

ADFS-Anmeldeseite anpassen – Logo, Texte, Hilfelinks



WISSEN

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

› [Active Directory Federation Services](#)

BERATUNG

Anmeldeseite, Themes und Texte aufräumen – und dabei gleich den Weg Richtung Entra ID mitdenken.

› [Consulting zu ADFS](#)

SCHULUNG

Im Labor: Themes exportieren, Texte je Sprache pflegen, onload.js anhängen, ohne die Farm zu zerlegen.

› [ADFS-Schulungen](#)

Die ADFS-Anmeldeseite ist die Visitenkarte deiner Firma. Jeden Morgen schauen Hunderte Kolleginnen und Kollegen darauf, Partner sehen sie bei jeder Freigabe, und externe Dienstleister bekommen sie zu Gesicht, bevor sie überhaupt wissen, wie euer Empfang aussieht. Und was steht auf dieser Visitenkarte? In erstaunlich vielen Farmen: das Logo von vor zwei Rebrandings, eine Illustration, die irgendwer 2014 aus dem Marketing-Ordner gefischt hat, der Satz „Sign in with your organizational account“ in tadellosem Englisch für eine rein deutschsprachige Belegschaft – und ein Hilfelink, der auf ein Intranet-Portal zeigt, das seit drei Jahren abgeschaltet ist. Wenn die Anmeldeseite ein Mensch wäre, hätte sie Schulterpolster und ein Faxgerät.

Das Gute: Du brauchst weder einen Webdesigner noch Mut zur Lücke, um das zu ändern. ADFS bringt seit Windows Server 2012 R2 Cmdlets mit, mit denen du Logo, Illustration, Texte und Links sauber anpasst, je Sprache und seit ADFS unter Windows Server 2016 sogar je Anwendung. Wenn du also nach „adfs anmeldeseite anpassen“ suchst, bekommst du hier den Weg vom Standard-Theme zu einem gepflegten Auftritt: Web-Theme kopieren und exportieren, Bilder tauschen, Texte und Hilfelinks setzen, einen „Kennwort vergessen?“-Link unterbringen und für einzelne Relying Parties abweichen. Am Ende steht ein Kapitel über JavaScript im onload.js, denn dort beginnt die Zone, in der Anpassungen gern beim nächsten Update Kopfschmerzen machen. Die Grundlagen zu Farm, Relying Parties und Web Application Proxy setzen wir voraus – sie stehen auf der Übersichtsseite [Active Directory Federation Services](#).

FAKTEN – ADFS-Anmeldeseite anpassen auf einen Blick

Zwei Werkzeugkästen: Web-Themes (Logo, Illustration, Stylesheet, onload.js) und Web-Content (Texte, Links, Fehlermeldungen).

Das mitgelieferte Theme heißt „default“. Eigene Anpassungen gehören in eine Kopie, die du mit Set-AdfsWebConfig -ActiveThemeName aktivierst.

Microsoft empfiehlt fürs Logo 260 × 35 Pixel bei 96 dpi und höchstens 10 KB, für die Illustration 1420 × 1080 Pixel bei 96 dpi als JPG oder PNG mit 200 bis 500 KB.

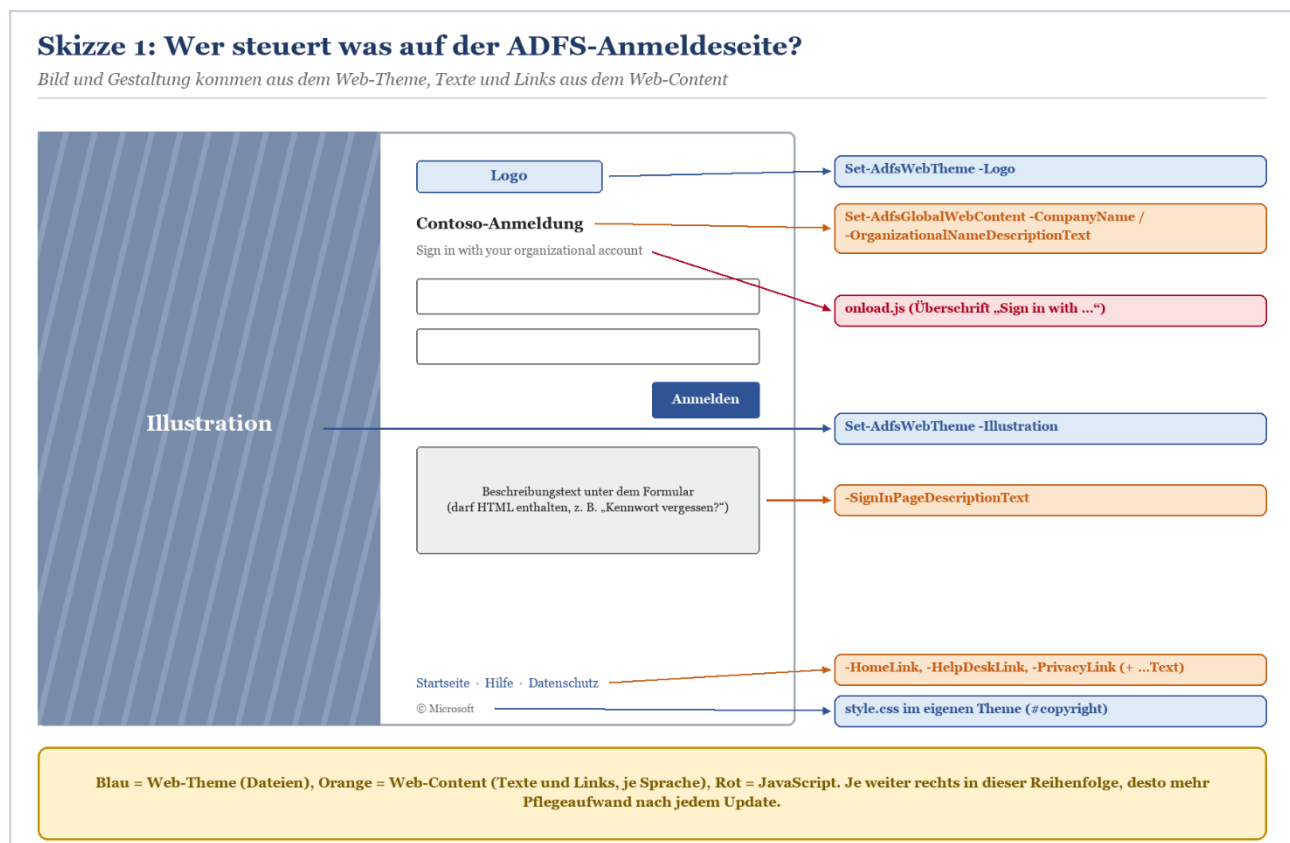
Texte und Bilder kennen einen Locale-Parameter. Erst den länderneutralen Wert pflegen („de“), dann bei Bedarf den länderspezifischen („de-AT“).

