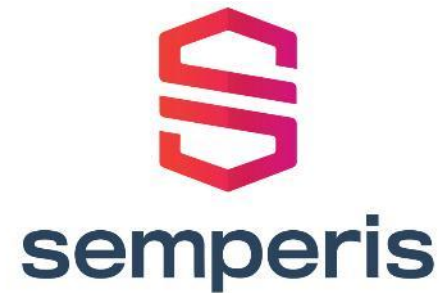


# Ordnung in PowerShell-Skripte bringen, ohne gleich Entwickler zu werden

Bechtle MVP Community Week | 24.04.2026

Vielen Dank!

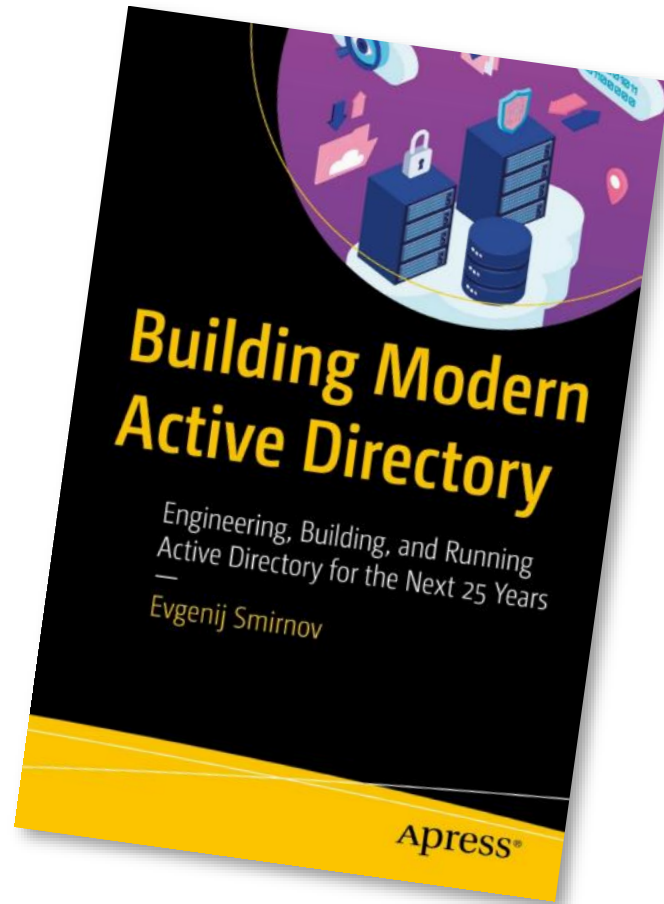


## Get-SpeakerInfo -Brief

- **Name:** Evgenij Smirnov
- **YearOfBirth:** 1972
- **JobTitle:** Principal Solutions Architect
- **Employer:** Semperis
- **MVP:** { Identity & Security, PowerShell }
- **Certifications:** { MCSE, MCSA, VCP, VCAP, VCIX, CCA, QCIC }
- **UserGroups:** { WSUG-B, EXUSG, PSUGB }
- **BlueSkyHandle:** @it-pro-berlin.de



## Kleine Werbung am Rande...



**Heinemann Verlag**  
Im Dialog mit Spezialisten.

Suchbegriff ...

0,00 €\*

Abonnements Sonderhefte Einzelhefte Digital Mehrfachlizenzen Bücher **Seminare** Extras

Administrator Website

Seminare

### Online-Intensivseminar Active-Directory-Security



1.785,00 €\*

Preis inkl. MwSt. [Lernaktivitäten](#)

Hybridpreise bei Moonster-Registrierung  
Stellen Sie bereits registriert sein, zum Login

- Veranstaltungsort
- Inhalt verständlich

Veranstaltungsort/-datum

ONLINE: 30. Juni - 02. Juli 2025

1

In den Warenkorb

[Zum Merkzettel hinzufügen](#)

Produktnummer: ITA-23-SEM-2823.2

<https://ad2049.com>

Nächste Gruppe: 01.07 – 03.07.2026

Kleine Werbung am Rande...



