

MS FSLogix-Profile

1. Was ist FSLogix *Profilmanagement für Windows-Umgebungen?*

FSLogix ist eine Microsoft-Lösung, die Benutzerprofile in virtuelle Festplatten (VHDX) auslagert. Beim Login wird der Container gemountet – das Profil ist sofort vollständig verfügbar, unabhängig vom Server. Dadurch werden Anmeldezeiten drastisch verkürzt und Serverumzüge zum Kinderspiel.

1.1 Hauptfunktionen

- Profile Container: Komplettes Benutzerprofil in einer VHDX-Datei
- Office Container: Separater Container für Outlook-Cache, Teams, OneDrive
- Application Masking: Anwendungen pro Benutzer/Gruppe ein-/ausblenden
- Java Version Control: Verschiedene Java-Versionen pro Anwendung

2. Vorteile und Nachteile

2.1 Vorteile

Vorteil	Beschreibung
Schnelle Anmeldung	Profil wird gemountet statt kopiert – Login in Sekunden statt Minuten
Serverunabhängigkeit	User kann sich an beliebigem Server anmelden, Profil folgt automatisch
Einfache Migration	Serverwechsel ohne Profilmigration – Container bleibt gleich
Vollständiges Profil	Im Gegensatz zu Roaming Profiles wird ALLES gespeichert
Outlook-Performance	OST-Datei bleibt erhalten, kein Neuaufbau bei jedem Login
Einfaches Backup	Eine VHDX-Datei pro User = einfache Sicherung
Dynamische Größe	VHDX wächst nur bei Bedarf, belegt nicht sofort Maximalgröße

2.2 Nachteile

Nachteil	Beschreibung
Single-Session	Standardmäßig nur eine gleichzeitige Anmeldung pro User
Storage-Abhängigkeit	Netzwerkfreigabe muss hochverfügbar und performant sein
Keine echte Synchronisation	Bei gleichzeitiger Anmeldung an zwei Geräten keine Zusammenführung (letzte Abmeldung gewinnt)
VHDX-Verwaltung	Monitoring für Füllstand erforderlich (siehe Kapitel 6)
Lizenzierung	Nicht für alle Szenarien kostenlos (siehe Kapitel 3)

3. Lizenzierung und Kosten

3.1 Kostenlos enthalten bei

Lizenz/Produkt	Anmerkung
RDS-CALs	Remote Desktop Services – der klassische Terminalserver
VDA-Lizenzen	Virtual Desktop Access für VDI-Umgebungen
Microsoft 365 E3/E5	Enterprise-Pläne
Microsoft 365 A3/A5	Education-Pläne
Microsoft 365 Business Premium	Für KMU mit bis zu 300 Usern
Windows 10/11 Enterprise	Per User oder Device
Windows 10/11 Education	Bildungseinrichtungen

3.2 Nicht kostenlos / Nicht lizenziert

Szenario	Status
Windows Pro ohne M365	Nicht lizenziert – auch nicht für Domänen-Clients
Standalone-PCs ohne Domäne/Azure AD	Nicht offiziell unterstützt
Reine File-Server-Nutzung	Nicht lizenziert

3.3 Wo erwerben?

FSLogix selbst ist kostenlos – du benötigst lediglich eine der oben genannten Lizenzen. Der Download erfolgt direkt bei Microsoft:

<https://aka.ms/fslogix-latest>

Hinweis für Server + Clients: Wenn du normale Server mit Clients (nicht RDS) betreibst und keine M365 Business Premium oder Enterprise-Lizenzen hast, ist FSLogix nicht lizenziert. Alternative: Microsoft 365 Business Premium enthält FSLogix-Rechte.

4. Szenarien und gleichzeitige Anmeldung

4.1 Terminalserver (RDS) – Einzelner Server

Bei einem einzelnen Terminalserver gibt es keine Probleme mit Doppel-Anmeldungen. Windows lässt standardmäßig nur eine Session pro User zu. Meldet sich ein User erneut an, landet er in seiner bestehenden Session.

4.2 Terminalserver-Farm (mehrere Server)

Bei mehreren Terminalservern muss verhindert werden, dass ein User sich parallel an verschiedenen Servern anmeldet. Das regelt der RD-Verbindungsbroker (Connection Broker):

- Der Broker merkt sich, wo jeder User angemeldet ist
- Bei erneuter Verbindung wird der User zu seinem bestehenden Server geleitet
- Zusätzlich per GPO absichern:

Computerkonfiguration ? Administrative Vorlagen ? Windows-Komponenten ? Remotedesktopdienste ? Remotedesktop-Sitzungshost ? Verbindungen:

Remotedesktopbenutzer auf eine Sitzung beschränken -> Aktiviert

Wichtig: FSLogix selbst regelt KEINE Sessions – das ist Aufgabe von Windows/RDS. FSLogix kümmert sich nur um das Profil.

4.3 Normale Server mit Clients (kein RDS)

Hier gibt es keinen zentralen Broker. Wenn ein User sich an PC-A anmeldet und dann an PC-B:

- FSLogix Standard-Modus: Der erste Rechner hat den Container exklusiv. Am zweiten Rechner wird ein temporäres lokales Profil erstellt – Änderungen gehen verloren.
- Read-Only Modus (ProfileType=3): Beide können gleichzeitig arbeiten, aber Änderungen werden nicht zurückgeschrieben.

Für echtes gleichzeitiges Arbeiten mit Synchronisation ist FSLogix nicht gedacht. Hierfür eignet sich eher:

- Klassische Ordnerumleitung auf Netzlaufwerk
- OneDrive Known Folder Move
- Kombination aus beidem

5. Installation und Konfiguration

5.1 Voraussetzungen

- Windows Server 2016 oder neuer / Windows 10/11
- Dateifreigabe für Profile (SMB)
- Ausreichend Storage (ca. 5-30 GB pro User, dynamisch)
- Hyper-V-Rolle oder integrierte VHD-Unterstützung

5.2 Schritt 1: Freigabe vorbereiten

Erstelle eine Freigabe auf deinem Fileserver, z.B.:

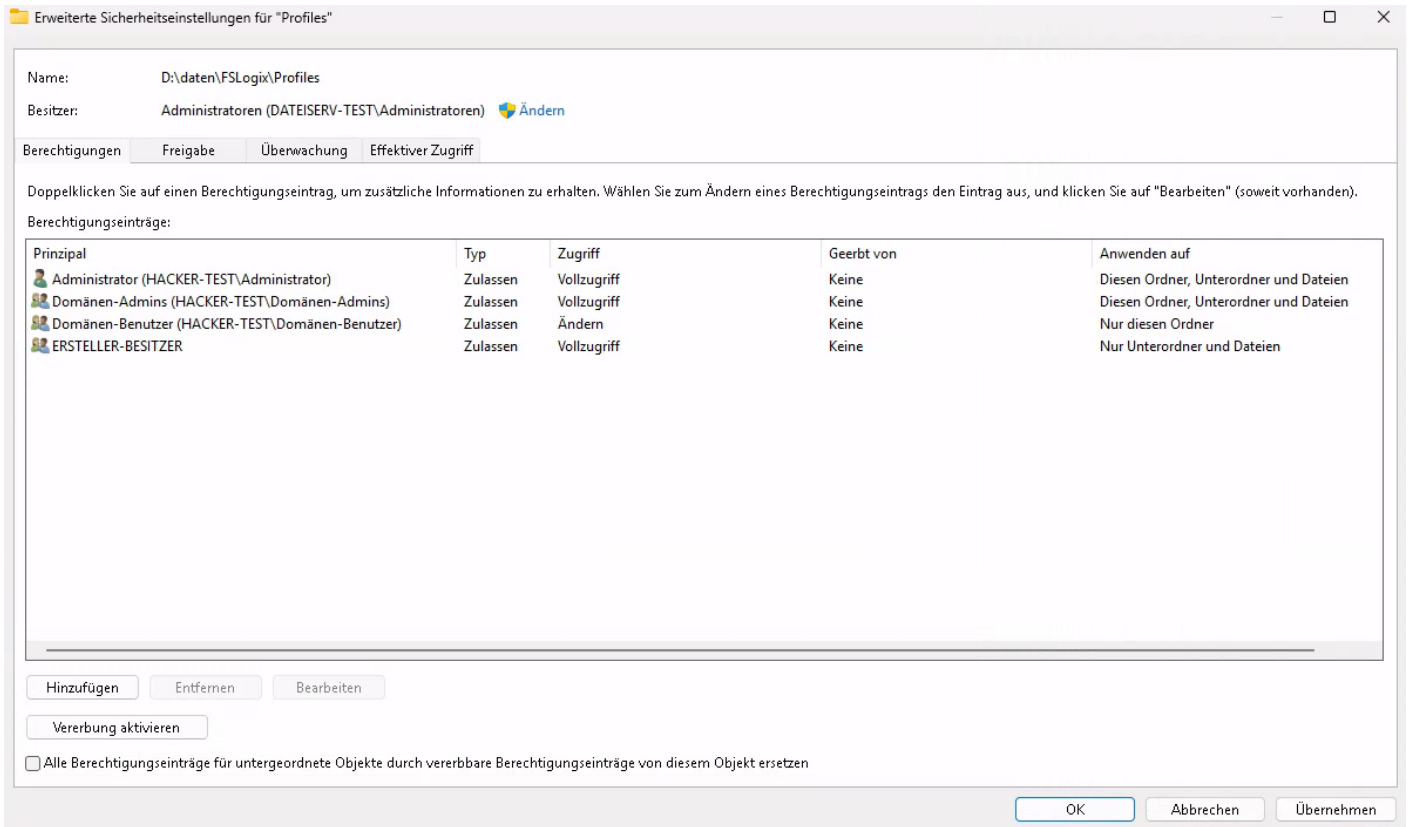
\\fileserver\FSLogix\Profiles

Man kann auch gleich auf Profiles den Namen legen, wenn man möchte ;-)

NTFS-Berechtigungen:

Benutzer/Gruppe	Berechtigung	Anwenden auf
SYSTEM	Vollzugriff	Diesen Ordner, Unterordner und Dateien
Domain Admins	Vollzugriff	Diesen Ordner, Unterordner und Dateien
Domain Users	Ändern	Nur diesen Ordner
Creator Owner	Vollzugriff	Nur Unterordner und Dateien

Beispiel Screenshot:



Freigabeberechtigungen: Jeder ? Vollzugriff (NTFS regelt den Rest)

5.3 Schritt 2: FSLogix Agent installieren

<https://aka.ms/fslogix-latest>

Auf dem Terminalserver/Remotedesktop Client die Datei FSLogixAppsSetup.exe ausführen. Standard-Installation reicht aus.

