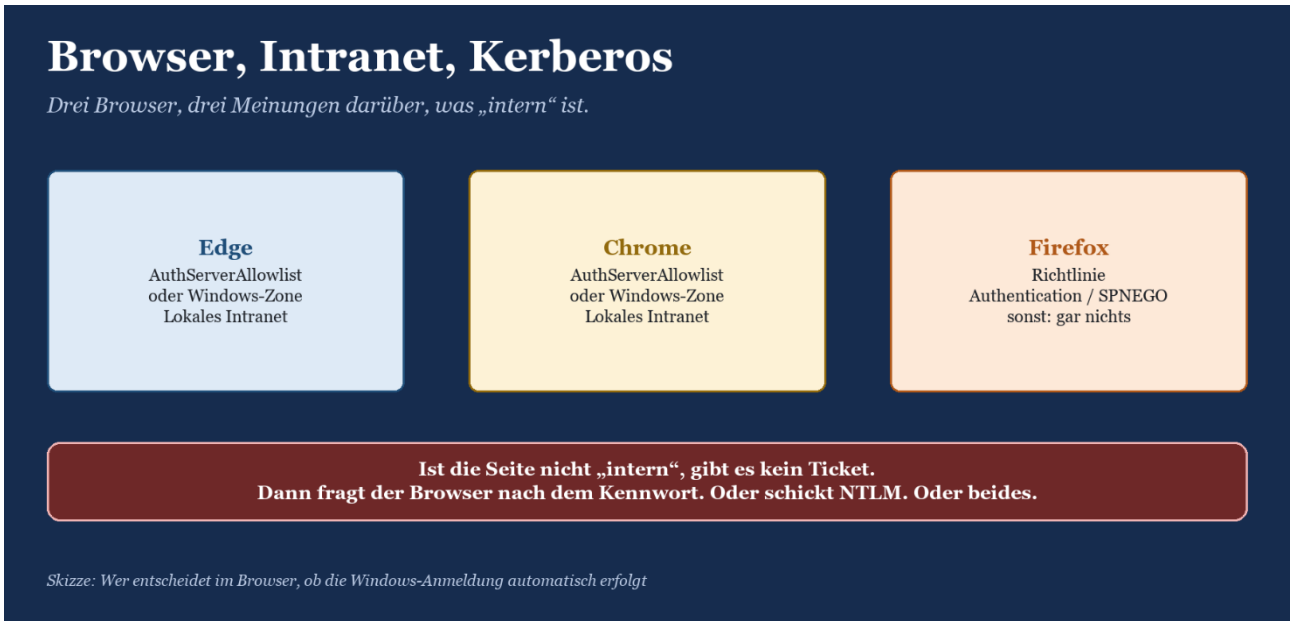
Edge, Chrome und Firefox: Browser für Kerberos im Intranet einstellen



WISSEN
Was der NTLM Diagnostiker erfasst, wie er arbeitet und was er dir zeigt.
› **NTLM Diagnostiker**

BERATUNG
Webanwendungen, Browser und SPNs gemeinsam auf Kerberos bringen: Wir planen das mit dir.
› **Beratung zur NTLM-Ablösung**

SCHULUNG
Ereignisse lesen, Verursacher finden, Ausnahmen begründen: praxisnah für dein Team.
› **Schulung zum NTLM Diagnostiker**

Es gibt diese eine Frage, die jede Helpdesk-Kollegin und jeder Helpdesk-Kollege kennt: „Warum will das Intranet heute mein Kennwort?“ Die ehrliche Antwort lautet meist: weil jemand den Link mit vollem Namen verschickt hat. Gestern stand in der Adresszeile http://intranet, heute http://intranet.musterwerk.local, und für den Browser sind das zwei völlig verschiedene Welten. Die eine ist „intern“ und bekommt die Windows-Anmeldung geschenkt, die andere ist „Internet“ und bekommt einen Anmeldedialog. Und wenn jemand dort brav Benutzername und Kennwort eintippt, landet die Anmeldung nicht selten bei NTLM.