Seit ADFS unter Windows Server 2016: eigene Texte und eigene Themes pro Relying Party mit Set-AdfsRelyingPartyWebContent und Set-AdfsRelyingPartyWebTheme.

Alle Anpassungen landen in der Konfigurationsdatenbank und gelten damit für die ganze Farm. Der Web Application Proxy liefert sie unverändert aus.

Bestandsaufnahme: Was du an der Anmeldeseite überhaupt ändern kannst

Bevor du Logos durch die Gegend schiebst, lohnt sich ein Blick auf die Aufteilung, die ADFS selbst vornimmt. Alles, was eine Datei ist – Logo, Illustration, Stylesheet, Skript –, gehört zum Web-Theme. Alles, was ein Text oder ein Link ist, gehört zum Web-Content. Diese Trennung ist nicht nur Ordnungsliebe der Entwickler in Redmond, sie bestimmt auch, mit welchem Cmdlet du arbeitest und ob die Änderung sprachabhängig ist.



Skizze 1: Die Elemente der ADFS-Anmeldeseite und die Werkzeuge, die sie steuern

Element	Werkzeug	Ebene	Je Sprache?
Logo	Set-AdfsWebTheme -Logo	Theme, global oder je RP	ja, per Locale im Hashtable
Illustration (linke Bildfläche)	Set-AdfsWebTheme -Illustration	Theme, global oder je RP	ja, per Locale im Hashtable
Farben, Schriften, Copyright	style.css, eingespielt mit Set-AdfsWebTheme -StyleSheet	Theme	möglich, selten nötig
Firmenname (nur ohne Logo sichtbar)	Set-AdfsGlobalWebContent -CompanyName	Content, global oder je RP	ja
Beschreibung unter	-SignInPageDescriptionText	Content, global oder je RP	ja

dem Formular		je RP	
Startseite, Hilfe, Datenschutz	-HomeLink, -HelpDeskLink, -PrivacyLink und die zugehörigen ...Text-Parameter	Content, global oder je RP	ja
Fehlerseite	-ErrorPageDescriptionText, -ErrorPageGenericErrorMessage, -ErrorPageSupportEmail	Content, global oder je RP	ja
Seite „Kennwort ändern“	-UpdatePasswordPageDescriptionText	Content, nur global	ja
Überschrift über den Eingabefeldern	nur über onload.js	Theme	nur per Skriptlogik

Tabelle 1: Elemente der Anmeldeseite und ihre Werkzeuge

Erst schauen, was schon da ist

In einer Farm, die seit Jahren läuft, ist die spannendste Frage oft nicht „Wie ändere ich das?“, sondern „Wer hat hier schon was geändert?“. Drei Cmdlets geben dir die Antwort in zehn Sekunden: welches Theme aktiv ist, welche Themes existieren und welche Texte für welche Sprache gesetzt sind. Wer dabei auf ein Theme namens „Neu2_final_ECHT“ stößt, hat soeben ein Stück Firmengeschichte ausgegraben.

```
# Welches Theme ist aktiv?
Get-AdfsWebConfig | Select-Object ActiveThemeName

# Welche Themes gibt es überhaupt?
Get-AdfsWebTheme | Select-Object Name, IsBuiltinTheme

# Welche Texte und Links sind je Sprache gepflegt?
Get-AdfsGlobalWebContent | Format-List Locale, CompanyName, HelpDeskLink, HelpDeskLinkText,
    HomeLink, PrivacyLink, SignInPageDescriptionText

# Gibt es schon Abweichungen pro Anwendung?
Get-AdfsRelyingPartyWebContent
Get-AdfsRelyingPartyWebTheme
```

Bestandsaufnahme: aktives Theme, vorhandene Themes, globale und anwendungsspezifische Inhalte

WICHTIG – Vorher sichern, nicht hinterher bereuen

Exportiere das aktuell aktive Theme mit Export-AdfsWebTheme in einen Ordner, bevor du etwas änderst, und notiere dir die Ausgabe von Get-AdfsGlobalWebContent. Damit hast du einen Rückweg, der ohne Wiederherstellung der Konfigurationsdatenbank auskommt. Für den großen Rückweg gibt es trotzdem eine ordentliche Sicherung – wie die aussieht, zeigt [ADFS sichern und wiederherstellen – Rapid Restore Tool in der Praxis](#).

Was interne Nutzer gar nicht zu Gesicht bekommen

Eine Sache vorweg, damit du nicht stundenlang an einem Text feilst, den niemand liest: Wer im internen Netz per Integrierter Windows-Authentifizierung angemeldet wird, sieht die Formularseite gar nicht. Microsoft weist beim Parameter -SignInPageDescriptionText ausdrücklich darauf hin. Die gesamte Visitenkarte richtet sich also vor allem an externe Zugriffe über den Web Application Proxy, an Browser ohne Kerberos-Unterstützung und an alle, bei denen Kerberos klemmt. Wenn interne Kolleginnen und Kollegen plötzlich deine schöne neue Formularseite sehen, ist das kein Lob für dein Design, sondern ein Hinweis auf ein SPN- oder DNS-Problem – dazu mehr in [SPN, DNS und Kerberos für ADFS – warum die interne Anmeldung auf Formular zurückfällt](#).

Web-Theme kopieren, exportieren und anpassen

Das mitgelieferte Theme „default“ ist der Werkzustand. Microsofts eigene Kurzanleitungen setzen Logo und Illustration zwar gern direkt darauf, aber das ist ungefähr so, als würdest du die Originalurkunde mit Textmarker bearbeiten. Spätestens beim onload.js ist ohnehin Schluss: Dessen Inhalt im Standard-Theme lässt sich nicht ändern. Der saubere Weg führt immer über eine Kopie, die du exportierst, bearbeitest, wieder einspielst und dann aktivierst.



Skizze 2: Der Weg vom Standard-Theme zum eigenen Theme – mit eingebautem Notausgang

Kopie anlegen und exportieren

```
# 1. Kopie des Standard-Themes anlegen
New-AdfsWebTheme -Name "Contoso" -SourceName "default"

# 2. Kopie in einen Arbeitsordner exportieren
Export-AdfsWebTheme -Name "Contoso" -DirectoryPath "C:\AdfsTheme\Contoso"
```

Eigenes Theme als Kopie anlegen und zur Bearbeitung exportieren

Im Exportordner findest du unter anderem das Stylesheet unter css\style.css und das Skript unter script\onload.js. Lege den Ordner in deine Versionsverwaltung oder wenigstens auf eine Freigabe mit Änderungsprotokoll. Der Kollege, der das Theme in fünf Jahren anfassen muss, wird dir dankbar sein – oder es zumindest nicht verfluchen. Das ist im Admin-Leben fast dasselbe.

