

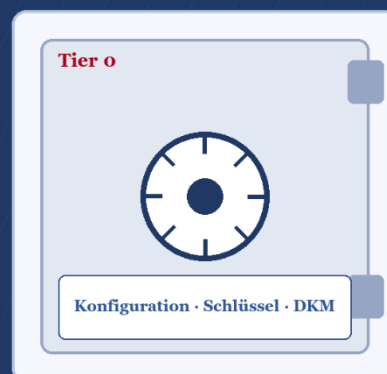
ADFS sichern und wiederherstellen – Rapid Restore Tool in der Praxis

ADFS · Sicherung und Wiederherstellung

Ein Backup, das nie zurückgespielt wurde, ist ein Gerücht

Was du sichern musst, wie das ADFS Rapid Restore Tool arbeitet und wie du die Farm auf neuer Hardware zurückholst.

boddenberg.de



WISSEN

Grundlagen, Architektur und alle Praxisbeiträge rund um ADFS an einem Ort.

› [Active Directory Federation Services](#)

BERATUNG

Du willst wissen, ob deine Sicherung im Ernstfall trägt? Wir spielen sie mit dir zurück, bevor es ernst wird.

› [Consulting zu ADFS](#)

SCHULUNG

Farm sichern, Server abreißen, Farm zurückholen – im Labor, wo der Puls ruhig bleiben darf.

› [ADFS-Schulungen](#)

Es gibt zwei Sorten von ADFS-Administratoren: die, die ihre Farm schon einmal wiederherstellen mussten, und die, denen das noch bevorsteht. Die erste Sorte erkennst du daran, dass sie bei dem Wort „Sicherung“ kurz zusammenzuckt und dann sehr genau nachfragt, wo das Kennwort liegt. Die zweite Sorte sagt Sätze wie „Die VM wird doch jede Nacht gesichert“ – und schläft gut. Noch.

Wer „adfs backup wiederherstellen“ in die Suchmaschine tippt, sitzt selten entspannt mit einem Kaffee vor dem Testsystem. Meist ist gerade ein Server gestorben, ein Update hat die Datenbank zerlegt oder jemand hat im Active Directory „aufgeräumt“. Dieser Beitrag hilft dir in beiden Lagen: vorher, damit du weißt, was gesichert werden muss und wie du es mit dem ADFS Rapid Restore Tool erledigst, und hinterher, damit du die Farm auf neuer Hardware zurückholst, ohne dabei die halbe Konfiguration zu verlieren. Die Grundlagen zu ADFS, Farm und Proxy findest du auf der Übersichtsseite [Active Directory Federation Services](#) – hier geht es ausschließlich um Sicherung und Wiederherstellung.

FAKTEN – ADFS-Sicherung auf einen Blick

Das ADFS Rapid Restore Tool ist ein kostenloses Werkzeug von Microsoft. Es sichert die ADFS-Konfiguration ohne vollständige Sicherung von Betriebssystem oder Systemstatus und spielt sie auf eine frische Installation zurück.

Es läuft mit ADFS ab Windows Server 2016 und braucht .NET Framework 4.6 oder neuer. Die aktuelle Version 2.0.2464.1 stammt von Dezember 2023.

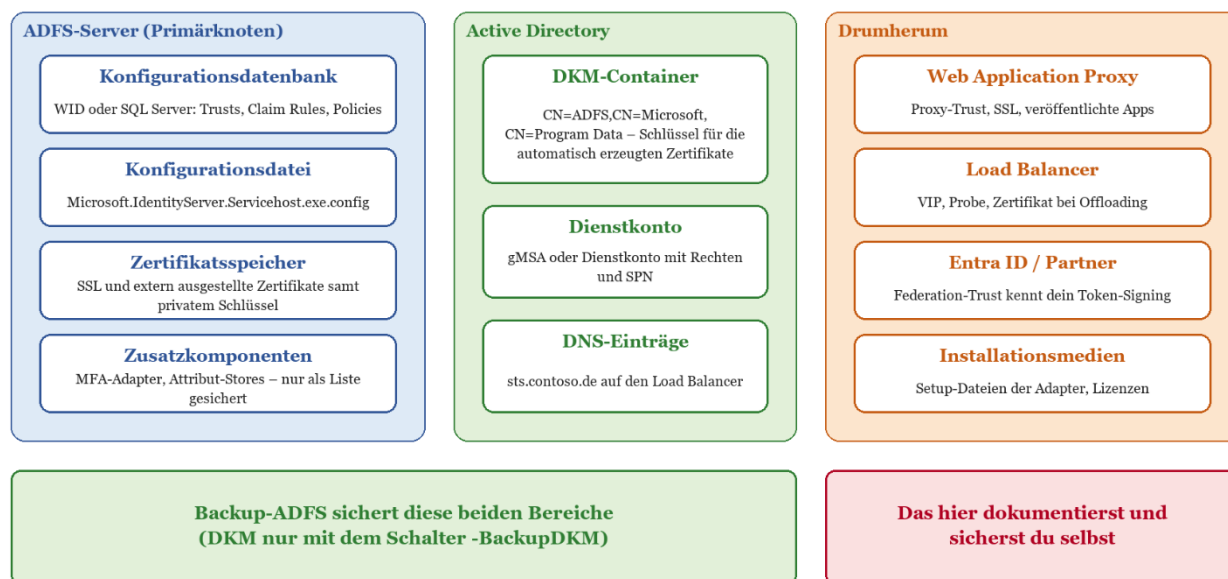
Gesichert werden die Konfigurationsdatenbank (WID oder SQL Server), die Konfigurationsdatei, eine Liste der Zusatzkomponenten, SSL- und extern ausgestellte Zertifikate mit privatem Schlüssel und – mit -BackupDKM – der DKM-Container aus dem Active Directory.

Jede Datei der Sicherung wird mit AES-256 verschlüsselt. Der Schlüssel wird aus dem Kennwort abgeleitet, das du beim Sichern angibst. Ohne dieses Kennwort ist die Sicherung Datenmüll.

Die Wiederherstellung muss auf derselben ADFS-Version laufen und dasselbe AD-Konto als Dienstkonto verwenden.

Skizze 1: Die drei Orte, an denen deine Farm wohnt

ADFS-Server, Active Directory und alles drumherum – das Rapid Restore Tool deckt nur die ersten beiden ab



Skizze 1: Die Konfiguration lebt auf dem ADFS-Server, die Schlüssel teilweise im Active Directory – und vieles Wichtige liegt ganz woanders.

