **[-fn** *Vorname***] [-ln** *Nachname***] [-display** *Anzeigename***] [-desc** *Beschreibung***] [-pwd** *Passwort* **| *] [-disabled {*yes* | *no*}]**

Setzt neben diversen Eigenschaften eines Benutzers das Kennwort und ermöglicht das Aktivieren bzw. Deaktivieren des Kontos.

quota *QuotaDN* **[-qlimit** *Limit***] [-desc** *Beschreibung***]**

Setzt Beschränkung (`-1` für unbegrenzt) oder Beschreibung eines Verzeichnis-Quota-Objekts.

partition *PartitionsDN* **[-qdefault** *Limit***] [-qtmbstnwt** *Prozent***]**

`-qdefault` setzt die Standard-Quota einer Verzeichnispartition, die für alle Benutzer/Gruppen gilt, für die nicht explizit Quoten definiert wurden. `-qtmbstnwt` legt fest, zu welchem Anteil Objekte, die als »tombstoned« gekennzeichnet sind, bei der Berechnung der Quota berücksichtigt

werden. Wenn dieser Wert beispielsweise 50 % beträgt, kann ein Benutzer bei einem Quota-Limit von 100 entweder 100 normale oder 200 Tombstoned-Objekte besitzen.

Für detailliertere Informationen zu den äußerst umfangreichen Optionen dieses Befehls sei auf dessen Hilfe verwiesen.

---

dsmove

dsmove *Objekt-DN* **[-s** *DC* | **-d** *Domäne***] [-u** *Benutzer***] [-p** *Kennwort* | ***] [-q] [-uc** | **-uco** | **-uci]**

Verschiebt ein Objekt innerhalb einer Active-Directory-Domäne und/oder benennt es um (Verschieben über Domänengrenzen hinweg ist mit `movetree` möglich). Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter DC bzw. eine bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe. Für die Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

dsmove *ObjektDN* **[-newname** *NeuerRDN***] [-newparent** *NeuerElternDN***]**

Ändert den Namen (RDN) des Objekts oder verschiebt es in der Verzeichnishierarchie. Beide Optionen können zusammen verwendet werden.

---

dsquery

dsquery *Objekttyp* **{*StartObjekt*** | ***domainroot*** | ***forestroot*****}**
[-o {*dn* | ***rdn*** | ***samid*** | ***upn*****}] [-scope {*base*** | ***onelevel*** | ***subtree*****}]**
[-limit *Limit***] [-gc] [-s** *DC* | **-d** *Domäne***] [-u** *Benutzer***]**
[-p *Kennwort* | ***] [-q] [-l] [-uc** | **-uco** | **-uci]**

Sucht nach Objekten im Active Directory, die den angegebenen Bedingungen genügen. Es wird ab einem angegebenen *StartObjekt* gesucht, in der

ganzen Domäne (Standardwert) oder der Gesamtstruktur (Forest). `-o` legt fest, welches Attribut gefundener Objekte ausgegeben wird. Mit `-scope` wird die Suchtiefe festgelegt: nur das angegebene Objekt, eine Ebene darunter oder der ganze Tree (Standardwert). Die maximale Anzahl zurückzugebender Ergebnisse wird mit `-limit` begrenzt (`0`: alle gefundenen, Standardwert: `100`). Bei Angabe von `-gc` wird die Suche im globalen Katalog durchgeführt. Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter DC bzw. eine bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe, `-l` erstellt die Ausgabe in Listen- statt in Tabellenform. Für Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

Objekttypen und ausgewählte Optionen

`computer [-name` *`CN`*`**][-samid** *`sAMAccountName`*`**][-inactive** *`Wochen`*`**]`**
`[stalepwd` *`Tage`*`**][-disabled]`**

Findet Computer anhand eines oder mehrerer der Kriterien: *`CN`* oder *`sAMAccountName`* (diese können ein Sternchen als Wildcard enthalten), Anzahl inaktiver Wochen, Anzahl der Tage, in denen das Kennwort nicht geändert wurde, deaktiviertes Konto.

`contact [-name` *`CN`*`**][-desc** *`Beschreibung`*`**]`**

Findet Kontakte anhand von *`CN`* und/oder *`Beschreibung`* (beide können ein Sternchen als Wildcard enthalten).

`group [-name` *`CN`*`**][-samid** *`sAMAccountName`*`**]`**

Findet Gruppen anhand von *`CN`* und/oder *`sAMAccountName`* (beide können ein Sternchen als Wildcard enthalten).

`ou [-name` *`CN`*`**][-desc** *`Beschreibung`*`**]`**

Findet Organisationseinheiten anhand von *`CN`* und/oder *`Beschreibung`* (beide können ein Sternchen als Wildcard enthalten).

`server [-forest][-domain` *`DNSDomänenName`*`**][-site** *`Standort`*`**]`**
`[-name` *`CN`*`**][-isgc][-hasfsmos {`*`schema`*` | `*`name`*` | `*`infr`*` | `*`pdc`*` | `*`rid`*`**}]`**

Findet Domänencontroller im ganzen Wald (Forest)/in der angegebenen Domäne/in einem Standort anhand einer oder mehrerer der folgenden

Eigenschaften: *CN* (kann Wildcards enthalten), globaler Katalog, Halten einer FSMO-Rolle (für `infr`, `pdc` und `rid` wird die durch den Parameter `-domain` angegebene Domäne verwendet).

site [-name *CN*`]`**[-desc** *Beschreibung*`]`

Findet Standorte anhand von *CN* und/oder *Beschreibung* (beide können ein Sternchen als Wildcard enthalten).

user [-name *CN*`]` **[-pname** *Phonetischer Name*`]` **[desc** *Beschreibung*`]` **[-upn** *UPN*`]` **[-samid** *sAMAccountName*`]`**[-inactive** *Wochen*`]` **[-stalepwd** *Tage*`]` **[-disabled]**

Findet Benutzer anhand eines oder mehrerer der Kriterien: *CN*, *UPN* oder *SAMAccountName*, *Name/Phonetischer Name* (diese können ein Sternchen als Wildcard enthalten), Beschreibungstext, Anzahl Wochen ohne Logon, Anzahl der Tage, in denen das Passwort nicht geändert wurde, deaktiviertes Konto.

quota [-acct *Konto*`]`**[-qlimit** *Limit*`]` **[-desc** *Beschreibung*`]`

Findet Quota-Objekte anhand des zugeordneten Kontos (als DN oder in der Form *Domäne\Benutzer*), des Limits oder der Beschreibung (Letztere kann ein Sternchen als Wildcard enthalten).

partition [-part *CN*`]`

Findet Verzeichnispartitionen anhand ihres *CN* (kann ein Sternchen als Wildcard enthalten).

*** [-filter** *LDAP-Filter*`]` **[-attr** *Attributliste* **|*]**

Findet beliebige Objekte im Active Directory anhand von Kriterien, die in einem LDAP-Suchfilter (siehe »LDAP-Suchfilter«) definiert werden. Die in der Attributliste (Standardwert: DN) angegebenen Attribute jedes Objekts werden zurückgegeben. Ein Stern liefert alle gesetzten Attribute zurück.

Für detailliertere Informationen zu den äußerst umfangreichen Optionen dieses Befehls sei auf dessen Hilfe verwiesen.

---

dsrm

dsrm *ObjektDN* **[-subtree [-exclude]] [-noprompt] [-s** *DC* | **-d** *Domäne***] [-u** *Benutzer***] [-p** *Kennwort* | ***] [-q] [-uc** | **-uco** | **-uci]**

Löscht das angegebene Objekt im Active Directory. Falls das Objekt Kindobjekte enthält, löscht `-subtree` den ganzen Baum. In Kombination mit `-exclude` werden nur die Kindobjekte gelöscht. Die Option `noprompt` unterdrückt die Aufforderung zur Bestätigung jeglicher Objektlöschvorgänge. Falls angegeben, wird anstelle der Anmeldedomäne ein bestimmter DC bzw. eine bestimmte Domäne mit gegebenenfalls angegebenen Anmeldeinformationen angesprochen. `-q` (*quiet mode*) unterdrückt jegliche Ausgabe, `-l` erstellt die Ausgabe in Listen- statt in Tabellenform. Für Eingabe (`-uci`), Ausgabe (`-uco`) oder beide (`-uc`) kann der UNICODE-Zeichensatz verwendet werden.

PoSh: Nach Installation der RSAT-Tools beinhaltet PowerShell im Modul *ActiveDirectory* zahlreiche Cmdlets, die die Aufgaben der zuvor aufgeführten Befehle weitestgehend übernehmen können. Einen ersten Überblick über die auf diesem Weg verfügbaren Cmdlets verschaffen Sie sich mit dem Aufruf von:

```
get-command -module ActiveDirectory
```

---

esentutl

esentutl [Operation**]** *Dateiname* **[***Optionen***]**

Beinhaltet Datenbankwerkzeuge für die in Windows integrierte *Extensible Storage Engine* (ESE).

Operationen

/d Datenbankname [*Optionen***]**

Führt die Offlinedefragmentierung von Systemdatenbanken durch, beispielsweise der Active-Directory-Datenbank. Dabei verweist der Dateiname auf die zu defragmentierende Datenbankdatei. `/t` legt einen

Namen für die temporäre Datenbank fest, `/p` verhindert, dass die ursprüngliche Datenbank am Ende des Vorgangs mit der defragmentierten Version überschrieben wird. Durch `/b` erzeugen Sie eine Sicherungskopie mit dem angegebenen Namen. Die numerischen Parameter bestimmen die Seitengröße der Datenbank, so entspricht `/2` einer Seitengröße von 2k. `/v` bewirkt die erweiterte Ausgabe von Informationen.

`/r Logdateiname [`*`Optionen`***`]`**

Führt die Wiederherstellung mittels der angegebenen Logdatei (mit dreibuchstabigem Namen) aus.

`/g Datenbankname [`*`Optionen`***`]`**

Prüft die Integrität der angegebenen Datenbankdatei.

`/k Dateiname[`*`Optionen`***`]`**

Vergleicht die Prüfsummen einer Datenbank, Prüfsummendatei oder Logdatei.

`/g Datenbankname [`*`Optionen`***`]`**

Prüft die Integrität der angegebenen Datenbankdatei.

`/p Datenbankname [`*`Optionen`***`]`**

Repariert eine beschädigte Datenbankdatei.

`/m[`*`Modus-Modifizierer`***`]`** ***`Dateiname`*** **`[`***`Optionen`***`]`**

Erzeugt formatierte Ausgaben verschiedener Datenbankdateitypen.

`/y Quelldatei [`*`Optionen`***`]`**

Kopiert eine Datenbank oder Logdatei.

---

gpresult

`gpresult [/s` ***`System`*** **`[/u`** ***`Benutzer`*** **`[/p`** ***`Kennwort`*** **`]]]`**
`[/user` ***`Zielbenutzer`*** **`] [`***`Optionen`***`]`**

Zeigt die effektiven Gruppenrichtlinieneinstellungen (RSOP, *Resultant Set Of Policies*) für den aktuellen oder den angegebenen ***`Zielbenutzer`*** und/oder für das lokale oder angegebene ***`System`*** an. Die Verbindung zu ***`System`*** kann unter Verwendung von ***`Benutzer`*** und ***`Kennwort`*** hergestellt werden.

Optionen

`/scope {`*`user`*` | `*`computer`*`}`

Beschränkt die Ausgabe auf Benutzer- bzw. Computereinstellungen.

`/h | /x` *`Datei`* **`[/f]`**

Speichert die Ausgabe in *`Datei`* entweder im HTML- (`/h`) oder im XML-Format (`/x`), wobei bestehende Dateien nur bei Angabe von `/f` überschrieben werden.

`/r`

Gibt eine RSOP-Zusammenfassung aus.

`/v | /z`

Erzeugt eine ausführliche (`/v`) oder sehr ausführliche (`/z`) Ausgabe. Die ausführliche Ausgabe wird empfohlen.

Hinweis: Programme zur Verwendung von Gruppenrichtlinien sind nicht Bestandteil der Home-Editionen des Clientbetriebssystems. Dazu zählt auch Windows 11, wenn es sich nicht um die Professional- oder die Enterprise-Edition handelt.

---

gpupdate

`gpupdate [/target:{computer | user}] [/f | /force] [/wait:{`*`Wert`*`}] [/logoff] [/boot] [/sync]`

Aktualisiert die Gruppenrichtlinien auf dem lokalen System. Optional lässt sich die Aktualisierung auf den Computer- oder Benutzerteil der Richtlinien mit `/target` beschränken. `/force` erzwingt auch die Aktualisierung unveränderter Einstellungen. Durch Angabe eines Werts für `/wait` in Sekunden können Sie festlegen, wie lange die Richtlinienverarbeitung dauern darf, bis die Eingabeaufforderung wieder freigegeben wird. Nach Ablauf der Wartezeit (Standard: 600 Sekunden, `0` bedeutet ohne Wartezeit, `-1` hebt die Wartezeitbegrenzung auf) wird die Verarbeitung der Gruppenrichtlinien im Hintergrund fortgesetzt. Durch `/logoff` kann die Abmeldung des Benutzers nach der Aktualisierung der Gruppenrichtlinieneinstellungen erzwungen werden, mit `/boot` wird ein Neustart des Systems initiiert. Diese Aktionen

können für Richtlinien erforderlich sein, die nicht im Hintergrund verarbeitet werden können. Durch `/sync` bewirken Sie eine synchrone Ausführung der nächsten Richtlinienanwendung im Vordergrund.

PoSh: Das Cmdlet `invoke-gpupdate` stellt die PowerShell-Alternative zum Programm `gpupdate` dar.

---

klist

`klist [-lh LogonId.HighPart] [-li LogonId.LowPart] tickets | tgt | purge | sessions | kcd_cache | get {spn} | add_bind Domäne DC | query_bind | purge_bind`

Zeigt Kerberos-Ticketinformationen an (aktuelles Ticket oder das ticketgenehmigende Ticket) oder löscht alle zwischengespeicherten Tickets (`purge`). Sitzungsinformationen können angezeigt werden (`sessions`) und ebenso Informationen zu zwischengespeicherten Tickets (`kcd_cache`).

---

ksetup

`ksetup` *`Optionen`*

Konfiguriert einen Windows-Computer für die Verwendung eines MIT-Kerberos-Servers zur Benutzerauthentifizierung anstelle einer Active-Directory-Domäne.

Wichtige Optionen

`[/Domain` *`Domäne`***`] [/Server`** *`Server`***`]`**

Gibt die passende Domäne und/oder den Server zur Durchführung des Befehls an.

`/SetRealm` *`DNS-Domäne`*

Gibt den gewünschten Kerberos-Bereich an.

`/RemoveRealm` *`Realmname`*

Löscht alle Informationen zum angegebenen Bereich aus der Registrierung.

/MapUser *Prinzipal Name*

Verbindet den angegebenen Kerberos-Prinzipal mit einem lokalen Namen. Ohne Angabe des lokalen Namens wird die Zuordnung für den Prinzipal gelöscht.