<https://ntlminator.com>

- > AUFTRAG: AD VERTEIDIGEN
- > WO: HANNOVER, DEUTSCHLAND
- > VON: 2026-05-04
- > BIS: 2026-05-08
- > WIRD DEINE FIRMA UEBERLEBEN?
- > DIE UHR LAEUFT...
- > \_
- > BUCHUNG: [ADGATOR.ORG/BOOTCAMP](https://adgator.org/bootcamp)

Nächste Gruppe: 04.05 – 08.05.2026

## In dieser Session

- **Warum skripten Menschen? Etwas Verhaltenspsychologie**
- **Warum ist es ein Problem? Etwas IT-Betriebserfahrung**
- **An welchen Schrauben kann man drehen?**
- **Wie bekommt man alle möglichst zufrieden?**

## NICHT in dieser Session

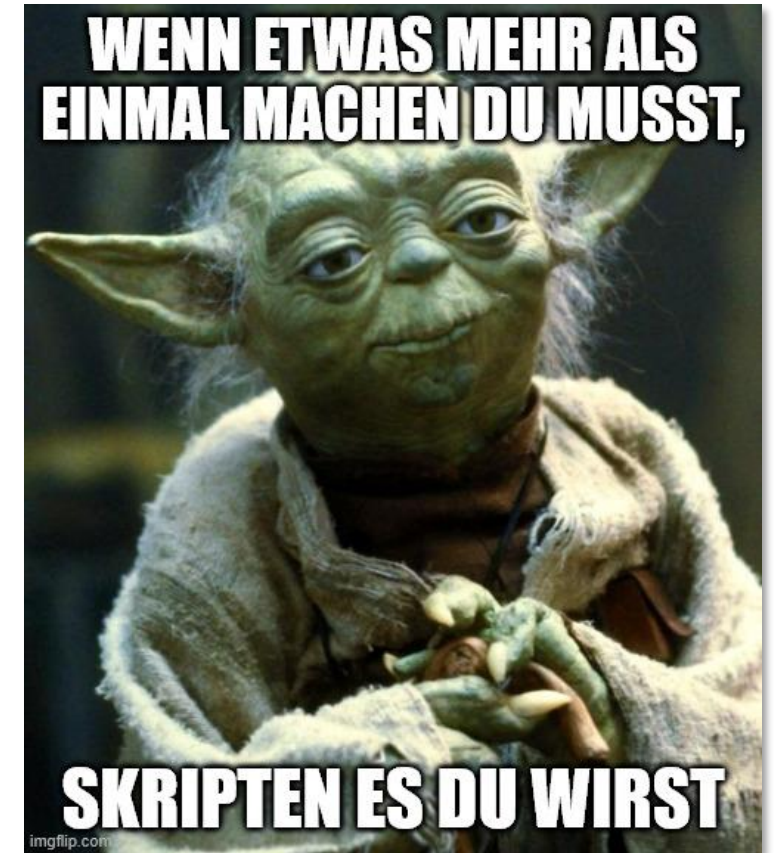
- **Nutzung probabilistischer Text-Extrusion™ zur Erstellung von Skript-Code**
  - Wenn man das 10x schnell hintereinander ausspricht, versteht man sofort, warum „KI“ so ein Marketing-Erfolg ist 😊
- **Best Practices jeder Art, was die Gestaltung des Code angeht**
  - Skript-Sprachen sind geduldig, besonders PowerShell
  - Das, was man \*als Organisation\* gut beherrscht, ist richtig
- **Cross-Plattform-Sachen**
  - Wer letztes Jahr dabei war, ist bestens im Bilde 😊
- **Auch nicht: Wie \*ich\* das aufziehen würde, wenn \*ich\* das Sagen hätte**
  - Denn: mich gibt es nur einmal, und ich arbeite nicht bei euch
  - Wenn ihr das trotzdem hören wollt → PSConfEU im Juni in Wiesbaden

