

Passwort-Spray und Brute Force auf ADFS erkennen

ADFS · Sicherheitsereignisse lesen

Viele Konten, wenige Versuche, verteilte IPs – so sieht ein Angriff aus, den keiner bemerkt

Passwort-Spray und Brute Force in den ADFS-Logs, in Entra ID und in Connect Health aufspüren – mit PowerShell statt Bauchgefühl.



ADFS

203.0.113.x · 198.51.100.x · ...

boddenberg.de

WISSEN

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

› [Active Directory Federation Services](#)

BERATUNG

Dein Sicherheitsprotokoll ist voller 1203er und keiner weiß, ob einer davon gepasst hat? Wir schauen mit.

› [Consulting zu ADFS](#)

SCHULUNG

Spray simulieren, Logs auswerten, Gegenmaßnahmen üben – im Labor statt mitten im Vorfall.

› [ADFS-Schulungen](#)

Der klassische Brute-Force-Angriff ist der Dorfpolizist unter den Angriffen: laut, plump, leicht zu fassen. Einer hämmert hundertmal auf dasselbe Konto ein, das Konto wird gesperrt, der Helpdesk bekommt einen Anruf, alle wissen Bescheid. Passwort-Spray ist das Gegenteil. Da probiert jemand „Herbst2026!“ an 3.000 Konten – einmal pro Konto, verteilt auf ein paar Hundert IP-Adressen, schön gleichmäßig über die Nacht. Kein Konto wird gesperrt, kein Alarm geht los, und am Montagmorgen meldet sich ein Benutzer aus der Buchhaltung, dessen Postfach seit Samstag Rechnungen mit geänderter Bankverbindung verschickt. Spätestens dann interessiert sich plötzlich jeder für die Sicherheitsereignisse deiner ADFS-Farm.

Dieser Beitrag beantwortet die Frage, die Admins in genau diesem Moment googeln: Wie erkenne ich Passwort-Spray auf ADFS? Du lernst, wie die drei typischen Angriffsmuster in den Protokollen aussehen, welche Ereignisse du brauchst und wie du sie einschaltest, wie du mit PowerShell aus Tausenden Ereignissen in wenigen Minuten einen Befund machst und was Entra ID, Entra ID Protection und Entra Connect Health dazu beitragen – und was nicht. Die Gegenmaßnahme Extranet Smart Lockout behandelt ein eigener Beitrag, den Rundumblick auf Härtung, MFA und Golden SAML liefert [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#). Die Einordnung von ADFS insgesamt findest du auf der Übersichtsseite [Active Directory Federation Services](#).

FAKTEN — Passwort-Spray auf ADFS in Kürze

Jeder fehlgeschlagene Kennwortversuch erzeugt bei eingeschaltetem Auditing ab ADFS in Windows Server 2016 das Ereignis 1203 (Fresh Credential Validation Error) im Sicherheitsprotokoll, Quelle „AD FS Auditing“. Die erfolgreiche Prüfung ist das Ereignis 1202.

Diese beiden Ereignisse gehören schon zur Auditstufe „Basic“, die in ADFS ab Windows Server 2016 Standard ist. Das Sicherheitsauditing selbst ist dagegen ab Werk ausgeschaltet.

Bei föderierten Domänen prüft ADFS das Kennwort. Fehlversuche landen deshalb nicht in den Anmeldeprotokollen von Entra ID – es sei denn, Entra Connect Health trägt sie dorthin.

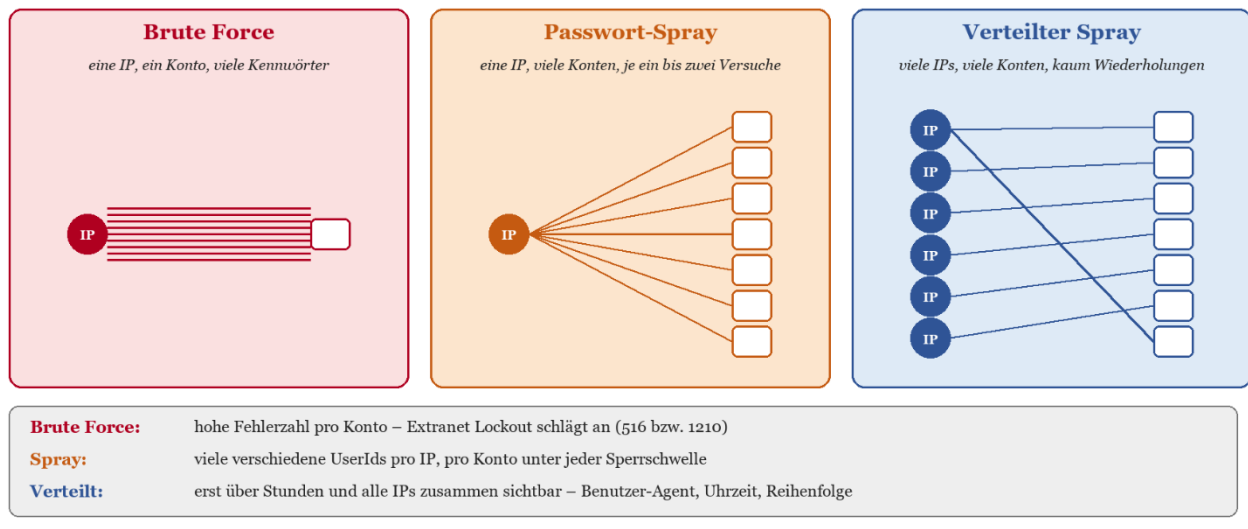
Für Kontosperrungen durch Extranet Lockout bzw. Extranet Smart Lockout dokumentiert Microsoft die Ereignisse 516 und 1210.

Wie Spray und Brute Force im Log aussehen

Bevor du Skripte baust, lohnt ein Blick auf die Form des Angriffs. Alle Varianten haben dasselbe Ziel – ein gültiges Kennwort zu finden –, hinterlassen aber völlig unterschiedliche Spuren. Wer nur nach gesperrten Konten sucht, findet den Brute-Force-Angriff und übersieht die beiden anderen. Das ist ungefähr so, als würdest du Einbrecher nur dann bemerken, wenn sie die Tür eintreten.

Skizze 1: Drei Angriffsmuster, drei Fingerabdrücke im Log

Links die Angreifer-IPs, rechts die Benutzerkonten – die Form der Linien verrät das Muster



Skizze 1: Brute Force, Passwort-Spray und verteilter Spray unterscheiden sich in der Verteilung von IPs und Konten.