Was du bei ADFS sichern musst – und warum die VM-Sicherung nicht reicht

Eine ADFS-Farm ist kein einzelner Server mit einer Datenbank, sondern ein Geflecht aus drei Orten. Auf dem ADFS-Server liegen Konfigurationsdatenbank, Konfigurationsdatei und Zertifikate. Im Active Directory liegen das Dienstkonto und – bei der Standardkonfiguration – der Schlüssel, mit dem ADFS seine automatisch erzeugten Zertifikate verschlüsselt. Und drumherum stehen Web Application Proxy, Load Balancer, DNS und die Gegenseiten deiner Vertrauensstellungen, vor allem Entra ID. Wer nur einen dieser Orte sichert, hat am Ende ein sehr teures Puzzle mit fehlenden Teilen.

Konfigurationsdatenbank und Konfigurationsdatei

In der Konfigurationsdatenbank steht alles, was du in den letzten Jahren mühsam zusammengekllickt hast: Relying Party Trusts, Claims Provider Trusts, Claim Rules, Access Control Policies, Endpunkte, Anmeldeseiten-Anpassungen und die Farm-Eigenschaften. Ob die Datenbank in der Windows Internal Database oder auf einem SQL Server liegt, spielt für das Rapid Restore Tool keine große Rolle – es kann sogar zwischen beiden wechseln. Die Unterschiede beider Varianten beschreibt der Beitrag [WID oder SQL Server als ADFS-Konfigurationsdatenbank](#).

Die Konfigurationsdatei Microsoft.IdentityServer.Servicehost.exe.config im ADFS-Verzeichnis enthält unter anderem die Verbindungszeichenfolge zur Datenbank. Sie ist klein, unscheinbar und genau deshalb die Datei, die in jeder improvisierten Sicherung fehlt.

Zertifikate inklusive privater Schlüssel

ADFS arbeitet mit drei Zertifikatsrollen: dem SSL-Zertifikat für die Kommunikation (Service Communications), dem Token-Signing-Zertifikat und dem Token-Decrypting-Zertifikat. Ein

Zertifikat ohne privaten Schlüssel ist für ADFS ungefähr so nützlich wie ein Haustürschlüssel aus Pappe. Für die Sicherung heißt das: Die privaten Schlüssel müssen exportierbar sein, und das Konto, das die Sicherung ausführt, muss sie lesen dürfen.

Bei den Token-Zertifikaten gibt es zwei Welten. Nutzt du AutoCertificateRollover, erzeugt ADFS die Zertifikate selbst und legt sie verschlüsselt in der Konfigurationsdatenbank ab. Den Schlüssel dafür verwaltet der Distributed Key Manager im Active Directory – dazu gleich mehr. Hast du die Token-Zertifikate dagegen bei einer Zertifizierungsstelle ausstellen lassen, liegen sie im Zertifikatsspeicher des Computers, und das Tool sichert sie von dort. Wie du die Zertifikate erneuerst und was dabei schiefgehen kann, steht in [ADFS Token-Signing-Zertifikat erneuern – mit und ohne AutoCertificateRollover](#) und [SSL-Zertifikat auf ADFS und Web Application Proxy tauschen](#).

```
# Welche Zertifikate nutzt die Farm - und sind die Schlüssel exportierbar?
Get-AdfsCertificate | Select-Object CertificateType, IsPrimary, Thumbprint, StoreLocation,
StoreName

Get-AdfsSslCertificate

# Für extern ausgestellte Zertifikate im Computerspeicher:
Get-ChildItem Cert:\LocalMachine\My | Where-Object HasPrivateKey |
    Select-Object Subject, Thumbprint, NotAfter,
        @{n="Exportierbar"; e={ try { $_.PrivateKey.CspKeyContainerInfo.Exportable } catch {
"CNG - im MMC prüfen" } }}
```

Bestandsaufnahme der Zertifikate vor der ersten Sicherung.

WICHTIG – Nicht exportierbarer Schlüssel = keine Sicherung

Wurde das SSL-Zertifikat irgendwann mit nicht exportierbarem Schlüssel importiert, kann auch das Rapid Restore Tool es nicht sichern. Prüfe das jetzt, nicht im Notfall. Die Lösung ist meist banal: Zertifikat samt Schlüssel als PFX aus der ursprünglichen Quelle beschaffen oder neu ausstellen lassen und dieses Mal exportierbar importieren.

Der DKM-Container im Active Directory

Jetzt kommt der Teil, den viele erst kennenlernen, wenn er weg ist. Bei der Standardkonfiguration verschlüsselt ADFS die automatisch erzeugten Token-Signing- und Token-Decrypting-Zertifikate mit einem Schlüssel, den der Distributed Key Manager (DKM) im Active Directory ablegt. Der Container liegt unterhalb von CN=ADFS,CN=Microsoft,CN=Program Data in deiner Domäne, darunter ein Container mit einer GUID als Namen. Welchen genau deine Farm nutzt, verrät die Eigenschaft CertificateSharingContainer.

```
# Welcher DKM-Container gehört zu meiner Farm?
(Get-AdfsProperties).CertificateSharingContainer
# Beispielausgabe:
# CN=8f2c...,CN=ADFS,CN=Microsoft,CN=Program Data,DC=contoso,DC=local
```

Den DKM-Container der Farm ermitteln.

Löscht jemand diesen Container – etwa weil „Program Data“ im AD verdächtig nach Altlast klingt –, kann ADFS die privaten Schlüssel seiner Token-Zertifikate nicht mehr entschlüsseln. Die Farm stellt dann keine Tokens mehr aus, und zwar für niemanden. Das ist der Moment, in dem du dich freust, den AD-Papierkorb aktiviert und den Container zusätzlich gesichert zu haben. Das Rapid Restore Tool exportiert ihn mit dem Schalter -BackupDKM über ldfde samt allen Unterobjekten.

WARNUNG – Wer den DKM-Container lesen kann, hält den Generalschlüssel

Der DKM-Container ist kein harmloses Konfigurationsobjekt. Zusammen mit der Konfigurationsdatenbank reicht er, um die privaten Token-Signing-Schlüssel zu rekonstruieren – die

Grundlage für Golden-SAML-Angriffe. Leserechte darauf gehören ausschließlich dem ADFS-Dienstkonto und den Tier-o-Administratoren. Wie Angreifer genau das ausnutzen, beschreibt der Beitrag [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#).

Was kein Tool für dich sichert

Das Rapid Restore Tool sichert von Zusatzkomponenten wie MFA-Adaptern, eigenen Attribut-Stores oder lokalen Claims Provider Trusts nur eine Liste. Die Installationsdateien, Lizenzen und Konfigurationen dieser Komponenten musst du selbst vorhalten. Dasselbe gilt für die Umgebung: Web Application Proxy, Load Balancer, DNS-Einträge und die Federation-Konfiguration in Entra ID gehören nicht zur ADFS-Datenbank. Dokumentiere sie – am besten so, dass jemand anderes als du sie um drei Uhr nachts versteht.