# Warum skripten Menschen?

Etwas Verhaltenspsychologie

## Warum Menschen skripten (ein paar Thesen)

- **Instant Gratification – schneller, billiger und nachhaltiger\* als Schuhe**
- **„Wer nicht weiß, programmiert“**
  - Das ist nicht zu verallgemeinern, denn oft gibt es *\*wirklich\** keinen anderen Weg...
  - ...somit ist jedes Skript *\*immer\** entweder ein fehlendes Feature in einem System oder eine Lücke in einem Prozess – oder beides!
- **„Wenn Du etwas mehr als einmal machen musst, skripte es“**
  - Skripte sind kodifiziertes Wissen – *nomen est omen*
  - Kommentierte Skripte sind besser als halbherzige Doku, denn eine Dokumentation zeigt, wie es sein sollte, Skripte aber zeigen, wie es ist\*



# Warum ist es ein Problem?

Etwas IT-Betriebserfahrung

## Ungetesteten Code mit hohen Rechten ausführen? Aber natürlich!

- **Eigener schlampig geschriebener nicht ausreichend getesteter Code kann bereits großen Aufwand verursachen...**
- **Zufällige Skripte auf GitHub / StackOverflow / meinem Blog / MCSEBoard könnten direkt destruktiv und auch so gemeint sein!**

## Wo war nochmal die letzte Version von dem Skript XY?

- **Benutzerprofil vom Kollegen?**
  - Der ist nicht mehr da...
- **Mein eigenes Benutzerprofil, aber auf einer anderen Maschine?**
  - Welche war es nochmal?
- **Vermeintlich gut abgelegt?**

```
\\files.firma.com\IT\Projekte\2022\01-Dokumentenmanagement\Organisation\00-DATENPFLEGE\  
2024-05 Migration der Verlinkung\Nachträglich\Schmidt-Skript\  
20240617_Update_Ordner.Okttober2025.Final.NutzeDies.ps1
```

## Welche Änderungen habe ich gestern gemacht? Da hat's noch getan...

- **Einmal [Ctrl]+S auf dem falschen Screen, und 6 Stunden Arbeit sind futsch...**
  - ... ohne dass man es sofort bemerkt hat
- **A/B-Vergleiche sind nur möglich, wenn es A und B gleichzeitig gibt!**
  - Lineares Bearbeiten ohne Versionskontrolle maskiert alle Änderungen ☹️
  - Da intern in Skripten oft Vorgaben und sogar Credentials hinterlegt werden, ist ein Prüfsummen-Vergleich grundsätzlich nicht möglich!