Muster	Was der Angreifer tut	Fingerabdruck im ADFS-Log	Wer es bemerkt
Brute Force	ein Konto, viele Kennwörter, meist eine oder wenige IPs	viele 1203 für dieselbe UserId; Sperrereignis 516 bzw. 1210	Extranet Lockout, Helpdesk, jeder
Passwort-Spray	ein bis drei gängige Kennwörter gegen sehr viele Konten	eine IP mit Dutzenden bis Hunderten verschiedener UserIds, je ein bis zwei Versuche	Auswertung pro IP, Risky-IP-Bericht
Verteilter Spray („low and slow“)	wie Spray, aber über viele IPs und Stunden verteilt	pro IP und pro Konto unauffällig; auffällig sind Takt, Uhrzeit, Benutzer-Agent und Reihenfolge der Konten	nur die Auswertung über alle IPs und die Zeit
Credential Stuffing	geleakte Paare aus Benutzername und Kennwort, je Konto ein Versuch	wenige Fehlschläge, dafür Erfolge (1202) von fremden IPs	Abgleich der Erfolge mit verdächtigen IPs

Tabelle 1: Angriffsmuster und ihre Spuren

Die Merkmale, auf die es ankommt

Ein Spray ist im Kern eine Statistikfrage. Kein einzelnes Ereignis ist verdächtig – ein Tippfehler sieht genauso aus wie ein Angriffsversuch. Verdächtig wird erst die Verteilung. Die folgenden Merkmale haben sich in der Praxis bewährt, und für jedes gibt es weiter unten ein PowerShell-Beispiel:

- Viele verschiedene Konten pro IP: Eine Büro-IP mit 40 Tippfehlern von 40 verschiedenen Kollegen an einem Montagmorgen ist denkbar. Eine fremde IP mit 300 Konten in einer Nacht nicht.
- Wenige Versuche pro Konto: Ein Spray bleibt bewusst unter jeder Sperrschwelle. Ein Verhältnis von etwa einem Versuch pro Konto ist das Markenzeichen.
- Konten, die es nicht gibt: Angreifer raten Benutzernamen aus Namenslisten, alten Leaks und dem Format deiner E-Mail-Adressen. Ausgeschiedene Kolleginnen und Kollegen, Tippvarianten und Allerweltsnamen tauchen auf.
- Reihenfolge: Konten in alphabetischer Folge oder in der Reihenfolge des Adressbuchs sind kein Zufall.
- Takt und Uhrzeit: Automatisierte Angriffe arbeiten gleichmäßig. Menschen tippen sich morgens um acht vertan, nicht jede Nacht von zwei bis fünf alle 90 Sekunden.
- Benutzer-Agent und Endpunkt: Ein exotischer oder veralteter User-Agent, der in deinem Normalbetrieb nicht vorkommt, oder Anfragen an WS-Trust-Endpunkte für Benutzername und Kennwort, die sonst kaum jemand nutzt.

WICHTIG – Ohne echte Client-IP ist jede Auswertung wertlos

Alle Auswertungen pro IP stehen und fallen damit, dass ADFS die tatsächliche Adresse des Clients sieht. Der Web Application Proxy reicht sie im Header x-ms-forwarded-client-ip weiter. Steht vor dem Proxy ein Load Balancer, der per Source-NAT die eigene Adresse einsetzt, kommt bei ADFS für jeden Versuch dieselbe interne IP an – und deine schöne Spray-Erkennung meldet den Load Balancer als Angreifer.

Microsoft weist im Risky-IP-Bericht von Connect Health ausdrücklich darauf hin: Tauchen dort private Adressen oder die Adresse des Load Balancers auf, gibt der Load Balancer die Client-IP nicht weiter.

› [ADFS hinter dem Load Balancer – Health Probe, SNI und Persistenz](#)

Was in Ereignis 1203 steht

Seit ADFS in Windows Server 2016 ist das Auditing deutlich aufgeräumt: Statt eines Dutzends Ereignisse pro Anfrage gibt es in der Stufe „Basic“ höchstens fünf. Für die Spray-Erkennung zählen vor allem zwei: 1203 für die gescheiterte und 1202 für die erfolgreiche Prüfung frischer Anmeldedaten. Beide decken WS-Trust, WS-Federation, SAML-P und den OAuth-Autorisierungsendpunkt ab. Die Nutzdaten stecken als XML im zweiten Parameter des Ereignisses. Darin findest du unter anderem die eingegebene Benutzerkennung (UserId), die IP-Adressen (IpAddress), den angesprochenen Endpunkt und den User-Agent.

Eine Besonderheit hat das Feld IpAddress: Es kann mehrere Adressen enthalten, durch Komma getrennt. Microsoft dokumentiert die Reihenfolge x-ms-forwarded-client-ip, x-forwarded-for, x-ms-proxy-client-ip. Für die Auswertung ist also in aller Regel die erste Adresse die interessante. Die UserId ist übrigens das, was jemand eingetippt hat – nicht das, was es im Verzeichnis gibt. Das ist kein Fehler, sondern ein Geschenk: Genau daran erkennst du geratene Konten.

Ereignis	Protokoll	Bedeutung für die Erkennung
1203	Sicherheit, AD FS Auditing	Kennwort oder Benutzername falsch – das Rohmaterial jeder Spray-Analyse
1202	Sicherheit, AD FS Auditing	Anmeldedaten erfolgreich geprüft – der Gegencheck auf Treffer
516	Sicherheit	Konto wegen zu vieler Fehlversuche im Extranet gesperrt (mit Client-IP und Fehlerzahl)
1210	Sicherheit	Sperre durch Extranet Smart Lockout
515	Sicherheit	Konto war gesperrt, und es wurde das richtige Kennwort

		eingetragen – möglicherweise kompromittiert
411, 342	Sicherheit bzw. Admin	aus ADFS unter Windows Server 2012 R2 bekannt: fehlgeschlagene Authentifizierung; 411 in neueren Versionen nur in der Stufe „Verbose“

Tabelle 2: Die relevanten Ereignisse auf den ADFS-Servern

WARNUNG – Ereignis 515 ist kein Rauschen

Wenn ADFS meldet, dass für ein gesperrtes Konto das korrekte Kennwort eingegeben wurde, hat entweder der Benutzer selbst nach fünf Fehlversuchen endlich richtig getippt – oder jemand anderes kennt das Kennwort. Prüfe in jedem Fall die IP. Kommt sie nicht aus deinem Netz und nicht vom Benutzer, behandle das als Vorfall.

Voraussetzungen: Auditing einschalten und Spuren sichern

Der häufigste Befund in Sicherheitsvorfällen rund um ADFS ist nicht „wir haben den Angriff übersehen“, sondern „wir hätten ihn sehen können, wenn das Auditing an gewesen wäre“. Die Auditstufe „Basic“ ist zwar ab Werk gesetzt, aber ADFS darf ohne passende lokale Richtlinie gar nicht ins Sicherheitsprotokoll schreiben. Drei Dinge müssen zusammenkommen, und zwar auf jedem ADFS-Server der Farm. Auf den Proxys ist das nicht nötig.

Schritt	Wo	Was
1	Lokale Sicherheitsrichtlinie oder GPO	Das ADFS-Dienstkonto (bei gMSA das Konto mit \$ am Ende) hat das Benutzerrecht „Generieren von Sicherheitsüberwachungen“.
2	auditpol	Unterkategorie „Application Generated“ für Erfolg und Fehler aktiviert.
3	ADFS-Eigenschaften	Erfolgs- und Fehlerüberwachung in den Federation-Service-Eigenschaften (Reiter „Ereignisse“) eingeschaltet, Auditstufe mindestens „Basic“ – für Connect Health „Verbose“.