Baustein	Wo er liegt	Sichert Backup-ADFS?	Was du zusätzlich tust
Konfigurationsdatenbank	WID oder SQL Server	Ja	Bei SQL mit Always On oder Merge-Replikation: SQL-Sicherung statt Tool
Konfigurationsdatei	ADFS-Verzeichnis	Ja	–
SSL-Zertifikat	Computerspeicher	Ja, wenn der Schlüssel exportierbar ist	PFX zusätzlich getrennt verwahren
Extern ausgestellte Token-Zertifikate	Computerspeicher	Ja, mit exportierbarem Schlüssel	Ablaufdaten überwachen
Automatisch erzeugte Token-Zertifikate	Datenbank, Schlüssel im DKM	Ja, mit -BackupDKM	AD-Papierkorb aktivieren
DKM-Container	Active Directory	Nur mit -BackupDKM	Leserechte prüfen
MFA-Adapter, Attribut-Stores	ADFS-Server	Nur als Liste	Setup-Dateien und Konfiguration ablegen
Dienstkonto oder gMSA	Active Directory	Nein	Kennwort oder gMSA-Berechtigungen dokumentieren
Web Application Proxy	DMZ	Nein	Veröffentlichungen exportieren, SSL-PFX sichern
Load Balancer, DNS	Netzwerk	Nein	Konfiguration im Netzwerk-Backup
Federation-Trust in Entra ID	Cloud	Nein	Nach dem Restore abgleichen

Tabelle 1: Was zur Farm gehört und wer es sichert.

Warum die nächtliche VM-Sicherung trotzdem nicht reicht

Eine Image-Sicherung des ADFS-Servers ist nicht falsch, sie ist nur nicht genug. Sie enthält weder den DKM-Container noch den Proxy, und beim Zurückspielen eines einzelnen Knotens einer WID-Farm holst du dir einen Server mit veraltetem Stand und womöglich gestörter Vertrauensstellung zur Domäne zurück. Das Rapid Restore Tool hat einen anderen Ansatz: Es sichert die Konfiguration, nicht den Server. Damit kannst du auf beliebiger Hardware neu anfangen – sofern Windows-Version und ADFS-Version zur Sicherung passen.

Verfahren	Stärke	Schwäche	Einsatz
ADFS Rapid Restore Tool	Konfiguration, Zertifikate und DKM in einer verschlüsselten Sicherung, Restore auf neuer Hardware	Keine Unterstützung für SQL Always On und Merge-Replikation, Zusatzkomponenten nur als Liste	Standard für jede Farm
VM- oder Image-Sicherung	Schnell zurück, wenn genau dieser Server kaputt ist	Kein DKM, Risiko veralteter Stände in der Farm	Ergänzung, nicht Ersatz

SQL-Sicherung der ADFS-Datenbanken	Bewährte Verfahren deines DBA-Teams	Zertifikate und DKM fehlen	Pflicht bei SQL Always On
Export per PowerShell	Lesbare Dokumentation einzelner Objekte	Kein vollständiger Restore möglich	Zusätzlich für Vergleich und Doku

Tabelle 2: Sicherungsverfahren im Vergleich.

Das ADFS Rapid Restore Tool in der Praxis

Das Tool heißt intern ADFS Rapid Recreation Tool, kommt als MSI aus dem Microsoft Download Center und installiert ein PowerShell-Modul mit genau zwei Cmdlets: Backup-ADFS und Restore-ADFS. Die offizielle Beschreibung mit Versionshistorie findest du unter [AD FS Rapid Restore Tool auf Microsoft Learn](#). Wichtig für deinen Betrieb: Mit Version 1.0.73.0 hat Microsoft die Verschlüsselung FIPS-konform umgestellt. Sicherungen aus älteren Versionen lassen sich mit der aktuellen Version nicht mehr zurückspielen. Wer noch einen Ordner mit Sicherungen von 2017 hütet, hütet also ein Museum.

Installation und Voraussetzungen

- ADFS ab Windows Server 2016, .NET Framework 4.6 oder neuer.
- Bei WID muss die Sicherung auf dem Primärknoten laufen. Get-AdfsSyncProperties zeigt dir, welche Rolle der Server hat.
- Das ausführende Konto braucht mindestens lokale Administratorrechte.
- Für den DKM-Container braucht es zusätzlich Domänen-Admin-Rechte, delegierten Zugriff auf den Container oder die Anmeldedaten des Dienstkontos. Bei einem gMSA kannst du keine Anmeldedaten übergeben – hier zählt der Zugriff auf den Container.
- Seit Version 1.0.75.0 erkennt Backup-ADFS selbst, ob der aktuelle Kontext den DKM-Container lesen darf. Damit sind automatisierte Sicherungen ohne Domänen-Admin möglich.

```
# Rolle des Servers prüfen - nur der Primärknoten einer WID-Farm darf sichern
Get-AdfsSyncProperties | Select-Object Role, PrimaryComputerName, LastSyncTime, LastSyncStatus

# Modul nach der Installation des MSI laden
Import-Module 'C:\Program Files (x86)\ADFS Rapid Recreation Tool\ADFSRapidRecreationTool.dll'
Get-Command -Module ADFSrapidRecreationTool
```

Vorbereitung auf dem Primärknoten.

Backup-ADFS: die erste Sicherung

Die Sicherung ist erfreulich unspektakulär. Du gibst Speicherort, Kennwort und einen Kommentar an, setzt -BackupDKM und wartest. Für jede Sicherung legt das Tool unterhalb des Speicherpfads einen eigenen Ordner nach dem Muster adfsBackup_ID_Datum-Uhrzeit an. Der Kommentar erscheint später bei der Auswahl im Restore – schreib also etwas hinein, das dir nachts hilft, zum Beispiel „vor Einspielen der Claim Rules für die Personal-App“, und nicht „test2“.

```
# Kennwort interaktiv abfragen statt im Klartext in die Konsole tippen
$pw = Read-Host -AsSecureString "Kennwort für die ADFS-Sicherung"
$klartext = [System.Net.NetworkCredential]::new("", $pw).Password

Backup-ADFS -StorageType "FileSystem" `
    -StoragePath "\\sicherung-t0.contoso.local\adfs$\\" `
    -EncryptionPassword $klartext `
    -BackupComment "Vor Änderung CR-0815: neue Relying Party für das Reisekostenportal"
    -BackupDKM

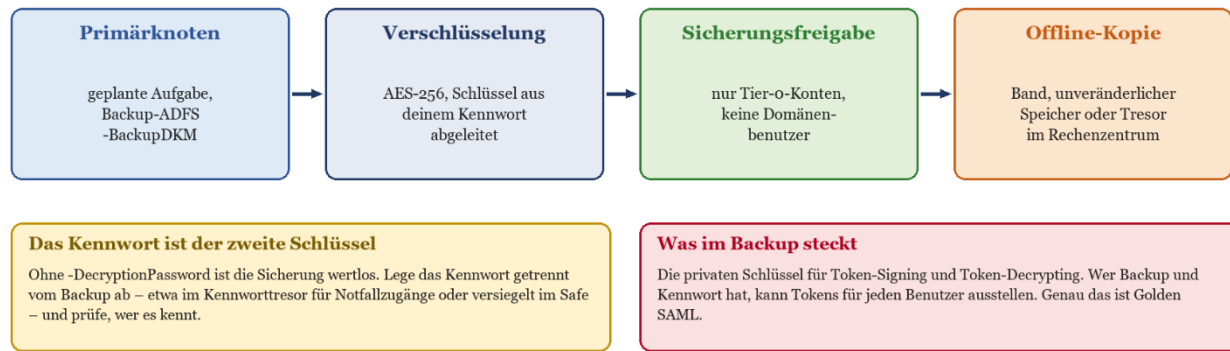
Remove-Variable klartext
```

Manuelle Sicherung vor einer Änderung, inklusive DKM-Container.