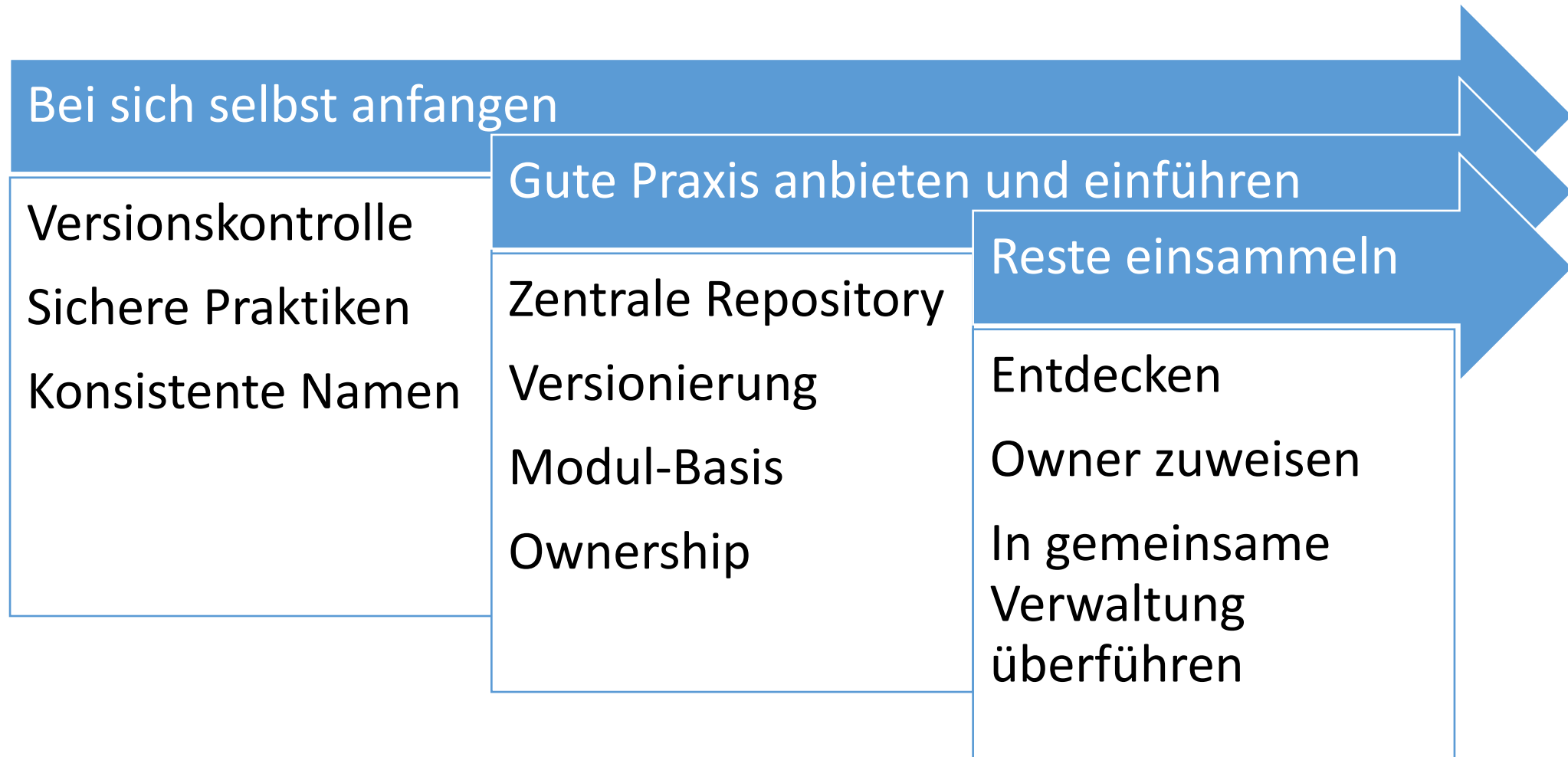
## Der Kollege™ erfindet immer wieder das Rad neu!

- **Gemeinsame Richtlinien und das Wiederverwenden von Code durchzusetzen, ist in „Enterprise Scripting“-Umgebungen sehr schwierig**
  - In der Software-Entwicklung eine Selbstverständlichkeit!
- **Angebote funktionieren immer besser als Zwang!**
  - Module in hoher Qualität bereitstellen, kann schon der erste wichtige Schritt sein...
  - Kommentierte Code-Snippets im Wiki veröffentlichen statt in Dateien ablegen

# Wie kriegt man das in den Griff?

Oder: Wie machen das die Profis?

# Das Chaos ordnen, den Müll beseitigen



## Keine Angst vor Git!

### Git bedeutet **NICHT**:

- Ich muss neue Werkzeuge\* lernen
  - ...insbesondere GitHub
- Ich muss meinen Skript-Code ins Internet stellen
- Ich muss zwingend probabilistische Text-Extrusion™ benutzen
- Andere können (und müssen) meinen unfertigen Code sehen
- Jemand muss bestätigen, bevor ich mein Skript laufen lassen kann

### Git bedeutet **vielmehr**:

- Ich kann Versionen fixieren
- Ich kann Unterschiede zwischen Versionen, zumindest bei Textdateien, sofort sehen
- Ich kann zwischen Version springen, ohne Änderungen zu verlieren!
- ...und wenn ich Lust habe und das Team mitzieht, stehen mir all die anderen Sachen auch noch offen!