Logo und Illustration tauschen

Die Bildwechsel sind der dankbarste Teil der Übung: zwei Zeilen PowerShell, sofort sichtbar, und das Marketing hält dich für einen Zauberer. Halte dich dabei an die Maße, die Microsoft empfiehlt. Ein Logo mit 2.000 Pixeln Breite wird zwar skaliert, aber die Seite lädt über den Web Application Proxy jedes Mal ein Bild, das größer ist als manche Anwendung dahinter.

Bild	Empfohlenes Format	Dateigröße	Praxistipp
Logo	260 × 35 Pixel bei 96 dpi, PNG	höchstens 10 KB	Transparenter Hintergrund, Querformat. Hochformat-Logos wirken auf dem Streifen wie eine Briefmarke.
Illustration	1420 × 1080 Pixel bei 96 dpi, JPG oder PNG	200 bis 500 KB	Ruhiges Motiv ohne Text. Auf schmalen Bildschirmen wird die Illustration ausgeblendet, Botschaften darin gehen verloren.

Tabelle 2: Microsofts Empfehlungen für Logo und Illustration – und was sie in der Praxis bedeuten

```
# Bilder für alle Sprachen (invariante Locale "")
```

```
Set-AdfsWebTheme -TargetName "Contoso" `
-Logo @{Locale=""; Path="C:\AdfsTheme\Contoso\bilder\logo.png"} `
-Illustration @{Locale=""; Path="C:\AdfsTheme\Contoso\bilder\illustration.jpg"}
```

Logo und Illustration in das eigene Theme einspielen

TIPP – Kein Logo? Dann eben Text

Solange im aktiven Theme kein Logo hinterlegt ist, zeigt ADFS den Firmennamen aus Set-AdfsGlobalWebContent -CompanyName an. Für eine Testfarm oder eine Übergangszeit ist das völlig in Ordnung und deutlich würdevoller als ein unscharfes Logo, das jemand aus einer E-Mail-Signatur ausgeschnitten hat.

Stylesheet anpassen und Copyright ausblenden

Farben, Abstände und Schriften steuerst du über die style.css im Exportordner. Ein Klassiker, den Microsoft sogar selbst dokumentiert, ist das Ausblenden des Microsoft-Copyrights am Seitenende. Dazu suchst du in der Datei den Abschnitt #copyright und ergänzt display:none. Ändere am Stylesheet nur, was du wirklich brauchst, und kommentiere deine Stellen. Bei größeren Updates vergleichst du dann die Datei aus einem frischen Export des Standard-Themes mit deiner Fassung und siehst sofort, wo sich etwas verschoben hat.

```
/* in css\style.css des eigenen Themes */
#copyright {color:#696969; display:none;} /* Contoso: Copyright ausgeblendet */
```

Die einzige CSS-Zeile, die in fast jeder Farm auftaucht

```
Set-AdfsWebTheme -TargetName "Contoso" `
-StyleSheet @{Locale=""; Path="C:\AdfsTheme\Contoso\css\style.css"}
```

Geändertes Stylesheet zurück in das Theme spielen

Aktivieren – und wieder zurück

Solange das neue Theme nicht aktiv ist, sieht niemand etwas davon. Das ist dein Vorteil: Du kannst in Ruhe einspielen und erst dann umschalten, wenn alles beisammen ist. Die Aktivierung wirkt farmweit, weil Themes in der Konfigurationsdatenbank liegen. Bei einer Farm mit Windows Internal Database führst du die Set-Cmdlets auf dem primären Server aus; warum das so ist und was bei SQL Server anders läuft, steht in [WID oder SQL Server als ADFS-Konfigurationsdatenbank](#).

```
# Eigenes Theme aktivieren
Set-AdfsWebConfig -ActiveThemeName "Contoso"

# Notausgang: zurück zum Werkszustand
Set-AdfsWebConfig -ActiveThemeName "default"
```

Umschalten und Rückweg – beides eine Zeile

TIPP – Testen mit privatem Browserfenster

Browser cachen Bilder und Stylesheets der Anmeldeseite gern länger, als dir lieb ist. Teste nach jeder Änderung in einem privaten Fenster und von außen über den Web Application Proxy. Sonst suchst du eine halbe Stunde nach einem Fehler, den nur dein Browser-Cache kennt.

Texte und Links: Firmenname, Beschreibung, Hilfe und „Kennwort vergessen“

Der Web-Content ist der Teil, in dem die meisten Farmen am schlimmsten aussehen – nicht weil er schwer zu pflegen wäre, sondern weil ihn bei der Einrichtung niemand auf dem Zettel hatte. Alle

Texte und Links setzt du mit Set-AdfsGlobalWebContent, und zwar je Sprache über den Parameter -Locale. Wer -Locale weglässt, schreibt in die invariante Sprache – dazu gleich mehr, denn genau dort liegt die häufigste Stolperfalle.

Firmenname und Beschreibung

```
Set-AdfsGlobalWebContent -Locale "de" `
  -CompanyName "Contoso" `
  -OrganizationalNameDescriptionText "Zentrale Anmeldung für Contoso-Anwendungen" `
  -SignInPageDescriptionText "<p>Melde dich mit deiner Firmen-E-Mail-Adresse an, z. B.
vorname.nachname@contoso.de.</p>"
```

Firmenname, Unterzeile und Beschreibungstext für deutschsprachige Browser

Der Beschreibungstext darf HTML enthalten, Microsoft zeigt das in den eigenen Beispielen mit Absätzen und Links. Nutze das maßvoll: ein Absatz, vielleicht ein Link, fertig. Wer hier eine halbe Intranetseite mit Tabellen und Inline-Styles unterbringt, baut sich eine Wartungsbaustelle, die beim nächsten Theme-Wechsel garantiert schief aussieht. Ein sinnvoller Inhalt ist übrigens der Hinweis auf das erwartete Anmeldeformat. Damit sparst du dem Helpdesk einen guten Teil der Anrufe, in denen jemand „Mein Kennwort geht nicht“ sagt und eigentlich „Ich tippe meinen alten Domännennamen ein“ meint.