Dieser Beitrag zu edge kerberos intranet zeigt dir, wie Edge, Chrome und Firefox entscheiden, ob sie eine Seite automatisch mit deiner Windows-Anmeldung bedienen, und wie du das per Gruppenrichtlinie so einstellst, dass dabei Kerberos herauskommt und nicht NTLM: Intranetzone und Zonenzuordnung, die Richtlinie für zulässige Authentifizierungsserver in Edge und Chrome, die Firefox-Richtlinie Authentication und der Unterschied zwischen Kurznamen und vollständigem Namen, der in fast jeder Umgebung für Verwirrung sorgt.

Warum das jetzt Eile hat: Seit dem Windows-Update vom 13.10.2026 blockieren Windows 11 ab 24H2 und Windows Server 2025 NTLMv1 für Single Sign-on standardmäßig, und mit der nächsten Windows-Server-Version ist NTLM standardmäßig abgeschaltet. Den Zeitplan findest du in NTLM-Abschaltung: Microsofts Fahrplan und was er für dich bedeutet. Webanwendungen sind dabei ein dankbares Feld: Die Serverseite ist oft schnell auf Negotiate umgestellt, der Browser ist das Bauteil, das am häufigsten übersehen wird.

FAKTEN · Das Wichtigste in Kürze

Edge und Chrome antworten auf eine Negotiate-Aufforderung nur dann automatisch, wenn der Server in der Richtlinie AuthServerAllowlist steht. Ist sie nicht gesetzt, versuchen sie zu erkennen, ob der Server im Intranet liegt; unter Windows heißt das: Zone „Lokales Intranet“.

Windows ordnet Namen ohne Punkt (http://intranet) automatisch der Intranetzone zu, vollständige Namen mit Punkt (http://intranet.musterwerk.local) dagegen der Internetzone, solange du sie nicht zuordnest.

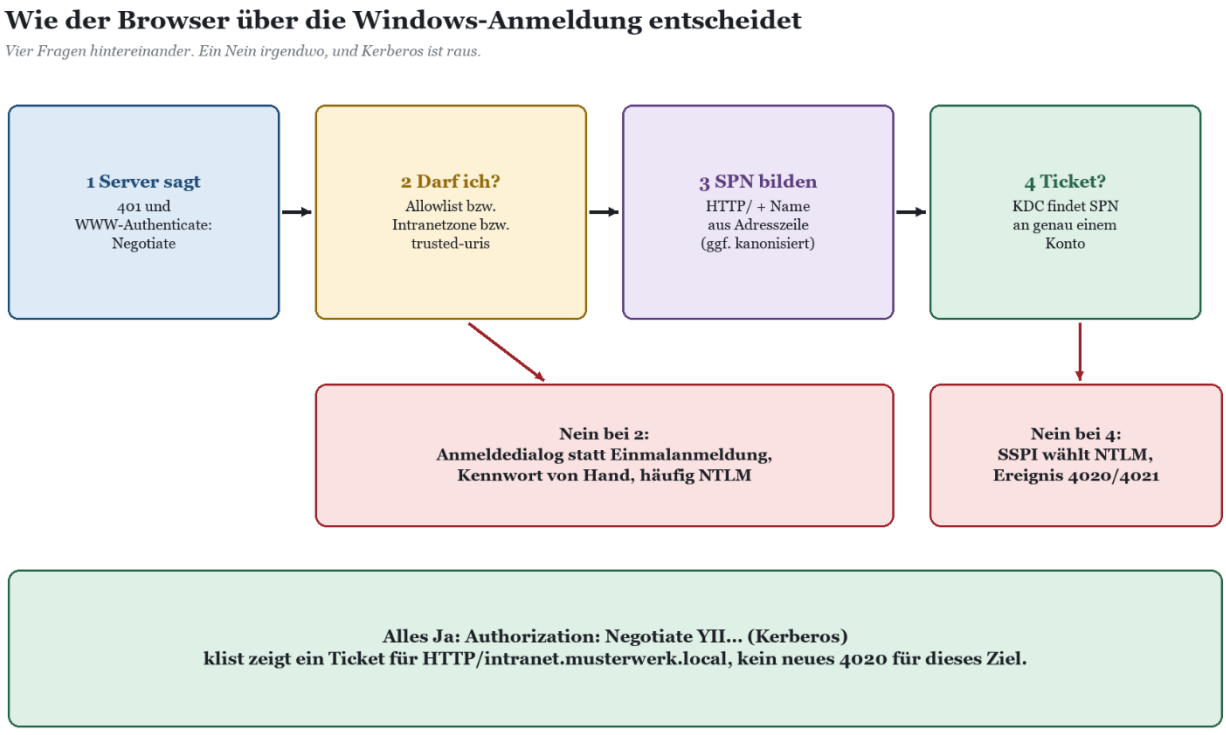
Firefox nutzt die Windows-Zonen nicht. Er braucht die Richtlinie Authentication mit einer SPNEGO-Liste, und Kurznamen lässt er standardmäßig gar nicht zu.