{*/AddKDC* | */DelKDC*} *Bereich KDCName*

Fügt eine KDC-Adresse des angegebenen Bereichs hinzu oder löscht sie.

{*/AddKPasswd* | */DelKPasswd*} *Bereich Server*

Fügt einen Kerberos-Passwortserver hinzu oder löscht ihn.

/ChangePassword *Alt Neu*

Ändert das Kennwort des Anwenders via `KPasswd`. Benötigt `/Domain`.

/SetComputerPassword *Kennwort*

Setzt das Kennwort für den lokalen Computer.

---

ktpass

ktpass /out *Datei* **/princ** *Benutzer* **[/pass** *Kennwort*| ***]**
[+rndpass] /mapuser [/crypto DES-CBC-CRC] [/crypto DES-CBC-MD5] [/crypto RC4-HMAC-NT] [/crypto AES256-SHA1] [/crypto AES128-SHA1] [/crypto all]

Dieses unter Windows Server verfügbare Programm generiert eine Kerberos-Schlüsseldatei für MIT-Kerberos-Interoperabilität (kann mit einer bestehenden Datei */etc/krb5.keytab* auf Unix-Systemen zusammengeführt werden). Der Hauptzweck dieses Befehls ist das Erstellen von Kontozuordnungen für Unix-basierte Kerberos-Dienste, um Active-Directory-basierte KDCs (Schlüsselverteilungscenter) zu nutzen.

Die Optionen definieren den Namen der Ausgabedatei sowie den Kerberos-Namen mit Kennwort. Verwenden Sie anstelle eines Kennworts ein Sternchen (*), um das Kennwort an der Eingabeaufforderung einzugeben, oder anstelle des Parameters `/pass +rndpass`, um ein per Zufallsgenerator erzeugtes Kennwort zu verwenden. `/mapuser` erstellt die Zuordnung des

Kerberos-Namens zum lokalen Konto. Dabei wird das Benutzerkonto mit dem dazugehörigen Unix-Dienst verbunden, der dem Kerberos-Namen entspricht. Die Option `/crypto` ändert das verwendete Verschlüsselungsschema. Weitere Optionen entnehmen Sie bitte der Hilfe zum Befehl.

---

net accounts

`net accounts` *`Optionen`* **`[/domain]`**

Verändert die Kennwortrichtlinie des lokalen Systems oder der Domäne.

Optionen

`/minpwlen:`*`n`*

Setzt den Minimalwert der Passwortlänge auf *n* Zeichen. Der Standardwert ist für eine neu eingerichtete Domäne 7, auf eigenständigen Servern und PCs 0, und der Wert kann ohne weitere Richtlinienänderungen zwischen 0 und 14 liegen.

`/maxpwage:`*`n`*

Stellt das Passwortalter auf maximal *n* Tage ein. Der Standardwert ist 42, und der Wert kann zwischen 1 und 999 liegen. Wenn `unlimited` statt *n* angegeben wurde, wird keine Kennwortänderung aufgrund des Alters erzwungen.

`/minpwage:`*`n`*

Stellt das Intervall zwischen Kennwortänderungen ein. Der Standardwert ist 0, und der Wert kann abhängig vom Wert für das maximale Passwortalter zwischen 0 und 998 liegen.

`/uniquepw:`*`n`*

Es werden *n* bereits genutzte Passwörter gespeichert. Der Standardwert ist für Domänenserver und -clients 24, für eigenständige Server und PCs 0, und der Wert kann zwischen 0 und 24 liegen.

`/forcelogoff:{`*`Minuten`* **`| no}`**

Zwingt den Benutzer nach *Minuten* zur Abmeldung, sobald die erlaubte Anmeldezeit überschritten ist. `no` ist der Standardwert und deaktiviert das erzwungene Abmelden.

PoSh: Zum Setzen von Kennwortrichtlinien in Active-Directory-Domänen finden sich in PowerShell die Cmdlets `set-addefaultdomainpasswordpolicy` und `set-adfinegrainedpasswordpolicy`. Folgender Befehl setzt die Mindestlänge der Kennwörter in der Domäne *domaene.de* auf zwölf Zeichen und erzwingt »komplexe« Kennwörter:

```
set-addefaultdomainpasswordpolicy -identity domaene.de
-minpasswordlength 12 -complexityenabled $true
```

---

netdom

`netdom` *`Operation`* **`[`***`Optionen`***`]`**

Verwaltet Domänen und Vertrauensstellungen.

Optionen

`/domain:`*`Domäne`* **`[/userd:[`***`Domäne`***`\]`***`Benutzer`* **`/passwordd:`***`Kennwort`* **`| *]`**

Gibt die zu verwaltende Domäne an. Falls ein bestimmter Domänencontroller verwendet werden soll, geben Sie diesen mit der Syntax *Domäne**Servername* an. Optional kann der Benutzerkontext mit Kennwort zur Authentifizierung angegeben werden (* fordert zur Eingabe des Kennworts an der Eingabeaufforderung auf).

`/usero:[`*`Domäne`***`\]`***`Benutzer`* **`/passwordo:`***`Kennwort`* **`| *`**

Gibt einen Benutzernamen und ein Kennwort zur Authentifizierung am zu bearbeitenden Computer an (* fordert zur Eingabe des Passworts an der Eingabeaufforderung auf).

`/securepasswordprompt`

Verwendung eines sicheren Pop-ups zur Eingabe der Anmeldeinformationen. Wird nur verwendet, wenn * als Kennwortwert

angegeben wurde.

Wichtige verfügbare Operationen

netdom join *Computer* **/d:***Domäne* **[/ou:***OU-DN***] [/reb:[***s***]]**

Fügt *Computer* der angegebenen Organisationseinheit in *Domäne* hinzu und erstellt das Computerkonto, falls es noch nicht existiert. /reb veranlasst einen Neustart des Computers nach *s* Sekunden (Standard: 20).

Das Schlüsselwort add kann anstelle von join verwendet werden, um das Computerkonto in der Domäne zu erstellen, ohne *Computer* der Domäne hinzuzufügen. Die Option /dc kann verwendet werden, um ein Computerkonto für einen Domänencontroller zu erstellen.

netdom move *Computer* **/d:***Domäne* **[/ou:***OU-DN***] [/reb:[***s***]]**

Verschiebt *Computer* in die angegebene Organisationseinheit in *Domäne*. /reb veranlasst einen Neustart des Computers nach *s* Sekunden (Standard: 20).

netdom renamecomputer *Computer* **/NewName:***NeuerName* **[/force] [/reb:[***s***]]**

Benennt einen Computer sowohl lokal als auch in der Domäne um. Sowohl der NetBIOS- als auch der DNS-Hostname werden geändert. Mit der Option /ud muss ein Benutzername zur Authentifikation an der Domäne angegeben werden. /force unterbindet die Aufforderung zur Bestätigung durch den angemeldeten Benutzer.

netdom remove *Computer* **/d:***Domäne* **[/reb:[***s***]]**

Entfernt ein Computerkonto (keinen Domänencontroller) aus einer Domäne.

netdom {verify | reset} *Computer* **/d:***Domäne* **[/reb:[***s***]]**

Verifiziert die sichere Verbindung zwischen einem Domänenmitglied und einem Domänencontroller (verify) oder setzt diese zurück (reset).

netdom query *Eintrag* **/d:***Domäne* **[/verify] [/reset]**

Fragt Informationen von der angegebenen Domäne ab. *Eintrag* ist eines der folgenden Schlüsselwörter: workstation, server, dc, ou, trust (Vertrauensstellung), pdc (PDC-Emulator) oder fsmo (FSMO-

Rolleninhaber). `trust` akzeptiert die Option `/direct`, mit der nur die direkt erstellten Vertrauensstellungen angezeigt werden, nicht die inbegriffenen.

`/verify` verifiziert die Funktion des sicheren Kanals, der für Vertrauensstellungen verwendet wird. `/reset` synchronisiert diese Verbindung neu.

`netdom trust` *`Vertrauende-Domäne`* **`/d:`***`Vertraute-Domäne [Option]`*

Verwaltet Vertrauensstellungen. *`Option`* gibt die durchzuführende Aktion an. Die möglichen Aktionen sind unter anderem `/add` (hinzufügen), `/remove` (entfernen), `/force` (erzwingen) – was bei `/remove` optional ist und gewaltsam entfernt bzw. das Entfernen erzwingt –, `/verify` (überprüfen) und `/two-way`. Vertrauensstellungen können auch mit einer nicht auf Windows basierenden Kerberos-Domäne aufgebaut werden. Dazu verwenden Sie folgende Optionen: `/add` (hinzufügen), `/realm` (Bereich), `/passwordt:`*`pwd`* (Passwort für eine neue Vertrauensstellung) und `/transitive` (für Nicht-Windows-Domänen standardmäßig deaktiviert). Die Option `/kerberos` kann mit der Option `/verify` verbunden werden, um den Befehl auf Nicht-Windows-Domänen auszuführen.

Eine nützliche Anwendung ist das Zurücksetzen des Computerkennworts, wenn dessen Vertrauensstellung zur Domäne verloren ging. Melden Sie sich dazu mit einem Administratorkonto am Computer an und führen Sie folgenden Befehl aus, wobei `server` für einen Domänencontroller steht und `user` für einen Domänenbenutzer mit dem Recht, das Computerkennwort zu ändern:

`netdom resetpwd /s:`*`server`* `/ud:`*`user`* `/pd:*`

PoSh: In PowerShell lässt sich diese Aufgabe mit dem Cmdlet `reset-computermachinepassword` erledigen. Die Syntax hierzu lautet:

```
reset-computermachinepassword [-credential
<PSCredential>] [-server Domänencontroller]
```

Einen Test auf eine funktionierende Vertrauensstellung können Sie mit dem Cmdlet `test-computersecurechannel` durchführen, eine Reparatur mit

dem zusätzlichen Parameter `-repair`.

---

ntdsutil

`ntdsutil`

Ermöglicht besondere Wartungsaufgaben am Active Directory, unter anderem eine autoritäre Wiederherstellung und einen erzwungenen Transfer von FSMO-Rollen (wenn ein Rolleninhaber nicht mehr startet). Die Funktionalität dieses Befehls entspricht im Wesentlichen einer Kombination aus `dsdbutil` und `dsmgmt`, daher sei auf die Beschreibung dieser Befehle verwiesen.

---

nltest

`nltest [`*`Optionen`*`]`

Ermöglicht die Abfrage umfangreicher Informationen zur Domänenkonfiguration, von denen hier nur die wichtigsten beschrieben sind. Zudem kann auch mit diesem Befehl der sichere Kanal von Servern geprüft und zurückgesetzt werden. Weitere Informationen erhalten Sie in der Hilfe zum Befehl.

Optionen

`/server:`*`Computer`*

Gibt den abzufragenden Server an.

`/dclist:`*`Domäne`*

Ermittelt die Domänencontroller von *`Domäne`*.

`/dcname:`*`Domäne`*

Ermittelt den PDC-Emulator von *`Domäne`*.

`/dsgetdc:`*`Domäne`* **`[`***`Optionen`***`]`**

Ermittelt diejenigen Domänencontroller von *`Domäne`*, die die in

Optionen angegebenen Eigenschaften aufweisen. Es können folgende angegeben werden: /pdc (PDC-Master), /ds (versucht, einen DC zu finden), /dsp (gibt DCs zurück), /gc (globaler Katalog), /kdc (KDC-Dienst muss laufen), /timeserv (Zeitserver), /gtimeserv (zuverlässiger Zeitserver), /netbios (*Domäne* ist ein NetBIOS-Name), /dns (*Domäne* ist ein DNS-Name), /ip (IP-Adresse wird ermittelt), /force (ermittelt alle Daten neu, Cache wird ignoriert), /writable (muss beschreibbar sein, gibt keine NT4-BDCs zurück), /avoid-self (der DC, auf dem nltest ausgeführt wird, wird nicht zurückgegeben), /ldaponly (zurückgegebene Server müssen nicht zwingend DCs, sondern nur LDAP-Server sein), /backg (zeigt nur gecachte Daten an), /site:*SiteName* (sucht bevorzugt in *Site-Name*), /ret_dns (es werden DNS-Namen zurückgegeben), /ret_netbios (es werden NetBIOS-Namen zurückgegeben).

/dsgetsite

Ermittelt den Standort des abgefragten Computers.

/dsgetsitecov

Gibt die Standorte zurück, die der angegebene Domänencontroller abdeckt.

/parentdomain

Ermittelt den Namen der übergeordneten Domäne.

/dsregdns

Registriert alle DC-spezifischen DNS-Einträge. Muss auf dem zu registrierenden DC ausgeführt werden. Mit der Option /server muss ein DNS-Server angegeben werden.

/dsderegdns:*FQDN-DC-Name*

Entfernt DC-spezifische DNS-Einträge des vollqualifiziert angegebenen DC. Mit der Option /server muss ein DNS-Server angegeben werden.

/shutdown:*Grund Sekunden*

Fährt den Server aus angegebenem Grund herunter.

/shutdown_abort

Bricht das Herunterfahren des Systems ab.

redircmp

redircmp *OU-DN*

Ändert den Standardspeicherort für neu erzeugte Computerkonten auf die als *Distinguished Name* angegebene Organisationseinheit im Active Directory.

redirusr

redirusr *OU-DN*

Ändert den Standardspeicherort für neu erzeugte Benutzerkonten auf die als *Distinguished Name* angegebene OU.

PoSh: Mit PowerShell können Sie die Standardspeicherorte für neu erzeugte Computer- und Benutzerkonten zwar anzeigen, aber nicht ohne die genannten Befehle ändern.

Zur Anzeige dieser Standardspeicherorte verwenden Sie den folgenden Befehl:

```
get-addomain | select computerscont*, userscont* | fl
```

rendom

rendom */Aktion* **[/dc:{***DC* | *Domäne***}] [/user:***Benutzer***] [/pwd:***Kennwort* | ***]**

Kann alle zur Umbenennung einer Domäne oder einer Gesamtstruktur nötigen Aktionen durchführen. Mit den optionalen Parametern kann ein bestimmter DC oder ein beliebiger DC einer bestimmten Domäne angesprochen werden (`/dc`). Auch kann ein anderes Benutzerkonto als das aktuelle zur Herstellung der Verbindung verwendet werden.

Zur Umbenennung einer Domäne wird zunächst mit `rendom /list` die Datei *domainlist.xml* erzeugt, die Namensinformationen aller Domänen der Gesamtstruktur enthält. Nachdem die Namen in der XML-Datei in einem

Texteditor manuell abgeändert wurden, zeigt der Befehl `rendom /showforest`, wie die neue Struktur aussehen würde. Mit `rendom /upload` wird aus der XML-Datei ein Skript erzeugt, das in die Konfigurationspartition des Domain Naming Master hochgeladen wird. Zusätzlich wird im aktuellen Verzeichnis die Statusdatei *dclist.xml* erzeugt, die einen Eintrag pro DC enthält. Diese Datei sollte nach jeder weiteren Operation auf Fehler überprüft werden.