Neben FileSystem kennt das Tool den Speichertyp Azure, der die Sicherung in einen Azure-Storage-Container schreibt. Dafür übergibst du Kontoname und Zugriffsschlüssel des Speicherkontos. Das funktioniert, aber überleg dir gut, ob ein Speicherkonto-Schlüssel in einem Skript auf deinem ADFS-Server die Tier-o-Regeln erfüllt, die wir weiter unten besprechen. In den meisten Umgebungen ist eine streng geschützte Freigabe mit Offline-Kopie die ehrlichere Lösung.

Skizze 2: Vom Primärknoten bis in den Tresor

Backup-ADFS verschlüsselt jede Datei mit AES-256 – der Schutz der Ablage bleibt trotzdem dein Job



Skizze 2: Die Sicherung ist verschlüsselt – aber erst die geschützte Ablage und das getrennt verwahrte Kennwort machen sie sicher.

Sicherung automatisieren

Eine Sicherung, die nur entsteht, wenn jemand daran denkt, entsteht nie. Richte eine geplante Aufgabe auf dem Primärknoten ein, die täglich oder mindestens nach jeder Änderung sichert. Das Kennwort legst du dabei nicht als Klartext in das Skript, sondern verschlüsselt mit der Windows-Datenschnittstelle DPAPI: Export-Clixml schreibt ein Credential-Objekt so, dass nur dasselbe Konto auf demselben Computer es wieder lesen kann.

```

# Einmalig, angemeldet als das Konto der geplanten Aufgabe:
Get-Credential -UserName "adfsbackup" -Message "Kennwort für die ADFS-Sicherung" |
Export-Clixml -Path "C:\ADFSBackup\kennwort.xml"

# ---- Skript C:\ADFSBackup\Sichern-Adfs.ps1 ----
Import-Module 'C:\Program Files (x86)\ADFS Rapid Recreation Tool\ADFSRapidRecreationTool.dll'

if ((Get-AdfsSyncProperties).Role -eq "SecondaryComputer") { return } # nur der Primärknoten sichert

$ziel = "\\sicherung-t0.contoso.local\adfs$"
$cred = Import-Clixml -Path "C:\ADFSBackup\kennwort.xml"

Backup-ADFS -StorageType "FileSystem" -StoragePath $ziel `
-EncryptionPassword $cred.GetNetworkCredential().Password `
-BackupComment ("Automatisch {0:yyyy-MM-dd HH:mm}" -f (Get-Date)) `
-BackupDKM

# Aufbewahrung: Sicherungen älter als 90 Tage entfernen, mindestens 10 behalten
Get-ChildItem -Path $ziel -Directory -Filter "adfsBackup_*" |
Sort-Object CreationTime -Descending | Select-Object -Skip 10 |
Where-Object CreationTime -lt (Get-Date).AddDays(-90) |
Remove-Item -Recurse -Force

```

Automatisierte tägliche Sicherung mit Aufbewahrungsregel.

TIPP – Das Logfile ist dein Beweis

Für jede Sicherung und jede Wiederherstellung schreibt das Tool ein Protokoll nach

%LOCALAPPDATA%\ADFSRapidRecreationTool – also in das Profil des ausführenden Kontos. Lass dein Monitoring dort nach dem Ergebnis schauen oder prüfe zumindest, ob jeden Tag ein neuer Ordner entsteht.

Wie du ADFS insgesamt im Blick behältst, steht in [ADFS überwachen – Health Checks, Diagnose-Toolbox und Entra Connect Health](#).

Ergänzend: lesbare Exporte für Vergleich und Doku

Die Sicherung des Rapid Restore Tools ist verschlüsselt und nur für den Restore gedacht. Für die Frage „Was hat sich seit letztem Monat an den Claim Rules geändert?“ hilft sie dir nicht. Dafür lohnt sich ein zusätzlicher Export der wichtigsten Objekte als XML. Er ersetzt keine Sicherung, ist aber Gold wert, wenn du nach einem Restore prüfen willst, ob alles da ist – und er ist eine hervorragende Grundlage für [Relying Parties inventarisieren – die Bestandsaufnahme vor der Entra-Migration](#).

```
$stamp = Get-Date -Format "yyyy-MM-dd"
$pfad = "\\sicherung-t0.contoso.local\adfs\doku\$stamp"
New-Item -ItemType Directory -Path $pfad -Force | Out-Null

Get-AdfsProperties | Export-Clixml "$pfad\Farm.xml"
Get-AdfsRelyingPartyTrust | Export-Clixml "$pfad\RelyingPartyTrusts.xml"
Get-AdfsClaimsProviderTrust | Export-Clixml "$pfad\ClaimsProviderTrusts.xml"
Get-AdfsAccessControlPolicy | Export-Clixml "$pfad\AccessControlPolicies.xml"
Get-AdfsApplicationGroup | Export-Clixml "$pfad\ApplicationGroups.xml"
Get-AdfsEndpoint | Export-Clixml "$pfad\Endpunkte.xml"
Get-AdfsCertificate | Export-Clixml "$pfad\Zertifikate.xml"
Get-AdfsGlobalAuthenticationPolicy | Export-Clixml "$pfad\Authentifizierung.xml"
```

