

PowerShell Cheatsheet für Windows

PRAXISLEITFADEN | Stand: 27. Oktober 2026 | Kostenlos

1. Umgebung prüfen

Vorgehen: Ermittle PowerShell-Version, Pfade und Befehlsauflösung vor Änderungen.

Beispiel: `$PSVersionTable.PSVersion`

`Get-Command php -All`

`Get-Location`

Prüfkriterium: Ausgeführte Programme und Versionen sind eindeutig.

Vermeiden: Alias und ausführbare Datei verwechseln.

2. Dateien finden

Vorgehen: Nutze `Get-ChildItem` mit Filter und gezieltem Pfad. Behandle Zugriffsfehler bewusst.

Beispiel: `Get-ChildItem D:\ -Filter php.exe -Recurse -ErrorAction SilentlyContinue`

Prüfkriterium: Trefferliste enthält nachvollziehbare Fundorte.

Vermeiden: Ungefilterte rekursive Suche auf allen Laufwerken ohne Bedarf.

3. Prozesse und Dienste

Vorgehen: Lese Prozess- und Dienststatus zunächst nur aus; Änderungen nur mit ausreichenden Rechten.

Beispiel: `Get-Process php -ErrorAction SilentlyContinue`

`Get-Service | Where-Object Status -eq Running`

Prüfkriterium: Betroffene Prozesse sind vor Stop/Restart identifiziert.

Vermeiden: Prozesse nach ungenauem Namensmuster beenden.

4. Fehlerbehandlung

Vorgehen: Nutze `try/catch` und `-ErrorAction Stop` für kritische Operationen.

Beispiel: `try { Get-Item C:\missing -ErrorAction Stop } catch { Write-Warning $_.Exception.Message }`

Prüfkriterium: Fehler werden sichtbar behandelt.

Vermeiden: Fehler mit `SilentlyContinue` dauerhaft verbergen.

5. Sichere Automatisierung

Vorgehen: Teste destruktive Befehle mit `-WhatIf`, protokolliere Ergebnisse und setze keine Passwörter im Skript fest.

Beispiel: `Remove-Item .\temp -Recurse -WhatIf`

Prüfkriterium: Vor produktiver Ausführung ist der Effekt nachvollziehbar.

Vermeiden: Ungeprüfte Löschskripte automatisiert starten.

Fachquellen

Microsoft: PowerShell Docs – <https://learn.microsoft.com/en-us/powershell/>

Onlineversion: <https://www.cyberknecht.de/wissen/cheatsheets/powershell-cheatsheet.html>