Wurden die auf den Domain Naming Master geschriebenen Informationen auf alle DCs repliziert, wird mit dem Befehl `rendom /prepare` jeder DC kontaktiert und geprüft, ob er für die Umbenennung bereit ist. Nach `/prepare` muss der in *dclist.xml* gespeicherte Status jedes DC »Prepared« lauten. Die eigentliche Umbenennung wird mit `rendom /execute` ausgelöst. Jeder DC, auf dem die Operation erfolgreich ausgeführt wurde, startet danach automatisch neu. In *dclist.xml* muss deren Status nun »Done« lauten.

Nach erfolgreicher Umbenennung muss die Sperrung der Gesamtstruktur (gegen Hinzufügen von Domänen, DCs und Vertrauensstellungen) mit `rendom /end` aufgehoben werden. Anschließend werden vom Umbenennungsvorgang zurückgebliebene Verzeichnisattribute mit `rendom /clean` entfernt.

Hinweis: Die Umbenennung von Domänen ist ein komplexer und daher fehlerträchtiger Vorgang, der nicht leichtfertig durchgeführt werden sollte.

---

repadmin

`repadmin` *`Kommando Argumente`* **`[`***`Optionen`***`]`**

Verwaltet zahlreiche Aspekte der AD-Replikation. Der Befehl bietet umfangreiche Optionen, von denen hier nur die wichtigsten aufgeführt werden. Für weitergehende Informationen sei auf die Hilfe zum Befehl verwiesen (`repadmin /?`), die unter anderem verschiedene Möglichkeiten auflistet, Detailinformationen abzurufen.

Bei vielen Kommandos können einzelne Domänencontroller oder Listen von als Verzeichnisdienstagenten (*Directory Service Agents*, DSAs) bezeichneten DCs angegeben werden. Neben IP-Adressen können dafür DNS-Namen verwendet werden, wobei das Sternchen (`*`) als Platzhalter dient. Zusätzlich können Domänencontroller anhand weiterer Kriterien bestimmt werden, von denen hier als wichtigstes `site:`*Standortname* erwähnt sei, was alle DCs eines AD-Standorts spezifiziert. Wird kein DSA angegeben, bezieht sich das Kommando auf das lokale System.

Optionen

`/u:`*`Domäne\Benutzer`* **`/pw:`***`Kennwort`***`| *`**

Definiert einen Benutzernamen mit Passwort zur Authentifizierung. Anstelle des Passworts kann ein Sternchen (`*`) angegeben werden, um an der Eingabeaufforderung nach dem Passwort gefragt zu werden.

Einfachere Kommandos

`/kcc [`*`DSA-Liste`***`] [/async]`**

Veranlasst die sofortige Neuberechnung der eingehenden Replikationstopologie durch den *Knowledge Consistency Checker* (KCC) auf den durch *`DSA-Liste`* angegebenen DCs. Optional wird der Vorgang asynchron durchgeführt.

`/queue [`*`DSA-Liste`***`]`**

Zeigt die Warteschlange der eingehenden Replikationsanforderungen an.

`/replicate` *`Ziel-DSA-Liste Quell-DSA-Name Verzeichnispartition`*

Löst die sofortige Replikation einer Verzeichnispartition von einem Quell-DC auf einen oder mehrere Ziel-DCs aus. Ist die Partition im Ziel schreibgeschützt, muss zusätzlich `/readonly` angegeben werden.

`/replsingleobj` *`Ziel-DSA-Liste Quell-DSA-Name Objekt-DN`*

Repliziert ein einzelnes Objekt von einem Quell-DC zu einem oder mehreren Ziel-DCs.

`/replsummary` *`DSA-Liste`*

Zeigt eine Zusammenfassung des Replikationsstatus einer Liste von DCs.

/rodcpwdrepl *Ziel-RODC-Liste Quell-DC Benutzer1-DN* **[***Benutzer2-DN***...]**

Repliziert die Kennwörter eines oder mehrerer Benutzer von einem Quell-DC auf einen oder mehrere RODCs.

/showattr *DSA-Liste Objekt-DN* **[/atts:***Attribut1***,***Attribut2***,...]**

Ermittelt den Wert aller oder bestimmter (/atts) Attribute eines Objekts auf den mit *DSA-Liste* angegebenen DCs.

/showrepl *DSA-Liste* **[***Namenskontext***] [/csv] [/errorsonly]**

Zeigt den Status eingehender Replikationen für die DCs der *DSA-Liste* an, optional begrenzt auf einen angegebenen Namenskontext. Die Ausgabe kann auf Fehler beschränkt werden (/errorsonly) sowie im CSV-Format erfolgen.

/showutdvec *DSA-Liste* **[***Namenskontext***] [***Optionen***]**

Zeigt die höchste vergebene USN (Aktualisierungssequenznummer) des Zieldomänencontrollers.

/syncall *DSA-Liste* **[***Namenskontext***] [***Optionen***]**

Repliziert die DCs einer *DSA-Liste* mit allen Replikationspartnern im gleichen Standort. Ohne Angabe eines Namenskontexts wird die Konfigurationspartition repliziert. An *Optionen* stehen unter anderem zur Verfügung: Replikation aller Namenskontexte (/A), auch standortübergreifende Replikation (/e), nur Replikation zwischen benachbarten Servern (/j).

/prp

Verwaltet die Kennwortrichtlinie für schreibgeschützte Domänencontroller (RODCs). Das Kommando kann sich auf einen einzelnen RODC beziehen, der dann durch seinen Namen bezeichnet wird. Stattdessen kann für *RODC-Name* auch ein Sternchen (*) angegeben werden, wodurch sich das Kommando auf alle RODCs bezieht. Dieser Platzhalter ist nur beim Verschieben (move) nicht erlaubt. Sicherheitsprinzipale (Benutzer, Gruppen, Computer) werden als *Distinguished Names* (DNs) angegeben.

view *RODC-Name* **{***Liste* **|** *Benutzer***}**

Zeigt entweder die aktuelle Kennwortreplizierungsrichtlinie für einen

Benutzer oder die auf einer der folgenden vier Listen enthaltenen Sicherheitsprinzipale (Benutzer/Computer): die vom RODC authentifizierten Prinzipale (`auth2`), die Prinzipale, deren Kennwörter vom RODC verwaltet werden (`reveal`), Prinzipale, deren Kennwörter vom RODC zwischengespeichert werden dürfen (`allow`), und solche, deren Kennwörter nicht zwischengespeichert werden dürfen (`deny`).

`add` *`RODC-Name`* **`allow`** *`Prinzipal`*

Fügt einen Sicherheits-Prinzipal der Zulassungsliste eines oder aller ROCDs hinzu. Die angegebenen RODCs dürfen Kennwörter der Mitglieder der Liste zwischenspeichern.

`delete` *`RODC-Name`* **{*`allow`* | *`auth2`*}** **{***`Prinzipal`* | `/all`}**

Löscht einen oder alle Sicherheits-Prinzipale von der Zulassungsliste (`allow`) oder löscht die Liste der Prinzipale, die vom RODC authentifiziert wurden (`auth2`). Bei letzterer Option ist nur `/all` zulässig.

`move` *`RODC-Name Gruppe`* **`[/noauth2cleanup] [/users_only | /comps_only]`**

Verschiebt alle Prinzipale aus der `auth2`-Liste in eine Gruppe, die erstellt wird, falls sie noch nicht besteht. Zusätzlich wird die Gruppe in die Zulassungsliste des RODC eingefügt. Optional werden die Prinzipale kopiert statt verschoben (`/noauth2cleanup`), oder es werden entweder nur Benutzer (`/users_only`) oder nur Computerkonten berücksichtigt (`/comps_only`).

PoSh: PowerShell bietet innerhalb des Active-Directory-Moduls zahlreiche Cmdlets zur Verwaltung von Replikation und Topologie. Diese beinhalten `ADReplication` als Namensbestandteil.

---

secedit

Dient der Erstellung und Anwendung von Sicherheitsvorlagen, mit denen sich die Sicherheitseinstellungen eines Computers konfigurieren lassen.

`secedit /export [/db` *`Datenbankdatei`***`] /cfg`** *`Vorlagendatei`*

[/areas *bereich1* **[***bereich2***] [...]] [/mergedpolicy][/log** *Logdatei***]**

Exportiert die Sicherheitseinstellungen aus *Datenbankdatei* oder die lokalen Systemeinstellungen in die *Vorlagendatei*. Wird /mergedpolicy angegeben, werden die effektiven Einstellungen (Domäne und lokal) exportiert, ansonsten nur die lokalen Einstellungen. Dabei kann die Ausgabe mit /areas auf bestimmte Bereiche beschränkt werden.

secedit /import /db *Datenbankdatei* **/cfg** *Vorlagendatei* **[/areas** *bereich1* **[***bereich2***] [...]] [/overwrite][/log** *Logdatei***] [/quiet]**

Importiert die Sicherheitseinstellungen aus *Vorlagendatei* in *Datenbankdatei*, die bei Angabe von /overwrite vorher geleert wird. Mit /areas kann der Import auf bestimmte Bereiche beschränkt werden.

secedit /configure /db *Datenbankdatei***[/cfg** *Vorlagendatei***] [/areas** *bereich1* **[***bereich2***] [...]] [/overwrite][/log** *Logdatei***] [/quiet]**

Importiert, falls angegeben, die Sicherheitseinstellungen aus *Vorlagendatei* in *Datenbankdatei*, die bei Angabe von /overwrite vorher geleert wird. Die resultierenden Sicherheitseinstellungen werden anschließend angewendet. Mit /areas kann der Import auf bestimmte Bereiche beschränkt werden.

secedit /generaterollback /cfg *Vorlagendatei* **/rbk** *Rollbackvorlage* **[/log** *Logdatei***][/quiet]**

Erstellt eine *Rollbackvorlage*, die zum Rückgängigmachen der in *Vorlagendatei* enthaltenen Einstellungen verwendet werden kann.

secedit /analyze /db *Datenbankdatei* **[/cfg** *Vorlagendatei***] [/overwrite] [/log** *Logdatei***][/quiet]**

Vergleicht die aktuellen Sicherheitseinstellungen mit denen einer *Datenbankdatei*, in die gegebenenfalls vor dem Vergleich die *Vorlagendatei* importiert wird. Die Ergebnisse werden in einen gesonderten Bereich der Datenbank geschrieben und können mit dem MMC-Snap-in *Sicherheitskonfiguration und analyse* eingesehen werden.

secedit /validate *Vorlagendatei*

Überprüft die Syntax der *Vorlagendatei*.

Optionen

/db *Datenbankdatei*

Gibt eine *Datenbankdatei* (Endung: *SDB*) für die Operation an. Die Datei wird erstellt, falls sie beim Import noch nicht existiert.

/cfg *Vorlagendatei*

Gibt eine *Vorlagendatei* (Endung: *INF*) für die Operation an.

/areas *Bereich1 [Bereich2]* **...**

Beschränkt die Operation auf einen oder mehrere der folgenden Bereiche: `securitypolicy` (Kontorichtlinien, Überwachungsrichtlinien, Ereignisprotokolleinstellungen und Sicherheitsoptionen), `group_mgmt` (eingeschränkte Gruppen), `user_rights` (Benutzerrechte und Privilegien), `regkeys` (Berechtigungen in der Registrierung), `filestore` (Berechtigungen im Dateisystem), `services` (Einstellungen für Systemdienste).

/log *Protokolldatei*

Schreibt das Protokoll der Operation in *Protokolldatei*. Falls `/log` nicht angegeben wurde, wird das Protokoll in *%SystemRoot%\security\logs\scesrv.log* gespeichert.

/quiet

Es erfolgen keine Sicherheitsabfragen zur Bestätigung.

---

setspn

setspn *Optionen* **[-p]**

Verwaltet *Service Principal Names* (Dienstprinzipalnamen) im Active Directory. Mit `-p` wird die Anzeige des Fortschritts unterdrückt, was beim Umleiten der Ausgabe in eine Datei sinnvoll ist.

Optionen

[-*c* | -*u*] *Konto*

Gibt an, ob das angegebene Konto ein Computerkonto (`-c`) oder ein Benutzerkonto (`-u`) ist. Wird der Parameter nicht angegeben, wird das angegebene Konto als Computerkonto interpretiert, sofern ein entsprechender Computer vorhanden ist, andernfalls als Benutzername.

-l *Konto*

Zeigt alle für ein Computer- oder Benutzerkonto registrierten SPNs an.

-q *SPN*

Prüft, ob der angegebene SPN im Verzeichnis existiert.

-t *Domäne*

Sucht in der angegebenen Domäne und kann mehrfach angegeben werden. Durch Angabe von `-f` wird die Suche auf die Gesamtstruktur ausgedehnt, zu der die angegebene Domäne gehört.

-x

Sucht nach doppelten SPNs.

-s [-f] *SPN Kontoname*

Fügt einen beliebigen SPN für *`Kontoname`* hinzu, nachdem auf Duplikate geprüft wurde. Optional kann die Überprüfung mit `-f` von der Domäne auf die Gesamtstruktur ausgedehnt werden.

-d *SPN Konto*

Löscht einen beliebigen SPN für *`Konto`*.

-r *Computer*

Setzt die für *`Computer`* registrierten SPNs auf die Standardwerte (*`HOST/Computername`* und *`HOST/Computer-FQDN`*) zurück.

PoSh: Mit den Cmdlets `get-adcomputer` und `set-adcomputer` können Sie sich die Liste der Dienstprinzipalnamen eines Systems anzeigen lassen bzw. ändern.

Die Syntax für die Anzeige lautet wie folgt:

```
Get-ADComputer -Identity Computername
-Properties ServicePrincipalNames |
Select-Object -ExpandProperty ServicePrincipalNames
```

Ein Beispiel für die Bearbeitung hingegen wäre:

```
Set-ADComputer -ServicePrincipalNames
@{Add='WSMAN/testdc01','WSMAN/testdc01.domaene.de'}
```

Als Parameter wird hier ein Array mit einem oder mehreren Dienstprinzipalnamen und dem zugeordneten System sowie einem der Schlüsselwörter `Add`, `Remove` oder `Replace` mitgegeben, um SPNs hinzuzufügen, zu entfernen oder zu ändern.

Cluster

cluster

Verwaltet Microsoft-Cluster. Der Befehl kann auf allen aktuellen Windows-Versionen ausgeführt werden, um lokale oder entfernte Cluster zu administrieren. Da dieser Befehl explizit als veraltet gekennzeichnet ist, werden Details zur Syntax an dieser Stelle nicht behandelt.

Hinweis: Auch in Server 2025 ist der Befehl zu Migrationszwecken noch enthalten, wurde aber bereits mit Windows Server 2016 durch entsprechende PowerShell-Cmdlets ersetzt.