Der SPN entsteht aus dem Namen in der Adresszeile, bei Edge und Chrome nach DNS-Kanonisierung. Fehlt er, gibt es kein Ticket, und der Browser fällt auf NTLM zurück.

Ob es wirklich Kerberos ist, zeigen klist und die Ereignisse 4020/4021 auf dem Client.

Wie der Browser entscheidet, ob er sich anmeldet

Integrierte Windows-Authentifizierung im Browser läuft über eine Verhandlung. Der Webserver antwortet auf die erste Anfrage mit 401 und dem Kopf WWW-Authenticate: Negotiate. Der Browser muss jetzt zwei Dinge entscheiden. Erstens: Darf ich mich bei diesem Server überhaupt automatisch mit der Windows-Anmeldung melden? Zweitens: Für welchen Dienstnamen hole ich mir ein Ticket? Erst wenn beide Fragen sauber beantwortet sind, schickt er ein Kerberos-Token. Wie die Serverseite aussieht, also Negotiate statt NTLM im IIS, steht in NTLM in IIS und Webanwendungen: Negotiate statt NTLM.



Skizze: Entscheidungsweg bei integrierter Windows-Authentifizierung im Browser, vereinfacht

Geht die erste Frage schief, bekommt die Anwenderin oder der Anwender einen Anmeldedialog. Geht die zweite schief, weil der SPN fehlt oder auf einem anderen Namen steht, merkt niemand etwas: Windows wählt innerhalb von Negotiate still NTLM, die Seite öffnet sich, und das Problem

taucht erst auf, wenn NTLM abgeschaltet ist. Dann allerdings gründlich. Der stille Fall ist der gefährlichere, und genau den findest du nur in den Ereignissen.

Die Windows-Zonen: altes Erbe, aktuelle Wirkung

Die Sicherheitszonen stammen aus der Zeit des Internet Explorers, sind aber nicht mit ihm gestorben. Windows verwaltet sie weiterhin, und Edge sowie Chrome fragen sie unter Windows ab, wenn keine eigene Allowlist gesetzt ist. Für die automatische Anmeldung zählt im Wesentlichen die Zone „Lokales Intranet“: Dort ist standardmäßig eingestellt, dass die Anmeldung automatisch nur in der Intranetzone erfolgt. Vertrauenswürdige Sites und Internet bekommen standardmäßig einen Dialog.