Startseite, Hilfe und Datenschutz

Unten auf der Seite gibt es drei Linkplätze: Startseite, Hilfe und Datenschutz. Jeder hat einen Parameter für das Ziel und einen für den Anzeigetext. Microsoft weist auf einen feinen Punkt hin: Den Anzeigetext musst du nur setzen, wenn du vom Standard abweichen willst. Der Standardtext ist für alle Browsersprachen übersetzt. Sobald du ihn überschreibst, gilt deine Fassung – und du bist ab dann selbst für jede Sprache zuständig.

```
Set-AdfsGlobalWebContent -Locale "de" `
  -HomeLink "https://intranet.contoso.de/" -HomeLinkText "Zum Intranet" `
  -HelpDeskLink "https://hilfe.contoso.de/anmeldung" -HelpDeskLinkText "Hilfe zur Anmeldung"
  -PrivacyLink "https://www.contoso.de/datenschutz" -PrivacyLinkText "Datenschutz"
```

Die drei Linkplätze am Seitenende für deutschsprachige Browser

WARNUNG – Interne Ziele für externe Besucher

Ein Hilfelink auf eine Adresse, die nur im internen Netz auflöst, ist für genau die Menschen nutzlos, die die Formularseite überhaupt sehen: externe Zugriffe. Prüfe jeden Link von außen, ohne VPN. Für Partner, die über einen Claims Provider Trust kommen, gilt das doppelt – Hintergründe in [Claims Provider Trust – Partnerorganisationen anbinden](#).

Der „Kennwort vergessen?“-Link

Einen eigenen Parameter für „Kennwort vergessen“ gibt es in ADFS nicht. Die einfachste und von Microsoft selbst gezeigte Lösung: Du baust den Link als HTML in den Beschreibungstext ein. Microsofts Beispiel verweist für die Microsoft-365-Relying-Party auf die Self-Service-Kennwortzurücksetzung von Entra ID; genauso gut kann dort euer eigenes Portal zur Kennwortrücksetzung stehen. Wenn der Link direkt unter dem Kennwortfeld sitzen soll statt im Beschreibungstext, landest du beim onload.js – siehe letztes Kapitel.

```
Set-AdfsGlobalWebContent -Locale "de" `
  -SignInPageDescriptionText ("<p>Melde dich mit deiner Firmen-E-Mail-Adresse an.</p>" +
  "<p><a href='https://kennwort.contoso.de/'>Kennwort vergessen?</a></p>")
```

Kennwort-vergessen-Link als HTML im Beschreibungstext

Verwechsle das nicht mit der ADFS-eigenen Seite zum Kennwortändern unter `/adfs/portal/updatepassword/`. Die ist für Menschen gedacht, die ihr altes Kennwort noch kennen, es aber ändern müssen – etwa weil es abgelaufen ist und kein Firmennetz in Reichweite ist. Den Endpunkt aktivierst du in der ADFS-Verwaltung unter „Endpunkte“, für externen Zugriff zusätzlich „Auf Proxy aktivieren“, danach ist ein Neustart des ADFS-Dienstes nötig.

	Kennwort vergessen	Kennwort abgelaufen oder ändern
Wer braucht es?	Jemand, der das Kennwort nicht mehr weiß	Jemand, der das alte Kennwort kennt
Was bietet ADFS?	Nichts Eigenes – du verlinkst ein Rücksetzungsportal	Endpunkt <code>/adfs/portal/updatepassword/</code>
Wie bindest du es ein?	HTML-Link im Beschreibungstext oder per <code>onload.js</code>	Endpunkt aktivieren, ggf. auf dem Proxy freigeben, Dienst neu starten
Text anpassen	<code>-SignInPageDescriptionText</code>	<code>-UpdatePasswordPageDescriptionText</code>

Tabelle 3: Zwei Kennwortszenarien, die gern in einen Topf geworfen werden

WARNUNG – Rücksetzungsportal ist ein Angriffsziel

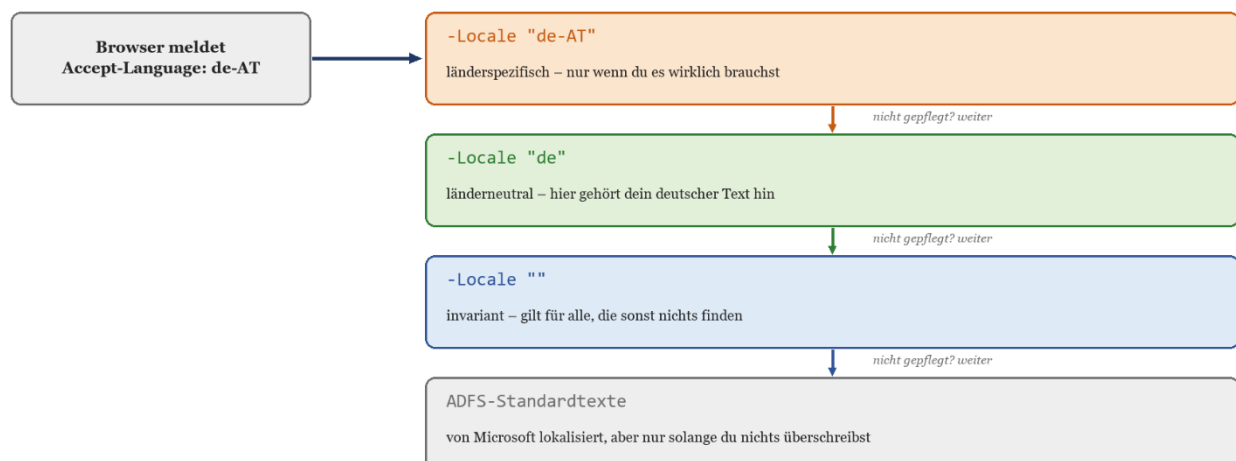
Ein Self-Service-Portal zur Kennworrücksetzung ist praktisch – und für Angreifer ebenso. Es gehört mit MFA und sauberer Identitätsprüfung abgesichert, sonst verlinkst du von deiner Anmeldeseite aus die Hintertür. Wie du die ADFS-Anmeldung selbst gegen Passwort-Spray schützt, zeigt [Extranet Smart Lockout – ADFS gegen Passwort-Spray schützen](#).

Texte je Sprache pflegen

ADFS liest die Sprache aus dem Browser und sucht dann von spezifisch nach allgemein: erst „de-AT“, dann „de“, dann die invariante Sprache, zuletzt die Standardtexte. Microsoft empfiehlt ausdrücklich, zuerst die länderneutrale Fassung zu pflegen und erst danach länderspezifische. Das Entscheidende ist aber ein anderer Satz aus der Dokumentation: Sobald du etwas anpasst, hat deine Anpassung Vorrang. Wer also einen deutschen Text in die invariante Sprache schreibt, beglückt damit auch die Kollegen in der Niederlassung in Lyon.

Skizze 3: Welcher Text erscheint? Die Sprachkette von ADFS

Der Browser schickt seine Sprache mit, ADFS sucht von spezifisch nach allgemein



Die klassische Falle: Du setzt `-HelpDeskLinkText "Hilfe"` nur für `""`. Ab sofort lesen auch Kollegen in Lyon und Leeds „Hilfe“ – denn deine Anpassung schlägt den mitgelieferten, übersetzten Standardtext.