PoSh: Um unter Windows Server 2025 Zugriff auf die Datei *cluster.exe* zu bekommen, müssen Sie das Feature *Failovercluster-Befehlsschnittstelle* installieren. Das können Sie beispielsweise mit folgendem PowerShell-Befehl tun:

```
add-windowsfeature rsat-clustering-cmdinterface
```

PowerShell enthält eine umfangreiche Sammlung von Cmdlets zur Administration von Clustern. Sie finden sie durch eine Suche nach dem Namensbestandteil `cluster`. Installieren Sie sie, indem Sie das Feature `rsat-clustering-powershell` installieren. Es folgen einige Beispiele:

`get-cluster -domain` *`domaene.de`*

Zeigt die Cluster in der angegebenen Domäne an.

`get-cluster | format-list –property *`

Zeigt die Eigenschaften der Cluster in der aktuellen Domäne.

`(get-cluster).dynamicquorum = 1`

Aktiviert die dynamische Quorum-Funktionalität für den Cluster.

Für die Verwaltung von Clusterknoten können Sie unter anderem das Cmdlet `get-clusternode` verwenden.

get-clusternode -name *Cluster1*

Zeigt Namen und Status jedes Knotens in *Cluster1*.

get-clusternode *node1* **| get-clusterresource**

Zeigt die dem Knoten *node1* zugehörigen Ressourcen an.

remove-clusternode -name *node3* **-force**

Entfernt den Knoten *node3* aus dem Cluster, ohne auf eine Bestätigung zu warten.

Für die Verwaltung von Clustergruppen befinden sich die Cmdlets `add-clustergroup`, `get-clustergroup`, `move-clustergroup`, `remove-clustergroup`, `start-clustergroup` und `stop-clustergroup` im Funktionsumfang von PowerShell.

Mit dem Cmdlet `get-clusternetwork` verwalten Sie Clusternetzwerke in der Windows PowerShell.

get-clusternetwork

Zeigt den Status eines oder mehrerer Netzwerke im Cluster an.

(get-clusternetwork –name "*Clusternetz* **1").name = "***Clusternetz 3***"**

Benennt das Netzwerk *Clusternetz 1* in *Clusternetz 3* um.

Das Cmdlet `get-clusternetworkinterface` unterstützt bei der Verwaltung der Netzwerkschnittstellen.

get-clusternetworkinterface

Zeigt die mit dem Netzwerk assoziierten Netzwerkschnittstellen und deren Status an.

Zur Verwaltung von Clusterressourcen sowie ihrer Typen und Abhängigkeiten bietet PowerShell 17 verschiedene Cmdlets. Diese finden Sie durch eine Suche nach dem Namensbestandteil `clusterresource`.

Zum Erzeugen von clusterspezifischen Logdateien enthält PowerShell die Cmdlets `get-clusterlog` zum Erzeugen von clusterspezifischen Logdateien und `set-clusterlog` zum Anpassen von Einstellungen wie Größe der Logdateien und Detailtiefe.

set-clusterlog -size *1024* **-level** *2*

Setzt die Maximalgröße der Logdatei auf 1.024 MByte und die Detailtiefe

auf 2.

---

nlb

Verwaltet Network-Load-Balancing-Cluster. Dieser Befehl ist erst verfügbar, nachdem NLB installiert wurde. In Windows Server 2025 werden PowerShell-Cmdlets für die Verwaltung von NLB-Clustern bereitgestellt, und *nlb.exe* wird nur noch aus Kompatibilitätsgründen mitgeliefert. Zur Administration entfernter Rechner muss dort zunächst Remote Control aktiviert werden, was jedoch sicherheitstechnisch problematisch ist.

nlb {suspend | resume | start | stop | drainstop | query} [*Cluster*[:*Host*] | all [local | global]]

Führt einen Befehl für einen Cluster (auf einem angegebenen Rechner), alle Cluster des lokalen Computers oder alle globalen Computer aus, die Teil des Clusters sind. `suspend` und `resume` pausieren bzw. reaktivieren einen Cluster. `start`/`stop` startet bzw. stoppt den Cluster. `drainstop` veranlasst den Cluster, keine neuen Verbindungen mehr anzunehmen, aber bestehende Verbindungen abzuarbeiten. `query` fragt den Status des Clusters ab.

nlb {enable | disable | drain} [*VIP*[:*Port* | :*all*] | all [:*Port* | :*all*]] [*Cluster*[:*Host*] | all [local | global]]

Aktiviert, deaktiviert oder lässt eine Regel auslaufen (keine neuen Verbindungen), deren Portbereich den angegebenen Port enthält. Der erste Satz optionaler Parameter spezifiziert alle oder bestimmte virtuelle IPs. Der optionale Parametersatz findet sich identisch im übergeordneten Abschnitt.

nlb queryport [*VIP*:] *Port* [*Cluster*[:*Host*] | all [local | global]]

Zeigt Informationen über eine Portregel an.

nlb {reload | display | params} [*Cluster* | all]

Lädt Clusterparameter aus der Registrierung (`reload`), zeigt ausführliche Informationen zu NLB-Parametern und Clusterstatus (`display`) oder die aktuelle NLB-Konfiguration an, abgefragt vom Kernel-Treiber (`params`).

nlb ip2mac *Cluster*

Zeigt die MAC-Adresse des Clusters an.

PoSh: Um die Werkzeuge zum Verwalten von NLB auf Windows Server 2025 zu installieren, führen Sie folgenden PowerShell-Befehl aus:

```
add-windowsfeature rsat-nlb
```

Danach können Sie sowohl auf *nlb.exe* als auch auf jene PowerShell-Cmdlets zugreifen, die `nlb` als Namensbestandteil haben.

Remotedesktopdienste

Die meisten der Befehle in diesem Abschnitt sind unter Windows Server 2025 nach Installation der Rolle *Remotedesktopdienste* ⟶ *Remotedesktop-Sitzungshost* verfügbar.

---

change logon

`change logon {/enable | /disable | /query | /drain | /drainuntilrestart}`

Aktiviert bzw. deaktiviert die Anmeldung von Clientsitzungen (`/enable` und `/disable`) bzw. fragt den aktuellen Status ab (`/query`). Bei Serverbetriebssystemen kann zusätzlich der *drain*-Modus eingeschaltet werden, der keine neuen Verbindungen zulässt, sondern nur Wiederverbindungen zu vorhandenen Sitzungen. Das kann dauerhaft (`/drain`) oder bis zum nächsten Neustart (`/drainuntilrestart`) aktiviert werden. Der Befehl `chglogon` ist in Funktion und Syntax identisch.

---

change port

Konfiguriert die Zuordnung (Mapping) serieller Ports. Der Befehl `chgport` entspricht diesem Befehl in Funktion und Syntax.

`change port` *`Port1=Port2`*

Ordnet *`Port1 Port2`* zu.

`change port /d` *`Port1`*

Löscht eine Zuordnung.

`change port /query`

Zeigt aktuelle Zuordnungen an.

change user

change user {/execute | /install | /query}

Schaltet den Remotedesktopdienst eines Servers in den Ausführungs- bzw. Installationsmodus oder zeigt den aktuellen Modus an. Der Befehl `chgusr` ist identisch.

logoff

logoff [*Sitzungsname* | *Sitzungskennung***]**
[/server:*servername***] [/v] [/vm]**

Beendet eine Benutzersitzung lokal, auf dem angegebenen Server oder in einem virtuellen Computer ohne weitere Rückfrage. Die zu beendende Sitzung kann durch ihren Namen oder ihre Kennung identifiziert werden. Beide Informationen lassen sich über die Terminal-Diensteverwaltung ermitteln. Die Sitzungskennung kann auch im Task-Manager angezeigt werden: *Ansicht* → *Spalten auswählen* → *Sitzungskennung*. Ohne Angabe von Name oder Kennung wird der aktuelle Benutzer abgemeldet. Bei Verwendung der Option `/vm` muss immer die Sitzungskennung angegeben werden.

msg

msg {*BenutzerName* | *SitzungsID* | *SitzungsName* | *@DateiName* | *****} [/server:***ServerName***] [/time:***Timeout***] [/v][/w]**
Nachrichtentext

Sendet eine Textnachricht an Benutzer des lokalen oder angegebenen Servers. Eine Datei mit einer Liste von Benutzernamen, Sitzungs-IDs oder Sitzungsnamen kann angegeben werden. Ein Stern als Adressat steht für alle Benutzer des Servers. Falls kein Timeout angegeben wird, bleibt die Nachricht 60 Sekunden auf dem Bildschirm des Adressaten. `/w` wartet auf

eine Bestätigung durch den Benutzer, `/v` zeigt ausführliche Informationen an.

PoSh: PowerShell besitzt im Gegensatz zum Versand von E-Mail-Nachrichten kein natives Cmdlet, um Pop-up-Nachrichten an andere Benutzer zu versenden, kann aber *msg.exe* als Werkzeug verwenden, um diese Art Nachrichten selektiv zu versenden. Zudem gibt es Drittanbieterprojekte im Web, die sich dieser Thematik annehmen.

---

mstsc

mstsc [*Verbindungsdatei* | **/v:***Servername***[**:*Port***]] [/admin] [/restrictedadmin] [/remoteguard] [/f] [/w:***Breite* **/h:***Höhe***] [/public] [/span] [**/multimon**] [/shadow] [/control] [/noconsentprompt] [/migrate] [/edit** *Verbindungsdatei***]**

Stellt eine Remotedesktopverbindung zum angegebenen Server oder gemäß den Einstellungen einer RDP-Verbindungsdatei her. `/admin` verbindet zur Konsolensitzung (nur möglich unter Serverbetriebssystemen), `/restrictedadmin` nutzt in Kombination damit einen eingeschränkten Verwaltungsmodus, bei dem keine Anmeldeinformationen zum Zielgerät übermittelt werden. `/remoteguard` verbindet ähnlich wie `/restrictedadmin`, ermöglicht jedoch die Authentifizierung weiterer Anforderungen durch die Weiterleitung an das Gerät, von dem aus die Sitzung gestartet wurde. `/f` schaltet in den Vollbildmodus, `/w` und `/h` erlauben die Angabe der gewünschten Auflösung.

Durch Angabe von `/public` wird die Option zum Speichern des Kennworts deaktiviert (gedacht zur Nutzung auf öffentlich zugänglichen Maschinen). Mit `/span` erstreckt sich das RDP-Fenster über mehrere Bildschirme, `/multimon` konfiguriert das Bildschirmlayout der Remotedesktopsitzung identisch mit der aktuellen clientseitigen Konfiguration.

`/shadow [Sitzungs-ID]` ermöglicht das Anzeigen der gewählten Sitzung, `/control` deren Steuerung, `/noconsentprompt` unterbindet, dass der an der Zielsitzung angemeldete Benutzer beidem zustimmen muss.

`/migrate` erstellt aus alten, mit dem Client-Verbindungsmanager erstellten Verbindungsdateien neue RDP-Dateien. `/edit` öffnet vorhandene RDP-Dateien zur Bearbeitung mit der grafischen Variante von `mstsc`.

---

query process

query process [*Benutzername* | **/id:***SitzungsID* | *Sitzungsname* | *ProzessID* | *Programmname.exe* | *****] [/server:***ServerName***]**

Fragt Informationen über laufende Prozesse im Kontext eines Benutzers, einer Sitzung, eines bestimmten Programms oder zu einer Prozess-ID auf *ServerName* ab. Der Befehl `qprocess` ist ein Alias dazu.

PoSh: Mittels `get-process` können ausführliche oder selektive Informationen zu ausgeführten Prozessen abgerufen werden.

---

query session

query session [*BenutzerName* | *SitzungsID* | *SitzungsName***] [/server:***ServerName***] [/mode] [/flow] [/connect][/counter] [**/vm**]**

Zeigt Informationen zu einer bestimmten oder allen Sitzungen auf *ServerName* an. `/mode` zeigt aktuelle Leitungseinstellungen an, `/flow` aktuelle Flusssteuerungseinstellungen, `/connect` die Verbindungseinstellungen, `/counter` verschiedene Zähler (Gesamtzahl an Sitzungen usw.). `/vm` gibt Informationen zu Sitzungen in virtuellen Computern aus.

---

query termserver

query termserver [*ServerName***] [/domain:***DomänenName***] [/address] [/continue]**

Zeigt Remotedesktop-Hostserver im gesamten Netzwerk (ohne Parameter) oder in einer Domäne an. Bei Angabe eines Servernamens kann mit `/address` dessen Netzwerkadresse ermittelt werden. `/continue` schaltet das Warten auf eine Eingabe nach jeder Bildschirmseite ab. Der Befehl `qappsrv` ist ein Alias dazu.

---

query user

`query user [`*`BenutzerName`*** `|` ***`SitzungsID`*** `|` ***`SitzungsName`***`]`**
`[/server:`*`ServerName`***`]`**

Zeigt alle an einem Server angemeldeten Benutzer oder folgende Informationen zu einem angegebenen Benutzer: Benutzername, Sitzungsname, Sitzungs-ID, Sitzungsstatus, Leerlaufzeit, Anmeldezeit. Der Befehl `quser` stellt einen Alias dazu dar.

PoSh: Das Cmdlet `Get-RDUserSession` kann vergleichbare Informationen bereitstellen, allerdings nur für Systeme einer Remotedesktopdienste-Bereitstellung.

---

rdpsign

`rdpsign /sha256` ***`Hash`*** **`[/q | /v] [/l]`** ***`RDP-Datei`***

Signiert eine RDP-Datei mit einem Zertifikat aus dem Zertifikatsspeicher (des Computers oder Benutzers), das durch seinen Hash (ohne Leerzeichen) angegeben wird. Bei dem Vorgang wird die Originaldatei überschrieben. Der Parameter `/l` bewirkt einen Testlauf, bei dem die Datei nicht verändert wird. Optional kann der stille (`/q`) oder der ausführliche Modus (`/v`) verwendet werden.

---

reset session

reset session [*SitzungsID* | *SitzungsName***]**
[/server:*ServerName***][/v]**

Zurücksetzen (Löschen) einer Sitzung. `/v` zeigt ausführliche Informationen an. Der Alias dazu ist der Befehl `rwinsta`.

---

shadow

shadow {*Sitzungsname* **|** *Sitzungs-ID***} [/server:***Servername***]**
[/v]

Ermöglicht die Remotesteuerung einer aktiven, durch *Sitzungsname* oder *Sitzungs-ID* bestimmten Sitzung eines anderen Benutzers auf einem Remotedesktop-Sitzungshostserver.

---

tscon

tscon {*Sitzungs-ID* | *Sitzungsname***}**
[/dest:*Zielsitzungsname***] [/password:***Kennwort***][/v]**

Verbindet die aktuelle Sitzung oder eine durch *Sitzungs-ID* oder *Sitzungsname* angegebene Sitzung mit *Zielsitzungsname*, wobei die Quellsitzung getrennt wird. Falls die Zielsitzung einem anderen Benutzer gehört, muss dessen Kennwort angegeben werden. `/v` zeigt ausführliche Informationen an.