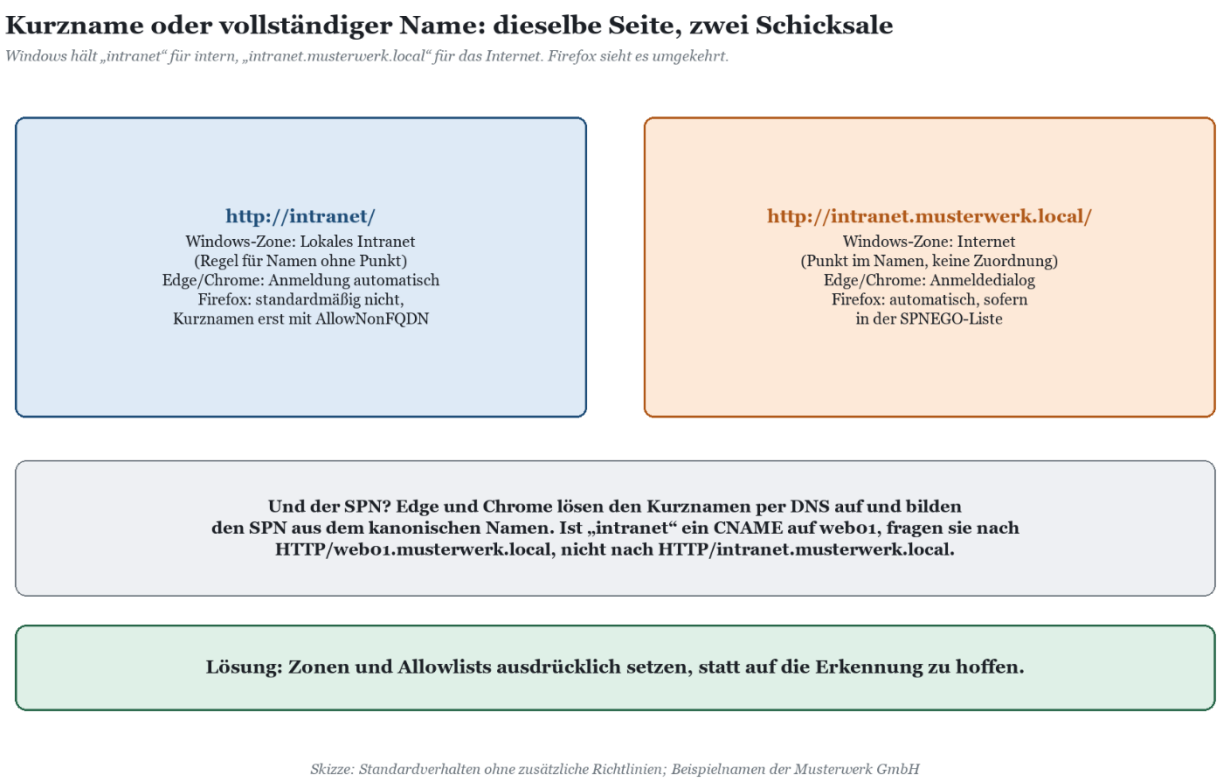
Zone	Wert in der Zuordnungsliste	Automatische Anmeldung (Standard)	Bemerkung
Lokales Intranet	1	ja	die Zone, in die deine internen Webanwendungen gehören
Vertrauenswürdige Sites	2	nein, Anmeldedialog	für Kerberos im Intranet die falsche Wahl
Internet	3	nein, Anmeldedialog	Standard für alles mit Punkt im Namen ohne Zuordnung
Eingeschränkte Sites	4	nein	für das Thema hier ohne Bedeutung

WARNUNG · Vertrauenswürdig ist nicht intern

Ein Klassiker aus alten Anleitungen: Intranetadressen in die Zone „Vertrauenswürdige Sites“ eintragen, weil das so schön vertrauensvoll klingt. Für die automatische Anmeldung bringt das nichts, im Gegenteil: Die Seite fällt aus der Intranetzone heraus, und der Anmeldedialog kommt zurück. Interne Webanwendungen gehören in Zone 1.

Kurzname gegen vollständigen Namen

Jetzt zum eigentlichen Rätsel. Windows erkennt das Intranet ohne Zuordnung über einige Faustregeln, und die bekannteste lautet: Ein Name ohne Punkt ist intern. http://intranet landet deshalb in der Intranetzone, und Edge und Chrome melden dich automatisch an. http://intranet.musterwerk.local enthält Punkte, also könnte es aus Sicht von Windows genauso gut eine Adresse im Internet sein, und es landet in der Internetzone. Ergebnis: Anmeldedialog. Weitere Faustregeln wie „Sites, die den Proxyserver umgehen“ helfen manchmal, sind aber von der Proxykonfiguration abhängig und damit kein Fundament, auf das du bauen willst.



Skizze: Standardverhalten ohne zusätzliche Richtlinien; Beispielnamen der Musterwerk GmbH

Firefox dreht das Ganze um. Er kennt die Windows-Zonen nicht und meldet sich nur bei Servern automatisch an, die in seiner SPNEGO-Liste stehen. Kurznamen ohne Punkt lässt er standardmäßig überhaupt nicht zu, solange du das nicht ausdrücklich erlaubst. In einer gemischten Umgebung funktioniert also bei den einen der Kurzname und bei den anderen der vollständige Name. Willkommen im Helpdesk.