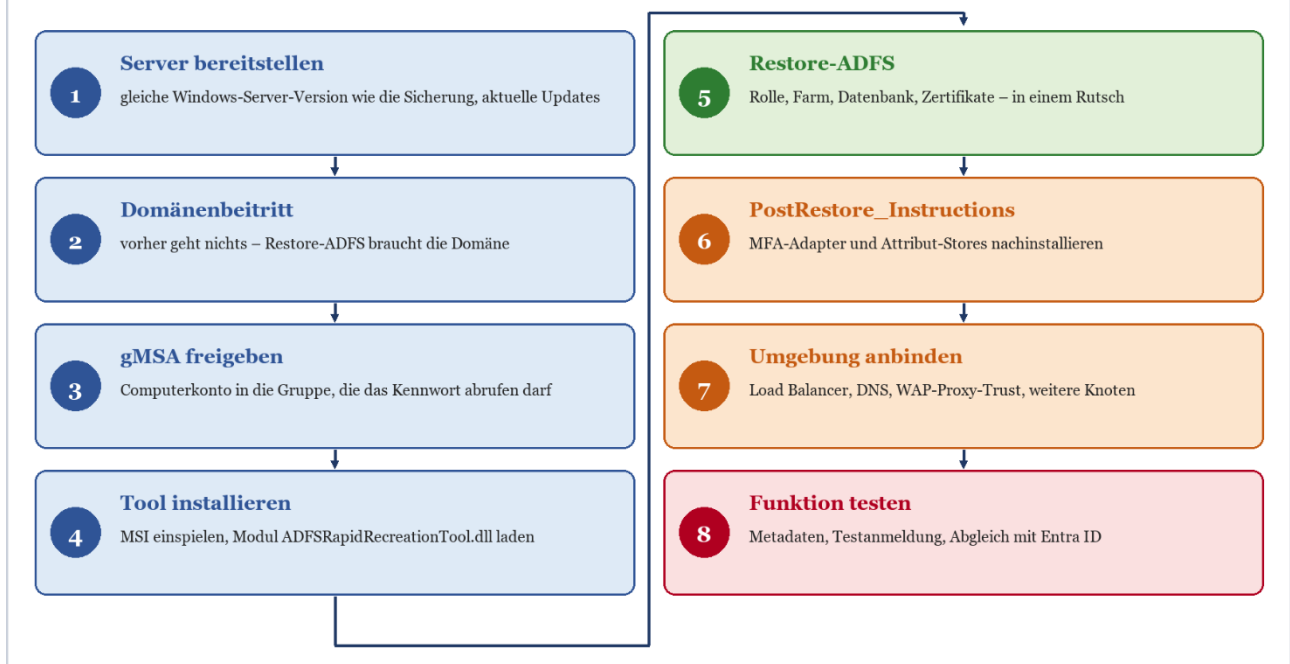
Lesbarer Konfigurationsexport als Ergänzung zur Sicherung.

Wiederherstellung auf neuer Hardware mit Restore-ADFS

Der Tag ist da. Der Server ist weg – Hardware tot, Hypervisor verschlüsselt, Update in den Abgrund gerissen, such dir etwas aus. Restore-ADFS legt auf einem frischen Server mit Install-AdfsFarm eine neue Farm an und spielt Konfiguration, Datenbank und Zertifikate zurück. Fehlt die ADFS-Rolle, installiert das Cmdlet sie selbst. Liegen im Speicherort mehrere Sicherungen, fragt es dich, welche du willst – mit Datum, Uhrzeit und deinem Kommentar.

Skizze 3: Wiederherstellung auf neuer Hardware in acht Schritten

Gleiche ADFS-Version, gleiches Dienstkonto, gleicher Dienstname – der Server selbst darf neu sein



Skizze 3: Der Restore selbst ist ein Befehl – die Schritte davor und danach entscheiden, ob es funktioniert.

Vorbereitung: der neue Server

Drei Dinge müssen stimmen, bevor du Restore-ADFS überhaupt aufrufst. Erstens die Version: Die Wiederherstellung muss auf einem ADFS-Server derselben Version laufen wie die Sicherung. Hast du auf Windows Server 2019 gesichert, stellst du auf Windows Server 2019 wieder her und hebst erst danach an – wie das sauber geht, zeigt [ADFS-Farm upgraden – Farm Behavior Level anheben ohne Ausfall](#). Zweitens die Domäne: Der Server muss Mitglied der Domäne sein. Drittens das Dienstkonto: Die Farm muss mit demselben AD-Konto laufen wie vorher.

Bei einem gMSA heißt Punkt drei konkret: Das Computerkonto des neuen Servers muss das gMSA-Kennwort abrufen dürfen. Ist der alte Server über eine Gruppe berechtigt, nimm den neuen in dieselbe Gruppe auf und starte ihn neu, damit das Kerberos-Ticket die neue Gruppenmitgliedschaft enthält. Details dazu im Beitrag [gMSA als ADFS-Dienstkonto einrichten und umstellen](#).

```

# Auf einem Domänencontroller oder mit RSAT: wer darf das gMSA-Kennwort abrufen?
Get-ADServiceAccount -Identity "gmsa-adfs" -Properties
PrincipalsAllowedToRetrieveManagedPassword |
    Select-Object -ExpandProperty PrincipalsAllowedToRetrieveManagedPassword

# Neuen Server in die berechtigte Gruppe aufnehmen
Add-ADGroupMember -Identity "ADFS-Server" -Members "ADFS03$"

# Auf dem neuen Server nach dem Neustart:
Test-ADServiceAccount -Identity "gmsa-adfs"
    
```

gMSA für den neuen Server freigeben und testen.

Restore-ADFS ausführen

Für den Restore brauchst du ein Konto, das gleichzeitig lokaler Administrator und Domänen-Administrator ist. Wechselst du bei der Gelegenheit von SQL Server auf WID oder umgekehrt, gibst du mit -DBConnectionString die neue Datenbank an – „WID“ oder eine SQL-Verbindungszeichenfolge. Den Schalter -RestoreDKM setzt du nur, wenn du in ein neues Active

Directory zurücksicherst, in dem der DKM-Container fehlt. Im bestehenden AD ist er in aller Regel noch da, und du lässt ihn in Ruhe.

```
Import-Module 'C:\Program Files (x86)\ADFS Rapid Recreation Tool\ADFSRapidRecreationTool.dll'

$pw = Read-Host -AsSecureString "Kennwort der ADFS-Sicherung"

# Restore in das bestehende AD, Farm mit gMSA, Datenbank als WID
Restore-ADFS -StorageType "FileSystem" `
    -StoragePath "\\sicherung-t0.contoso.local\adfs$" `
    -DecryptionPassword ([System.Net.NetworkCredential]::new("", $pw).Password) `
    -GroupServiceAccountIdentifier 'CONTOSO\gmsa-adfs$' `
    -DBConnectionString "WID"

# Variante: Restore in ein komplett neues AD (Labor, Totalverlust) - DKM gleich mit
zurückspielen
# Restore-ADFS ... -RestoreDKM
```

Wiederherstellung auf einem neuen Server.

WICHTIG — Erst die PostRestore_Instructions, dann der Dienststart

Nach dem Restore legt das Tool unter Umständen eine Datei PostRestore_Instructions ab. Darin steht, welche Authentifizierungsanbieter, Attribut-Stores und lokalen Claims Provider Trusts du von Hand nachinstallieren musst, bevor du den ADFS-Dienst startest. Klassiker ist der MFA-Adapter. Fehlt er, startet die Farm zwar, aber jede Anmeldung mit MFA-Pflicht scheitert.

Wie du den Entra-MFA-Adapter einrichtest, beschreibt [ADFS mit Entra-MFA koppeln – Adapter, Zertifikat und Fallstricke](#).