5.4 Schritt 3: Konfiguration per Registry (Ich würde aber Schritt 5.5 per GPO bevorzugen) wurde gelöscht, GPO ist einfacher, deswegen bei 5.5 weiter machen.

5.5 Konfiguration per GPO (Alternative zur Registry empfohlen)

Statt der Registry-Konfiguration kannst du FSLogix auch zentral per Gruppenrichtlinie steuern. Das ist besonders bei mehreren Servern praktisch.

GPO-Vorlagen installieren

Die ADMX-Vorlagen liegen nach der FSLogix-Installation unter:

C:\Program Files\FsLogix\Apps\fslogix.admx

C:\Program Files\FsLogix\Apps\fslogix.adml

Kopiere die Dateien in den Central Store (falls vorhanden) oder lokal:

Central Store (empfohlen, denn bei mehreren DCs wird automatisch synchronisiert)

```
\\domain.local\SYSVOL\domain.local\Policies\PolicyDefinitions\fslogix.admx
```

```
\\domain.local\SYSVOL\domain.local\Policies\PolicyDefinitions\de-DE\fslogix.adml
```

Falls die Verzeichnisse nicht bestehen, manuell anlegen

Einfach den Ordner manuell erstellen:

```
\\domain.local\SYSVOL\domain.local\Policies\PolicyDefinitions
```

Dann noch den Sprachordner:

```
\\domain.local\SYSVOL\domain.local\Policies\PolicyDefinitions\de-DE
```

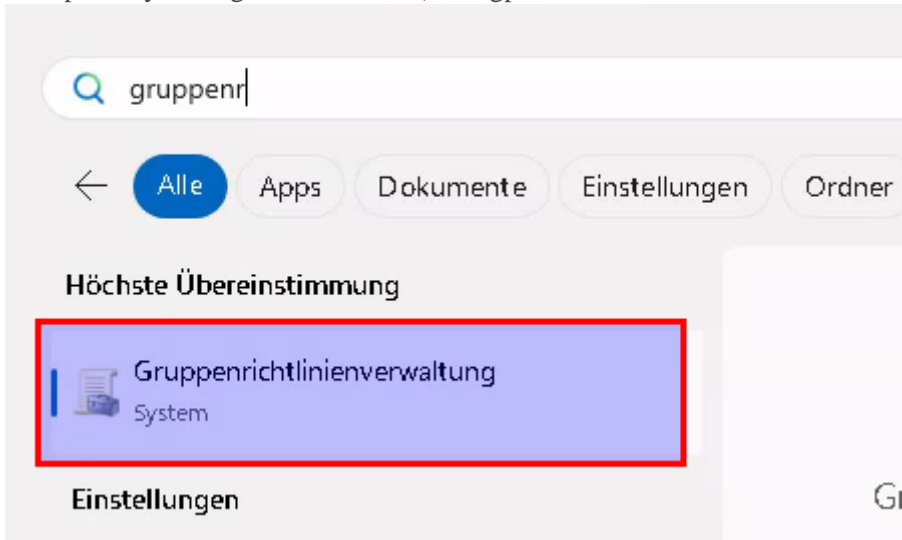
Oder lokal auf dem DC (Ich würde Central Store bevorzugen, siehe oben)

```
C:\Windows\PolicyDefinitions\fslogix.admx
```

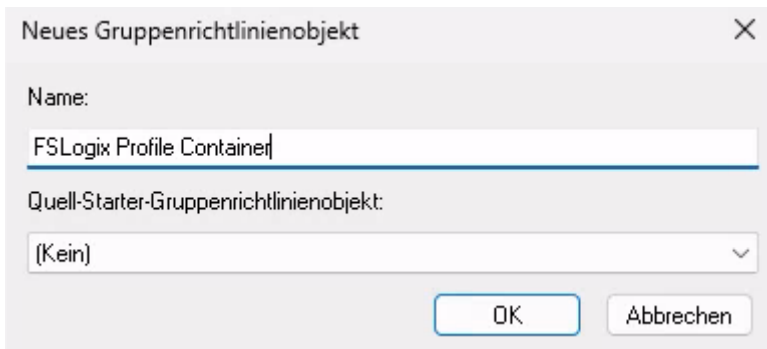
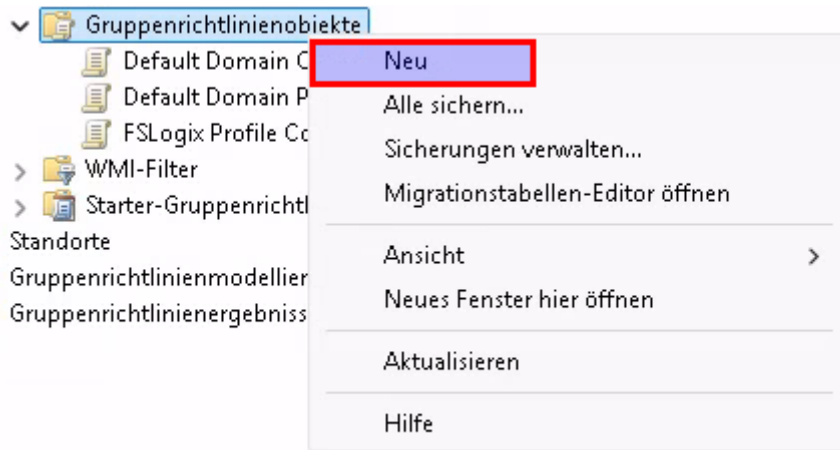
```
C:\Windows\PolicyDefinitions\de-DE\fslogix.adml
```

Nachdem die Dateien kopiert sind (lokal oder Central Store) GPO erstellen und verknüpfen

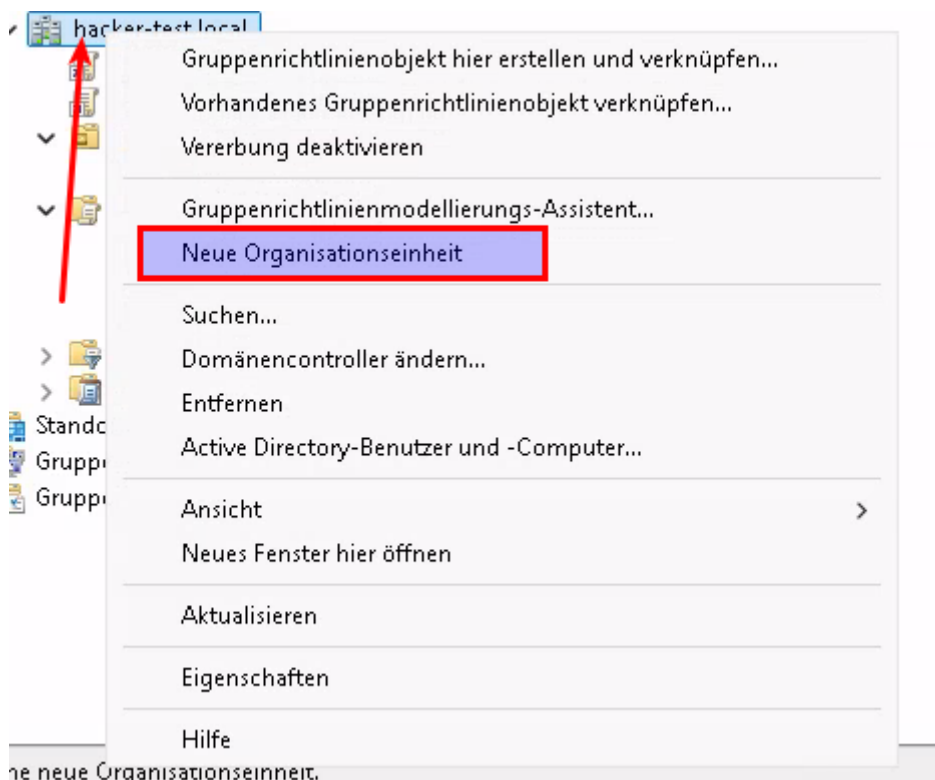
1. Group Policy Management öffnen (nicht gpedit.msc, das ist lokal, nicht domänenbezogen)