---

tsdiscon

tsdiscon {*Sitzungs-ID* | *Sitzungsname***} [/server:***Servername***]**
[/v] [/vm]

Trennt die aktuelle oder angegebene Sitzung auf *Servername*. `/v` zeigt ausführliche Informationen an, `/vm` trennt die Verbindung für die Sitzung auf dem Server oder innerhalb des virtuellen Computers. Ohne Angabe eines

Parameters wird die lokale Windows-Sitzung gesperrt.

---

tskill

tskill {*ProzessID* | *Prozessname***} [/server:***Servername***]**
[/id:{*Sitzungs-ID* | **/a}][/v]**

Beendet einen Prozess in einer angegebenen oder allen (/a) Sitzungen auf *Servername*. /v zeigt ausführliche Informationen an.

---

tsprof

Kopiert die Remotedesktopdienste-Konfiguration von einem Benutzer zu einem anderen oder setzt den Remotedesktopdienste-Profilpfad.

tsprof /update /domain:*DomänenName* | **/local**
/profile:*Profilpfad BenutzerName*

Ändert den Remotedesktopdienste-Profilpfad eines lokalen oder Domänenbenutzers nach *Profilpfad*.

tsprof /q /domain:*DomänenName* | **/local** *BenutzerName*

Zeigt den Remotedesktopdienste-Profilpfad eines lokalen oder Domänenbenutzers an.

tsprof /copy /domain:*DomänenName* | **/local**
[/profile:*Profilpfad***]**
QuellBenutzer ZielBenutzer

Kopiert die Remotedesktopdienste-Konfiguration von *QuellBenutzer* nach *ZielBenutzer*, wobei optional der angegebene *Profilpfad* für *ZielBenutzer* gesetzt wird.

Installation, Deployment, Updates

msiexec

Die Kommandozeilenschnittstelle des Windows Installer. Im Folgenden werden die wichtigsten Optionen dieses mächtigen Befehls beschrieben. Für eine vollständige Beschreibung sei auf die Hilfe zum Befehl und das Windows Installer SDK verwiesen.

msiexec /i *MSI-Paket* **[TRANSFORMS=***MST-Datei1*;*MST-Datei2*;...**]**

Installiert *MSI-Paket*. Optional können Transformationsdateien angewendet werden.

msiexec /a *MSI-Paket*

Führt eine administrative Installation in einer Netzwerkfreigabe durch (das MSI-Paket muss sich bereits am gewünschten Ziel befinden).

msiexec {/uninstall | /x} {*MSI-Paket* | *Produkt-GUID***}**

Deinstalliert die durch den Paketnamen oder die GUID angegebene Software.

Optionen

/quiet

Hintergrundmodus ohne Benutzerinteraktion.

/passive

Unbeaufsichtigter Modus, nur die Statusleiste wird angezeigt.

/q[n|b|r|f]

Art der Benutzeroberfläche: Mit `n` wird keine Benutzeroberfläche angezeigt, mit `b` eine einfache, mit `r` eine reduzierte und mit `f` die

vollständige Benutzeroberfläche der Installationsroutine.

/norestart | /promptrestart | /forcerestart

Legt das Neustartverhalten nach Abschluss der Installation fest: Bei /`norestart` wird kein Neustart des Systems durchgeführt, bei Angabe von /`promptrestart` wird der Anwender gefragt, und bei /`forcerestart` wird der Computer immer neu gestartet.

/log *Protokolldatei*

Ein Protokoll mit Status- und Fehlermeldungen wird in die angegebene Datei geschrieben.

Eigenschaft1=*Wert1 Eigenschaft2*=*Wert2[...]*

Setzt paketspezifische Eigenschaften auf den jeweils angegebenen Wert. Damit können Einstellungen vorgenommen werden, die sonst beim Benutzer abgefragt würden, z.B. der Pfad zum Zielverzeichnis.

---

pnputil

pnputil [*Optionen***]**

Verwaltet Treiberpakete im Windows Driver Store. Unter anderem können Treiber zum Driver Store hinzugefügt werden, die dann von Benutzern ohne Administratorrechte installiert werden können. Unter Server Core kann dieser Befehl zur Installation von Treibern in der Eingabeaufforderung verwendet werden. Treiber werden durch die Angabe ihrer INF-Datei spezifiziert, die alle zur (De)Installation nötigen Angaben enthält. Da dieser Befehl im Funktionsumfang beträchtlich erweitert wurde, kann an dieser Stelle nur ein kurzer Überblick angeboten werden.

Optionen

/add-driver {*INF-Datei* **| *.inf*} [/install]**

Fügt einen Treiber oder alle Treiber (*.*inf*) aus diesem Verzeichnis dem Driver Store hinzu und installiert sie gegebenenfalls (/`install`).

/delete-driver *INF-Datei* **[/uninstall] [/force] [/reboot]**

Löscht einen Treiber aus dem Driver Store. Die zusätzlichen Optionen sind `/uninstall` zum Deinstallieren des Treibers, `/force` erzwingt das Löschen auch bei aktueller Verwendung durch Geräte, und `/reboot` erzwingt einen Neustart zum Abschluss der Deinstallation.

`/export-driver {`*`oem#.inf`*` | *} [`*`Zielverzeichnis`*`]`

Exportiert den Treiber mit der genannten *inf*-Datei oder alle Treiber in das angegebene Zielverzeichnis.

`/enum-classes [`*`Optionen`*`]`

Listet alle Geräteeinrichtungsklassen des Systems auf.

`/enum-containers [`*`Optionen`*`]`

Listet alle Gerätecontainer im System auf.

`/enum-drivers [/class` *`Name`* **`|`** *`GUID`*`] [/files] [/ids] [/devices] [/format {`*`txt`*` | `*`xml`*` | `*`csv`*`}] [/output-file [`*`Dateiname`*`]]`**

Zeigt Treiberpakete von Drittanbietern im Treiberspeicher.

`/enum-devices [`*`Optionen`*`]`

Listet alle Geräte im System auf.

`/enum-devicetree [`*`Optionen`*`]`

Listet die Gerätestruktur vergleichbar zum Geräte-Manager auf.

`/enum-interfaces [`*`Optionen`*`]`

Listet alle Geräteschnittstellen im System auf.

`/disable-device [`*`Optionen`*`]`

Deaktiviert angegebene Geräte.

`/enable-device [`*`Optionen`*`]`

Aktiviert angegebene Geräte.

`/restart-device [`*`Optionen`*`]`

Startet angegebene Geräte neu.

`/remove-device [`*`Optionen`*`]`

Entfernt angegebene Geräte vom System.

PoSh: Mit den Cmdlets `get-pnpdevice`, `disable-pnpdevice`, `enable-pnpdevice` und `get-pnpdeviceproperty` stellt PowerShell grundlegende Funktionen zum Verwalten von Geräten und Schnittstellen bereit.

sysprep

sysprep [*/quiet***] [***/generalize***] [***/audit***] [***/oobe***] [***/reboot*** | ***/shutdown*** | ***/quit***] [***/unattend:Dateiname***] [***/mode:vm***]**

Bereitet das lokale System zum Klonen bzw. Duplizieren vor. `sysprep` generalisiert das System für eine spätere Verteilung oder für die Installation auf einer neuen Hardwareplattform. Mit `/generalize` werden folgende Anpassungen des Systems vorgenommen:

- Löschen der Aktivierungsinformationen.
- Löschung aller Einträge in der Ereignisanzeige.
- Löschung aller Wiederherstellungspunkte.
- Deaktivierung des lokalen Kontos *Administrator* und Löschung seines Profils.
- Entfernen von PnP-Gerätetreibern, die während der Installation hinzugefügt wurden.
- Beendigung der Domänenmitgliedschaft.

Der Parameter `/generalize` muss zusammen mit `/oobe` oder `/audit` verwendet werden. `/oobe` führt nach dem Neustart zur Willkommensseite analog zur Neuinstallation eines Rechners, `/audit` ermöglicht den Neustart für weitere Tests und Anpassungen der Systemkonfiguration.

Die Parameter `/reboot` (Neustart), `/shutdown` (Herunterfahren) und `/quit` (nur `sysprep` beenden) steuern die Aktion des Betriebssystems nach Abschluss der Systemvorbereitung durch `sysprep`. Der Parameter `/mode:vm` dient der Systemvorbereitung von Abbildern, die als virtuelle Maschinen ausgerollt werden. Da hierbei die Treiber unangetastet bleiben, sollte die Plattform des Hostsystems identisch sein.

Durch eine Antwortdatei, deren Pfad im Parameter `unattend` angegeben wird, können die obigen Systemanpassungen und weitere Einstellungen verändert werden.

Hinweis: Das Programm befindet sich im Verzeichnis

%SystemRoot%\System32\sysprep.

---

usoclient

Dieser Befehl ersetzt den Vorgänger *wuauclt.exe*, der in früheren Windows-Versionen mit den Parametern `/detectnow /updatenow` eine Updatesuche und -installation von der Eingabeaufforderung aus starten konnte. `usoclient` (steht für *Update Session Orchestrator Client*) übernimmt diese Aufgabe, enthält sich aber auch jeglicher Bildschirmausgabe. Daher können Sie sich in der Datei *C:\Windows\SoftwareDistribution\ReportingEvents.log* oder in der Aufgabenplanung unter *Microsoft* ⟶ *Windows* ⟶ *Update Orchestrator* über den Erfolg der Ausführung kundig machen.

`usoclient {startscan | startinteractivescan}`

Beginnen die Suche nach Updates. Auf einigen Systemen funktioniert nur einer der Befehle, zum Testen schauen Sie am besten zeitgleich auf die grafische Oberfläche für Windows Update.

`usoclient {startdownload | scaninstallwait}`

Mit einem dieser Parameter lösen Sie das Herunterladen ausstehender Updates aus.

`usoclient startinstall`

Dieser Parameter sorgt für die Installation der heruntergeladenen Updates.

`usoclient restartdevice`

Mit diesem Befehl schließlich können Sie den Rechner neu starten, aufgrund der Intransparenz des Befehls sollten Sie allerdings eine Mindestwartezeit von 30 Minuten vor dem Neustart einplanen. (Auch dieser Parameter funktioniert nicht immer zuverlässig.)

PoSh: Im separat zu installierenden Modul `pswindowsupdate` finden Sie etliche Cmdlets zum Abfragen, Installieren, Entfernen und Unterbinden der Installation von Windows-Updates.

Skripte und Batchdateien

Skripte und Batchdateien dienen dazu, eine Abfolge mehrerer Befehle durch einen einzigen Aufruf auszuführen. Batchdateien, auch als »Stapelverarbeitungsdateien« bekannt, sind Textdateien, die einen oder mehrere Befehle enthalten und mit den Dateierweiterungen *.cmd* oder *.bat* gespeichert werden.

CALL

CALL *Datei* | :*Marke* **[***Argumente***]**

Führt die angegebene Batchdatei aus oder springt zu *Marke*.

---

choice

choice [/c *Auswahloptionen***] [/cs] [/n] [/t** *Timeout* **/d** *Standard***] /m** *Text*

Zeigt *Text* an und fordert den Anwender zur Auswahl einer Option auf. Die Variable ERRORLEVEL wird auf die Nummer der gewählten Option gesetzt (bei 1 beginnend), bei einem Abbruch mit *Strg+C* oder *Strg+Pause* auf 0, bei einem Fehlerzustand auf 255. Daher ist es erforderlich, die Auswertung bei Verwendung der ERRORLEVEL-Werte absteigend auszuführen.

Optionen

/c *Auswahloptionen*

Definiert die wählbaren Buchstaben (Standard: JN).

/cs

Unterscheidet zwischen Groß- und Kleinschreibung.

/n

Zeigt die wählbaren Buchstaben nicht an. Normalerweise werden sie, durch Kommata getrennt, in eckigen Klammern angezeigt.

/t *Timeout* **/d** *Standard*

Wählt die Option *Standard* nach *Timeout* Sekunden, wenn keine Eingabe erfolgt ist.

CLS

CLS

Löscht den bisherigen Bildschirminhalt.

PoSh: Das Cmdlet `Clear-Host` führt dasselbe für den Ausgabebereich einer PowerShell-Sitzung durch. Alternativ können Sie auch einen der Aliase `clear` oder `cls` verwenden.

---

cscript

cscript [*Skript***] [***Optionen***]**

Führt ein Skript unter dem Windows Scripting Host aus und zeigt dessen Ausgabe im Fenster der Eingabeaufforderung an (im Gegensatz zu `wscript`). Beachten Sie, dass die Optionen des `cscript`-Befehls mit zwei Schrägstrichen (`//`) gekennzeichnet sind, um sie von den Optionen des Skripts zu unterscheiden.

cscript `//h:`*cscript*

Ändert den Standard-Skripthost auf `cscript`, um Skripte, die nur durch Eingabe des Skriptnamens gestartet werden, unter `cscript` statt unter `wscript` ausführen zu lassen.

PoSh: Mittels PowerShell können Sie bei Bedarf auch auf den Windows Scripting Host zurückgreifen. Folgender Zweizeiler erzeugt ein einfaches Pop-up am Bildschirm:

```
$wshshell = new-object -comobject wscript.shell
$wshell.popup("Meldungstext.",0,"Titelzeile")
```

ECHO

ECHO *Text*

Gibt den angegebenen *Text* auf Standard-Output aus. Zum Erzeugen einer Leerzeile wird `ECHO.` verwendet (Befehlsname mit angehängtem Punkt ohne Leerzeichen). Anstelle von *Text* kann auch eine Variable genutzt werden, beispielsweise `ECHO %COMPUTERNAME%` zur Anzeige des Rechnernamens.

ECHO [*ON* | *OFF*]

Aktiviert oder deaktiviert die Anzeige der ausgeführten Befehlszeile (Standard: aktiviert). Ohne Parameter zeigt `ECHO` die aktuelle Einstellung an.

EXIT

EXIT [/b [*Rückgabewert***]]**

Beendet eine Batchdatei (samt laufendem Befehlsinterpreter *cmd.exe* bzw. bei interaktiver Eingabe nur diesen) sofort. Siehe auch `goto :EOF`.

Innerhalb von Batchdateien ist es oft unerwünscht, das gesamte Fenster (*cmd.exe*) beim Beenden zu schließen. Dies kann mit dem Parameter `/b` unterbunden werden, wobei optional noch ein Rückgabewert angegeben werden kann (mit anderen Worten, der Error-level wird gesetzt).

FOR

Schleifenkonstrukt. Dieser äußerst mächtige Befehl bietet zahlreiche Optionen, von denen hier nur die wichtigsten beschrieben sind. Weitere Informationen können Sie der Hilfe (`help for`) entnehmen.