Nach dem Restore: die Farm wieder anbinden

Die Farm läuft auf einem Server – das ist ein Anfang, keine Farm. Jetzt kommen die Arbeiten, die in keiner Sicherung stecken:

- Weitere Knoten: Neue Server mit Add-AdfsFarmNode aufnehmen, wie in [ADFS-Farm installieren – Schritt für Schritt auf Windows Server 2022](#) beschrieben.
- Load Balancer und DNS: Den neuen Server in den Pool nehmen, alte Knoten entfernen, Probe prüfen. Hinweise dazu in [ADFS hinter dem Load Balancer – Health Probe, SNI und Persistenz](#).
- Web Application Proxy: Der Proxy vertraut der Farm über ein eigenes Zertifikat. Nach einem Restore musst du das Vertrauen häufig mit Install-WebApplicationProxy neu herstellen – Schritt für Schritt in [WAP-Proxy-Vertrauen erneuern – wenn der Web Application Proxy plötzlich nicht mehr will](#).
- Entra ID und Partner: Weil das Tool die Token-Signing-Schlüssel mitbringt, bleibt das Zertifikat dasselbe. Prüfe es trotzdem, bevor du Entwarnung gibst.

```
# Läuft die Farm, und mit welchen Knoten?
Get-AdfsFarmInformation
Get-Service adfssrv

# Metadaten über den Dienstenamen abrufen (läuft durch Load Balancer und SSL)
(Invoke-WebRequest -Uri "https://sts.contoso.de/FederationMetadata/2007-06/FederationMetadata.xml" -UseBasicParsing).StatusCode

# Kennt Entra ID dasselbe Token-Signing-Zertifikat? (Microsoft Graph PowerShell)
Connect-MgGraph -Scopes "Domain.Read.All"
$fed = Get-MgDomainFederationConfiguration -DomainId "contoso.de"
$adfs = Get-AdfsCertificate -CertificateType Token-Signing | Where-Object IsPrimary
$fed.SigningCertificate -eq [Convert]::ToBase64String($adfs.Certificate.RawData)
```

Funktionsprüfung nach der Wiederherstellung.

Liefert der Vergleich False, bist du nicht mehr im Restore-Thema, sondern im Zertifikatsthema. Früher stand an dieser Stelle in jeder Anleitung ein Aufruf aus dem MSOnline-Modul – das ist Geschichte. Wie du die Federation-Konfiguration heute mit Microsoft Graph PowerShell aktualisierst, zeigt [Microsoft-365-Federation-Trust nach Zertifikatswechsel aktualisieren](#).

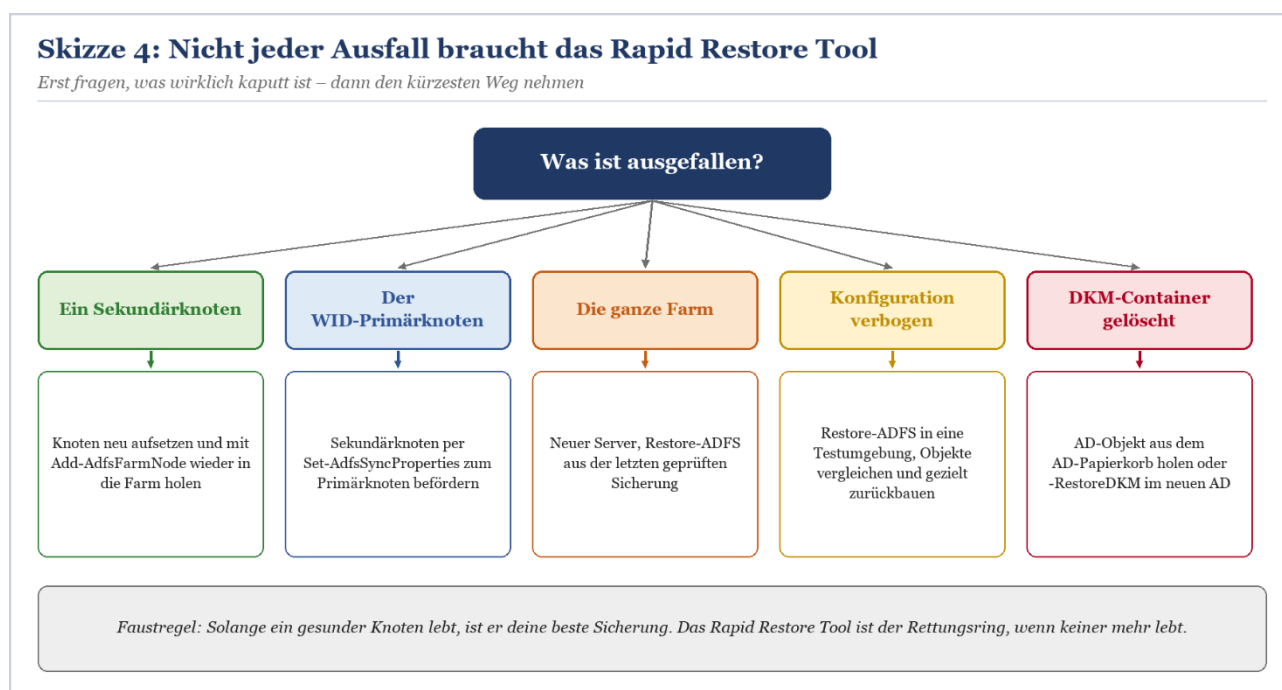
Nicht jeder Ausfall ist ein Fall für Restore-ADFS

Bevor du mit dem großen Besteck anrückst: In einer Farm mit mehreren Knoten ist ein gesunder Server die beste Sicherung, die du hast. Fällt nur ein Sekundärknoten aus, setzt du ihn neu auf und holst ihn mit Add-AdfsFarmNode zurück. Fällt der WID-Primärknoten aus, beförderst du einen Sekundärknoten mit Set-AdfsSyncProperties -Role PrimaryComputer und zeigst den übrigen Knoten den neuen Primärknoten. Das dauert Minuten statt Stunden.

```
# Auf dem Sekundärknoten, der neuer Primärknoten werden soll:
Set-AdfsSyncProperties -Role PrimaryComputer

# Auf jedem weiteren Sekundärknoten:
Set-AdfsSyncProperties -Role SecondaryComputer -PrimaryComputerName "adfs02.contoso.local"
```

WID-Sekundärknoten zum Primärknoten befördern.



Skizze 4: Fünf typische Ausfälle und der jeweils kürzeste Weg zurück.