Dazu kommt der SPN. Edge und Chrome lösen den Namen aus der Adresszeile per DNS auf und verwenden standardmäßig den kanonischen Namen für den SPN. Aus http://intranet wird so über das DNS-Suffix HTTP/intranet.musterwerk.local, und ist intranet ein CNAME auf web01, wird daraus HTTP/web01.musterwerk.local. Das funktioniert, wenn die Webanwendung unter dem Computerkonto von web01 läuft. Läuft sie unter einem Dienstkonto, an dem nur HTTP/intranet.musterwerk.local hängt, verschlüsselt der KDC das Ticket für das falsche Konto, und die Anmeldung scheitert oder fällt zurück. Die Richtlinie DisableAuthNegotiateCnameLookup schaltet die Kanonisierung ab; dann gilt der Name aus der Adresszeile. Die ganze Geschichte mit Aliasnamen steht in CNAME, Aliasnamen und hosts-Datei: warum Kerberos den Namen nicht findet.

Adresse in der Adresszeile	Windows-Zone ohne Zuordnung	Edge / Chrome ohne Allowlist	Firefox ohne Richtlinie	SPN (Edge/Chrome, Standard)
http://intranet/	Lokales Intranet	automatische Anmeldung	keine Anmeldung	HTTP/ + kanonischer Name aus DNS
http://intranet.musterwerk.local/	Internet	Anmeldedialog	keine Anmeldung	HTTP/intranet.musterwerk.local oder CNAME-Ziel
http://10.20.0.30/	je nach Regel und	uneinheitlich	keine Anmeldu	kein brauchbarer SPN, Ergebnis: NTLM

	Proxy		ng	
https://intranet.musterwerk.local:8443/	Internet	Anmeldedialog	keine Anmeldung	Port standardmäßig nicht im SPN

TIPP · Kein Port im SPN

Edge und Chrome bauen den Port standardmäßig nicht in den SPN ein. Für https://intranet.musterwerk.local:8443 fragen sie also nach HTTP/intranet.musterwerk.local. Die Richtlinie EnableAuthNegotiatePort ändert das. Lass sie aus, solange du keinen sehr guten Grund hast, sonst brauchst du für jeden Port einen eigenen SPN. Und ja: Auch für https heißt die Dienstklasse HTTP.

Intranetzone per Gruppenrichtlinie zuordnen

Der erste und für Edge und Chrome oft schon ausreichende Schritt ist eine saubere Zonenzuordnung. Du findest sie in der Gruppenrichtlinienverwaltung unter Computerkonfiguration (oder Benutzerkonfiguration), Administrative Vorlagen, Windows-Komponenten, Internet Explorer, Internetsystemsteuerung, Sicherheitsseite, Richtlinie „Liste der Site-zu-Zonenzuweisungen“. Lass dich vom Pfad nicht täuschen: Die Einstellung wirkt auf die Windows-Zonen und damit auch auf Edge und Chrome, wenn diese keine eigene Allowlist haben.

```
Liste der Site-zu-Zonenzuweisungen (Wertname = Site, Wert = Zone)

  *.musterwerk.local          1
  intranet                    1
  https://portal.musterwerk.local   1

Zonen: 1 = Lokales Intranet, 2 = Vertrauenswürdige Sites, 3 = Internet, 4 = Eingeschränkte
Sites
```

Ein Eintrag wie *.musterwerk.local erfasst alle Hosts der internen Domäne, egal ob mit http oder https. Willst du enger fassen, nennst du einzelne Hosts. Beachte, dass eine per Richtlinie gesetzte Liste die Einträge ersetzt, die Anwenderinnen und Anwender selbst in den Internetoptionen gepflegt haben; die Oberfläche ist dann gesperrt. Das ist meistens genau, was du willst.

WICHTIG · Computer oder Benutzer: nicht mischen

Die Zuordnungsliste gibt es unter Computer- und unter Benutzerkonfiguration. Pflegst du beide, wird es unübersichtlich, und je nach weiteren Einstellungen gewinnt die eine oder die andere. Leg dich auf eine Seite fest, in der Praxis meist die Computerkonfiguration, und dokumentiere das. Prüf das Ergebnis auf einem Pilotrechner mit gpresult /h, bevor du die Richtlinie breit verteilst.