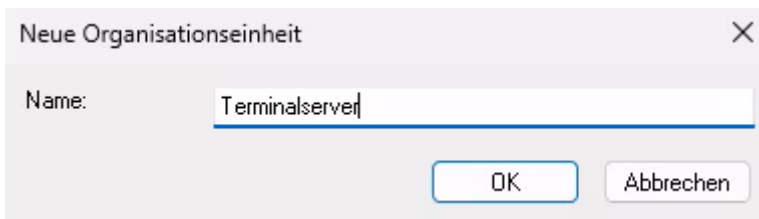


2. Neue GPO erstellen, z.B. "FSLogix Profile Container"

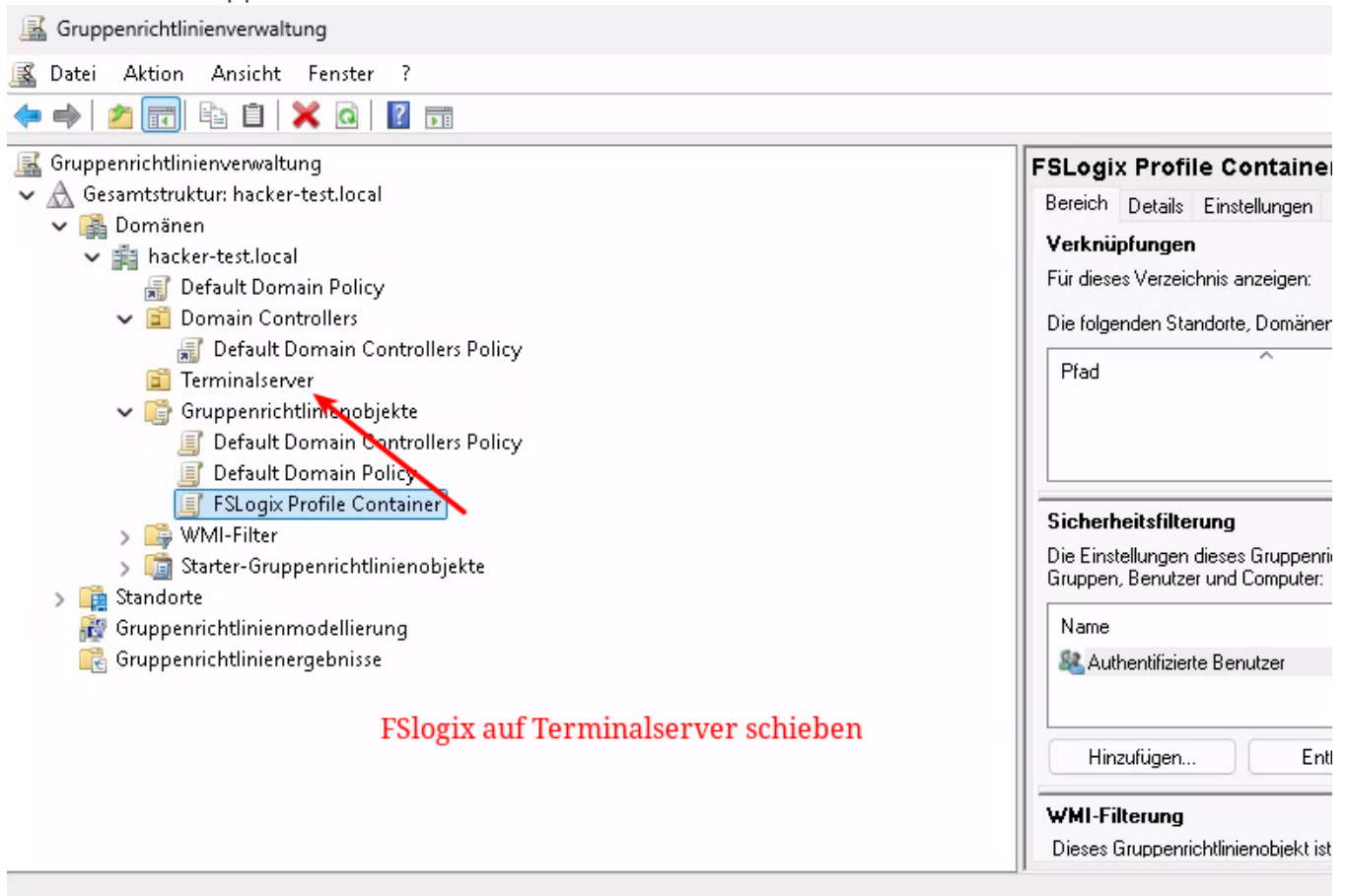


3. GPO mit der OU verknuepfen, in der die Terminalserver liegen, in unserem Fall eine neue OU Rechtsklick auf die Dömäne dann neue Organistaoneinheit

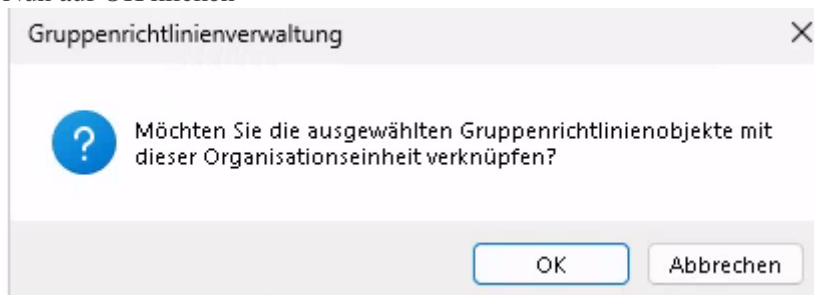


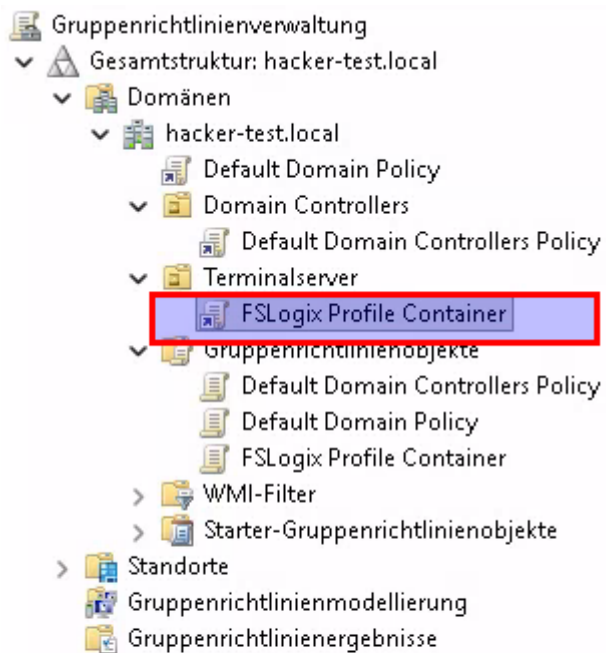


Nun unsere Gruppenrichtlinie auf Terminalserver ziehen



Nun auf OK klicken

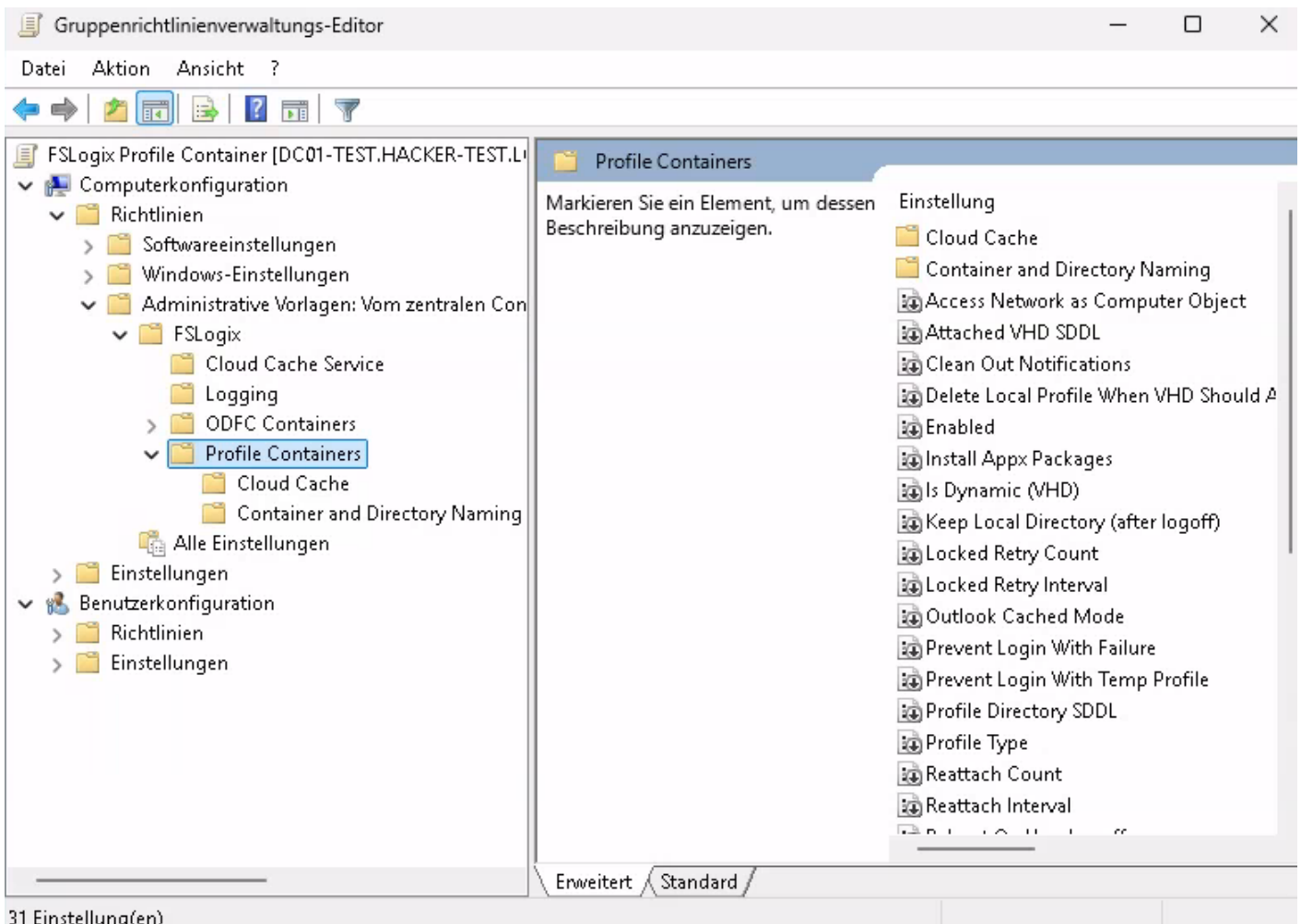




4. GPO bearbeiten

GPO-Einstellungen bearbeiten

Rechtsklick auf die GPO und bearbeiten auswählen:



Folgende Einstellungen konfigurieren:

Einstellung	Wert	Beschreibung
Enabled	Enabled + Dropdown auf "Enabled"	Aktiviert FSLogix (WICHTIG: Beides setzen!) Aber erst wenn die Profile migriert sind. Erstmal auf 0 lassen also Disabled, wenn Benutzer migriert werden sollen
VHD Locations	\\fileserver\FSGlobal\Profiles	Pfad zur Freigabe

Delete Local Profile When VHD Should Apply	Disabled	Lokales Profil loeschen wenn FSLogix greift: Auf Disabled setzen Ganz wichtig, damit vorhandene Profile nicht gelöscht werden: Sind lokale Profile vorhanden wir auch keine VHDX angelegt. Also verhalten so als gäbe es FSLogix gar nicht.
Size In MBs	30000	Maximale Groesse in MB (30 GB)
Prevent Login With Failure	Enabled	Anmeldung verhindern wenn Container nicht geladen werden kann
Prevent Login With Temp Profile	Enabled	Anmeldung mit temporaerem Profil verhindern, wenn schon woanders angemeldet ist
NoProfileAvailableMessage	Dein Profil ist bereits in Benutzung. Bitte melde dich zuerst am anderen Computer/Terminalserver ab	Nachricht für Prevent Login With Temp Profile, ansonsten kommt ne riesen Fehlerbox. Muss als Reg Key erstellt werden.
VolumeWaitTimeMS	20000	Legt fest, wie lange FSLogix maximal wartet, bis die VHDX-Datei verfügbar ist. Ist die Datei nach Ablauf der Zeit nicht verfügbar (z.B. weil sie von einem anderen Computer verwendet wird), schlägt der Login fehl. Default : 20000 (20 seconds)
RoamIdentity	Enabled	Speichert die Authentifizierungs-Tokens für Microsoft 365 im FSLogix-Container statt lokal. Ohne diese Einstellung werden die Anmeldedaten lokal auf dem Server gespeichert – der User muss dann auf jedem Server sein Passwort neu eingeben.

RoamCredentials nur Per Reg hinzufügar und das aus gutem Grund!

Speichert klassische Windows-Anmeldeinformationen (Benutzername/Passwort) im FSLogix-Container statt lokal. Diese Einstellung ist für ältere Authentifizierungsmethoden gedacht.

Wann aktivieren	Bei lokalem Exchange Server (On-Premises), IMAP/POP3 Konten, älteren Outlook-Versionen
Registry	HKLM\SOFTWARE\FSLogix\Profiles\RoamCredentials
Wert	1 = Aktiviert, 0 = Deaktiviert
GPO	Nicht verfügbar – nur per Registry konfigurierbar

Hinweis: Diese Einstellung gilt als veraltet. Die meisten modernen Umgebungen nutzen Microsoft 365 mit Web Account Manager (WAM) – dort ist stattdessen `RoamIdentity` die richtige Einstellung. `RoamCredentials` wird nur noch für Legacy-Umgebungen mit klassischer Authentifizierung benötigt.

Sicherheitswarnung: Bei aktiviertem `RoamCredentials` werden Passwörter im Netzwerk-Container gespeichert. Bei Diebstahl der VHDX-Datei könnten diese kompromittiert werden.

Powershell Befehl:

Enable:

```
New-ItemProperty -Path "HKLM:\SOFTWARE\FSLogix\Profiles" -Name "RoamCredentials" -Value 1 -PropertyType DWord -Force
```

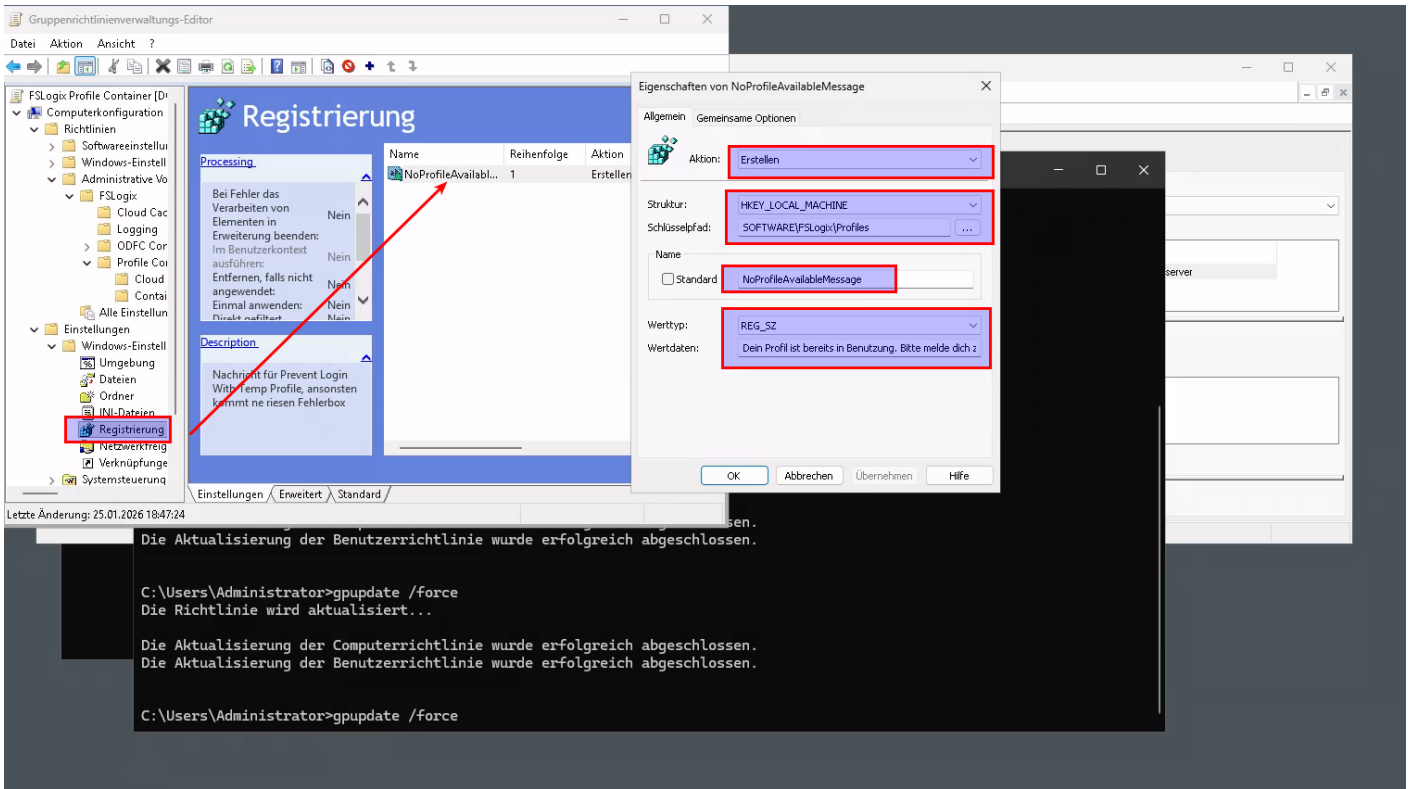
Disable:

```
New-ItemProperty -Path "HKLM:\SOFTWARE\FSLogix\Profiles" -Name "RoamCredentials" -Value 0 -PropertyType DWord -Force
```

NoProfileAvailableMessage nur Per Reg hinzufügar

Oder per GPO **Registry Preferences** verteilen:

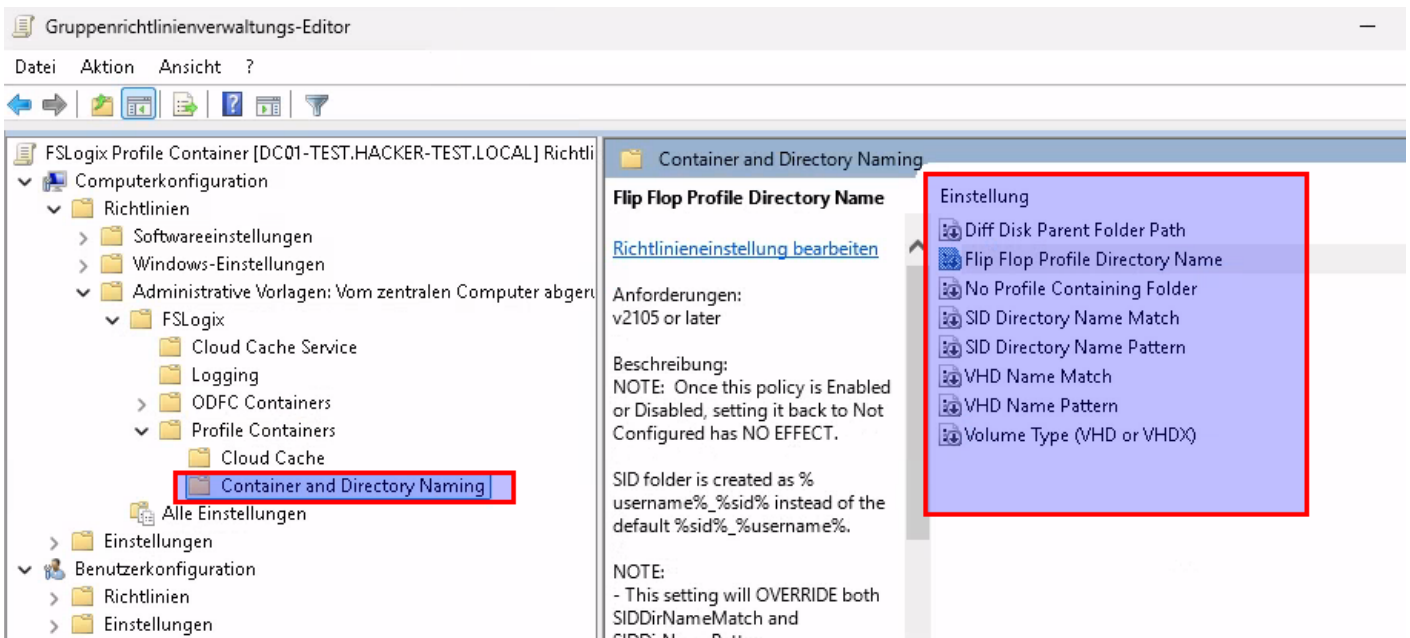
1. `gpmmc.msc` → deine FSLogix GPO bearbeiten
2. **Computerkonfiguration** → **Einstellungen** → **Windows-Einstellungen** → **Registrierung**
3. Rechtsklick → Neu → Registrierungselement
4. Eintragen:
 - Aktion: Erstellen
 - Struktur: HKEY_LOCAL_MACHINE
 - Schüsselpfad: SOFTWARE\FSLogix\Profiles
 - Wertname: NoProfileAvailableMessage
 - Werttyp: REG_SZ
 - Wertdaten: `Dein Profil ist bereits in Benutzung. Bitte melde dich zuerst am anderen Computer/Terminalserver ab.`