Wenn dieser Befehl in einer Batchdatei eingesetzt wird, muss `%%var` anstelle von `%var` verwendet werden. Beachten Sie, dass durch diesen Befehl Variablen bei unterschiedlicher Groß- und Kleinschreibung im Namen als unterschiedlich betrachtet werden.

FOR [/d] *%var* **in (***Dateiliste***) do** *Befehl*

Durchläuft *Dateiliste* (Wildcards sind zulässig) und speichert das jeweils aktuelle Listenelement in `%var` zur Verwendung durch *Befehl*.

Falls die angegebene Liste Wildcards enthält, werden bei Angabe von `/d` nur Verzeichnisse durchlaufen, keine Dateien.

FOR /l *%var* **in (***Start*,*Schritt*,*Ende***) do** *Befehl*

Durchläuft eine Schleife mit der Schrittweite *Schritt* von *Start* bis *Ende* und speichert den aktuellen Schleifenindex in `%var`.

`FOR /l` kann verwendet werden, um *Befehl* mehrfach auszuführen: `(1,1,5)` erzeugt die Reihe `1 2 3 4 5` und führt den Befehl demnach fünfmal aus. Falls eine bestimmte einfache Zahlenreihe für *Befehl* benötigt wird, kann diese durch Variation der Start-, Schritt- und Ende-Werte erzeugt werden: `(2,2,100)` gibt alle geraden Zahlen zwischen 2 und 100 aus.

FOR /f ["*Optionen***"]** *%var* **in (***Dateiliste***) do** *Befehl*

Durchläuft *Dateiliste* (Wildcards sind zulässig) und verarbeitet jede Datei nacheinander zeilenweise. Jede eingelesene Textzeile wird gemäß den angegebenen Optionen in Tokens (Bausteine) aufgespalten, die in `%var` und gegebenenfalls weiteren Variablen gespeichert werden. Ohne Angabe von Optionen wird die Zeile an Leerzeichen aufgespalten; `%var` nimmt das erste erhaltene Token auf (Zeilenbeginn bis zum ersten Leerzeichen bzw. bis zum Zeilenende).

Optionen

eol=*c*

Das Zeichen *c* wird als Zeilenende interpretiert. Alle darauf folgenden Zeichen werden ignoriert. Dient der Definition eines Zeichens, mit dem Kommentare in den Dateien gekennzeichnet werden.

skip=*n*

Die ersten *n* Zeilen zu Beginn jeder Datei werden übersprungen.

delims=*xxx*

Gibt einen Satz von Trennzeichen an, an denen die Zeile in Tokens aufgespalten wird (Standard: Tab und Leerzeichen).

tokens=*a,b,c-d*

Bezeichnet die Nummern der zu speichernden Tokens. Das können entweder einzelne Ziffern oder ein Bereich sein. Für die zusätzlichen Tokens werden automatisch weitere Variablen reserviert, die im Alphabet auf `%var` folgen.

Beispiel

```
FOR /f "eol=# tokens=1,3-5* delims=,;" %i in (datei.txt)
do echo %i %j %k %l %m
```

Setzt die Raute als Kommentarzeichen: Zeilen, die damit beginnen, werden ignoriert. Als Trennzeichen zwischen Tokens werden Komma und Semikolon verwendet. Die Variablen `%i` bis `%m` nehmen, in dieser Reihenfolge, die Tokens 1, 3, 4, 5 und den Rest der Zeile auf. Auf diese Weise können einzelne Spalten (z.B. einer CSV-Datei) herausgelöst und durch einen Befehl weiterverarbeitet werden.

---

forfiles

```
forfiles [/p <Verzeichnis>] [/m <Zeichenkette>] [/s] [/c "
<Befehl>"] [/d [{+|-}][{<Datum> | <Tage>}]]
```

Wendet den in Anführungszeichen einzuschließenden Befehl auf jede Datei einer Auswahl an (Standardaktion: Anzeigen des Dateinamens).

Optionen

/c "*Befehl***"**

Gibt den Befehl an, der ausgeführt werden soll. Nachfolgende Konstrukte

können unter Beachtung von Groß- und Kleinschreibung innerhalb von *Befehl* eingesetzt werden:

@FILE	aktueller Dateiname
@FNAME	Dateiname ohne Erweiterung
@EXT	Dateierweiterung
@PATH	Verzeichnispfad der Datei
@RELPATH	Verzeichnispfad der Datei relativ zum aktuellen Pfad
@ISDIR	TRUE, falls Verzeichnis, sonst FALSE
@FSIZE	Dateigröße
@FDATE	Änderungsdatum der Datei
@FTIME	Änderungszeit der Datei

Der Standardbefehl ist "cmd /c echo @FILE".

/p *Verzeichnis*

Gibt den Verzeichnispfad an, in dem die Suche beginnen soll (die Standardeinstellung ist das aktuelle Verzeichnis).

/m *Zeichenkette*

Wählt die Dateien aus, die der angegebenen Zeichenkette entsprechen. *Zeichenkette* kann Platzhalter enthalten (Standardeinstellung ist *).

/d [+|–]*tt.mm.jjjj* **|** *n*

Wählt nur Dateien aus, die zum angegebenen Datum modifiziert wurden (**–** bedeutet am/vorher und **+** bedeutet am/danach). Wird anstelle eines Datums eine Zahl angegeben, werden die Dateien ausgewählt, die vor mindestens *n* Tagen modifiziert (**–**) oder am bzw. mindestens *n* Tage nach dem aktuellen Datum modifiziert wurden (**+**).

/s

Führt den Befehl auch in Unterverzeichnissen aus.

/v

Erzeugt eine ausführliche Ausgabe.

GOTO

GOTO *Marke*

Springt zu der angegebenen Sprungmarke innerhalb der Batchdatei. Eigene Sprungmarken werden durch ein mit Doppelpunkt eingeleitetes Wort in einer separaten Zeile definiert.

GOTO *Fehler*

Springt zur Zeile mit dem Inhalt *Fehler*.

GOTO :*EOF*

Springt zum Ende der aktuellen Batchdatei, ohne dass die Marke *EOF* definiert werden muss. (Setzt aktivierte Befehlserweiterungen voraus.)

IF

Befehl zur Auswertung von Bedingungen. Der Sinn des logischen Ausdrucks wird in allen Fällen durch Verwendung des Schlüsselworts `not` umgekehrt. Wird mit `ELSE` ein alternativer Befehl angegeben, muss sich das Schlüsselwort `ELSE` in derselben Zeile wie der erste Befehl befinden, zudem sollte dieser mit Klammern umschlossen werden.

IF [not] *errorlevel n Befehl*

Führt *Befehl* aus, wenn der Wert der Variablen `errorlevel` größer oder gleich *n* ist bzw. nicht (`not`) größer oder gleich *n* ist.

IF [not] defined *Variable Befehl*

Führt *Befehl* aus, wenn die angegebene *Variable* definiert bzw. nicht definiert ist.

IF [not] exist *Datei (Befehl1)* **[ELSE** *Befehl2***]**

Führt *Befehl1* aus, wenn die angegebene *Datei* existiert bzw. nicht existiert; falls die Bedingung nicht zutrifft, wird *Befehl2* ausgeführt.

IF [/i] [not] *Zeichenkette1 Operator Zeichenkette2 Befehl*

Führt *Befehl* aus, wenn der Vergleich der Zeichenketten »wahr« bzw. »falsch« liefert. Die möglichen Operatoren sind folgende:

== | EQU

Gleich.

NEQ

Nicht gleich.

LSS

Kleiner als.

LEQ

Kleiner als oder gleich.

GTR

Größer als.

GEQ

Größer als oder gleich.

Alle Vergleichsoperatoren unterscheiden zwischen Groß- und Kleinschreibung. Mit `/i` deaktivieren Sie die Unterscheidung.

PAUSE

PAUSE

Fordert zum Drücken einer beliebigen Taste auf; die Verarbeitung der Batchdatei wird so lange angehalten.

POPD

POPD

Wechselt zum letzten mit `PUSHD` gespeicherten Verzeichnis und löscht alle von `PUSHD` erstellten temporären Laufwerkbuchstaben.

PROMPT

PROMPT *Text*

Ändert die Eingabeaufforderung in *Text*. *Text* kann unter anderem (siehe `prompt/?`) die folgenden Werte enthalten:

$D, $T
: Aktuelles Datum, aktuelle Uhrzeit.

$G
: Größer-Zeichen (>).

$N
: Aktueller Laufwerkbuchstabe.

$P
: Aktueller Laufwerkbuchstabe und Verzeichnis.

$S
: Leerzeichen.

$_
: Zeilenvorschub.

PUSHD

PUSHD [*Pfad***]**

Sichert das aktuelle Verzeichnis zum Abruf durch `POPD` und wechselt dann in das angegebene Verzeichnis. Bei Aufruf ohne Argumente wird die aktuelle Verzeichnisliste angezeigt.

REM

REM *was-auch-immer*

Leitet eine Kommentarzeile ein, die vom Befehlsinterpreter ignoriert wird.

SET

Zeigt Variablenwerte an und erlaubt ihre Veränderung.

SET [*var***]**

Zeigt den Wert der angegebenen oder aller definierten Variablen an.

SET *var=Zeichenkette*

Setzt den Wert der Variablen auf *Zeichenkette*, im Namen der Variablen sind Gleichheitszeichen nicht zulässig.

SET /a *var=Numerischer-Ausdruck*

Setzt den Wert der Variablen auf das Ergebnis des numerischen Ausdrucks. Hierbei werden zahlreiche Operatoren unterstützt.

SET /p *var=[Eingegebene Zeichenfolge]*

Fordert den Anwender zur Zuweisung eines Werts zur angegebenen Variablen auf.

Hinweis: Mit `cmd /v` können Sie die verzögerte Erweiterung von Umgebungsvariablen aktivieren oder deaktivieren. Diese ermöglicht das korrekte Arbeiten mit sich während einiger Skriptabläufe ändernden Variablen. Um diese veränderten Variablen anzusprechen, wird statt des %-Zeichens das Ausrufezeichen verwendet, also z.B. *!PATH!* anstelle *%PATH%*.

SETLOCAL... ENDLOCAL

`SETLOCAL` definiert den Anfang einer lokalen Umgebung innerhalb der Batchdatei. Die so begonnene lokale Umgebung wird durch `ENDLOCAL` beendet. Die vor dem Aufruf von `SETLOCAL` gültigen Werte sind dann wieder aktiv.

SHIFT

SHIFT [/*n***]**

Verschiebt die Argumente einer Batchdatei um eine Stelle nach vorn, beginnend beim Argument *n* (falls angegeben), das entfällt.

timeout

`timeout /t` *`n`* **`[/nobreak]`**

Entspricht `pause`, wartet jedoch maximal *`n`* Sekunden auf einen Tastendruck. Wird als Zeit der Wert `-1` angegeben, wird ohne Begrenzung gewartet. Bei Angabe von `/nobreak` wird die angegebene Zeit immer abgewartet, ein Tastendruck also ignoriert.

TITLE

`TITLE` *`Zeichenfolge`*

Setzt den Fenstertitel der aktiven Instanz der Eingabeaufforderung (*cmd.exe*).

waitfor

Sendet Signale an ein entferntes System oder wartet auf solche. Ein Signal kann eine beliebige Zeichenkette sein. `waitfor` kann verwendet werden, um voneinander abhängige Aktionen auf verschiedenen Computern in der richtigen Reihenfolge auszuführen.

`waitfor [/t` *`Timeout`***`]`** *`Signal`*

> Wartet auf das angegebene Signal. Optional kann mit `/t` eine maximale Wartezeit angegeben werden. Ein Signal ist eine maximal 225 Zeichen lange Zeichenkette, die aus den Zeichen a–z, A–Z, 0–9 sowie den ASCII-Zeichen mit den Codes 128–255 bestehen darf.

`waitfor /si` *`Signal`* **`[/s`** *`Computer`***`] [/u`** *`Domäne\Benutzer`***`] [/p`** *`Kennwort`***`]**

> Sendet das angegebene Signal entweder per Broadcast an alle Computer der Domäne oder an den angegebenen Computer, gegebenenfalls im Kontext des angegebenen Benutzerkontos.

wscript

`wscript [`*`Skript`*`] [`*`Optionen`*`]`

Führt ein Skript unter dem Windows Scripting Host aus und zeigt jede Zeile von dessen Ausgabe in einem eigenen Dialogfenster an (im Gegensatz zu `cscript`). Beachten Sie, dass die Optionen des `wscript`-Befehls mit zwei Schrägstrichen (`//`) gekennzeichnet sind, um sie von den Optionen des Skripts zu unterscheiden.

`wscript //h:`*`wscript`*

Ändert den Standard-Skripthost auf `wscript`, um Skripte, die nur durch Eingabe des Skriptnamens gestartet werden, unter `wscript` statt unter `cscript` ausführen zu lassen. Das ist die Standardeinstellung.

Zertifikate

certreq

`Verwaltet Zertifikatsanforderungen. Hier sind ausgewählte Optionen beschrieben. certreq -v -? liefert eine ausführliche Hilfe.`

`certreq [-submit] [Optionen] [Anforderungsdatei-Ein [Zertifikatdatei-Aus [Zertifikatkettendatei-Aus [Vollständige Antwortdatei-Aus]]]]`

Sendet eine Zertifikatsanforderung an eine CA (*Certificate Authority*, Zertifizierungsstelle). Falls die Eingabedatei (*Request-FileIn*) nicht angegeben wird, erscheint ein Öffnen-Dialog zur Auswahl.

`certreq -retrieve [Optionen] Anforderungs-ID [Zertifikatdatei-Aus [Zertifikatkettendatei-Aus [Vollst. Antwortdatei-Aus]]]`

Holt die Antwort auf eine vorherige Zertifikatsanforderung. Die benötigte *RequestID* wird von `certreq -submit` angezeigt.

`certreq –new [Optionen] [Richtliniendatei-Ein [Anforderungsdatei-Aus]]`

Erstellt die Zertifikatsantragsdatei *RequestFileOut* aus den Informationen der INF-Datei *PolicyFileIn*.

`certreq -accept [Optionen] [ZertKettedatei-Ein | VollstAntwDatei-Ein | ZertDatei-Ein]`

Akzeptiert ein Zertifikat oder eine Antwort von der CA.

`certreq -policy [Optionen] [Anforderungsdatei-Ein [Richtliniendatei-Ein [Anforderungsdatei-Aus [PKCS10-Datei-Aus]]]]`

Erstellt einen Antrag zur Kreuzzertifizierung von CAs oder für eine untergeordnete Zertifizierungsstelle aus einem Antrag oder CA-Zertifikat.

`certreq -sign [Optionen] [Anforderungsdatei-Ein [Anforderungsdatei-Aus]]`

Signiert einen Antrag zur Kreuzzertifizierung von CAs oder für eine

untergeordnete Zertifizierungsstelle.

Optionen

`-AdminForceMachine`

Anträge werden im Kontext des lokalen Systems verschickt.