Tabelle 3: Damit 1202 und 1203 überhaupt geschrieben werden

```
# Auditstufe und Ereignistypen der Farm prüfen
Get-AdfsProperties | Select-Object AuditLevel, LogLevel

# Erfolgs- und Fehlerüberwachung sicherstellen (LogLevel enthält die Liste)
Set-AdfsProperties -LogLevel Errors, Warnings, Information, SuccessAudits, FailureAudits

# Für Connect Health und ausführliche Forensik
Set-AdfsProperties -AuditLevel Verbose

# Auf JEDEM ADFS-Server: Unterkategorie prüfen bzw. einschalten
auditpol.exe /get /subcategory:"Application Generated"
auditpol.exe /set /subcategory:"Application Generated" /success:enable /failure:enable
```

Listing 1: Auditing prüfen und einschalten

TIPP – Lass das Sicherheitsprotokoll nicht verhungern

Mit eingeschaltetem Auditing füllt sich das Sicherheitsprotokoll eines viel genutzten ADFS-Servers schnell. Ist es zu klein, überschreibt Windows die Ereignisse von Freitagabend, bevor du am Montag nachsiehst. Vergrößere das Protokoll per Gruppenrichtlinie deutlich – oder, besser, leite die Ereignisse an ein SIEM oder per Connect Health nach Entra ID weiter. Ein Angriff, dessen Spuren schon überschrieben sind, ist für die Analyse nicht passiert.

Wie du Admin-Log, Debug-Tracing und Auditing generell liest, beschreibt [ADFS-Ereignisprotokolle lesen – Admin-Log, Debug-Tracing und Auditing](#).

WARNUNG — Gruppenrichtlinien, die heimlich zurückdrehen

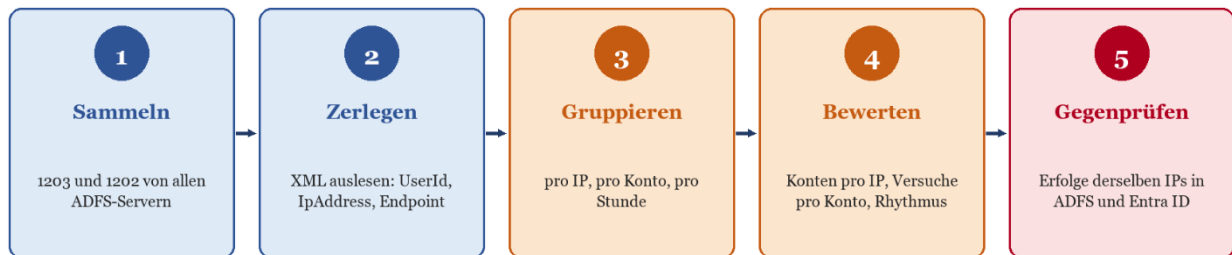
Microsoft warnt in der Dokumentation zu Connect Health ausdrücklich: Eine Gruppenrichtlinie kann das ADFS-Auditing deaktivieren. Klassiker ist eine zentrale Audit-Richtlinie, die „Application Generated“ nicht kennt und bei jeder Aktualisierung wieder abschaltet. Prüfe mit auditpol nach einem gpupdate, ob deine Einstellung überlebt hat.

Auswertung mit PowerShell: vom Rohereignis zum Befund

Die Ereignisanzeige mit ihrem Filterdialog ist für einen Spray ungefähr so geeignet wie ein Teelöffel zum Ausschöpfen eines Kellers. Du brauchst Zahlen pro IP, pro Konto und pro Stunde – und zwar über alle Server der Farm. Die folgenden Beispiele bauen aufeinander auf und verwenden nur Bordmittel von Windows, das ADFS-Modul und Microsoft Graph PowerShell. Die Servernamen sind Beispiele aus der Domäne contoso.local.

Skizze 3: Vom Ereignis zum Befund in fünf Schritten

Was die PowerShell-Beispiele in diesem Beitrag der Reihe nach tun



Schritt 5 entscheidet über deinen Tag: Ein Spray ohne Treffer ist Rauschen. Ein Spray mit Treffer ist ein Sicherheitsvorfall.

Die Schwellen in Schritt 4 sind Startwerte. Miss erst eine ruhige Woche, dann weißt du, was bei dir normal ist.

Skizze 3: Sammeln, zerlegen, gruppieren, bewerten, gegenprüfen – die Reihenfolge der folgenden Listings.

Schritt 1 und 2: Ereignisse sammeln und zerlegen

Das erste Skript holt 1202 und 1203 der letzten 24 Stunden von allen ADFS-Servern und zerlegt das XML direkt auf dem jeweiligen Server. Das spart Bandbreite und vermeidet Ärger mit deserialisierten Ereignisobjekten. Die Felder liest es über ihren lokalen Namen aus, damit Namensräume im XML nicht stören. Schau dir vor dem ersten Lauf ein einzelnes 1203 in der Ereignisanzeige an (Reiter „Details“, XML-Ansicht) – dann weißt du, welche Felder deine ADFS-Version tatsächlich liefert.

```

$servers = 'adfs01.contoso.local', 'adfs02.contoso.local'
$start = (Get-Date).AddHours(-24)

$events = Invoke-Command -ComputerName $servers -ArgumentList $start -ScriptBlock {
    param($start)
    function Feld($xml, $name) {
        $n = $xml.SelectSingleNode("//*[@local-name()='<`>$name'<`>"]")
        if ($n) { $n.InnerText }
    }
    $filter = @{
        LogName = 'Security'
        ProviderName = 'AD FS Auditing'
    }
}

```

```

        Id          = 1202, 1203
        StartTime   = $start
    }
    Get-WinEvent -FilterHashtable $filter -ErrorAction SilentlyContinue | ForEach-Object {
        $x          = [xml]$_ .Properties[1].Value
        $ipKette    = Feld $x 'IpAddress'
        [pscustomobject]@{
            Zeit      = $_.TimeCreated
            Server    = $env:COMPUTERNAME
            EventId   = $_.Id
            Benutzer  = (Feld $x 'UserId')
            IP        = (($ipKette -split ',')[0]).Trim()
            IpKette   = $ipKette
            Endpunkt  = (Feld $x 'Endpoint')
            Agent     = (Feld $x 'UserAgentString')
        }
    }
}

$fail = $events | Where-Object EventId -eq 1203
"{0} Fehlversuche, {1} erfolgreiche Prüfungen" -f $fail.Count, ($events.Count - $fail.Count)

```

Listing 2: 1202 und 1203 von allen Farmservern einsammeln

TIPP – Interne Adressen gleich aussortieren

Anfragen aus dem Intranet gehen direkt an die ADFS-Server und tragen interne Adressen. Für die Spray-Suche interessieren dich zunächst die externen. Ein schlichter Filter genügt: `$fail | Where-Object { $_.IP -notmatch '^(10\.|192\.168\.|172\.(1[6-9]|2\d|3[01])\.)' }`. Bleiben danach kaum Ereignisse übrig, obwohl Extranet-Zugriffe stattfinden, ist das wieder der Load Balancer, der die Client-IP verschluckt.