Wichtig: Bei "Enabled" muss sowohl der Status auf "Enabled" gesetzt werden ALS AUCH im Dropdown darunter nochmal "Enabled" ausgewaehlt werden. Sonst greift FSLogix nicht!

Optionale Einstellungen

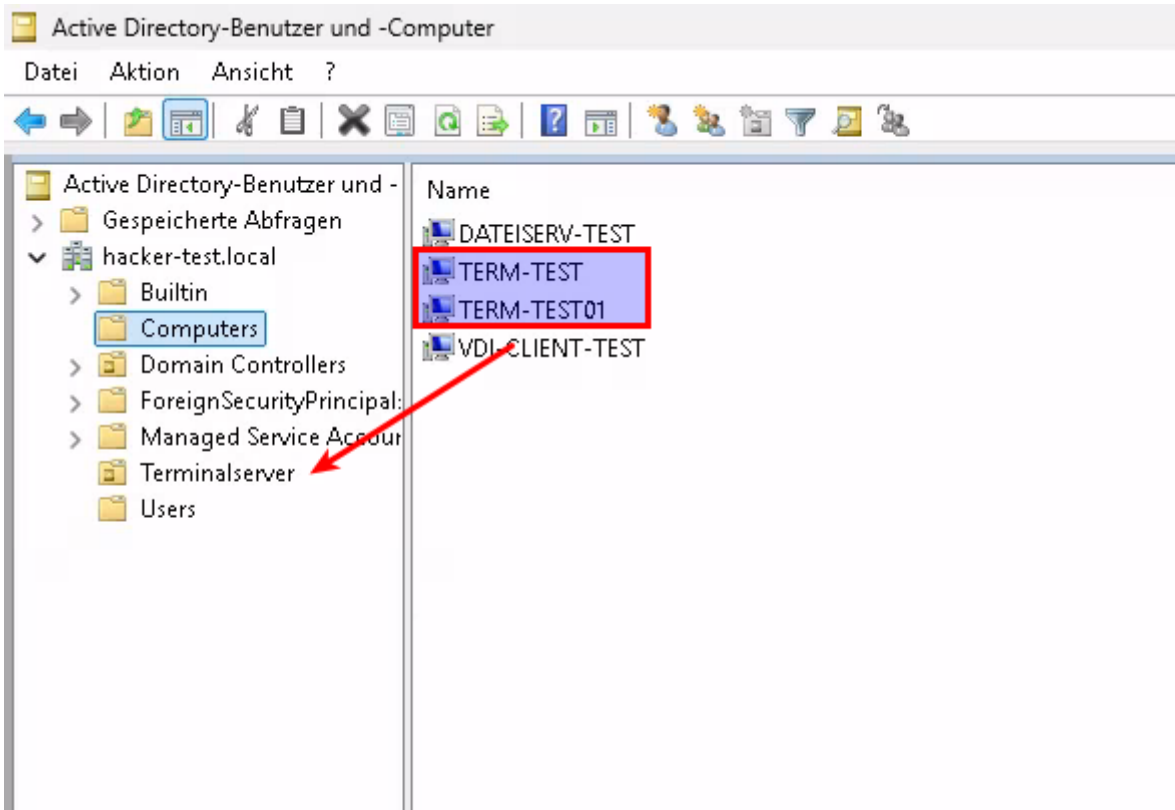
Unter Profile Containers -> Container and Directory Naming:



Einstellung	mpfehlung	Beschreibung
Flip Flop Profile Directory Name	Enabled	Ordner heissen username_SID statt SID_username

Virtual Disk Type	VHDX	Modernerer Format, besser fuer PowerShell-Verwaltung
-------------------	------	--

Nun noch im Active Directory Benutzer und Computer die Terminalserver auf unsere OU Terminal ziehen



GPO aktualisieren

Nach dem Konfigurieren auf dem DC und Terminalserver(n) die GPO aktualisieren oder Server neu starten.:

```
gpupdate /force
```

Nun können wir überprüfen ob die Gruppenrichtlinie auch drin ist.
Auf den Terminalserver(n) überprüfen.

```
gpresult /r
```

Ausgabe:

```
Betriebssystem Microsoft (R) Windows (R) Gruppenrichtlinienergebnis-Tool v2.0
```

```
© Microsoft Corporation. Alle Rechte vorbehalten.
```

```
Am 25.01.2026 um 15:05:48 erstellt
```

RSOP-Daten für HACKER-TEST\administrator auf TERM-TEST: Protokollmodus

Betriebssystemkonfiguration: Mitgliedsserver

Betriebssystemversion: 10.0.26100

Standortname: Default-First-Site-Name

Roamingprofil:Nicht zutreffend

Lokales Profil: C:\Users\administrator.HACKER-TEST

Langsame Verbindung? Nein

COMPUTEREINSTELLUNGEN

CN=TERM-TEST,OU=Terminalserver,DC=hacker-test,DC=local

Letzte Gruppenrichtlinienanwendung: 25.01.2026, um 15:05:33

Gruppenrichtlinienanwendung von: DC01-test.hacker-test.local

Schwellenwert für langsame Verbindung:500 kbps

Domänenname: HACKER-TEST

Domänentyp: Windows 2008 oder höher

Angewendete Gruppenrichtlinienobjekte

FSLogix Profile Container

Default Domain Policy

Richtlinien der lokalen Gruppe

Der Computer ist Mitglied der folgenden Sicherheitsgruppen

Administratoren

Jeder

Benutzer

NETZWERK

Authentifizierte Benutzer

Diese Organisation

TERM-TEST\$

Domänencomputer

Von der Authentifizierungsstelle bestätigte ID

Systemverbindlichkeitsstufe

BENUTZEREINSTELLUNGEN

CN=Administrator,CN=Users,DC=hacker-test,DC=local

Letzte Gruppenrichtlinienanwendung: 25.01.2026, um 15:05:34

Gruppenrichtlinienanwendung von: DC01-test.hacker-test.local

Schwellenwert für langsame Verbindung:500 kbps

Domänenname: HACKER-TEST

Domänentyp: Windows 2008 oder höher

Angewendete Gruppenrichtlinienobjekte

Nicht zutreffend

Folgende herausgefilterte Gruppenrichtlinien werden nicht angewendet.

Richtlinien der lokalen Gruppe

Filterung: Nicht angewendet (Leer)

Der Benutzer ist Mitglied der folgenden Sicherheitsgruppen

Domänen-Benutzer

Jeder

Benutzer

Administratoren

INTERAKTIVE REMOTEANMELDUNG

INTERAKTIV

Authentifizierte Benutzer

Diese Organisation

LOKAL

Richtlinien-Ersteller-Besitzer

Domänen-Admins

Organisations-Admins

Schema-Admins

Von der Authentifizierungsstelle bestätigte ID

Abgelehnte RODC-Kennwortreplikationsgruppe

Hohe Verbindlichkeitsstufe

Der entscheidene Teil:

Nun testen der Einstellungen:

```
Get-ItemProperty -Path "HKLM:\SOFTWARE\FSLogix\Profiles"
```

Ausgabe:

```
Enabled                : 1
VHDLocations           : \\dateiserv-test\Profiles
SizeInMBs              : 30000
VolumeType             : VHDX
FlipFlopProfileDirectoryName : 1
DeleteLocalProfileWhenVHDShouldApply : 0
PreventLoginWithFailure : 1
PreventLoginWithTempProfile : 1
PSPath                 :
Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SOFTWARE\FSLogix\Profiles
PSParentPath           : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SOFTWARE\FSLogix
PSChildName            : Profiles
PSDrive                : HKLM
PSProvider             : Microsoft.PowerShell.Core\Registry
```

Hinweis: Registry und GPO koennen parallel existieren. GPO-Einstellungen ueberschreiben Registry-Werte. Fuer Konsistenz empfiehlt sich EINE Methode.

5.6 Schritt 4: Testen

1. Server neu starten (oder Dienst frxsvc neu starten)
2. Mit einem Testuser anmelden
3. Auf der Freigabe prüfen ob VHDX erstellt wurde

Kontrolle am Client per CMD:

```
frx list-redirects
```

Zeigt an, ob das Profil korrekt gemountet ist.