Was die Zone nicht löst

Die Zone entscheidet nur über die erste Frage: ob der Browser sich automatisch meldet. Ob dabei Kerberos oder NTLM herauskommt, entscheidet der SPN. Eine Seite in der Intranetzone mit fehlendem SPN ist der unauffälligste NTLM-Verursacher überhaupt, denn niemand bekommt einen Dialog, und alles läuft. Prüf deshalb für jede interne Webanwendung, ob ihr SPN existiert und an genau einem Konto steht. Hintergrund liefert Fehlender SPN: wenn Kerberos den Dienst nicht findet.

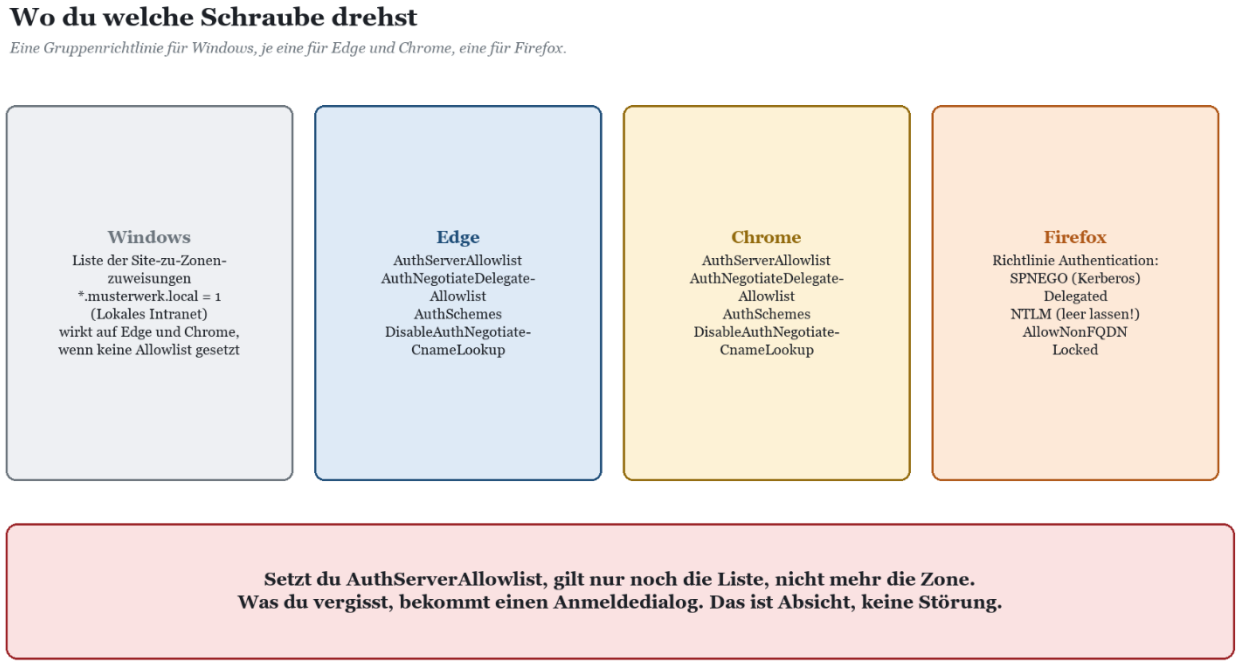
```
# Gibt es den SPN, und an welchem Konto hängt er?
setspn -Q HTTP/intranet.musterwerk.local
```

```
# Doppelte SPNs in der Gesamtstruktur suchen
setspn -X -F

# SPN an das Dienstkonto des Anwendungspools setzen (prüft vorher auf Duplikate)
setspn -S HTTP/intranet.musterwerk.local MUSTERWERK\svc-intranet
```

Edge, Chrome und Firefox per Richtlinie einstellen

Die Zonenzuordnung ist ein guter Anfang, aber nicht das Ende. Edge und Chrome bringen eigene Richtlinien für die HTTP-Authentifizierung mit, die Vorrang vor der Zonenerkennung haben, und Firefox braucht ohnehin eine eigene Einstellung. Der Vorteil eigener Richtlinien: Sie gelten genau so, wie du sie schreibst, unabhängig von Faustregeln und Proxykonfiguration.