`-any`

Erzwingt Übermittlung des Codierungstyps.

`-attrib` *`Attribute`*

Spezifiziert Attribute in der Form *Name1:Wert1\nName2:Wert2*.

`-binary`

Ausgabedateien werden statt Base64-codiert im Binärformat erstellt.

`-cert` *`CertID`*

Spezifiziert das Signierungszertifikat anhand des allgemeinen Namens (CN) oder von Serien-Nr., SHA-1-Schlüssel oder Hash.

`-config [`*`CAComputer\CAName`***`]`**

Ohne Verwendung von `-config` erscheint eine Dialogbox zur Abfrage der zu verwendenden CA. Wird `-config` ohne den Namen einer CA angegeben, wird die Standard-CA verwendet.

`-crl`

Fügt den Base64-codierten Ausgabedateien *CertChainFileOut* und *RequestFileOut* Zertifikatssperrlisten (CRLs) hinzu.

`-f`

Überschreibt vorhandene Dateien ohne Nachfrage.

`-q`

Stiller Modus: Es werden keine interaktiven Dialoge angezeigt.

`-policyserver` *`Richtlinienserver`*

URL oder ID des Richtlinienservers, alternativ `*` für GUI-Auswahl.

`-rpc`

Weist die AD-Zertifikatsdienste an, RPC statt DCOM zu verwenden.

PoSh: Mit dem Cmdlet `get-certificate` können Sie mit der PowerShell Zertifikate anfordern und installieren oder die Zertifikate von früheren

Anträgen abholen.

---

certutil

Bietet eine Vielzahl von Optionen zur Arbeit mit CAs, Zertifikaten, Schlüsseln usw. Hier ist lediglich eine Auswahl der Optionen beschrieben. `certutil v -?` liefert eine sehr ausführliche Hilfe.

`certutil {-encode | -decode | -decodehex}` *`Eingabedatei Ausgabedatei`*

En- bzw. decodiert *`Eingabedatei`* von/nach Base64 bzw. Hex.

`certutil -hashfile` *`Datei`*

Berechnet den SHA-1-Hash einer *`Datei`*.

`certutil -deny` *`Anforderungs-ID`*

Verweigert eine ausstehende Anfrage.

`certutil -resubmit` *`Anforderungs-ID`*

Übermittelt eine ausstehende Anfrage erneut.

`certutil -setAttributes` *`Anforderungs-ID Attribute`*

Setzt die Attribute für eine ausstehende Anfrage.

Format: *Name1:Wert1\nName2:Wert2*.

`certutil -revoke` *`Seriennummer(n)`* **`[`***`Grund`***`]`**

Sperrt die als kommaseparierte Liste angegebenen Zertifikate bzw. hebt eine Sperrung wieder auf, wenn als *`Grund`* `-1` angegeben wird.

`certutil -dump`

Gibt Informationen zu einer CA aus.

`certutil -CAinfo` *`Information`*

Zeigt bestimmte oder alle (`*`) Informationen zu einer Zertifizierungsstelle an.

`certutil -ca.cert` *`Ausgabedatei`*

Zeigt das CA-Zertifikat an und schreibt es in *`Ausgabedatei`*.

`certutil -ca.chain` *`Ausgabedatei`*

Zeigt die Zertifizierungsstellen-Zertifikatskette an und schreibt sie in *`Ausgabedatei`*.

certutil -getCRL *Ausgabedatei*

Schreibt die Zertifikatssperrliste in *Ausgabedatei*.

certutil -renewCert [*CAComputer\ÜbergeordneteCA***]**

Erneuert das CA-Zertifikat bei der übergeordneten Zertifizierungsstelle.

certutil {-*schema* | -*view*}

Gibt das Schema bzw. die Zertifikatansicht einer CA aus.

certutil -backup *Sicherungsverzeichnis* **[incremental]**

Sichert die Zertifikatsdienste in das angegebene Verzeichnis – falls angegeben, inkrementell, ansonsten wird eine Vollsicherung durchgeführt.

certutil -restore *Sicherungsverzeichnis*

Stellt die Zertifikatsdienste aus einer Sicherung wieder her.

certutil -store [*Zertifikatsspeicher***]**

Gibt die in einem Zertifikatsspeicher enthaltenen Zertifikate aus. Es können unter anderem diverse lokale und per LDAP erreichbare Speicher angesprochen werden.

certutil -addStore *Zertifikatsspeicher Eingabedatei*

Fügt *Zertifikatsspeicher* ein Zertifikat oder eine Sperrliste hinzu.

certutil -delStore *Zertifikatsspeicher Eingabedatei*

Löscht ein Zertifikat oder eine Sperrliste aus *Zertifikatsspeicher*.

certutil -dsPublish *Eingabedatei* **[***Zertifikatsspeicher***]**

Veröffentlicht ein Zertifikat oder eine Sperrliste im AD.

certutil -template

Zeigt die Vorlagen an.

certutil -templateCAs *Vorlage*

Zeigt die CAs für eine Vorlage an.

certutil -machineInfo *Domäne\Computername***$**

Zeigt Informationen zu einem Computerkonto an, unter anderem *Service Principal Names* (SPNs).

certutil -dcinfo [*verify* | *deletebad* | *deleteall*]

Zeigt Informationen zu Domänencontrollerzertifikaten an, die optional überprüft (`verify`) oder gelöscht (`deletebad`, `deleteall`) werden

können.

`certutil -key`

Zeigt die installierten Schlüsselcontainer sortiert nach CSPs an.

`certutil -delkey` *`Schlüsselcontainer`*

Löscht den angegebenen Schlüsselcontainer.

`certutil -dsdel` *`CA-Name`*

Löscht die im AD gespeicherten Daten zu einer CA. Neben dem Deinstallieren der Zertifikatsdienste muss dieser Befehl zum Löschen einer CA abgesetzt werden, um alle zugehörigen Informationen zu entfernen.

Optionen

`-config` *`CAComputer\CAName`*

Legt die zu verwendende Zertifizierungsstelle (CA) fest. Ohne diese Angabe wird die Standard-CA verwendet.

`-f`

Überschreibt vorhandene Dateien ohne Nachfrage.

`-gmt`

Zeigt Uhrzeiten in GMT an.

`-privatekey`

Auch Daten privater Schlüssel werden angezeigt.

`-seconds`

Zeigt Uhrzeiten inklusive Sekunden und Millisekunden an.

`-user`

Spezifiziert einen Benutzer- statt einen Computerzertifikatsspeicher.

`-v`

Zeigt ausführliche Informationen an.

PoSh: Sie finden zahlreiche Cmdlets zur Verwaltung der Zertifikatsdienste in Windows PowerShell. Direkt bezogen auf die Zertifikatsdienste im Active Directory finden Sie diese nach Installation der Rolle *Zertifizierungsstelle* mit dem Befehl `add-windowsfeature adcs-certauthority`. Führen Sie

zur Auflistung den Befehl `get-command -module adcsadministration` aus.

Andere Cmdlets rund um die Verwaltung von Zertifikaten enthalten den Namensbestandteil `certificate`, lassen sich also mit `get-command *certificate*` auffinden.

Die Wiederherstellungsumgebung

Windows PE stellt die Grundlage für die Wiederherstellungsumgebung (WinRE) dar. Diese enthält unter anderem eine vollwertige Eingabeaufforderung, an der sich viele der in diesem Buch beschriebenen Befehle nutzen lassen. Zusätzlich können nicht vom Explorer/Desktop abhängige grafische Programme gestartet werden, z.B. `regedit`. Mit dem Dialog *Datei* ⟶ *Öffnen* von Notepad lässt sich ein Dialogfeld verwenden, das es zumindest erlaubt, auf Laufwerkbuchstaben und Ordner visuell zuzugreifen, sie anzuzeigen oder per Kontextmenü darauf zuzugreifen, solange es sich nicht um versteckte Dateien/Ordner handelt und nichts ausgeführt/geöffnet werden soll. Dabei empfiehlt es sich, in der Auswahlliste *Dateityp* von *Text* auf *Alle Dateien* umzustellen.

Die Wiederherstellungsumgebung wird durch Anwählen folgender Optionen beim Booten von der Windows-DVD gestartet: *Sprachauswahl* ⟶ *Computerreparaturoptionen* ⟶ *Problembehandlung* ⟶*Erweiterte Optionen* ⟶ *Eingabeaufforderung*.

Optional kann die Wiederherstellungsumgebung auch auf der Festplatte installiert werden.

Konstrukte in Batchdateien

Die im Folgenden beschriebenen Konstrukte sind in Batchdateien nützlich, die vom Kommandozeileninterpreter *cmd.exe* ausgeführt werden.

:Marke

Sprungmarke (Ziel eines `goto`- oder `call`-Befehls).

%m

Das Argument mit der Nummer *m* (die Argumente 10 und höher können über den `shift`-Befehl ausgewertet werden).

%~cm

Das modifizierte Argument Nummer *m*. *c* kann einen der folgenden Werte annehmen, wenn nur ein Teil des Arguments ausgewertet werden soll:

f:

Vollständiger Pfad.

d:

Nur Laufwerkbuchstabe.

p:

Nur Pfad.

n:

Nur Dateiname.

x:

Nur Dateierweiterung.

s:

Es werden DOS-Namen im 8.3-Format verwendet (kann in Verbindung mit `n` und `x` eingesetzt werden).

$PATH:

Durchsucht die Umgebungsvariable `PATH` nach dem Argument und gibt die erste Übereinstimmung aus (vollständiger Pfad inklusive Dateiname). Wird keine Übereinstimmung gefunden, wird ein leerer String ausgegeben.

*%**

Alle angegebenen Argumente.

%var%

Wert der Variablen `var`. Es kann sich dabei um eine Skriptvariable oder eine Umgebungsvariable handeln.

errorlevel

Interne Variable, die den Rückgabewert (Fehlercode) des letzten Befehls enthält; siehe auch die Beschreibung des `if`-Befehls.

Windows PowerShell – Grundlagen

PowerShell ist der Nachfolger der bislang für Batchdateien und Skripte verwendeten Betriebssystemkomponenten *cmd.exe* und Windows Scripting Host (siehe `cscript` und `wscript`). Sie ist seit Windows 7/Server 2008 R2 im Betriebssystem enthalten, für ältere Versionen kann sie von Microsoft heruntergeladen und installiert werden.

Durch die Objektorientierung von PowerShell können Objekte Auskunft über sich selbst geben und zu ihren Fähigkeiten »befragt« werden. Dazu wird der Befehl `get-member` verwendet, der im folgenden Beispiel die Eigenschaften und Methoden eines Prozessobjekts (konkret: des PowerShell-Prozesses) auflistet:

```
get-process powershell | get-member
```

Leistungsstarke Befehle zum Filtern und Sortieren erleichtern die Auswahl der benötigten Daten: `where-object` filtert eine Objektliste, wobei die Variable `$_` stets das aktuelle Objekt der automatisch in einer Schleife durchlaufenen Liste repräsentiert. `sort-object` sortiert eine Liste von Objekten, `select-object` wählt gewünschte Objekteigenschaften zur Weiterverarbeitung aus, `format-table` gibt Daten tabellarisch aus:

```
Get-Process | Where-Object {$_.handlecount -gt 1000} |
Sort-Object cpu -desc | Select-Object
processname,path,cpu,workingset | Format-Table -auto
```

In dieser Befehlskette (Pipeline) wird die Liste der laufenden Prozesse ermittelt, auf Prozesse mit mehr als 1.000 geöffneten Handles gefiltert und absteigend nach CPU-Nutzung sortiert. Dann werden die gewünschten Eigenschaften der einzelnen resultierenden Prozesse angegeben: Name, Pfad, CPU- und Speichernutzung. Schließlich werden die ermittelten Daten tabellarisch formatiert mit automatisch ermittelter Spaltenbreite ausgegeben.

Variablen sind in PowerShell nichts anderes als benannte Objekte. Sie werden mit einem vorangestellten Dollarzeichen gekennzeichnet und können neben üblichen Datentypen wie Zahlen, Zeichenketten, Arrays und Hashes (Dictionaries) beliebige Objekte wie z.B. Datumsobjekte aufnehmen:

```
[System.DateTime] $date = get-date
$date.Subtract([system.datetime] "01.01.2025").days
```

Die Variable `$date` wird im Beispiel auf das aktuelle Systemdatum gesetzt. Der zweite Befehl subtrahiert davon ein früheres (oder künftiges) Datum und gibt ein TimeSpan-Objekt zurück. Dessen Eigenschaft `days` enthält die Anzahl der Tage, die zwischen den Daten liegen.

Neben Variablen werden Funktionen und die in verschiedenen Programmier- und Skriptsprachen üblichen Kontrollstrukturen (`if, switch, do while, do until, for, foreach`) unterstützt.

Gängige Operatoren zur Addition, Subtraktion usw. (`+-*/%`) sind genauso enthalten wie kombinierte Zuweisungsoperatoren (z.B. Addition und anschließend Zuweisung: `+=`). Vergleiche werden, um Verwechslungen zu vermeiden, nicht durch das Gleichheitszeichen ausgeführt, sondern durch eigene Operatoren: `-eq` (Gleichheit), `-ne` (Ungleichheit), `-gt` (größer als), `-ge` (größer gleich), `-lt` (kleiner als), `-le` (kleiner gleich).

In PowerShell sind etliche Aliase vordefiniert, die aus der Eingabeaufforderung oder Linux bekannte Befehlsnamen den PowerShell-Kommandos zuordnen. Somit sind folgende Zeilen möglich:

```
cd HKLM:
dir SOFTWARE
```

Zunächst wird in das die Registrierung darstellende »Laufwerk« *HKEY_LOCAL_MACHINE* gewechselt, und anschließend werden die Unterschlüssel von *HKLM\Software* ausgegeben. Eine nach Namen sortierte Liste der vordefinierten Aliase gibt `Get-Alias | Sort-Object Name` aus. Aliase verhalten sich oftmals nicht exakt wie der Originalbefehl, mit dem sie durch gleiche Namenswahl Gleichheit suggerieren.

Zum Erlernen von PowerShell ist es hilfreich, anstelle der Eingabeaufforderung die PowerShell-Konsole oder eine PowerShell-Instanz

im Terminal standardmäßig zu öffnen und auch Alltagsaufgaben gezielt damit anzugehen.

PowerShell-Skripte sind Textdateien, die analog zu Stapelverarbeitungsdateien PowerShell-Befehle enthalten. Der Unterschied ist die Dateinamenserweiterung, die für ein PowerShell-Skript *.ps1* lautet. Die Ausführung unsignierter PowerShell-Skripte muss auf jedem PC zunächst freigegeben werden, da sie aus Sicherheitsgründen standardmäßig deaktiviert ist.