Skizze 3: Die Sprachkette – von der Browsersprache bis zum mitgelieferten Standardtext

```
# Deutsch
Set-AdfsGlobalWebContent -Locale "de" -HelpDeskLinkText "Hilfe zur Anmeldung" `
    -SignInPageDescriptionText "<p>Melde dich mit deiner Firmen-E-Mail-Adresse an.</p>"

# Englisch
Set-AdfsGlobalWebContent -Locale "en" -HelpDeskLinkText "Sign-in help" `
    -SignInPageDescriptionText "<p>Sign in with your company e-mail address.</p>"

# Invariant: neutraler Fallback für alle übrigen Sprachen
Set-AdfsGlobalWebContent -Locale "" -HelpDeskLink "https://hilfe.contoso.de/anmeldung"
```

Texte je Sprache – der Link selbst kann sprachneutral bleiben, die Beschriftung nicht

Strategie	Wann sinnvoll	Risiko
Nur „de“ pflegen	Rein deutschsprachige Belegschaft	Englische Browser sehen Standardtexte, deine Links aber nur, wenn sie zusätzlich in „de“ stehen.
„de“ und „en“ pflegen, Links in „de“	Gemischte Belegschaft, internationale Partner	Gering – der Standard für die meisten Firmen
Alles in „de“ auf Deutsch	Eigentlich nie	Alle anderen Sprachen sehen deutschen Text, auch Partner im Ausland
Zusätzlich „de-AT“, „de-CH“	Unterschiedliche Hotlines oder Rechtstexte je Land	Pflegeaufwand – jede Änderung mehrfach

Tabelle 4: Sprachstrategien für den Web-Content

TIPP – Umlaute in PowerShell-Skripten

Windows PowerShell 5.1 liest Skriptdateien ohne Byte Order Mark als ANSI. Aus „Hilfe zur Anmeldung für Bürokräfte“ wird dann auf der Anmeldeseite eine Buchstabensuppe. Speichere deine Anpassungsskripte als UTF-8 mit BOM. Microsoft selbst empfiehlt für Unicode-Eingaben die PowerShell ISE – die speichert von Haus aus passend.

Fehlerseiten nicht vergessen

Die Fehlerseite ist die Seite, die Menschen in ihrer schlechtesten Laune sehen. Trotzdem steht dort in vielen Farmen nur die nackte Activity-ID. Mit -ErrorPageDescriptionText, -ErrorPageGenericErrorMessage, -ErrorPageAuthorizationErrorMessage und -ErrorPageSupportEmail gibst du ihr einen Ausweg: wer zuständig ist, wie man ihn erreicht und dass die Activity-ID bitte mitgeschickt werden soll. Gerade der Hinweis auf die Activity-ID spart dir später Zeit, wenn du die passenden Einträge im Ereignisprotokoll suchst. Welche Fehlerbilder hinter solchen Seiten typischerweise stecken, zeigt [ADFS-Anmeldung schlägt fehl – die häufigsten Ursachen und ihre Lösung](#).

```
Set-AdfsGlobalWebContent -Locale "de" `
    -ErrorPageDescriptionText "<p>Bei der Anmeldung ist ein Fehler aufgetreten. Bitte schick die angezeigte Activity-ID an den Support.</p>" `
    -ErrorPageAuthorizationErrorMessage "Du hast für diese Anwendung keine Berechtigung. Bitte wende dich an die Fachabteilung." `
    -ErrorPageSupportEmail "support@contoso.de"
```

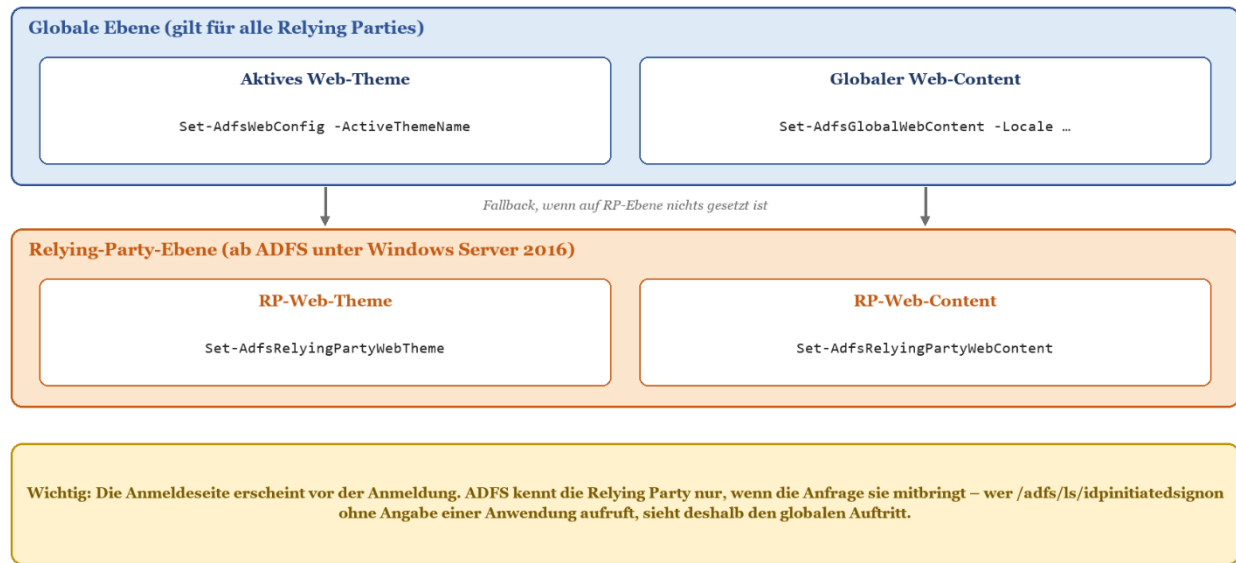
Fehlerseite mit Ansprechpartner statt Ratlosigkeit

Relying-Party-spezifische Anpassungen

Manchmal reicht ein Auftritt für alle nicht. Die Lohnabrechnung soll auf die Lohnbuchhaltung als Ansprechpartner verweisen, das Partnerportal braucht einen englischen Hinweis und das Logo der Tochtergesellschaft, und für Microsoft 365 soll der Kennwort-Link auf die Entra-Self-Service-Rücksetzung zeigen. Seit ADFS unter Windows Server 2016 kannst du dafür sowohl Texte als auch komplette Themes an eine einzelne Relying Party hängen. Was dort gesetzt ist, gewinnt; was fehlt, kommt weiter von der globalen Ebene.