6. Profilgröße überwachen und automatisch vergrößern

Ein volles Profil verhindert die Anmeldung! Daher ist Monitoring wichtig.

6.1 Manuelle Vergrößerung

User muss abgemeldet sein. Dann per PowerShell:

```
# VHDX auf 50 GB vergrößern
Resize-VHD -Path "\\fileserv\FSLogix\Profiles\user_SID\Profile_user.VHDX" -SizeBytes 50GB
```

Die Partition wächst beim nächsten Login automatisch mit.

6.2 Standardgröße für neue Profile ändern

Registry-Wert anpassen (Beispiel 50 GB = 50000 MB = 0xC350):

```
"SizeInMBs"=dword:0000c350
```

6.3 Automatisches Monitoring und Resize

Siehe beiliegendes PowerShell-Skript: [FSLogix-AutoResize.ps1 download](#)

FSLogix-AutoResize.ps1

```
# FSLogix VHDX Auto-Resize Script mit Mail-Benachrichtigung
# Prüft alle Profile und vergrößert automatisch wenn > 85% voll
#
# Einrichtung:
# 1. Skript anpassen (Pfade, Mail-Einstellungen)
# 2. Speichern unter C:\Scripts\FSLogix-AutoResize.ps1
# 3. Log-Ordner anlegen: mkdir C:\Logs
# 4. Scheduled Task erstellen (siehe unten)
#
# Scheduled Task per PowerShell erstellen:
# $action = New-ScheduledTaskAction -Execute "PowerShell.exe" -Argument "-NoProfile -
ExecutionPolicy Bypass -File C:\Scripts\FSLogix-AutoResize.ps1"
# $trigger = New-ScheduledTaskTrigger -Daily -At "06:00"
# $principal = New-ScheduledTaskPrincipal -UserId "SYSTEM" -LogonType ServiceAccount -
RunLevel Highest
# Register-ScheduledTask -TaskName "FSLogix AutoResize" -Action $action -Trigger $trigger -
Principal $principal

#region Konfiguration
#
=====
=====
# HIER ANPASSEN
#
```

```

=====
=====

# Pfad zu den FSLogix Profilen
$ProfilePath = "\\filesERVER\FsLogix\Profiles"

# Ab welchem Füllstand vergrößern (in Prozent)
$ThresholdPercent = 85

# Um wie viel GB vergrößern
$GrowByGB = 10

# Maximale Größe (Obergrenze)
$MaxSizeGB = 100

# Log-Datei
$LogFile = "C:\Logs\FsLogix-AutoResize.log"

# Mail-Konfiguration
$MailSettings = @{
    SmtPserver = "mail.example.de"      # SMTP-Server
    Port       = 587                    # 587 für STARTTLS, 465 für SSL, 25 für unverschlüsselt
    UseSsl     = $true                  # SSL/TLS verwenden
    Username   = "monitoring@example.de" # SMTP-Benutzername
    Password   = "DEIN-PASSWORT-HIER"   # SMTP-Passwort
    From       = "monitoring@example.de" # Absender
    To        = "admin@example.de"      # Empfänger (mehrere: "a@x.de","b@x.de")
}

#
=====
=====
# AB HIER NICHTS MEHR ÄNDERN
#
=====
=====
#endregion

#region Funktionen
function Write-Log {
    param(
        [string]$Message,
        [string]$Level = "INFO"
    )
}

```

```

# Log-Verzeichnis erstellen falls nicht vorhanden
$logDir = Split-Path -Path $LogFile -Parent
if (-not (Test-Path $logDir)) {
    New-Item -ItemType Directory -Path $logDir -Force | Out-Null
}

$timestamp = Get-Date -Format "yyyy-MM-dd HH:mm:ss"
$logEntry = "[$Level] $timestamp - $Message"

# In Datei schreiben
$logEntry | Out-File -FilePath $LogFile -Append -Encoding UTF8

# Auch auf Konsole ausgeben
switch ($Level) {
    "ERROR" { Write-Host $logEntry -ForegroundColor Red }
    "WARNING" { Write-Host $logEntry -ForegroundColor Yellow }
    "RESIZE" { Write-Host $logEntry -ForegroundColor Green }
    default { Write-Host $logEntry }
}
}

function Send-AlertMail {
    param(
        [string]$Subject,
        [string]$Body
    )

    try {
        $securePassword = ConvertTo-SecureString $MailSettings.Password -AsPlainText -Force
        $credential = New-Object System.Management.Automation.PSCredential
($MailSettings.Username, $securePassword)

        $mailParams = @{
            SmtpServer = $MailSettings.SmtpServer
            Port = $MailSettings.Port
            UseSsl = $MailSettings.UseSsl
            Credential = $credential
            From = $MailSettings.From
            To = $MailSettings.To
            Subject = $Subject
            Body = $Body
            BodyAsHtml = $true
            Encoding = [System.Text.Encoding]::UTF8
        }
    }
}

```

```
Send-MailMessage @mailParams
Write-Log "Mail erfolgreich gesendet: $Subject"
}
catch {
    Write-Log "Mail-Versand fehlgeschlagen: $_" -Level "ERROR"
}
}

function Get-HtmlReport {
    param(
        [array]$Warnings,
        [array]$Resizes,
        [array]$Errors
    )

    $html = @"
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <style>
        body {
            font-family: 'Segoe UI', Arial, sans-serif;
            margin: 20px;
            background-color: #f5f5f5;
        }
        .container {
            max-width: 800px;
            margin: 0 auto;
            background-color: white;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 2px 4px rgba(0,0,0,0.1);
        }
        h2 {
            color: #333;
            border-bottom: 2px solid #4472C4;
            padding-bottom: 10px;
        }
        h3 {
            margin-top: 25px;
        }
        table {
            border-collapse: collapse;
            width: 100%;
```

```
    margin-bottom: 20px;
}
th, td {
    border: 1px solid #ddd;
    padding: 10px 12px;
    text-align: left;
}
th {
    background-color: #4472C4;
    color: white;
}
tr:nth-child(even) {
    background-color: #f9f9f9;
}
tr:hover {
    background-color: #f1f1f1;
}
.warning {
    background-color: #fff3cd !important;
}
.error {
    background-color: #f8d7da !important;
}
.success {
    background-color: #d4edda !important;
}
.info-box {
    background-color: #e7f3ff;
    border-left: 4px solid #4472C4;
    padding: 10px 15px;
    margin-bottom: 20px;
}
.footer {
    margin-top: 30px;
    padding-top: 15px;
    border-top: 1px solid #ddd;
    font-size: 12px;
    color: #666;
}
</style>
</head>
<body>
<div class="container">
<h2>☐ FSLogix Auto-Resize Report</h2>
```

```

<div class="info-box">
  <strong>Server:</strong> $env:COMPUTERNAME<br>
  <strong>Zeitpunkt:</strong> $(Get-Date -Format "dd.MM.yyyy HH:mm:ss")<br>
  <strong>Profilpfad:</strong> $ProfilePath
</div>

```

```
"@
```

```
# Zusammenfassung
```

```
$html += "<p><strong>Zusammenfassung:</strong> "
```

```
$summaryParts = @()
```

```
if ($Resizes.Count -gt 0) { $summaryParts += "□ $($Resizes.Count) Profile vergrößert" }
```

```
if ($Warnings.Count -gt 0) { $summaryParts += "△ $($Warnings.Count) Warnungen" }
```

```
if ($Errors.Count -gt 0) { $summaryParts += "□ $($Errors.Count) Fehler" }
```

```
$html += ($summaryParts -join " | ") + "</p>"
```

```
# Vergrößerte Profile
```

```
if ($Resizes.Count -gt 0) {
```

```
  $html += @"
```

```
  <h3>□ Automatisch vergrößerte Profile</h3>
```

```
  <table>
```

```
    <tr>
```

```
      <th>Profil</th>
```

```
      <th>Alte Größe</th>
```

```
      <th>Neue Größe</th>
```

```
      <th>Belegt</th>
```

```
    </tr>
```

```
"@
```

```
  foreach ($r in $Resizes) {
```

```
    $html += "<tr class='success'><td>$($r.Name)</td><td>$($r.OldSize)
```

```
GB</td><td>$($r.NewSize) GB</td><td>$($r.UsedPercent)%</td></tr>"
```

```
  }
```

```
  $html += "</table>"
```

```
}
```

```
# Warnungen
```

```
if ($Warnings.Count -gt 0) {
```

```
  $html += @"
```

```
  <h3>△ Maximum erreicht - Manuelle Prüfung erforderlich!</h3>
```

```
  <p style="color: #856404;">Diese Profile haben die maximale Größe von $MaxSizeGB GB erreicht und können nicht weiter automatisch vergrößert werden.</p>
```

```
  <table>
```

```
    <tr>
```

```
      <th>Profil</th>
```

```
      <th>Aktuelle Größe</th>
```

```

        <th>Belegt</th>
    </tr>
"@
    foreach ($w in $Warnings) {
        $html += "<tr class='warning'><td>$($w.Name)</td><td>$($w.Size)
GB</td><td>$($w.UsedPercent)%</td></tr>"
    }
    $html += "</table>"
}

# Fehler
if ($Errors.Count -gt 0) {
    $html += @"
<h3>␣ Fehler bei der Verarbeitung</h3>
<table>
    <tr>
        <th>Profil</th>
        <th>Fehlermeldung</th>
    </tr>
"@
    foreach ($e in $Errors) {
        $html += "<tr class='error'><td>$($e.Name)</td><td>$($e.Error)</td></tr>"
    }
    $html += "</table>"
}

# Footer
$html += @"
<div class="footer">
    Diese E-Mail wurde automatisch generiert vom FSLogix Auto-Resize Script.<br>
    Schwellwert: ${ThresholdPercent}% | Vergrößerung: +${GrowByGB} GB | Maximum:
${MaxSizeGB} GB
</div>
</div>
</body>
</html>
"@

    return $html
}
#endregion

#region Hauptlogik
Write-Log "=== Starte FSLogix Auto-Resize ==="