<https://git-scm.com/cheat-sheet>

Git für einen alleine und für einen engen Kreis



makeameme.org

## Wird die Execution Policy uns retten?

- **Execution Policy ist ein gutes Feature für die Selbstdisziplin**
  - So wie ins Fitness-Studio gehen, keine Schokolade essen, früh schlafen gehen...
- **Sie ist aber weder ein Fremd-Disziplinierungsmittel noch ein Security Feature!**
  - Es gibt mindestens 15 Wege, Execution Policy zu umgehen 😊
  - Und wenn Kollegen das täglich machen müssen, dann haben sie auch Adminrechte...
- **Die einzige Einstellung, die dabei wirklich Sinn ergibt, ist ATTSigned**
  - Nicht alle Produkte werden damit klarkommen...
  - ... und das kann erst realisiert werden, wenn alle Vorgabewerte aus Skripten raus sind
- **Diese Maßnahme sollte am Ende der Verbesserungs-Strecke stehen, nicht am Anfang!**

## Die Tool-Dichotomie: PS ISE vs. VS Code

- So sehr uns manche glauben lassen wollen, PS7 wäre ein “Upgrade” von PS5.1...
- ...das Gros von On-Prem-Skripten wird unter Windows PowerShell laufen.
- Das Problem dabei:
  - Die PowerShell ISE ist überall, beherrscht aber kein Git
  - VSCode ist ein exzellenter grafischer Git-Client, ist aber auf PowerShell 7 getrimmt
- Lösung?
  - Für Puristen: einen Aufsatz für ISE suchen, der Git beherrscht und noch supported ist (Hint: ISESteroids gehört nicht dazu, auch wenn’s andere Dinge beherrscht)
  - Für Vorwärtsgewandte: ISE Mode in VSCode (nur Farben) + Terminal Switch  
Die Sprachprüfung ist immer noch auf PowerShell 7 ausgelegt
  - Für Tool-Sammler: 3<sup>rd</sup> Party – oder ist es schon 4<sup>th</sup> Party?

## Empfohlenen Code zentral verteilen

- **Eigenes Repository als Default (kann auch eine DFS-Freigabe sein)**
  - PSGallery austragen ist eine sehr gute Idee, da dort keine Garantie für Authentizität besteht
- **Eine Auswahl an Modulen überall vorinstallieren und aktuell halten**
  - Das ist das Angebot, diese Module in den eigenen Skripten zu nutzen
  - Auch 3<sup>rd</sup> Party-Module lassen sich im privaten Repository veröffentlichen

Auswahl von Modulen mit Bordmitteln verteilen



makeameme.org

## Wissen, was ausgeführt wird (i)

- **PowerShell bietet zwei Bordmittel zur Verfolgung des ausgeführten Code:**
  - Transcript → Textdateien
  - Script Block Logging → Windows Event Log
- **Beide sind per Gruppenrichtlinie konfigurierbar**
  - Auch für PowerShell 7 → ADMX wird mitinstalliert
  - Beste Einstellung für PS7 = „Use Windows PowerShell policy setting“
  - Policy für User und Computer verfügbar → Computer gewinnt!
- **Beide waren aus Sicherheitssicht umstritten, da Secrets ggfls. drin sind**
  - An offensichtlichen Stellen werden sie ohnehin maskiert...
  - ...aber auch gerade aus Sicherheitssicht ist es eigentlich gut 😊
  - ...und das CIS ist dieser Auffassung inzwischen gefolgt!

## Wissen, was ausgeführt wird (ii)

### Script Block Logging:

- Nur Script Blocks werden geloggt, einzelne interaktive Befehle fehlen
- Verschiedene Logs + Event IDs für PS7 + WinPS
- Allein das Starten erzeugt
  - 24 Events (PS7)
  - 7 Events (WinPS)
- Dafür können die Logs
  - ins SIEM abgeleitet werden
  - nicht durch User verändert werden!

### PowerShell Transcript:

- Enthält alle Befehle, auch interaktive
- Dateien gehören dem ausführenden User
  - Und können durch diesen gelöscht werden
- Das Format eignet sich nicht sehr gut zum Extrahieren von nutzbaren Informationen...

**Wissen teilen!**

- **Interne Communities etablieren, wenn man bereits Champions hat**
- **Hat man keine Champions → PowerShell-MVPs sprechen gern auch bei einer internen User Group**
- **Best Practice ist das, was funktioniert**
  - ...außer Klartext-Credentials !!!!!!!!!!!!!!!!!!!!!!!

# Happy Frühjahrsputz!

Bei Fragen → einfach fragen 😊