Skizze 4: Zwei Ebenen – farmweit und pro Anwendung

Was an der Relying Party hängt, gewinnt. Was dort fehlt, kommt von der globalen Ebene



Skizze 4: Globale Ebene und Relying-Party-Ebene – Vorrang und Fallback

Eigene Texte pro Anwendung

```
Set-AdfsRelyingPartyWebContent -TargetRelyingPartyName "Microsoft Office 365 Identity Platform"
-Locale "de" `
  -SignInPageDescriptionText ("<p>Anmeldung für Microsoft 365.</p>" +
    "<p><a href='https://sspr.contoso.de/'>Kennwort vergessen?</a></p>")

Set-AdfsRelyingPartyWebContent -TargetRelyingPartyName "Lohnportal" -Locale "de" `
  -HelpDeskLink "https://hilfe.contoso.de/lohn" -HelpDeskLinkText "Fragen zur Abrechnung" `
  -ErrorPageAuthorizationErrorMessage "Zugriff nur für Personen mit Abrechnungsrolle."
```

Abweichende Texte und Links für zwei Relying Parties – sspr.contoso.de steht als Platzhalter für die Self-Service-Rücksetzung

Der Name hinter -TargetRelyingPartyName ist der Anzeigename des Relying Party Trusts, so wie ihn Get-AdfsRelyingPartyTrust ausgibt. Bei Microsoft 365 heißt er in nahezu jeder Farm „Microsoft Office 365 Identity Platform“, weil die Einrichtung ihn so anlegt. Bei eigenen Anwendungen lohnt sich ein kurzer Blick, bevor du dich über einen Tippfehler wunderst. Wie Relying Party Trusts angelegt und benannt werden, steht in [Relying Party Trust anlegen – per Metadaten und von Hand](#).

Eigenes Theme pro Anwendung

```
# Theme für das Partnerportal: Kopie, eigenes Logo, Zuweisung
New-AdfsWebTheme -Name "Partnerportal" -SourceName "Contoso"
Set-AdfsWebTheme -TargetName "Partnerportal" `
  -Logo @{Locale=""; Path="C:\AdfsTheme\Partner\logo-tochter.png"}
Set-AdfsRelyingPartyWebTheme -TargetRelyingPartyName "Partnerportal" -SourceWebThemeName
"Partnerportal"

# Prüfen, welche Relying Parties ein eigenes Theme haben
Get-AdfsRelyingPartyWebTheme | Select-Object Name
```

Ein eigenes Theme für eine einzelne Relying Party – als Kopie deines globalen Themes

Beachte, dass du mit -SourceWebThemeName die Inhalte eines Themes an die Relying Party überträgst. Plane nach jeder Änderung am Quell-Theme eine erneute Zuweisung ein und prüfe das Ergebnis im Browser, statt dich darauf zu verlassen, dass sich die Änderung von selbst durchreicht. Du kannst auch Logo, Illustration, Stylesheet oder onload.js direkt an Set-

AdfsRelyingPartyWebTheme übergeben, ohne ein separates Theme anzulegen. Für die Nachvollziehbarkeit ist ein benanntes Theme aber deutlich angenehmer.

Wieder zurück auf den globalen Auftritt

```
Remove-AdfsRelyingPartyWebTheme -TargetRelyingPartyName "Partnerportal"
Remove-AdfsRelyingPartyWebContent -TargetRelyingPartyName "Lohnportal" -Locale "de"
```

Anwendungsspezifische Anpassungen entfernen – danach greift wieder die globale Ebene

Szenario	RP-spezifisch sinnvoll?	Begründung
Eigene Hotline für eine Fachanwendung	ja, per Web-Content	Ein Link, ein Text – geringer Pflegeaufwand
Kennwort-Link für Microsoft 365 auf Entra SSPR	ja, per Web-Content	Gilt nur für Cloud-Identitäten, nicht für alle Anwendungen
Tochtergesellschaft mit eigenem Logo	ja, per Theme	Corporate Design je Gesellschaft
Jede Anwendung bekommt ihre eigene Farbe	nein	Verwirrt Nutzer und erschwert die Phishing-Erkennung
Eigenes onload.js je Anwendung	nur mit gutem Grund	Jedes Skript ist eine weitere Baustelle bei Updates

Tabelle 5: Wann sich anwendungsspezifische Anpassungen lohnen

WICHTIG – Einheitlichkeit ist ein Sicherheitsmerkmal

Eine Anmeldeseite, die überall gleich aussieht, unter derselben Adresse wie sts.contoso.de, trainiert deine Belegschaft darauf, Abweichungen zu bemerken. Zehn verschiedene Auftritte trainieren das Gegenteil – und machen es Phishing-Seiten leicht. Mehr zu Angriffen auf ADFS findest du in [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#).

Wenn du ohnehin gerade durch alle Relying Parties gehst, um zu entscheiden, welche einen eigenen Auftritt braucht, ist das die halbe Bestandsaufnahme für eine spätere Migration. Wie du die gleich vollständig machst, zeigt [Relying Parties inventarisieren – die Bestandsaufnahme vor der Entra-Migration](#).