Dies geschieht in der PowerShell-Konsole mit dem Befehl `set-executionpolicy` und einem der nachfolgenden Werte:

```
set-executionpolicy {Restricted | AllSigned |
RemoteSigned | Unrestricted}
```

Hierbei wird mit `RemoteSigned` dafür gesorgt, dass nur heruntergeladene Skripte von einem vertrauenswürdigen Anbieter signiert sein müssen. Damit ist beispielsweise der Schreiber von Skripten in der Lage, diese ohne Signatur lokal auszuführen. Die (unsichere) Alternative zur Ausführung von Skripten, die im Netzwerk gespeichert sind, wäre die Einstellung `Unrestricted`, die allenfalls zum Testen eine vorübergehende Lösung sein kann. Später sollten die selbst erstellten Skripte signiert werden, um zu verhindern, dass bösartige Skripte durch Malware aufgerufen werden. In Domänennetzwerken kann diese Einstellung per Gruppenrichtlinie verteilt werden.

Die Stärken der PowerShell kommt vor allem dann zum Tragen, wenn die Möglichkeiten von Stapelverarbeitungsdateien an ihre Grenzen stoßen. In diesen sind komplexere Auswertungen der Ergebnisse von klassischen Befehlen oftmals nicht möglich.

Sollte auf Anhieb kein passender PowerShell-Befehl zu finden sein oder die Lösung etwas zu komplex werden, kann jederzeit auch auf die klassische Alternative ausgewichen werden. Sowohl `invoke-expression` als auch der Call-Operator `&` ermöglichen solch einen direkten Aufruf:

```
invoke-expression "C:\windows\system32\tasklist.exe"
& "C:\windows\system32\tasklist.exe" /SVC
```

Der vollständige Befehlspfad zur ausführbaren Datei sollte mit angegeben

werden, um zu vermeiden, dass statt des gewünschten Befehls beispielsweise PowerShell-Aliase greifen.

Hinweis: Über Gruppenrichtlinien können Sie PowerShell-Skripte beispielsweise als Computerstart- oder Anmeldeskript konfigurieren. Alternativ lassen sie sich auch mit einer Batchdatei starten, die ihrerseits als normales Start- oder Anmeldeskript konfiguriert ist.

```
powershell -noprofile "x:\skript\start.ps1"
```

Dieser Befehl in einer Batchdatei ruft das Skript *start.ps1* im Pfad *x: \skript* auf und vermeidet dabei das Laden des lokalen Profils aus Performancegründen. Sollte Ihr Skript auf Inhalte des Profils angewiesen sein, lassen Sie den Parameter weg.

Hinweis: Vermeiden Sie, beide Startarten auf dieselbe Organisationseinheit anzuwenden, da es in diesem Fall zu erheblichen Verzögerungen bei der Abarbeitung kommen kann.

Zahlreiche interessante Module werden zum direkten Herunterladen direkt aus PowerShell heraus bereitgestellt. Dazu benötigen Sie zunächst das Modul `PowerShellGet`, das in Windows 11 und Windows Server 2025 bereits in einer älteren Version enthalten ist. Sie installieren es in einer administrativen PowerShell-Konsole mit dem folgenden Befehl:

```
Install-Module –Name PowerShellGet –Force
```

Um Module aus Onlinekatalogen mit dem darin enthaltenen Cmdlet `Install-Module` zu installieren, muss Ihnen mindestens der Name bekannt sein. Für die *PowerShell Community Extensions* ist dies **`pscx`**. Folgender Befehl sucht dieses Modul im Onlinekatalog PSGallery, lädt es herunter und installiert es, sofern dem nicht irgendwelche Konflikte entgegenstehen:

```
Find-Module –Name pscx | Install-Module
```

LDAP-Suchfilter

Eine Reihe von Tools, vornehmlich solche zur Abfrage des Active Directory, ermöglicht die Verwendung eines Filters, mit dem sich, ähnlich wie mit einer WHERE-Anweisung bei SQL, die gewünschten Objekte exakt finden lassen.

Ein LDAP-Suchfilter besteht aus einzelnen Bedingungen, die jeweils in runde Klammern eingeschlossen und mit den logischen Operatoren `&` (und), `|` (oder) bzw. `!` (nicht) verknüpft werden:

```
(&(objectClass=user)(sAMAccountName=fmeier))
```

Damit wird der Benutzer mit dem Anmeldenamen *fmeier* gefunden.

Eine Bedingung besteht aus einem Attribut, das mithilfe eines Operators mit einem Wert verglichen wird. Dabei stehen folgende Operatoren zur Verfügung: = (ist gleich), ~= (ist ungefähr gleich), <= (ist kleiner gleich) und >= (ist größer gleich).

Die Namen der Attribute sind im Schema definiert und können mit einem LDAP-Client, z.B. *ldp*, im Active Directory ermittelt werden.

Bei der Angabe der Werte kann ein Sternchen (*) als Wildcard verwendet werden.

Beispiele

Findet alle Benutzer, deren Nachname *meier* lautet und deren Vorname mit *f* beginnt:

```
(&(objectClass=user)(sn=meier)(givenName=f*))
```

Findet alle Benutzer, die einen Profilpfad eingetragen haben:

```
(&(objectClass=user)(profilePath=*))
```

Findet alle Benutzer, die keinen Profilpfad eingetragen haben:

```
(&(objectClass=user)(!(profilePath=*)))
```

Das gibt jedoch auch Computerobjekte zurück, da diese im Schema von der Klasse *Benutzer* abgeleitet sind. Eine weitere Bedingung filtert sie aus:

```
(&(objectClass=user)(!(objectClass=computer))(!
(profilePath=*)))
```

PoSh: Das `get-aduser`-Cmdlet kann LDAP-Suchfilter direkt verwenden. Folgender Zweizeiler zeigt jene Benutzerkonten, denen keine EMail-Adresse zugeordnet ist.

```
$filter = '(&(objectCategory=person) (objectClass=user)
(!mail=*))' get-aduser -ldapfilter $filter -prop *
```

Windows-GUI – Tipps und Tricks

Verwendung der Maus

Umschalt + linke Maustaste (auf Objekte)
: Wählt aufeinanderfolgende Objekte aus.

Strg + linke Maustaste (auf Objekte)
: Wählt mehrere nicht notwendigerweise aufeinanderfolgende Objekte aus.

Rechte Maustaste (auf ein Objekt)
: Zeigt das Kontextmenü des Objekts an, inklusive des Punkts *Eigenschaften*.

Umschalt + rechte Maustaste (auf ein Objekt)
: Zeigt je nach Objekttyp im Kontextmenü zusätzlich die Einträge *Als Pfad kopieren* und *PowerShell-Fenster hier öffnen*.

Alt + Doppelklick (auf ein Objekt)
: Öffnet die Registerkarte *Eigenschaften* des Objekts. *Alt+Enter* hat denselben Effekt.

Strg + Doppelklick (auf einen Ordner)
: Kehrt die Einstellung der Option *Jeden Ordner in einem eigenen Fenster anzeigen* um.

Tastenkombinationen

Alt+Pfeil links/rechts (im Dateiexplorer oder Edge)
: Navigiert zur letzten/nächsten besuchten Seite.

Alt+Leertaste

Öffnet das Systemmenü des aktiven Fensters, das unter anderem die Menüpunkte zum Minimieren, Maximieren und Verschieben enthält. Gerade Letzteres kann nützlich sein, wenn die Bildschirmauflösung beispielsweise eines Notebooks zu klein ist, damit das gerade offene Fenster die entscheidende Schaltfläche zeigen kann, oder wenn ein Fenster außerhalb der Anzeigegrenzen des Bildschirms geöffnet wird.

Alt+Tab/Alt+Umschalt+Tab

Wechselt zwischen geöffneten Apps und Fenstern vorwärts/rückwärts.

Rücktaste (im Explorer)

Eine Verzeichnisebene nach oben.

Strg+Esc

Öffnet das Startmenü.

Strg+Tab, Strg+Umschalt+Tab

»Blättert« innerhalb von Dialogfeldern in den Registerkarten nach vorne bzw. zurück.

Strg+Umschalt+Esc

Öffnet den Task-Manager.

Strg+Umschalt+Enter

Startet das ausgewählte Programm als Administrator.

F1

Öffnet die Hilfe des jeweiligen Programms.

F2 (auf ein ausgewähltes Objekt)

Umbenennen des Objekts.

F3 (im Dateiexplorer oder Edge)

Öffnet den Suchassistenten bzw. springt ins Suchfeld.

F4 (im Dateiexplorer und Edge)

Klappt die Adressleiste auf.

F5

Aktualisieren.

F6 (im Dateiexplorer und Edge)

Setzt den Fokus nacheinander auf den nächsten Frame (bzw. das nächste Unterfenster) und schließlich auf die Adressleiste.

F10

Selektiert das Dreipunktemenü in Edge (das Drücken von *Alt* hat denselben Effekt).

F11 (im Dateiexplorer und Edge)

Wechselt zwischen maximierter und normaler Darstellung.

Umschalt+Entf (ausgewählte Objekte)

Löscht unter Umgehung des Papierkorbs. Im Kontextmenü des Papierkorbs unter *Eigenschaften* kann dieses Verhalten dauerhaft eingestellt werden.

Win

Wechselt zwischen aktueller Windows-App (kann auch der Desktop sein) und dem Startmenü.

Win+D

Zeigt den Desktop (minimiert alle Programme) bzw. stellt den vorherigen Zustand wieder her.

Win+E

Öffnet den Explorer.

Win+I

Öffnet die App-Einstellungen, in denen mittlerweile viele der einst in der Systemsteuerung beheimateten Konfigurationsmöglichkeiten kontrolliert werden.

Win+, (Komma)

Zeigt die Desktopvorschau.

Win+L

Sperrt den Computer.

Win+M
: Minimiert alle offenen Fenster.

Win+Umschalt+M
: Kehrt eine *Win+M*-Operation um.

Win+P
: Anzeigemodus für Präsentationen (Duplizieren, Erweitern) auswählen.

Win+Pfeil links/Pfeil rechts
: Richtet aktives Fenster am linken/rechten Bildschirmrand aus.

Win+Pfeil hoch/Pfeil runter
: Maximiert das aktive Fenster bzw. schaltet es in den Fenstermodus.

Win+Pause
: Öffnet die Systeminfo in den Einstellungen.

Win+R
: Öffnet das Dialogfeld *Ausführen*.

Win+S
: Öffnet die Windows-Suche.

Win+Strg+D
: Erstellt einen neuen virtuellen Desktop.

Win+Strg+F4
: Schließt den aktuellen virtuellen Desktop.

Win+Strg+Pfeil links/rechts
: Wechselt den virtuellen Desktop.

Win+T
: Schaltet zwischen den Apps auf der Taskleiste um.

Win+Tab
: Öffnet die Task-Ansicht.

Win+U
: Öffnet Einstellungen für die *Barrierefreiheit*.

Win+V

Wechselt zwischen Benachrichtigungen.

Win+X

Öffnet das Quicklink-Menü zum schnellen Zugriff auf Systemeinstellungen.

x

Die Eingabe eines beliebigen Buchstabens auf dem Desktop wählt das nächste Symbol, dessen Name mit dem Buchstaben beginnt. Eine Wiederholung bewirkt die Auswahl des nächsten Symbols mit diesem Anfangsbuchstaben.

»God-Mode«

Den sogenannten »God-Mode«, der Ihnen schnellen Zugriff auf zahlreiche Systemfunktionen unter Windows gibt, erhalten Sie, indem Sie auf dem Desktop einen neuen Ordner anlegen und diesen in *Alle Aufgaben.{ED7BA470-8E54-465E-825C-99712043E01C}* umbenennen.

Symbole für Verknüpfungen

Symbole zum optischen Hervorheben der Verknüpfungen zu ausführbaren Dateien und Skripten finden sich in der standardmäßig angebotenen *shell32.dll* sowie in größerer Zahl unter anderem in den Systemdateien *imageres.dll*, *ieframe.dll* und *wmploc.dll*.

Windows im WWW

Blogs

Durch die große Zahl an Microsoft-Mitarbeitenden und weiteren Bloggern, die oft technisch sehr detaillierte Blogs betreiben, ist dieses Medium ausgezeichnet geeignet, sich in Themen einzuarbeiten und dauerhaft auf dem aktuellen Stand zu bleiben. Im Folgenden finden Sie einige persönliche Favoriten, wenngleich viele davon nur in englischer Sprache verfügbar sind.

- Ask the Directory Services Team (*https://techcommunity.microsoft.com/category/windows-server/blog/askds*)
- Sysinternals Site Discussion (*https://learn.microsoft.com/de-de/sysinternals)*
- Borns IT- und Windows-Blog (*https://www.borncity.com/blog/*)
- Storage at Microsoft (*https://techcommunity.microsoft.com/category/windows-server/blog/filecab*)
- PowerShell Community (*http://community.idera.com/powershell/*)
- Deskmodder-Blog und Forum (*https://www.deskmodder.de/blog/*)
- Deutschsprachige Community für Anwender und Entwickler (*https://administrator.de/*)

Informationen und Fehlersuche

- *https://support.microsoft.com/*
  (Microsoft-Produktunterstützung)
- *https://learn.microsoft.com/de-de/windows-server/administration/windows-commands/command-line-syntax-key*
  (Microsoft-Befehlszeilenreferenz)
- *https://learn.microsoft.com/de-de/*
  (Microsoft Learn – technische Dokumentationen und Schulungen
- *https://developer.microsoft.com/de-de/*
  (Microsofts Portal für Entwickler)
- *https://www.gruppenrichtlinien.de*
  (informative Website zu Gruppenrichtlinien)
- *https://answers.microsoft.com/de-de/windows*
  (deutschsprachige Microsoft-Foren für Heimanwender)
- *https://www.windowstipps.net*
  (Website des Autors, in Überarbeitung)

Softwarearchive

- *https://www.microsoft.com/de-de/download/*
  (Microsoft Download Center u.a. mit Hilfsprogrammen und Updates)
- *https://docs.microsoft.com/en-us/sysinternals/downloads/sysinternals-suite*
  (viele äußerst leistungsfähige Tools primär zur Überwachung und Analyse)
- *https://www.powershellgallery.com/*
  (zahlreiche PowerShell-Module und -Skripte)
- *https://download.cnet.com/windows/*
  (Free- und Sharewarearchiv)
- *https://sourceforge.net/*
  (freie Software)

- *https://github.com/*
  (Plattform für zahlreiche Open-Source-Projekte)
- *https://nirsoft.net/*
  (zahlreiche gelegentlich sehr nützliche, wenngleich nicht immer gern gesehene Hilfsprogramme)
- *https://www.starwindsoftware.com/starwind-v2v-converter/*
  (eine für die Konvertierung zu virtuellen PCs verschiedener Zielformate von physischen und virtuellen PCs verschiedener Quellformate nutzbare kostenlose Software)

Index

Symbole

Numerisch

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Rezensieren
Sie dieses Buch

Senden
Sie uns Ihre Rezension
unter **www.dpunkt.de/rez**

Erhalten
Sie Ihr Wunschbuch aus
unserem Verlagsangebot