Schritt 3 und 4: Gruppieren und bewerten

Jetzt wird gezählt. Die erste Auswertung schaut pro IP, wie viele Versuche gegen wie viele verschiedene Konten liefen. Eine IP mit vielen Konten und wenigen Versuchen pro Konto ist der Spray-Kandidat schlechthin. Die Schwellen im Beispiel – mindestens zehn Konten, höchstens drei Versuche pro Konto – sind Startwerte, keine Naturkonstanten. Lass die Auswertung eine ruhige Woche lang laufen, dann weißt du, was bei dir normal ist.

```

$proIp = $fail | Group-Object IP | ForEach-Object {
    $konten = @($_.Group.Benutzer | Sort-Object -Unique).Count
    $sorted = $_.Group | Sort-Object Zeit
    [pscustomobject]@{
        IP          = $_.Name
        Versuche    = $_.Count
        Konten      = $konten
        ProKonto    = [math]::Round($_.Count / [math]::Max($konten, 1), 1)
        Erster      = $sorted[0].Zeit
        Letzter     = $sorted[-1].Zeit
    }
}

# Spray-Kandidaten: viele Konten, wenige Versuche pro Konto
$proIp | Where-Object { $_.Konten -ge 10 -and $_.ProKonto -le 3 } |
    Sort-Object Konten -Descending | Format-Table -AutoSize

```

Listing 3: Auswertung pro IP-Adresse

Die zweite Auswertung dreht die Perspektive um und schaut pro Konto. Viele Fehlversuche aus wenigen IPs sind klassisches Brute Force – oder ein Mobiltelefon mit altem Kennwort im Mailprofil, das alle zehn Minuten tapfer weiterprobiert. Viele Fehlversuche aus vielen IPs gegen dasselbe Konto sprechen dagegen für einen verteilten Angriff auf ein lohnendes Ziel, typischerweise Geschäftsführung, Buchhaltung oder Admin-Konten.

```
$fail | Group-Object Benutzer | ForEach-Object {
    [pscustomobject]@{
        Benutzer = $_.Name
        Versuche = $_.Count
        IPs      = @($_.Group.IP | Sort-Object -Unique).Count
    }
} | Where-Object Versuche -ge 10 | Sort-Object Versuche -Descending | Format-Table -AutoSize
```

Listing 4: Auswertung pro Konto

Befund	Wahrscheinliche Ursache	Nächster Schritt
viele Konten, ca. 1 Versuch je Konto, eine IP	Passwort-Spray	Erfolge dieser IP prüfen (Listing 7), IP sperren
ein Konto, viele Versuche, eine IP	Brute Force oder Gerät mit altem Kennwort	User-Agent ansehen; Benutzer fragen, bevor du Alarm schlägst
ein Konto, viele Versuche, viele IPs	gezielter verteilter Angriff	Konto mit MFA absichern, Erfolge prüfen
viele Konten, viele IPs, gleichmäßiger Takt	verteilter Spray	Auswertung pro Stunde (Listing 5), User-Agent
viele unbekannte UserIds	geratene Konten aus Namenslisten	Abgleich mit dem Verzeichnis (Listing 6)

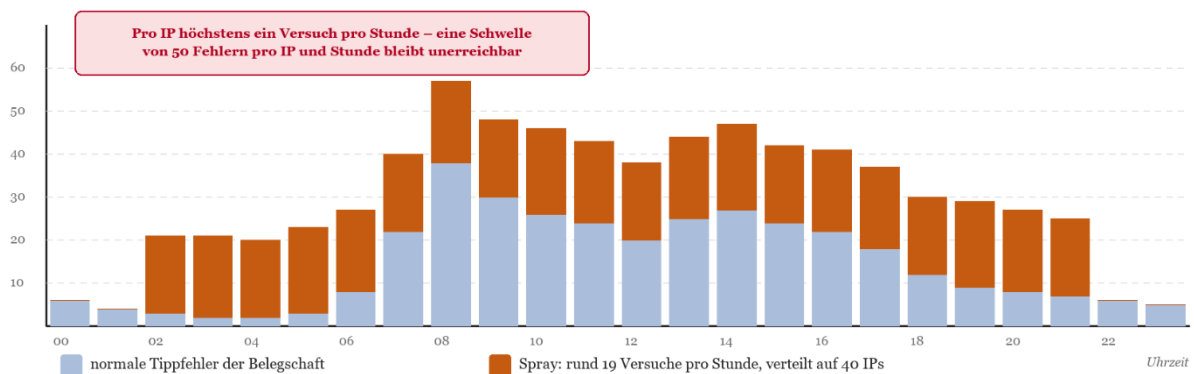
Tabelle 4: Befunde richtig einordnen

Den verteilten Spray sichtbar machen

Der unangenehmste Fall ist der, bei dem keine einzelne IP auffällt. Moderne Angreifer mieten sich dafür Proxy-Dienste mit Tausenden Adressen, gerne aus Privatkundennetzen. Pro IP zwei oder drei Versuche, pro Konto einer – da schlägt keine Schwelle an. Sichtbar wird so ein Angriff nur, wenn du die Zeit zur Achse machst und über alle IPs hinweg zählst.

Skizze 4: Ein verteilter Spray über 24 Stunden

Fehlversuche pro Stunde – einzeln betrachtet unter jeder Alarmschwelle, gemeinsam ein Muster



Keine IP überschreitet die Schwelle, kein Konto wird gesperrt. Auffällig sind nur der nächtliche Sockel, der gleichmäßige Takt und die Konten, die sonst nie Tippfehler haben. Das findest du nur, wenn du über alle IPs hinweg pro Stunde zählst.

Skizze 4: Kein Balken reißt die Schwelle – aber der nächtliche Sockel und der gleichmäßige Takt sind eindeutig.

```
# Fehlversuche, Konten und IPs pro Stunde
$fail | Group-Object { $_.Zeit.ToString('yyyy-MM-dd HH:00') } | ForEach-Object {
    [pscustomobject]@{
        Stunde = $_.Name
        Versuche = $_.Count
        Konten = @($_.Group.Benutzer | Sort-Object -Unique).Count
        IPs = @($_.Group.IP | Sort-Object -Unique).Count
    }
} | Sort-Object Stunde | Format-Table -AutoSize
```

```
# Welche User-Agents und Endpunkte tauchen bei den Fehlversuchen auf?
$fail | Group-Object Agent | Sort-Object Count -Descending | Select-Object -First 10 Count,
Name
$fail | Group-Object Endpunkt | Sort-Object Count -Descending | Select-Object Count, Name
```

Listing 5: Zeitverlauf, User-Agents und Endpunkte

Achte bei der Stundenliste auf das Verhältnis von Konten zu Versuchen. In einer normalen Nacht hast du wenige Fehlversuche von wenigen Konten – oft immer denselben, deren Smartphone ein altes Kennwort gespeichert hat. Bei einem verteilten Spray steigt die Zahl der Konten fast genauso stark wie die der Versuche, und die Zahl der IPs gleich mit. Bei den User-Agents lohnt sich der Blick auf alles, was in deinem Normalbetrieb nicht vorkommt. Bei den Endpunkten fallen besonders Anfragen an die WS-Trust-Endpunkte für Benutzername und Kennwort auf (usernamemixed): Diese Endpunkte haben keine Anmeldeseite, kein JavaScript und kein Captcha und sind deshalb bei Skripten beliebt. Ob du sie über den Proxy veröffentlichen musst, ist eine Härtungsfrage – mit dem Cmdlet Set-AdfsEndpoint und dem Parameter -Proxy lässt sich das pro Endpunkt steuern. Prüfe aber vorher, welche Anwendungen sie noch brauchen.