Skizze: Richtlinien für integrierte Authentifizierung je Browser; Namen wie in der Herstellerdokumentation

Edge und Chrome: AuthServerAllowlist

Beide Browser basieren auf Chromium und verwenden deshalb dieselben Richtliniennamen. Die wichtigste heißt AuthServerAllowlist. Microsoft beschreibt sie so: Integrierte Authentifizierung wird nur aktiviert, wenn Edge eine Authentifizierungsaufforderung von einem Proxy oder von einem Server in dieser Liste erhält. Mehrere Namen trennst du mit Kommas, Platzhalter mit Stern sind erlaubt. Ist die Richtlinie nicht gesetzt, versucht Edge zu erkennen, ob der Server im Intranet liegt, und antwortet nur dann. Für Edge findest du sie in der ADMX-Vorlage unter Microsoft Edge, Bereich HTTP-Authentifizierung; die Chrome-Vorlage enthält eine gleichnamige Richtlinie im Bereich für die HTTP-Authentifizierung. Ältere Chrome-Anleitungen nennen noch den früheren Namen AuthServerWhitelist.

```
Registrierung (per Gruppenrichtlinie gesetzt, nicht von Hand pflegen):

  HKLM\SOFTWARE\Policies\Microsoft\Edge     AuthServerAllowlist  (REG_SZ)
  HKLM\SOFTWARE\Policies\Google\Chrome      AuthServerAllowlist  (REG_SZ)

  Wert:  *.musterwerk.local,intranet

Prüfen im Browser:  edge://policy  bzw.  chrome://policy
```

Der Stern vor dem Domänennamen ohne Punkt, also *musterwerk.local, erfasst auch einbrotmusterwerk.local. Schreib deshalb lieber *.musterwerk.local und nimm die nackte Domäne musterwerk.local separat auf, falls du sie brauchst. Und denk an die Kurznamen: Setzt du die Allowlist, gilt nur noch sie. Wer bisher mit http://intranet gearbeitet hat, braucht den Kurznamen in der Liste, sonst kommt dort ab sofort ein Anmeldedialog. Eine Änderung greift erst nach einem Neustart des Browsers.

Richtlinie (Edge und Chrome)	Wirkung	Empfehlung im Intranet
AuthServerAllowlist	Server, für die integrierte Authentifizierung automatisch erfolgt	setzen: *.musterwerk.local und nötige Kurznamen
AuthNegotiateDelegateAllowlist	Server, an die der Browser Anmeldedaten per Kerberos-Delegierung weitergeben darf	nur setzen, wenn eine Anwendung Delegierung wirklich braucht; eng fassen
AuthSchemes	erlaubte Verfahren, Standard: basic, digest, ntlm, negotiate	nicht vorschnell ändern; ntlm entfernen wirkt auf alle Seiten
DisableAuthNegotiateCnameLookup	SPN aus dem eingegebenen Namen statt aus dem kanonischen DNS-Namen	nur bei Aliasnamen mit SPN am Dienstkonto, nach Test
EnableAuthNegotiatePort	Port wird in den SPN aufgenommen	nicht setzen

WARNUNG · AuthSchemes ist kein NTLM-Schalter für Mutige

Es ist verlockend, in AuthSchemes einfach ntlm wegzulassen. Das betrifft aber jede Seite und jeden Proxy, nicht nur die Intranetadresse, die du gerade im Blick hast. Und Negotiate mit NTLM im Innern ist davon nicht unbedingt erfasst, weil Windows den Rückfall innerhalb von Negotiate selbst aushandelt. Die gezielte Steuerung von NTLM gehört in die Restrict-NTLM-Richtlinien, siehe Restrict-NTLM-Richtlinien erklärt: Überwachen, Ausnahmen, Verweigern.

Firefox: Richtlinie Authentication

Firefox lässt sich unter Windows über die Gruppenrichtlinienvorlagen von Mozilla oder über eine policies.json im Installationsverzeichnis steuern. Für die Anmeldung zuständig ist die Richtlinie Authentication. Sie kennt eine Liste SPNEGO für Server, bei denen Firefox Kerberos über Negotiate verwenden darf, eine Liste Delegated für die Delegierung, eine Liste NTLM für die automatische NTLM-Anmeldung und die Option AllowNonFQDN, die Kurznamen ohne Punkt getrennt für SPNEGO und NTLM freigibt. Mit Locked verhinderst du, dass jemand die Werte in about:config überschreibt. Hinter den Kulissen setzt die Richtlinie unter anderem die Einstellungen network.negotiate-auth.trusted-uris, network.negotiate-auth.delegation-uris und network.negotiate-auth.allow-non-fqdn.

```
{
  "policies": {
    "Authentication": {
      "SPNEGO": ["musterwerk.local"],
      "AllowNonFQDN": { "SPNEGO": true },
      "Locked": true
    }
  }
}
```