```

```

# Prüfen ob Profilpfad erreichbar
if (-not (Test-Path $ProfilePath)) {
    Write-Log "Profilpfad nicht erreichbar: $ProfilePath" -Level "ERROR"

    $errorBody = Get-HtmlReport -Warnings @() -Resizes @() -Errors @(@{
        Name = "Profilpfad"
        Error = "Der Pfad $ProfilePath ist nicht erreichbar. Bitte Netzwerkverbindung und
Berechtigungen prüfen."
    })
    Send-AlertMail -Subject "FSLogix FEHLER: Profilpfad nicht erreichbar" -Body $errorBody
    exit 1
}

# Arrays für Report
$warnings = @()
$resizes = @()
$errors = @()

# Alle VHDX-Dateien finden
$vhdxFiles = Get-ChildItem -Path $ProfilePath -Recurse -Filter "*.vhd" -ErrorAction
SilentlyContinue

Write-Log "Gefundene Profile: $($vhdxFiles.Count)"

foreach ($vhd in $vhdxFiles) {
    try {
        # VHDX mounten (readonly) um Größe zu prüfen
        $mount = Mount-VHD -Path $vhd.FullName -ReadOnly -Passthru -ErrorAction Stop
        $partition = Get-Partition -DiskNumber $mount.DiskNumber | Where-Object { $_.Type -eq
"Basic" }

        if ($partition) {
            $volume = Get-Volume -Partition $partition
            $usedPercent = [math]::Round(((($volume.Size - $volume.SizeRemaining) /
$volume.Size) * 100, 1)
            $currentSizeGB = [math]::Round($volume.Size / 1GB, 1)

            Write-Log "Prüfe: $($vhd.Name) - $usedPercent% belegt ($currentSizeGB GB)"

            # Dismounten vor möglichem Resize
            Dismount-VHD -Path $vhd.FullName

            # Prüfen ob Schwellwert überschritten
            if ($usedPercent -ge $ThresholdPercent) {
                $currentMaxGB = [math]::Round((Get-VHD -Path $vhd.FullName).Size / 1GB, 1)
            }
        }
    }
}

```

```

$newSizeGB = $currentMaxGB + $GrowByGB

if ($newSizeGB -le $MaxSizeGB) {
    # Vergrößern
    Write-Log "RESIZE: $($vhdx.Name) von $currentMaxGB GB auf $newSizeGB GB" -
Level "RESIZE"
    Resize-VHD -Path $vhdx.FullName -SizeBytes ($newSizeGB * 1GB)

    $resizes += @{
        Name      = $vhdx.Name
        OldSize   = $currentMaxGB
        NewSize   = $newSizeGB
        UsedPercent = $usedPercent
    }
}
else {
    # Maximum erreicht
    Write-Log "WARNUNG: $($vhdx.Name) hat Maximum ($MaxSizeGB GB) erreicht!" -
Level "WARNING"

    $warnings += @{
        Name      = $vhdx.Name
        Size      = $currentMaxGB
        UsedPercent = $usedPercent
    }
}
}
else {
    Dismount-VHD -Path $vhdx.FullName
    Write-Log "Keine gültige Partition gefunden in: $($vhdx.Name)" -Level "WARNING"
}
}
catch {
    Write-Log "FEHLER bei $($vhdx.Name): $_" -Level "ERROR"

    # Sicherstellen dass VHD dismounted ist
    Dismount-VHD -Path $vhdx.FullName -ErrorAction SilentlyContinue

    $errors += @{
        Name = $vhdx.Name
        Error = $_.Exception.Message
    }
}
}

```

```

}

# Mail senden wenn es was zu berichten gibt
if ($warnings.Count -gt 0 -or $resizes.Count -gt 0 -or $errors.Count -gt 0) {

    # Betreff zusammenbauen
    $subject = "FSLogix Report: "
    $parts = @()

    if ($resizes.Count -gt 0) { $parts += "$($resizes.Count) vergrößert" }
    if ($warnings.Count -gt 0) { $parts += "$($warnings.Count) Warnungen" }
    if ($errors.Count -gt 0) { $parts += "$($errors.Count) Fehler" }

    $subject += $parts -join ", "

    # HTML-Report erstellen und senden
    $body = Get-HTMLReport -Warnings $warnings -Resizes $resizes -Errors $errors
    Send-AlertMail -Subject $subject -Body $body
}
else {
    Write-Log "Keine Aktionen erforderlich - alle Profile im grünen Bereich"
}

Write-Log "=== Durchlauf beendet ==="
#endregion

```

Das Skript:

- Prüft alle Profile auf Füllstand
- Vergrößert automatisch wenn über Schwellwert (Standard: 85%)
- Sendet E-Mail-Benachrichtigung bei Aktionen oder Fehlern
- Loggt alle Aktivitäten

Einrichtung als Scheduled Task:

```

$action = New-ScheduledTaskAction -Execute "PowerShell.exe" -Argument "-NoProfile -ExecutionPolicy Bypass -
File C:\Scripts\FsLogix-AutoResize.ps1"
$trigger = New-ScheduledTaskTrigger -Daily -At "06:00"
$principal = New-ScheduledTaskPrincipal -UserId "SYSTEM" -LogonType ServiceAccount -RunLevel Highest
Register-ScheduledTask -TaskName "FSLogix AutoResize" -Action $action -Trigger $trigger -Principal $principal

```

7. Bestehende Profile migrieren

7.1 Option A: Sauber neu anfangen

User meldet sich an, neues Profil wird erstellt. Outlook, Netzlaufwerke etc. müssen neu eingerichtet werden. Bei wenigen Usern vertretbar.

7.2 Option B: Profil übernehmen

Vorraussetzungen:

1. Das Scripte:
migrate2vhdx.ps1

```
2. # =====  
# FSLogix Profil-Migration  
# Mountet VHDX, kopiert Profildaten, trennt VHDX  
# =====  
  
# =====  
# Variablen anpassen  
# =====  
$username = "max.mustermann"  
$sid = "S-1-5-21-2630395647-2518157359-3528892432-1109"  
$driveLetter = "P"  
$profileShare = "\\dateiserv-test\Profiles"  
  
# =====  
# Ab hier nichts aendern  
# =====  
$vhdPath = "$profileShare\${username}_$sid\Profile_${username}.VHDX"  
$source = "C:\Users\${username}_migrate"  
$destination = "${driveLetter}:\Profile"  
  
Write-Host ""  
Write-Host "===== " -ForegroundColor  
Cyan  
Write-Host " FSLogix Profil-Migration" -ForegroundColor Cyan
```

```

Write-Host "=====" -ForegroundColor
Cyan
Write-Host ""
Write-Host "User:    $username" -ForegroundColor White
Write-Host "SID:      $sid" -ForegroundColor White
Write-Host "VHDX:     $vhdPath" -ForegroundColor White
Write-Host "Quelle:    $source" -ForegroundColor White
Write-Host "Ziel:     $destination" -ForegroundColor White
Write-Host ""

# Pruefen ob Quelle existiert
if (-not (Test-Path $source)) {
    Write-Host "FEHLER: Quellordner $source nicht gefunden!" -ForegroundColor Red
    Write-Host "Wurde das lokale Profil in _migrate umbenannt?" -ForegroundColor Yellow
    exit 1
}

# Pruefen ob VHDX existiert
if (-not (Test-Path $vhdPath)) {
    Write-Host "FEHLER: VHDX $vhdPath nicht gefunden!" -ForegroundColor Red
    Write-Host "Hat sich der User einmal angemeldet damit FSLogix die VHDX erstellt?" -
ForegroundColor Yellow
    exit 1
}

# VHDX mounten
Write-Host "[1/3] Monte VHDX..." -ForegroundColor Yellow
$diskpartMount = @"
select vdisk file="$vhdPath"
attach vdisk
select partition 1
assign letter=$driveLetter
"@
$diskpartMount | diskpart | Out-Null

# Kurz warten bis Laufwerk verfuegbar
Start-Sleep -Seconds 2

# Pruefen ob Mount erfolgreich

```

```

if (-not (Test-Path "${driveLetter}:\")) {
    Write-Host "FEHLER: Laufwerk ${driveLetter}: nicht verfuegbar!" -ForegroundColor Red
    Write-Host "VHDX konnte nicht gemountet werden." -ForegroundColor Yellow
    exit 1
}
Write-Host "    VHDX gemountet auf ${driveLetter}:" -ForegroundColor Green

# Daten kopieren
Write-Host "[2/3] Kopiere Profildaten..." -ForegroundColor Yellow
Write-Host "    Das kann je nach Profilgroesse einige Minuten dauern..." -ForegroundColor Gray
robocopy $source $destination /E /COPYALL /R:0 /W:0 /NP /NFL /NDL

if ($LASTEXITCODE -le 7) {
    Write-Host "    Kopieren erfolgreich!" -ForegroundColor Green
} else {
    Write-Host "    WARNUNG: Einige Dateien konnten nicht kopiert werden (Exit: $LASTEXITCODE)" -
ForegroundColor Yellow
}

# VHDX trennen
Write-Host "[3/3] Trenne VHDX..." -ForegroundColor Yellow
$diskpartDismount = @"
select vdisk file="$vhdPath"
detach vdisk
"@
$diskpartDismount | diskpart | Out-Null
Write-Host "    VHDX getrennt." -ForegroundColor Green

Write-Host ""
Write-Host "=====" -ForegroundColor Green
Write-Host " Migration abgeschlossen!" -ForegroundColor Green
Write-Host "=====" -ForegroundColor Green
Write-Host ""
Write-Host "Naechste Schritte:" -ForegroundColor Cyan
Write-Host "1. FSLogix aktivieren (Enabled = 1)" -ForegroundColor White
Write-Host "2. User $username kann sich jetzt anmelden" -ForegroundColor White
Write-Host "3. Nach erfolgreichem Test: $source loeschen" -ForegroundColor White