Geratene Konten erkennen

Ein Spray verrät sich oft durch Benutzernamen, die es nie gab oder nicht mehr gibt. Weil ADFS die eingegebene Kennung protokolliert, kannst du sie gegen das Verzeichnis abgleichen. Mit Microsoft Graph PowerShell geht das gegen Entra ID, sofern deine Benutzer mit UPN synchronisiert sind. Eingaben im Format DOMÄNE\Benutzer filtert das Beispiel heraus; die prüfst du bei Bedarf gegen das lokale AD.

```
Connect-MgGraph -Scopes 'User.Read.All'

$kennungen = $fail.Benutzer | Where-Object { $_ -match '@' } | Sort-Object -Unique
$unbekannt = foreach ($k in $kennungen) {
    $upn = $k -replace '\\', ''
    if (-not (Get-MgUser -Filter "userPrincipalName eq '$upn'" -ErrorAction SilentlyContinue))
    { $k }
}
"{0} von {1} angegriffenen Kennungen existieren nicht" -f @($unbekannt).Count,
@($kennungen).Count
```

Listing 6: Eingegebene Kennungen gegen Entra ID abgleichen

FAKTEN — Was der Anteil unbekannter Konten verrät

Bei normalen Tippfehlern liegt der Anteil unbekannter Kennungen niedrig – Menschen vertippen sich eher beim Kennwort als beim eigenen Namen. Steigt er sprunghaft, arbeitet jemand eine Namensliste ab.

Tauchen dagegen fast nur gültige Konten auf, kennt der Angreifer dein Verzeichnis ziemlich genau – etwa aus einem früheren Leak, dem Internetauftritt oder einem bereits übernommenen Postfach. Das ist die unangenehmere Nachricht.

Schritt 5: Gab es Treffer?

Alles bisher war Vorgeplänkel. Die eine Frage, die über deinen restlichen Tag entscheidet, lautet: Hat sich von einer der verdächtigen IPs jemand erfolgreich angemeldet? Dafür nimmst du die 1202-Ereignisse aus Listing 2 und legst sie neben die Spray-Kandidaten aus Listing 3.

```
$verdaechtig = ($proIp | Where-Object { $_.Konten -ge 10 }).IP
$treffer = $events | Where-Object { $_.EventId -eq 1202 -and $_.IP -in $verdaechtig }
$treffer | Sort-Object Zeit | Format-Table Zeit, Benutzer, IP, Endpunkt -AutoSize

# Kontoaktivität der Getroffenen in Extranet Smart Lockout ansehen
$treffer.Benutzer | Sort-Object -Unique | ForEach-Object {
```

```
Get-AdfsAccountActivity $_ | Format-List
}
```

Listing 7: Erfolgreiche Anmeldungen von verdächtigen IPs

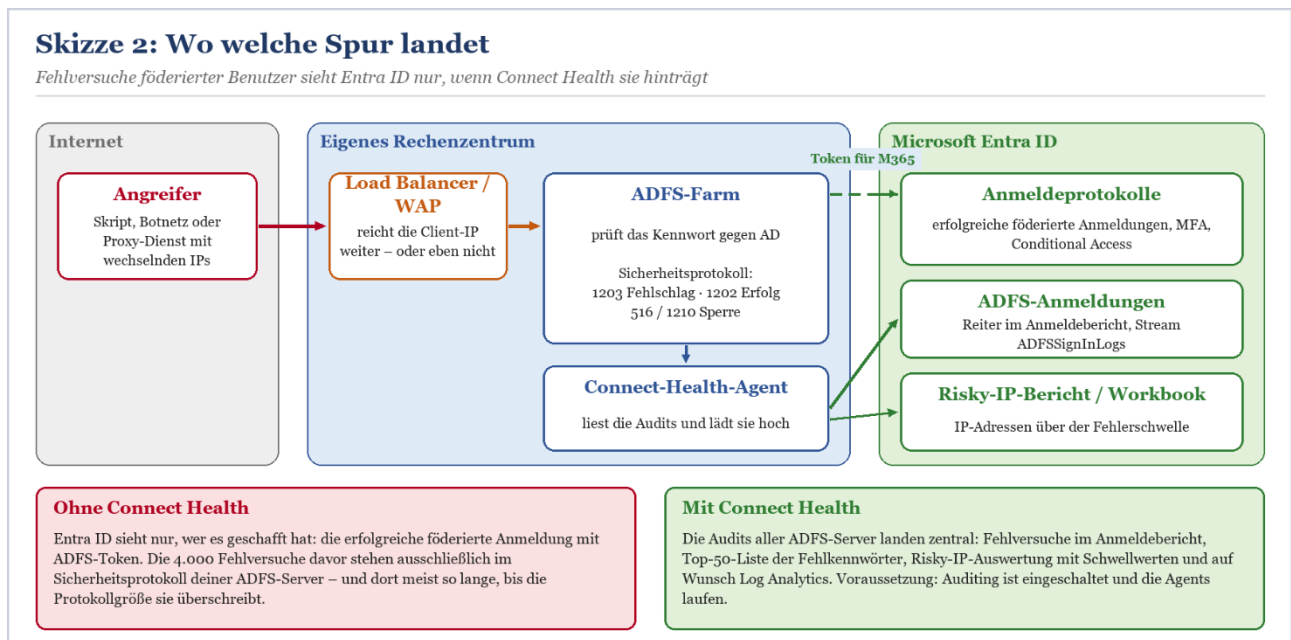
Get-AdfsAccountActivity steht dir zur Verfügung, wenn Extranet Smart Lockout im Log-Only- oder Enforce-Modus läuft. Es zeigt unter anderem die Fehlerzähler für vertraute und unbekannte Orte, den Zeitpunkt des letzten Fehlschlags und die Liste der vertrauten IPs, die bis zu 20 Einträge umfasst. Steht dort die IP des Angreifers, hat Smart Lockout sie nach dem erfolgreichen Versuch als vertraut gelernt – für dieses Konto schützt dich die Sperre dann nicht mehr. Microsoft beschreibt genau diesen Fall: Hat der Angreifer das Kennwort, blockiert Smart Lockout ihn nicht. Hilfreich sind dann nur noch ein neues Kennwort und MFA. Wie du Smart Lockout einrichtest und die Schwellen wählst, zeigt [Extranet Smart Lockout – ADFS gegen Passwort-Spray schützen](#).

WARNUNG – Erfolgreiche Prüfung heißt nicht erfolgreicher Zugriff – aber fast

Ein 1202 bedeutet: Kennwort korrekt. Ob der Angreifer danach an der MFA gescheitert ist oder durch eine Access Control Policy aufgehalten wurde, steht woanders – im 1200 bzw. 1201 derselben Anfrage (gleiche Activity ID) und, für Microsoft 365, in den Anmeldeprotokollen von Entra ID. Für deine Bewertung zählt beides: Ein bekanntes Kennwort ist ein kompromittiertes Kennwort, auch wenn die MFA heute gehalten hat.

