

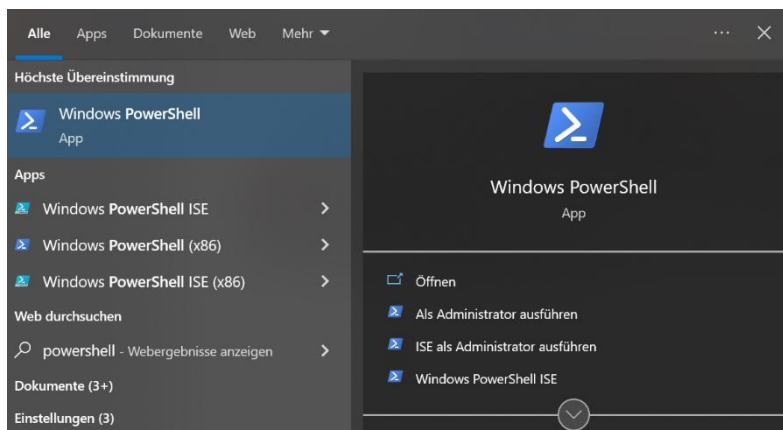


Informationen über das Betriebssystem sammeln

Informationen

Manche Einstellungen werden nicht über die graphische Benutzeroberfläche (GUI), sondern über eine textbasierte Schnittstelle (CLI – Command Line Interface) vorgenommen.

Der moderne Nachfolger der klassischen Windows Eingabeaufforderung *cmd.exe* ist die Windows PowerShell. Sie starten eine einfache PowerShell durch Eingabe von *powershell* im Suchfeld und Klick auf *Öffnen*¹.



Genauso lässt sich die PowerShell auch mit erhöhten Rechten ausführen, wenn sie durch Klick auf *Als Administrator ausführen* gestartet wird.

Bis zum Windows 10 Build 14791 war die Eingabeaufforderung *cmd.exe* das Standard-CLI für Windows. Seitdem wurde sie von der PowerShell abgelöst. Sämtliche Befehle, die in der Eingabeaufforderung verfügbar waren, sind mittlerweile auch in der PowerShell identisch aufrufbar.

Die meisten Übungen sehen die durchgängige Verwendung der PowerShell vor, weil sie den neuen Standard im Ökosystem von Microsoft dargestellt².

In der Praxis wird die Eingabeaufforderung oft jedoch noch verwendet; entweder aus Gewohnheit oder aufgrund vorhandener Skripte, die in der Batch-Language geschrieben sind oder weil das vorliegende Betriebssystem die PowerShell nicht (Windows XP) oder nur teilweise unterstützt wie es etwa bei Windows 7, Windows 8 und Windows Server 2008 oder Windows Server 2012 der Fall ist.

¹ Schnelle Alternative: Rechtsklick auf das Windowszeichen unten links und dann im Kontextmenü *Windows PowerShell* (bei Windows 11: *Windows-Terminal*) bzw. *Windows PowerShell (Administrator)* (bei Windows 11: *Windows-Terminal (Administrator)*) auswählen.

² PowerShell-Varianten sind auch für einige Linux-Distributionen verfügbar sowie als Azure Cloud Shell in den Cloud-Anwendungen von Microsoft die Standard-CLI.



Informationen über das Betriebssystem sammeln

Aufgabe 1

Starten Sie eine einfache Eingabeaufforderung oder PowerShell und eine zweite mit erhöhten Rechten. Notieren Sie jeweils, in welchem Verzeichnis Sie sich direkt nach dem Start befinden.



Aufgabe 2

Starten Sie eine Eingabeaufforderung mit erhöhten Rechten.

- Ermitteln Sie Ihre IPv4-Adresse mit dem Befehl *ipconfig*
- Verwenden Sie den Befehl *ping 127.0.0.1* um die Funktionalität Ihres TCP-Stacks zu überprüfen.
- Suchen Sie eine beliebige weitere IPv4-Adresse und verwenden Sie den Befehl *ping*, um die Verbindung zwischen den Rechnern zu überprüfen.

Aufgabe 3

Starten Sie eine PowerShell mit erhöhten Rechten.

- Ermitteln Sie Ihre IPv4-Adresse mit dem Cmdlet³ *Get-NetIPConfiguration*
-  Verwenden Sie den Befehl *Test-Connection 127.0.0.1* um die Funktionalität Ihres TCP-Stacks zu überprüfen.
-  Suchen Sie eine beliebige weitere IPv4-Adresse und verwenden Sie das Cmdlet *Test-Connection*, um die Verbindung zwischen den Rechnern zu überprüfen.
- (Bonus) Lassen Sie sich nur die IPv4-Adressinformationen zum Adapter *Ethernet* anzeigen, indem Sie den Parameter *-InterfaceAlias Ethernet* verwenden.

Aufgabe 4

Vergleichen Sie Ihre Arbeit mit der Eingabeaufforderung und der PowerShell und notieren Sie stichpunktartig Ihre Ergebnisse, z. B. welches Werkzeug Ihnen aus welchen Gründen besser gefällt.

³ Ein Cmdlet (sprich: Commandlet) ist ein spezieller Befehl innerhalb der Windows PowerShell.



Informationen über das Betriebssystem sammeln

Aufgabe 5 (Bonus)

Windows legt viele Informationen zur Konfiguration des Betriebssystems in vordefinierten Systemvariablen ab.

Der Wert einer Variablen kann auch mit der moderneren Windows PowerShell über das Cmdlet *Get-ChildItem* angezeigt werden wie im Beispiel mit der Variable *systemroot*. Beachten Sie, dass vor dem Namen der Variablen jeweils *Env:* eingegeben werden muss, um den Wert der Variablen auszugeben⁴.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

Lernen Sie das neue plattformübergreifende PowerShell kennen - https://aka.ms/pscore6

PS C:\Users\anr> Get-ChildItem Env:\systemroot

Name                           Value
----                           -
SystemRoot                     C:\windows

PS C:\Users\anr> _
```

Füllen Sie die folgende Tabelle zu den Werten einiger Systemvariablen mithilfe der Windows PowerShell aus.

Variable	Bedeutung	Wert
<i>systemroot</i>	Installationsverzeichnis von Windows	C:\windows
<i>computername</i>	Name des Rechners	
<i>Processor_Architecture</i>	x86 für 32-Bit-System, amd64 für 64-Bit-System	
<i>username</i>	Name des aktuellen Benutzers	
<i>userprofile</i>	Pfad zum Ordner des aktuellen Benutzers	

⁴ Bei *Env:* handelt es sich um ein virtuelles PowerShell-Laufwerk (PSDrive), in dem die Umgebungsvariablen gespeichert sind. Der Zugriff erfolgt wie bei einem gewöhnlichen Laufwerk, kann aber nur aus einer PowerShell Session heraus erfolgen. Alternativ ist die Ausgabe der Systemvariable *systemroot* auch per *Write-Output \$env:username* möglich; das Cmdlet *Write-Output* ist dabei eine möglichst direkte Entsprechung von *echo* in der Eingabeaufforderung, während *Get-ChildItem* Elemente in einem Verzeichnis auflistet wie der Befehl *dir* in der Eingabeaufforderung.