```

```
Write-Host ""
```

Kleines Hilfsscript um zu sehen, wo in der Registry die USER IDS auf welches Verzeichnis links:

```
Get-ChildItem "HKLM:\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList" | ForEach-Object {  
  
    $sid = $_.PSChildName  
    $path = (Get-ItemProperty $_.PSPath).ProfileImagePath  
    Write-Host "$sid -> $path"  
}
```

Ausgabe:

```
S-1-5-18 -> C:\WINDOWS\system32\config\systemprofile  
S-1-5-19 -> C:\WINDOWS\ServiceProfiles\LocalService  
S-1-5-20 -> C:\WINDOWS\ServiceProfiles\NetworkService  
S-1-5-21-2630395647-2518157359-3528892432-1109 -> C:\Users\max.mustermann.HACKER-TEST
```

3. SID herausbekommen in einer Powershell:

Wenn es ein lokaler Benutzer ist:

```
Get-LocalUser -Name "benutzername").SID
```

Wenn es ein Domänen Benutzer ist:

```
([System.Security.Principal.NTAccount]"DOMÄNE\BENUTZERNAME").Translate([System.Security.Principal.SecurityIdentifier]).Value
```

Ausgabe

```
S-1-5-21-2630395647-2518157359-3528892432-1109
```

Die user_SID würde dann so aussehen Benutzername_S-1-5-21-2630395647-2518157359-3528892432-1109

```
"C:\Program Files\FSLogix\Apps\frx.exe" copy-profile -filename  
"\\fileserver\FSLogix\Profiles\<<username>_<SID>\Profile_<username>.VHDX" -username "DOMAIN\username" -  
size-mbs 30000
```

Beispielpfad

```
\\dateiserv-test\Profiles\max.mustermann_S-1-5-21-2630395647-2518157359-3528892432-1107\Profile_max.mustermann.VHDX
```

Sauberer Ablauf für die Migration:

1. Die Scripte erstellt haben
2. Wenn credentials gesichert werden sollen, dann mit dem Benutzer nochmals anmelden und

```
rundll32.exe keymgr.dll,KRShowKeyMgr
```

ausführen, dann die Credentials im Benutzerverzeichnis sichern

3. FSLogix auf `Enabled = 1` setzen falls nicht schon geschehen
4. Lokales Benutzerprofil umbenennen in `<username>_migrate`
5. Benutzer anmelden, so das die VHDX erstellt wird.
6. Benutzer abmelden
7. SSID des Benutzers mit Befehl siehe oben herausbekommen und die Script anpassen
8. Das migrate script starten und danach dismount starten
9. Benutzer anmelden
10. Fertig

7.3 Option C: Migration mit Transwiz (Drittanbieter)

Für einfache GUI-basierte Migration:

4. Am alten Server Profil exportieren (.zip)
5. Am neuen Server importieren und User zuweisen
6. Danach FSLogix aktivieren

8. Troubleshooting

8.1 User kann sich nicht anmelden

- Freigabe erreichbar? Test: `dir \\fileserver\FSLogix\Profiles`
- VHDX voll? Manuell vergrößern (siehe Kapitel 6.1)
- VHDX bereits gemountet? Anderer Server/PC hält Container offen
- Berechtigungen korrekt? Creator Owner muss Vollzugriff haben

8.2 Profil wird nicht geladen (lokales Profil statt Container)

- Registry prüfen: Enabled = 1?
- Dienst läuft? services.msc ? FSLogix Apps Services
- Logs prüfen: C:\ProgramData\FsLogix\Logs

8.3 Anmeldung sehr langsam

- Netzwerk zum Fileserver prüfen (Latenz, Bandbreite)
- Antivirus-Ausnahmen für VHDX-Dateien?
- Storage-Performance (IOPS) ausreichend?

8.4 Nützliche Befehle

```
# Status prüfen
```

```
frx list-redirects
```

```
# Logs anzeigen
```

```
Get-Content C:\ProgramData\FsLogix\Logs\Profile*.log -Tail 50
```

```
# VHDX-Info abrufen
```

```
Get-VHD -Path "\\server\share\user_SID\Profile_user.VHDX"
```

9. Checkliste für die Einrichtung

Zum Abhaken:

Schritt	Erledigt
Lizenzierung geklärt (RDS-CAL, M365, etc.)	<input type="checkbox"/>
Freigabe erstellt mit korrekten Berechtigungen	<input type="checkbox"/>
FSLogix Agent installiert	<input type="checkbox"/>
Registry-Konfiguration importiert	<input type="checkbox"/>
Server/Client neu gestartet	<input type="checkbox"/>

Test-Anmeldung erfolgreich	<input type="checkbox"/>
VHDX auf Freigabe vorhanden	<input type="checkbox"/>
frx list-redirects zeigt gemountetes Profil	<input type="checkbox"/>
Monitoring-Skript eingerichtet	<input type="checkbox"/>
Scheduled Task für Auto-Resize erstellt	<input type="checkbox"/>
Bei RDS-Farm: Verbindungsbroker konfiguriert	<input type="checkbox"/>
Bei RDS-Farm: GPO für Single-Session aktiviert	<input type="checkbox"/>

10. ODFC-Container (Optional)

Der ODFC-Container (Office Data File Container) ist eine optionale Erweiterung von FSLogix speziell für Microsoft 365 und Exchange. Er ist KEIN Muss – wenn du nur den Profile Container nutzt, landen alle Office-Daten automatisch dort mit drin und alles funktioniert problemlos.

10.1 Was macht der ODFC-Container?

Der ODFC-Container speichert ausschließlich Daten von Microsoft 365 Anwendungen:

- Outlook Offline-Cache (OST-Datei) – kann schnell 10-20 GB werden
- OneDrive-Cache
- OneNote lokale Daten
- SharePoint-Cache
- Teams-Daten
- Office-Aktivierungsdaten

Wichtig: Der ODFC-Container ist optional! Ohne separate ODFC-Konfiguration landen alle Office-Daten einfach im normalen Profile Container – das funktioniert genauso gut.

10.2 Ein Container vs. Zwei Container

Variante	Vorteile	Nachteile
Nur Profile Container	Einfacher zu verwalten, weniger Komplexität, eine VHDX pro User	Office-Cache ist im Backup enthalten (braucht mehr Speicher)

Profile + ODFC Container	Office-Cache kann vom Backup ausgeschlossen werden, bessere Trennung	Zwei VHDXs pro User, mehr Verwaltungsaufwand
--------------------------	--	--

10.3 Wann lohnt sich der ODFC-Container?

Der separate ODFC-Container ist sinnvoll wenn:

- Du regelmässige Backups der Profile machst und den Office-Cache (nur Cache, kann neu aufgebaut werden) ausschliessen willst
- Du bereits eine andere Profilloesung hast (z.B. Citrix Profile Management) und nur den Office-Teil mit FSLogix loesen willst
- Du sehr grosse Outlook-Postfaecher hast (M365 E3/E5 bieten 100 GB Postfach, Outlook cached bis zu 50 GB)

Fuer kleinere Umgebungen (z.B. einzelner Terminalserver mit 25 Usern): Ein Container reicht voellig aus!

10.4 Konfiguration (falls gewuenscht)

Die Konfiguration erfolgt analog zum Profile Container per Registry oder GPO:

GPO-Pfad:

Computer Configuration -> Administrative Templates -> FSLogix -> ODFC Containers

Registry-Eintraege:

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\FSLogix\ODFC]
"Enabled"=dword:00000001
"VHDLocations"="\\fileserv\FSLogix\ODFC"
"VolumeType"="VHDX"
"SizeInMBs"=dword:00007530
```

Du kannst den gleichen Share wie fuer Profile verwenden oder einen separaten. FSLogix erstellt dann pro User:

- profile_username.vhdx – restliches Profil
- odfc_username.vhdx – nur Office/M365-Daten

10.5 Gleichzeitige Anmeldung mit ODFC

Bei gleichzeitigen Sessions (z.B. User an VDI + veröffentlichte App vom RDSH) nutzt FSLogix Differenzdateien.

Fuer ODFC gibt es eine separate Einstellung:

VHDAccessMode – regelt wie gleichzeitige Zugriffe auf den ODFC-Container behandelt werden

Modus	Verhalten
0 (Direct Access)	Erste Session hat exklusiven Zugriff, weitere Sessions bekommen temporären Container
1 (Difference Disk - Default)	Änderungen werden in Differenzdatei geschrieben, bei Abmeldung zusammengeführt
2 (Read-Only)	Nur lesender Zugriff, Änderungen werden verworfen

Empfehlung: Bei einem einzelnen Terminalserver brauchst du dir darüber keine Gedanken machen – da kann sich der User ohnehin nur einmal anmelden.

10.6 Outlook Cache begrenzen

Um zu verhindern, dass der Outlook-Cache die Container sprengt, kannst du per GPO begrenzen wie viele Monate Outlook cached:

GPO-Pfad:

User Configuration -> Administrative Templates -> Microsoft Outlook -> Account Settings -> Exchange -> Cached Exchange Mode

- "Use Cached Exchange Mode" -> Aktiviert

- "Sync Settings" -> z.B. 3 Monate oder 6 Monate

Das reduziert den Speicherbedarf des Outlook-Caches erheblich.

Stand: Januar 2026 / Erstellt mit Claude

Version #21

Erstellt: 24 Januar 2026 09:28:24 von Admin

Zuletzt aktualisiert: 25 Januar 2026 17:52:59 von Admin