Entra ID, ID Protection und Connect Health richtig nutzen

In Hybridumgebungen liegt die zweite Hälfte der Wahrheit in Entra ID. Hier lauert allerdings ein Denkfehler, den man in fast jedem Vorfall wiederfindet: „Wir haben doch Entra ID Protection, das hätte uns gewarnt.“ Bei föderierten Domänen prüft nicht Entra ID das Kennwort, sondern ADFS. Microsoft formuliert es im eigenen Leitfaden zur Untersuchung von Passwort-Spray unmissverständlich: Bei föderierter Authentifizierung liegen die erfolgreichen Anmeldungen in Entra ID, die fehlgeschlagenen beim Identitätsanbieter.



Skizze 2: Ohne Connect Health bleiben die Fehlversuche auf den ADFS-Servern. Entra ID sieht nur die Treffer.

Anmeldeprotokolle in Entra ID auswerten

Hat der Angreifer ein Kennwort erraten und sich an Microsoft 365 angemeldet, taucht die Anmeldung in den Anmeldeprotokollen von Entra ID auf – mit IP-Adresse, Anwendung, Client-

App, Conditional-Access-Ergebnis und Fehlercode. Damit beantwortest du die Folgefragen: Hat die MFA gegriffen? Wurde Conditional Access ausgelöst? Welche Anwendung wollte der Angreifer? Die alten Cmdlets aus den Modulen MSOnline und AzureAD, die früher in jeder Anleitung standen, sind ausgemustert. Der aktuelle Weg ist Microsoft Graph PowerShell.

```
Connect-MgGraph -Scopes 'AuditLog.Read.All', 'Directory.Read.All'

$ip = '203.0.113.45'
$von = (Get-Date).ToUniversalTime().AddDays(-3).ToString('yyyy-MM-ddTHH:mm:ssZ')

Get-MgAuditLogSignIn -Filter "ipAddress eq '$ip' and createdDateTime ge $von" -All |
    Select-Object CreatedDateTime, UserPrincipalName, AppDisplayName, ClientAppUsed,
        @{ n = 'Fehlercode'; e = { $_.Status.ErrorCode } },
        ConditionalAccessStatus, RiskLevelDuringSignIn |
    Sort-Object CreatedDateTime | Format-Table -AutoSize
```

Listing 8: Anmeldungen einer verdächtigen IP in Entra ID

Fehlercode 0 heißt: Anmeldung erfolgreich, der Angreifer ist drin. Steht dort ein Code, der eine MFA-Anforderung oder eine Sperre durch Conditional Access meldet, hat er das Kennwort, ist aber an der nächsten Hürde hängengeblieben – für dich trotzdem ein Grund, das Kennwort ändern zu lassen. Den Klartext zu jedem Code liefert die Spalte „Fehlerursache“ im Portal. Hast du einen Teil der Benutzer bereits per Staged Rollout auf Password Hash Sync oder Pass-through-Authentifizierung umgestellt, liegen deren Fehlversuche direkt in Entra ID. Dort steht dann typischerweise der Fehlercode 50126 für einen falschen Benutzernamen oder ein falsches Kennwort und 50053 für ein durch Smart Lockout gesperrtes Konto oder eine bekannte böartige IP. Wie dieser Umstieg geht, beschreibt [Microsoft 365 von Federated auf Managed umstellen – mit Staged Rollout](#).

Entra ID Protection: was es sieht und was nicht

Entra ID Protection kennt eine eigene Risikoerkennung „Password spray“ (riskEventType passwordSpray). Microsoft beschreibt sie so: Die Erkennung überwacht Spray-Muster über alle Mandanten hinweg und schlägt nur dann an, wenn ein Angreifer ein Kennwort erfolgreich validiert hat – erfolglose Versuche erzeugen keine Erkennung. Die Details gibt es nur mit Entra ID P2. Da bei föderierten Benutzern die Kennwortprüfung auf deiner ADFS-Farm stattfindet, solltest du dich nicht darauf verlassen, dass diese Erkennung einen Spray gegen ADFS meldet. Was Entra ID Protection sehr wohl bewertet, ist die anschließende Anmeldung bei Entra ID: unbekannte Anmeldeeigenschaften, anonyme oder böartige IP-Adressen und Erkenntnisse aus Microsofts Bedrohungsanalyse.

Erkennung	riskEventType	Lizenz	Nutzen bei ADFS-Föderation
Password spray	passwordSpray	P2	nur, wenn Entra ID die Kennwortprüfung sieht – bei reiner Föderation eingeschränkt
Unfamiliar sign-in properties	unfamiliarFeatures	P2	hoch: ungewohnte IP, ASN oder Gerät bei der Anmeldung an Entra ID
Malicious IP address	maliciousIPAddress	P2	mittel: IP mit hoher Fehlerquote oder schlechtem Ruf
Anonymous IP address	anonymizedIPAddress	Free/P1	mittel: Tor oder anonymisierende VPN-Dienste
Microsoft Entra threat intelligence	investigationsThreatIntelligence	Free/P1	hoch: bekanntes Angriffsmuster
Leaked credentials	leakedCredentials	Free/P1	nur mit Password Hash Sync für lokale Kennwörter

Tabelle 5: Risikoerkennungen und ihr Wert für föderierte Umgebungen

TIPP – Password Hash Sync als stiller Wachposten

Auch wenn deine Domäne föderiert bleibt, kannst du Password Hash Sync zusätzlich einschalten. Dann prüft Entra ID die Kennwörter deiner Benutzer gegen gefundene Leaks, und die Erkennung „Leaked credentials“ funktioniert – laut Microsoft ist PHS dafür bei lokalen Kennwörtern Voraussetzung. Nebenbei hast du einen Notausgang, falls die ADFS-Farm einmal ausfällt.

```
Connect-MgGraph -Scopes 'IdentityRiskEvent.Read.All', 'IdentityRiskyUser.Read.All'

# Spray-Erkennungen und Erkennungen zu einer bestimmten IP
Get-MgRiskDetection -Filter "riskEventType eq 'passwordSpray'" -All |
    Select-Object DetectedDateTime, UserPrincipalName, IPAddress, RiskLevel, RiskState

Get-MgRiskDetection -All | Where-Object IPAddress -eq '203.0.113.45' |
    Select-Object DetectedDateTime, UserPrincipalName, RiskEventType, RiskLevel

# Benutzer, die aktuell als riskant gelten
Get-MgRiskyUser -Filter "riskState eq 'atRisk'" -All |
    Select-Object UserPrincipalName, RiskLevel, RiskLastUpdatedDateTime
```

Listing 9: Risikoerkennungen und riskante Benutzer abfragen

Entra Connect Health: Fehlversuche zentral und mit Schwellen

Connect Health für ADFS ist das Bindeglied, das in Skizze 2 die Fehlversuche nach Entra ID trägt. Der Agent läuft auf den ADFS-Servern und den Proxys, liest die Audits und lädt sie hoch. Voraussetzung sind Entra ID P1 oder P2 und eingeschaltetes ADFS-Auditing, für die Nutzungsanalyse mit der Stufe „Verbose“. Für die Spray-Erkennung sind drei Funktionen interessant.