JavaScript im onload.js: mächtig, aber mit Nebenwirkungen

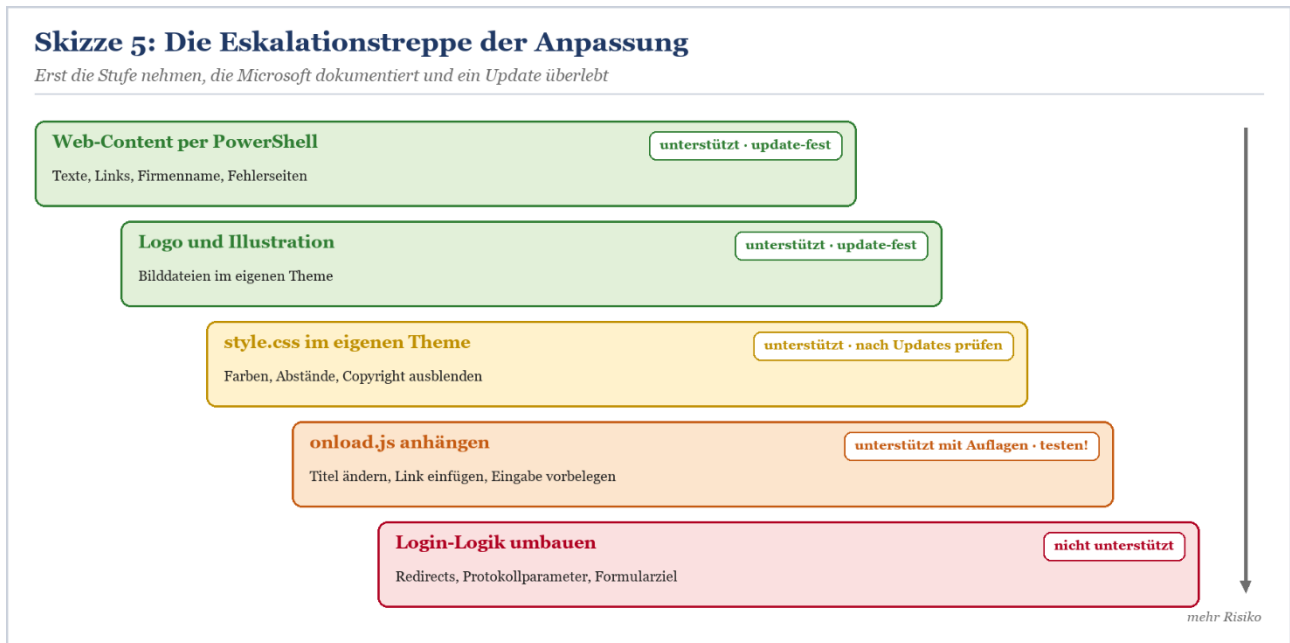
Alles, was die Cmdlets nicht hergeben, landet früher oder später im onload.js. Das Skript wird auf allen ADFS-Seiten ausgeführt: Anmeldeformular, Home Realm Discovery, MFA-Auswahl, Fehlerseite. Microsoft unterstützt solche Anpassungen ausdrücklich, aber mit Auflagen – und die Auflagen sind der Grund, warum dieses Kapitel mit einem roten Kasten beginnt.

WARNUNG – JavaScript-Anpassungen – bitte erst lesen, dann tippen

Microsoft unterstützt keine Anpassungen, die Weiterleitungen oder Protokollparameter verändern. Wer am Formularziel, an wctx, RelayState oder an den Redirects dreht, steht beim Support allein da. Das onload.js läuft auf jeder ADFS-Seite. Prüfe vor jedem Zugriff, ob das HTML-Element existiert, sonst wirft dein Skript auf der MFA-Seite Fehler, die du nur auf der Anmeldeseite getestet hast. Hänge deinen Code ans Ende des vorhandenen onload.js an. Der Originalinhalt kümmert sich um die Darstellung auf verschiedenen Geräten und soll unverändert bleiben. Lade keine externen Skripte, Schriften oder Tracker nach. Auf der Seite, auf der deine Kennwörter eingetippt werden, hat fremder Code nichts verloren – das ist kein Stilrat, sondern Angriffsfläche. Teste in einer Labor- oder Testfarm und nach jedem größeren Update erneut. Element-IDs sind keine zugesicherte Schnittstelle.

Skizze 5: Die Eskalationstreppe der Anpassung

Erst die Stufe nehmen, die Microsoft dokumentiert und ein Update überlebt



Skizze 5: Die Eskalationstreppe – je weiter unten, desto mehr Risiko und Pflegeaufwand

Ein sauberes Beispiel: Überschrift und Kennwort-Link

Das folgende Beispiel ändert die Überschrift über den Eingabefeldern – die lässt sich anders nicht anpassen – und setzt einen „Kennwort vergessen?“-Link direkt unter das Kennwortfeld. Es nutzt die Element-ID loginMessage und das Login-Objekt, die Microsoft selbst in seinen Beispielen verwendet, und prüft beides vor dem Zugriff.

```
// --- Contoso-Anpassungen, angehängt am Ende von onload.js ---

// Überschrift über den Eingabefeldern
var loginMessage = document.getElementById('loginMessage');
if (loginMessage) {
    loginMessage.innerHTML = 'Mit dem Contoso-Konto anmelden';
}

// "Kennwort vergessen?" unter dem Kennwortfeld
if (typeof Login !== 'undefined') {
    var pw = document.getElementById(Login.passwordInput);
    if (pw && !document.getElementById('contosoForgotPw')) {
        var a = document.createElement('a');
        a.id = 'contosoForgotPw';
        a.href = 'https://kennwort.contoso.de/';
        a.textContent = 'Kennwort vergessen?';
        a.style.display = 'block';
        a.style.marginTop = '8px';
        pw.parentNode.insertBefore(a, pw.nextSibling);
    }
}
```

onload.js-Erweiterung mit Existenzprüfung – Adressen sind Beispieldomänen

Beachte, dass dieser Text nicht über -Locale übersetzt wird. Wer mehrsprachig arbeitet, muss im Skript selbst die Browsersprache auswerten – ein weiterer Grund, den Kennwort-Link lieber über den sprachfähigen Beschreibungstext zu lösen, wenn die Position auf der Seite verhandelbar ist.

Skript einspielen

```
# ADFS ab Windows Server 2016
Set-AdfsWebTheme -TargetName "Contoso" -OnLoadScriptPath
```

```
"C:\AdfsTheme\Contoso\script\onload.js"
```

```
# So stand es früher für ADFS unter Windows Server 2012 R2 in jeder Anleitung
Set-AdfsWebTheme -TargetName "Contoso" `
  -AdditionalFileResource @{Uri="/adfs/portal/script/onload.js";
  Path="C:\AdfsTheme\Contoso\script\onload.js"}
```

```
Set-AdfsWebConfig -ActiveThemeName "Contoso"
```

onload.js in das eigene Theme laden und aktivieren

Nach Updates prüfen

Kumulative Updates und ein Anheben des Farm Behavior Level können das Markup der Anmeldeseite verändern. Plane deshalb nach jedem größeren Patch-Termin einen Kurztest von außen ein: Anmeldeseite, MFA-Seite, Fehlerseite, Kennwortändern. Wenn du eine Farm-Aktualisierung vor dir hast, gehört dieser Test in die Checkliste – den Rest der Vorbereitung beschreibt [ADFS-Farm upgraden – Farm Behavior Level anheben ohne Ausfall](#). Und wer einen MFA-Adapter einsetzt, sollte auch dessen Seiten im neuen Auftritt durchklicken; welche Besonderheiten die Entra-MFA-Anbindung mitbringt, steht in [ADFS mit Entra-MFA koppeln – Adapter, Zertifikat und Fallstricke](#).