Notfallplan, Wiederherstellungstest und Tier-o-Schutz

WARNUNG — Backup ohne Test ist Hoffnung

Eine Sicherung, die nie zurückgespielt wurde, ist kein Backup, sondern eine Behauptung. Ob das Kennwort stimmt, der Schlüssel exportierbar war, die Version passt und das gMSA auf dem neuen Server funktioniert, erfährst du nur durch einen echten Restore. Mach ihn in einem isolierten Labor mindestens einmal im Jahr und nach jedem größeren Umbau – und schreib die Zeit auf, die du gebraucht hast. Diese Zahl ist dein ehrliches Wiederherstellungsziel, nicht die aus dem Betriebshandbuch.

Der Notfallplan als Tabelle

Ein Notfallplan muss nicht schön sein, er muss nachts funktionieren. Die folgende Tabelle ist ein Gerüst, das du an deine Umgebung anpasst. Die Zeitangaben sind Beispielwerte aus der Praxis, keine Garantie – deine echten Werte liefert der Wiederherstellungstest.

Szenario	Erkennungsmerkmal	Sofortmaßnahme	Wiederherstellung	Beispielzeit
Sekundärknoten defekt	Load Balancer nimmt Knoten aus dem Pool	Knoten aus dem Pool lassen	Server neu, Add-AdfsFarmNode	1 Arbeitstag, kein Ausfall
WID-Primärknoten defekt	Änderungen nicht möglich, Sync-Fehler auf Sekundärknoten	Keine Konfigurationsänderungen	Sekundärknoten per Set-AdfsSyncProperties befördern	unter 1 Stunde
Ganze Farm verloren	Keine Anmeldung, alle Knoten tot	Notfallkommunikation, Entscheidung Restore	Neuer Server, Restore-ADFS, Umgebung anbinden	4 bis 8 Stunden
Konfiguration beschädigt	Einzelne Anwendungen scheitern nach Änderung	Änderung zurückdrehen, falls bekannt	Restore-ADFS im Labor, Vergleich, gezielt korrigieren	2 bis 4 Stunden
DKM-Container gelöscht	Token-Ausstellung scheitert auf allen Knoten	Löschvorgang bestätigen, nichts neu erzeugen	Container aus dem AD-Papierkorb oder AD-Sicherung	1 bis 2 Stunden
Verdacht auf Diebstahl der Sicherung	Unerklärlicher Zugriff auf die Ablage	Incident Response starten	Token-Zertifikate neu erzeugen, Trusts aktualisieren, Kennwörter wechseln	nach Lage

Tabelle 3: Notfallplan für die ADFS-Farm (Zeitangaben sind Beispielwerte).

Zum Plan gehört eine Liste mit allem, was du im Ernstfall brauchst und was nicht auf dem toten Server liegen darf: Speicherort der Sicherung, Kennwort im Notfalltresor, Installationsmedien der richtigen Windows-Version, Setup-Dateien der Zusatzkomponenten, SSL-PFX, Konfiguration von Load Balancer und Proxy sowie die Telefonnummern der Leute, die DNS und Firewall ändern dürfen. Wer das alles in einem Wiki auf einem Server pflegt, der sich per ADFS anmeldet, hat einen besonders schwarzen Humor.

Der Wiederherstellungstest im Labor

Der Test ist einfacher, als er klingt. Du brauchst ein isoliertes Netz mit einem Domänencontroller, einen Server in der richtigen Windows-Version und deine Sicherung. Im Labor ohne den echten DKM-Container nutzt du -RestoreDKM. Danach prüfst du die Liste aus Tabelle 4 ab und vergleichst das Ergebnis mit deinem XML-Export.

Prüfpunkt	Werkzeug	Erwartung
Dienst läuft	Get-Service adfssrv	Status Running
Farm-Eigenschaften	Get-AdfsProperties	Dienstname und Kennung wie in der Produktion
Relying Party Trusts vollständig	Get-AdfsRelyingPartyTrust	Anzahl und Namen wie im Export
Zertifikate da	Get-AdfsCertificate	Gleiche Fingerabdrücke wie in der Produktion
Zusatzkomponenten	PostRestore_Instructions	Alle Punkte erledigt
Anmeldung	Testbenutzer an einer Test-Relying-Party	Token wird ausgestellt
Zeitbedarf	Stoppuhr	Notiert und im Notfallplan eingetragen

Tabelle 4: Checkliste für den Wiederherstellungstest.

TIPP – Den Test zur Schulung machen

Lass beim Test nicht die Person ran, die das Skript geschrieben hat, sondern die Vertretung. Erst dann merkst du, welche Schritte nur in deinem Kopf stehen. Wer das unter Anleitung üben möchte, findet das passende Format in den [ADFS-Schulungen](#).

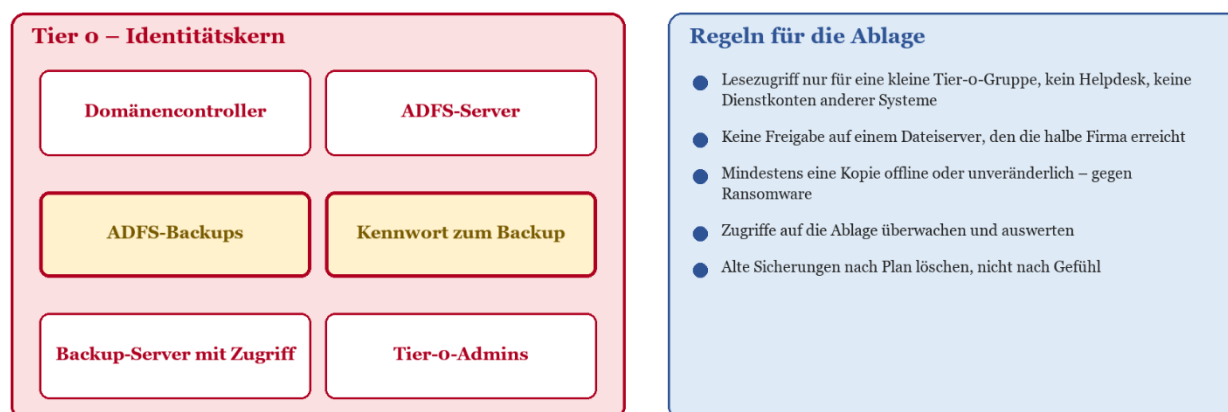
Das Backup als Tier-o-Asset schützen

Jetzt der Teil, der gern überlesen wird, weil er unbequem ist. Die Sicherung des Rapid Restore Tools enthält die privaten Schlüssel deiner Token-Zertifikate und auf Wunsch den DKM-Container. Wer Sicherung und Kennwort in die Finger bekommt, kann Tokens für jeden beliebigen Benutzer deiner Föderation ausstellen – für Microsoft 365, für jede angebundene SaaS-Anwendung, mit jeder Gruppenmitgliedschaft, die er sich wünscht. Das ist Golden SAML, nur mit freundlicher Unterstützung deines Backup-Konzepts.

Die Konsequenz: Die Sicherung gehört in dieselbe Schutzklasse wie Domänencontroller und ADFS-Server selbst. Das betrifft den Speicherort, die Konten mit Zugriff, den Backup-Server, der die Ablage weiterverarbeitet, und das Kennwort.