Funktion	Was sie liefert	Gut zu wissen
Bericht „Bad Password Attempts“	Top 50 der Konten mit Fehlversuchen, 30-Tage-Verlauf, letzte Client-IP	wird laut Microsoft alle 12 Stunden aktualisiert – nichts für den laufenden Angriff
Risky-IP-Bericht	IPs über einer Fehlerschwelle pro Stunde oder Tag, mit Zahl der Fehlversuche, Sperren und verschiedenen Konten	Standardschwellen: 50 Fehler pro Stunde, 100 pro Tag; Sperren 25 pro Stunde, 50 pro Tag. Microsoft verweist inzwischen auf die Workbook-Variante
ADFS-Anmeldungen in Entra ID	eigener Reiter im Anmeldebericht, Stream ADFSSignInLogs für Log Analytics	Grundlage für das Risky-IP-Workbook und eigene Abfragen; kann die Log-Analytics-Kosten deutlich erhöhen

Tabelle 6: Connect Health und die Spray-Erkennung

Die Standardschwellen des Risky-IP-Berichts sind bewusst grob – und genau deshalb findet er den verteilten Spray aus Skizze 4 nicht. Er ist gut für die laute Variante und als Frühwarnsystem, wenn du die E-Mail-Benachrichtigung einschaltest, die ab Werk aus ist. Für die leisen Varianten brauchst du eigene Auswertungen, entweder per PowerShell wie oben oder als Abfrage auf ADFSSignInLogs in Log Analytics bzw. Microsoft Sentinel. Ein angenehmes Detail im Anmeldebericht: Versucht jemand ein Konto, das es nicht gibt, steht beim Benutzer eine Null-GUID und im Feld „Anmeldebezeichner“ die eingetippte Kennung. Den Betrieb von Connect Health mit Agents, Warnungen und Diagnose behandelt [ADFS überwachen – Health Checks, Diagnose-Toolbox und Entra Connect Health](#).

WICHTIG – Connect Health ersetzt nicht dein eigenes Auditing

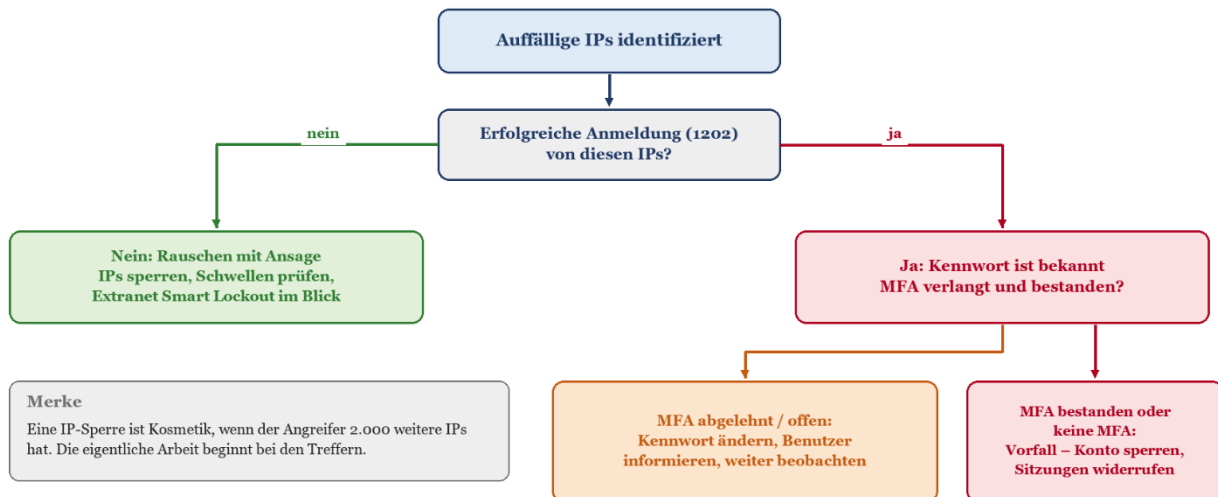
Der Agent kann nur hochladen, was ADFS protokolliert. Ist die Auditrichtlinie weg, sind auch die Berichte leer – Microsoft nennt fehlendes Auditing ausdrücklich als Grund, wenn im Risky-IP-Bericht nichts erscheint. Ein leerer Bericht ist also kein Beweis für Ruhe, sondern zuerst ein Grund, auditpol zu fragen.

Vom Befund zur Reaktion

Wenn die Auswertung einen Spray zeigt, ist der Reflex, sofort IPs zu sperren. Das ist nicht falsch, aber es ist der unwichtigste Teil. Ein Angreifer mit einem Proxy-Netz wechselt die Adresse schneller, als du Set-AdfsProperties tippen kannst. Entscheidend ist, ob er Treffer hatte und was er damit angefangen hat.

Skizze 5: Spray erkannt – und jetzt?

Die entscheidende Frage ist nicht, wie viele Versuche es waren, sondern ob einer gepasst hat



Skizze 5: Erst die Treffer klären, dann aufräumen – nicht umgekehrt.

```

# Auffällige Adressen und Netze in ADFS sperren (wirkt auch auf weitergeleitete Client-IPs)
Set-AdfsProperties -AddBannedIps '203.0.113.45', '198.51.100.0/24'
Get-AdfsProperties | Select-Object -ExpandProperty BannedIpList

# Getroffenes Konto: Sitzungen in Entra ID widerrufen und als kompromittiert markieren
Connect-MgGraph -Scopes 'User.RevokeSessions.All', 'IdentityRiskyUser.ReadWrite.All'
$u = Get-MgUser -UserId 'anna.beispiel@contoso.de'
Revoke-MgUserSignInSession -UserId $u.Id
Confirm-MgRiskyUserCompromised -UserIds $u.Id

# Lockout-Zähler in ADFS zurücksetzen, nachdem das Kennwort geändert wurde
Reset-AdfsAccountLockout 'anna.beispiel@contoso.de' -Location Unknown
    
```

Listing 10: Sofortmaßnahmen nach einem Treffer

Lage	Sofort	Danach
Spray ohne Treffer	IPs bzw. Netze sperren, Auswertung weiterlaufen lassen	Extranet Smart Lockout prüfen, Kennwortrichtlinie und gesperrte Kennwortlisten ansehen
Treffer, MFA hat gehalten	Kennwort ändern lassen, Benutzer informieren	prüfen, warum gerade dieses Kennwort erraten wurde – meist Jahreszeit plus Jahr
Treffer ohne MFA oder MFA bestanden	Konto sperren, Kennwort ändern, Sitzungen widerrufen	Postfachregeln, Weiterleitungen, App-Zustimmungen und neue MFA-Methoden des Kontos prüfen
Spray gegen Admin-Konten	wie oben, zusätzlich Admin-Anmeldungen der letzten Tage prüfen	Admin-Konten nicht über das Extranet anmelden lassen; Access Control Policy einschränken

Tabelle 7: Reaktion nach Lage

Für den letzten Punkt sind Access Control Policies das Mittel der Wahl: Eine Richtlinie, die bestimmten Gruppen nur aus dem Intranet oder nur mit MFA Zugriff gewährt, entzieht dem Spray gegen genau diese Konten die Grundlage – mehr dazu in [Access Control Policies in ADFS – Zugriffsregeln ohne Claim-Rule-Akrobatik](#). Und weil die beste Erkennung nichts nützt, wenn das erratene Kennwort allein reicht, gehört MFA für alle Extranet-Zugriffe zur Pflicht. Wie du den Entra-MFA-Adapter in ADFS anbindest, steht in [ADFS mit Entra-MFA koppeln – Adapter, Zertifikat und Fallstricke](#).