Test	Woran du einen Fehler erkennst	Typische Ursache
Anmeldeseite von außen	Altes Logo, fehlende Texte	Browser-Cache oder falsches Theme aktiv
Englischer Browser	Deutsche Texte oder Standardtexte statt deiner englischen	Text nur in „“ oder nur in „de“ gepflegt
MFA-Auswahlseite	Skriptfehler in der Konsole, verschobenes Layout	onload.js ohne Existenzprüfung
Fehlerseite	Leere Fläche statt Ansprechpartner	Error-Texte nur global, RP hat eigene leere Werte
Schmales Display	Abgeschnittenes Logo	Logo zu breit oder im Hochformat

Tabelle 6: Kurztest nach Änderungen und Updates

Fazit: Zehn Minuten Pflege für tausend Anmeldungen am Tag

Die ADFS-Anmeldeseite anzupassen ist kein Großprojekt, sondern Handwerk. Theme kopieren, exportieren, Logo und Illustration tauschen, Texte und Links je Sprache setzen, aktivieren – das schaffst du an einem Vormittag, inklusive Kaffee und Rückfrage beim Marketing, welches der drei Logos nun das richtige ist. Mit den Cmdlets für Relying Parties gibst du einzelnen Anwendungen einen eigenen Hilfelink oder ein eigenes Logo, ohne den Gesamtauftritt zu zerfasern. Und JavaScript bleibt das, was es sein sollte: die letzte Stufe, sauber angehängt, mit Existenzprüfung und ohne Fremdcode.

Eine Farm, die seit über zehn Jahren zuverlässig läuft, hat eine gepflegte Visitenkarte verdient – auch dann, wenn mittelfristig der Umzug nach Entra ID ansteht. Das, was du hier ordnest, brauchst du dort wieder: Logo, Hintergrundbild, Hinweistexte und den Kennwort-Link findest du in Entra ID als Unternehmensbranding wieder. Mehr dazu auf der Seite [Microsoft Entra ID](#) und im Beitrag [Microsoft 365 von Federated auf Managed umstellen – mit Staged Rollout](#). Wenn du Anmeldeseite, Themes und Texte gemeinsam mit uns aufräumen willst, unterstützen wir dich im [Consulting zu ADFS \(Active Directory Federation Services\)](#). Wer das lieber selbst im Labor ausprobiert, ist in den [ADFS-Schulungen](#) richtig. Die Anpassung der ADFS-Seiten und vieles mehr rund um den Betrieb der Farm vertieft außerdem das Buch – mehr dazu unter [ADFS in der Praxis – das Buch im Detail](#).

WEITER – Passend zum Thema

- › [SPN, DNS und Kerberos für ADFS – warum die interne Anmeldung auf Formular zurückfällt](#) – wenn interne Nutzer plötzlich die Formularseite sehen
- › [ADFS-Anmeldung schlägt fehl – die häufigsten Ursachen und ihre Lösung](#) – was hinter der Fehlerseite steckt
- › [Relying Party Trust anlegen – per Metadaten und von Hand](#) – Anzeigenamen und Identifier deiner Anwendungen
- › [ADFS-Farm upgraden – Farm Behavior Level anheben ohne Ausfall](#) – Anmeldeseite in die Upgrade-Checkliste aufnehmen
- › [Web Application Proxy: Architektur und Ablöse](#) – der Weg, auf dem externe Besucher deine Seite sehen
- › [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#) – warum fremdes JavaScript auf der Anmeldeseite tabu ist

FAQ: ADFS-Anmeldeseite anpassen

Wie ändere ich das Logo auf der ADFS-Anmeldeseite?

Lege mit New-AdfsWebTheme -Name Contoso -SourceName default eine Kopie des Standard-Themes an, spiele das Logo mit Set-AdfsWebTheme -TargetName Contoso -Logo @{Locale=""; Path="C:\...\logo.png"} ein und aktiviere das Theme mit Set-AdfsWebConfig -ActiveThemeName Contoso. Microsoft empfiehlt 260 × 35 Pixel bei höchstens 10 KB.

Kann ich das Standard-Theme „default“ direkt ändern?

Logo und Illustration lassen sich laut Microsofts Kurzanleitungen auch am Standard-Theme setzen, das onload.js dagegen nicht. Sauberer ist immer eine Kopie: Du hast einen unveränderten Rückweg und kannst jederzeit mit einer Zeile zum Werkszustand zurückschalten.

Muss ich die Anpassung auf jedem ADFS-Server und auf dem WAP wiederholen?

Nein. Themes und Web-Content liegen in der Konfigurationsdatenbank und gelten für die ganze Farm. Bei Windows Internal Database führst du die Cmdlets auf dem primären Server aus. Der Web Application Proxy reicht die Seiten nur durch.

Wie füge ich einen „Kennwort vergessen?“-Link hinzu?

Am einfachsten als HTML-Link im Parameter -SignInPageDescriptionText von Set-AdfsGlobalWebContent oder – nur für eine Anwendung – von Set-AdfsRelyingPartyWebContent. Soll der Link direkt unter dem Kennwortfeld stehen, brauchst du eine Erweiterung im onload.js eines eigenen Themes.

Warum sehen englischsprachige Nutzer meine deutschen Texte?

Weil die Texte in der invarianten Sprache „gesetzt sind. Anpassungen haben Vorrang vor den übersetzten Standardtexten. Pflege deutsche Texte mit -Locale "de", englische mit -Locale "en" und lege in „ nur sprachneutrale Werte wie Linkziele ab.

Warum sehen interne Nutzer meine neue Anmeldeseite nicht?

Weil sie per Integrierter Windows-Authentifizierung angemeldet werden und die Formularseite gar nicht erscheint. Das ist der gewünschte Zustand. Sehen sie die Formularseite doch, solltest du SPN, DNS und Browser-Einstellungen prüfen.

Kann jede Relying Party eine eigene Anmeldeseite bekommen?

Ja, ab ADFS unter Windows Server 2016. Texte setzt du mit Set-AdfsRelyingPartyWebContent, Logo, Illustration, Stylesheet und onload.js mit Set-AdfsRelyingPartyWebTheme. Was du dort nicht setzt, kommt von der globalen Ebene.

Wie entferne ich das Microsoft-Copyright?

Im exportierten Theme die style.css öffnen, beim Abschnitt #copyright display:none ergänzen, die Datei mit Set-AdfsWebTheme -StyleSheet einspielen und das Theme aktivieren. Microsoft dokumentiert genau dieses Vorgehen.

Ist JavaScript im onload.js von Microsoft unterstützt?

Ja, mit Einschränkungen. Anpassungen, die Weiterleitungen oder Protokollparameter verändern, werden nicht unterstützt. Eigener Code gehört ans Ende des vorhandenen onload.js, muss die Existenz von Elementen prüfen und sollte vorher in einer Testumgebung laufen.

Meine Umlaute erscheinen auf der Anmeldeseite als Zeichensalat. Was tun?

Speichere dein PowerShell-Skript als UTF-8 mit BOM oder arbeite in der PowerShell ISE. Windows PowerShell 5.1 liest Skripte ohne BOM als ANSI und schreibt die falsch interpretierten Zeichen dann brav in die Konfiguration.