Skizze 5: Das Backup gehört in dieselbe Liga wie der Domänencontroller

Wer die Sicherung lesen kann, kann deine Identität fälschen – also gelten Tier-o-Regeln



Skizze 5: Sicherung und Kennwort sind so schützenswert wie der Domänencontroller.

- Ablage auf einem Speicher, auf den nur eine kleine Gruppe von Tier-o-Konten zugreift – nicht auf dem allgemeinen Dateiserver und nicht auf einem Laufwerk, das dein Backup-System mit einem Konto voller Domänenrechte durchsucht.
- Kennwort getrennt vom Backup verwahren, etwa im Kennworttresor für Notfallzugänge. Wer Backup und Kennwort im selben Ordner ablegt, kann sich die Verschlüsselung auch sparen.
- Mindestens eine Kopie offline oder unveränderlich halten. Ransomware verschlüsselt gern zuerst die Sicherungen.
- Zugriffe auf die Ablage protokollieren und auswerten, Ausreißer lösen einen Alarm aus.
- Ausgemusterte Sicherungen nachweisbar löschen. Ein alter USB-Stick mit Backup ist genauso gefährlich wie ein neuer.

WEITER – Passende Beiträge zur Absicherung

- › [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#) – warum Token-Signing-Schlüssel der Hauptgewinn für Angreifer sind
- › [gMSA als ADFS-Dienstkonto einrichten und umstellen](#) – das Dienstkonto, das der Restore

wiederfinden muss

› [ADFS: Architektur, Einsatzszenarien und Hochverfügbarkeit](#) – Hochverfügbarkeit, damit der Restore die Ausnahme bleibt

Fazit: Langweilige Sicherungen, spannende Labortests – nicht umgekehrt

ADFS ist nicht tot. In vielen Unternehmen läuft die Farm seit über zehn Jahren und stellt jeden Tag zuverlässig Tokens aus. Genau deshalb verdient sie eine Sicherung, die im Ernstfall trägt. Die Zutaten sind überschaubar: das Rapid Restore Tool mit -BackupDKM als tägliche geplante Aufgabe, exportierbare Zertifikatsschlüssel, ein aktivierter AD-Papierkorb, eine Dokumentation von allem, was außerhalb der Datenbank lebt, und ein Notfallplan, der nicht auf dem Server liegt, den er retten soll. Dazu kommt der eine Punkt, den kein Werkzeug für dich erledigt: der regelmäßige Wiederherstellungstest. Und weil die Sicherung den Generalschlüssel deiner Identität enthält, behandelst du sie wie einen Domänencontroller.

Auch wer den Ausstieg aus ADFS bereits plant, braucht bis zum letzten Tag eine funktionierende Sicherung – spätestens beim Abschalten der Farm wirst du froh sein, den Stand vorher sauber festgehalten zu haben. Wie das geordnet abläuft, beschreibt [ADFS abschalten – die Farm sauber außer Betrieb nehmen](#). Wenn du deine Sicherung einmal gemeinsam mit uns zurückspielen oder einen Notfallplan aufbauen möchtest, findest du uns im [Consulting zu ADFS \(Active Directory Federation Services\)](#). Betrieb, Hochverfügbarkeit und Wiederherstellung vertieft außerdem das Buch – mehr dazu unter [ADFS in der Praxis – das Buch im Detail](#).

FAQ: ADFS sichern und wiederherstellen

Wie sichere ich ADFS richtig?

Mit dem ADFS Rapid Restore Tool und dem Cmdlet Backup-ADFS inklusive -BackupDKM, regelmäßig als geplante Aufgabe auf dem Primärknoten. Ergänzend dokumentierst du Zusatzkomponenten, Web Application Proxy, Load Balancer und DNS.

Wo bekomme ich das ADFS Rapid Restore Tool?

Als kostenloses MSI im Microsoft Download Center. Nach der Installation lädst du das Modul ADFS Rapid Recreation Tool.dll aus dem Programmverzeichnis und hast die Cmdlets Backup-ADFS und Restore-ADFS.

Was sichert Backup-ADFS genau?

Die Konfigurationsdatenbank, die Konfigurationsdatei, eine Liste installierter Zusatzkomponenten, SSL- und extern ausgestellte Zertifikate mit privatem Schlüssel sowie mit -BackupDKM den DKM-Container und damit die Schlüssel der automatisch erzeugten Token-Zertifikate.

Was ist der DKM-Container bei ADFS?

Ein Container im Active Directory unterhalb von CN=ADFS,CN=Microsoft,CN=Program Data. Er enthält den Schlüssel, mit dem ADFS die automatisch erzeugten Token-Signing- und Token-Decrypting-Zertifikate schützt. Ohne ihn kann die Farm keine Tokens mehr signieren.

Kann ich ADFS auf neuer Hardware wiederherstellen?

Ja, genau dafür ist Restore-ADFS gedacht. Voraussetzungen sind dieselbe ADFS-Version wie bei der Sicherung, ein Domänenbeitritt des neuen Servers und dasselbe AD-Konto als Dienstkonto.

Kann ich mit dem Rapid Restore Tool von SQL Server auf WID wechseln?

Ja. Beim Restore gibst du mit -DBConnectionString entweder „WID“ oder eine SQL-Verbindungszeichenfolge an. Bei SQL mit Merge-Replikation oder Always-On-Verfügbarkeitsgruppen wird das Tool allerdings nicht unterstützt.

Muss ich nach dem Restore den Federation-Trust in Entra ID aktualisieren?

In der Regel nicht, weil die Token-Signing-Schlüssel mit zurückkommen. Prüfe es trotzdem mit Get-MgDomainFederationConfiguration aus Microsoft Graph PowerShell, bevor du Entwarnung gibst.

Warum funktioniert meine alte ADFS-Sicherung nicht mehr?

Häufigste Ursachen sind ein falsches Kennwort, eine andere ADFS-Version auf dem Zielsystem oder eine Sicherung aus einer Tool-Version vor 1.0.73.0 – seit der Umstellung auf FIPS-konforme Verschlüsselung lassen sich solche Sicherungen nicht mehr einspielen.

Wie oft sollte ich ADFS sichern?

Mindestens täglich automatisiert und zusätzlich vor jeder Änderung an Trusts, Claim Rules oder Zertifikaten. Einen Wiederherstellungstest im Labor solltest du mindestens einmal im Jahr durchführen.

Warum ist eine ADFS-Sicherung ein Tier-0-Asset?

Weil sie die privaten Token-Signing-Schlüssel enthält. Wer Sicherung und Kennwort besitzt, kann gültige Tokens für beliebige Benutzer ausstellen. Deshalb gelten für Ablage, Zugriffskonten und Kennwort dieselben Regeln wie für Domänencontroller.