WARNUNG – Spray ist nicht Golden SAML

Passwort-Spray zielt auf die Kennwörter deiner Benutzer. Golden SAML zielt auf das Token-Signing-Zertifikat deiner Farm – wer das hat, braucht gar kein Kennwort mehr und taucht in keinem 1203 auf. Wenn deine Analyse auffällige Anmeldungen in Entra ID zeigt, zu denen es auf den ADFS-Servern gar keine passenden 1202 gibt, ist das ein Fall für die andere Baustelle.

› [ADFS-Sicherheit: Angriffe, Härtung, Golden SAML und MFA](#)

WEITER – Passende Beiträge zum Thema

- › [Extranet Smart Lockout – ADFS gegen Passwort-Spray schützen](#)
- › [ADFS-Ereignisprotokolle lesen – Admin-Log, Debug-Tracing und Auditing](#)
- › [ADFS überwachen – Health Checks, Diagnose-Toolbox und Entra Connect Health](#)
- › [ADFS hinter dem Load Balancer – Health Probe, SNI und Persistenz](#)
- › [Web Application Proxy – Architektur und Ablöse](#)
- › [Microsoft Entra ID](#)

Fazit: Zählen schlägt Hoffen

Passwort-Spray ist kein Hexenwerk, sondern Statistik – auf beiden Seiten. Der Angreifer setzt darauf, dass jedes einzelne Ereignis harmlos aussieht. Du gewinnst, indem du die Ereignisse zusammenzählst: pro IP, pro Konto, pro Stunde. Die Werkzeuge dafür sind vorhanden. ADFS schreibt mit 1202 und 1203 alles, was du brauchst, sobald das Auditing wirklich an ist. PowerShell macht daraus in Minuten einen Befund. Connect Health trägt die Fehlversuche nach Entra ID und liefert Schwellwert-Alarme, Entra ID und ID Protection bewerten, was nach dem erratenen Kennwort passiert.

Wichtig ist, die Lücken zu kennen: Entra ID sieht bei föderierten Domänen die Fehlversuche nicht von selbst, der Risky-IP-Bericht findet den leisen Spray nicht, und ein Load Balancer ohne weitergereichte Client-IP macht jede Auswertung pro IP zur Makulatur. ADFS läuft in vielen Unternehmen seit über zehn Jahren und wird das noch eine Weile tun. Mit sauberem Auditing, einer regelmäßigen Auswertung und MFA für das Extranet ist die Farm kein leichtes Ziel – bis du den Umstieg auf Entra ID in Ruhe geplant hast. Wenn du dabei oder bei der Analyse eines konkreten Vorfalls Unterstützung möchtest, findest du sie im [Consulting zu ADFS \(Active Directory Federation Services\)](#). Auditing, Smart Lockout und die Absicherung der Farm vertieft außerdem das Buch – mehr dazu unter [ADFS in der Praxis – das Buch im Detail](#). Und wer das Ganze einmal im Labor durchspielen will, statt es im Ernstfall zu lernen, ist in den [ADFS-Schulungen](#) richtig.

FAQ: Passwort-Spray auf ADFS erkennen

Welche Event-ID zeigt fehlgeschlagene Anmeldungen auf ADFS?

Ab ADFS in Windows Server 2016 ist es das Ereignis 1203 (Fresh Credential Validation Error) im Sicherheitsprotokoll mit der Quelle „AD FS Auditing“. Erfolgreiche Kennwortprüfungen protokolliert ADFS als 1202.

Warum sehe ich keine 1203-Ereignisse, obwohl Anmeldungen fehlschlagen?

Meist fehlt eine der drei Voraussetzungen: das Benutzerrecht „Generieren von Sicherheitsüberwachungen“ für das ADFS-Dienstkonto, die Unterkategorie „Application Generated“ in auditpol oder die Fehlerüberwachung in den ADFS-Eigenschaften. Auch eine Gruppenrichtlinie kann das Auditing wieder abschalten.

Wie unterscheide ich Passwort-Spray von Brute Force?

Brute Force zeigt viele Fehlversuche für ein Konto. Passwort-Spray zeigt viele verschiedene Konten mit je einem oder zwei Versuchen – pro IP oder verteilt über viele IPs. Zähle deshalb sowohl pro Konto als auch pro IP.

Sehe ich ADFS-Fehlversuche in den Entra-Anmeldeprotokollen?

Nicht von selbst. Bei föderierten Domänen prüft ADFS das Kennwort, die Fehlversuche bleiben auf den ADFS-Servern. Mit Entra Connect Health für ADFS erscheinen sie in einem eigenen Bereich des Anmeldeberichts und optional in Log Analytics.

Erkennt Entra ID Protection einen Spray gegen meine ADFS-Farm?

Darauf solltest du dich nicht verlassen. Die Erkennung „Password spray“ schlägt nur bei erfolgreich validierten Kennwörtern an und setzt voraus, dass Entra ID die Prüfung sieht. ID Protection bewertet aber die anschließende Anmeldung bei Entra ID, etwa über ungewohnte Anmeldeeigenschaften oder bösartige IPs.

Welche Schwellen nutzt der Risky-IP-Bericht von Connect Health?

Standardmäßig 50 Fehlversuche pro Stunde und 100 pro Tag aus Fehlkennwörtern und Sperren zusammen sowie 25 bzw. 50 Extranet-Sperren pro Stunde bzw. Tag. Du kannst die Werte anpassen. Die E-Mail-Benachrichtigung ist ab Werk aus.

Warum steht im Risky-IP-Bericht die Adresse meines Load Balancers?

Weil der Load Balancer die Client-IP nicht an den Web Application Proxy weitergibt. Dann sieht ADFS alle Anfragen von derselben Adresse, und jede Auswertung pro IP wird unbrauchbar.

Kann ich auf ADFS einzelne IP-Adressen sperren?

Ja, mit Set-AdfsProperties -AddBannedIps. Das wirkt auch auf die vom Proxy weitergeleiteten Client-IPs. Bei verteilten Angriffen mit wechselnden Adressen hilft das allerdings nur begrenzt.

Schützt Extranet Smart Lockout vor Passwort-Spray?

Es verhindert, dass Angreifer Konten aus unbekannten Orten sperren oder dort beliebig oft raten. Ein Spray mit einem Versuch pro Konto bleibt aber unter jeder Schwelle, und hat der Angreifer das richtige Kennwort, lässt Smart Lockout ihn durch. Erkennung und MFA bleiben deshalb nötig.

Was mache ich, wenn eine verdächtige IP ein gültiges Kennwort hatte?

Behandle das Konto als kompromittiert: Kennwort ändern, Sitzungen in Entra ID widerrufen, Anmeldungen und Postfachregeln prüfen. Hat die MFA gehalten, ist das Kennwort trotzdem bekannt und muss geändert werden.