Ein Eintrag musterwerk.local passt auf alle Hosts unter dieser Domäne. Die Liste NTLM lässt du leer, denn genau die willst du loswerden. AllowNonFQDN brauchst du nur, wenn Kurznamen noch in Umlauf sind; besser ist es, Lesezeichen und Verknüpfungen auf vollständige Namen umzustellen und die Option nach einer Übergangszeit wieder zu entfernen. Prüfen kannst du die wirksamen Richtlinien in Firefox unter about:policies.

Aufgabe	Edge	Chrome	Firefox
automatische Anmeldung freigeben	AuthServerAllowlist	AuthServerAllowlist	Authentication / SPNEGO
Kerberos-Delegierung	AuthNegotiateDelegateAllowlist	AuthNegotiateDelegateAllowlist	Authentication / Delegated
Kurznamen ohne Punkt	über Allowlist oder Intranetzone	über Allowlist oder Intranetzone	Authentication / AllowNonFQDN
Fallback ohne eigene Richtlinie	Windows-Zone Lokales Intranet	Windows-Zone Lokales Intranet	keine automatische Anmeldung
Kontrolle der wirksamen Werte	edge://policy	chrome://policy	about:policies

AUS DER PRAXIS · Musterwerk GmbH: der Firefox in der Konstruktion

Bei der Musterwerk GmbH ist das Intranet nach der Umstellung auf Kerberos für alle unauffällig, nur die Konstruktion beschwert sich. Dort läuft Firefox, weil ein CAD-Portal es „so empfiehlt“. Ohne Richtlinie bot Firefox nie Kerberos an, die Leute tippten seit Jahren ihr Kennwort in einen Anmeldedialog und niemand fand das seltsam. Eine policies.json mit SPNEGO-Liste später war das Kennwort aus dem Alltag verschwunden. Die Begeisterung hielt sich trotzdem in Grenzen, denn jetzt fehlte der Konstruktion ein Grund zum Meckern.

Ereignisse, die dir verraten, welcher Browser noch NTLM spricht

Edge und Chrome nutzen unter Windows für Negotiate die Windows-Schnittstelle SSPI, Firefox unter Windows ebenfalls. Fällt dabei NTLM, schreibt der Client ein ausgehendes Ereignis 4020 oder 4021 mit ProcessName msedge.exe, chrome.exe oder firefox.exe, auf älteren Systemen das klassische 8001. Dafür muss die NTLM-Überwachung aktiv sein; wie das geht, steht in NTLM-Auditing aktivieren: Schritt für Schritt, ein fertiges Skript für die Gruppenrichtlinie findest du unter Audit-GPO-Skript für die NTLM-Überwachung. Wichtig ist der Begründungstext im Feld NtlmUsageReason, die Reason-ID ist nur ein Hinweis, weil Nummer und Text in der Praxis nicht immer zusammenpassen. Die Begründungen erklärt NTLM-Reason-IDs erklärt: warum Windows nicht Kerberos benutzt hat.

```
# Ausgehende NTLM-Ereignisse von Browsern der letzten 7 Tage, nach Ziel und Begründung
Get-WinEvent -FilterHashtable @{ LogName = 'Microsoft-Windows-NTLM/Operational';
    Id = 4020,4021; StartTime = (Get-Date).AddDays(-7) } |
  ForEach-Object { $x = [xml]$_.ToXml(); $f = @{}
    $x.Event.EventData.Data | ForEach-Object { $f[$_.Name] = $_.'#text' }
    if ($f['ProcessName'] -match '(msedge|chrome|firefox)\.exe$') {
      [pscustomobject]@{ Browser = Split-Path $f['ProcessName'] -Leaf;
        Ziel = $f['TargetMachine']; Dienst = $f['TargetService'];
        Grund = $f['NtlmUsageReason'] } } } |
  Group-Object Browser, Ziel, Dienst, Grund | Sort-Object Count -Descending |
  Select-Object Count, Name
```

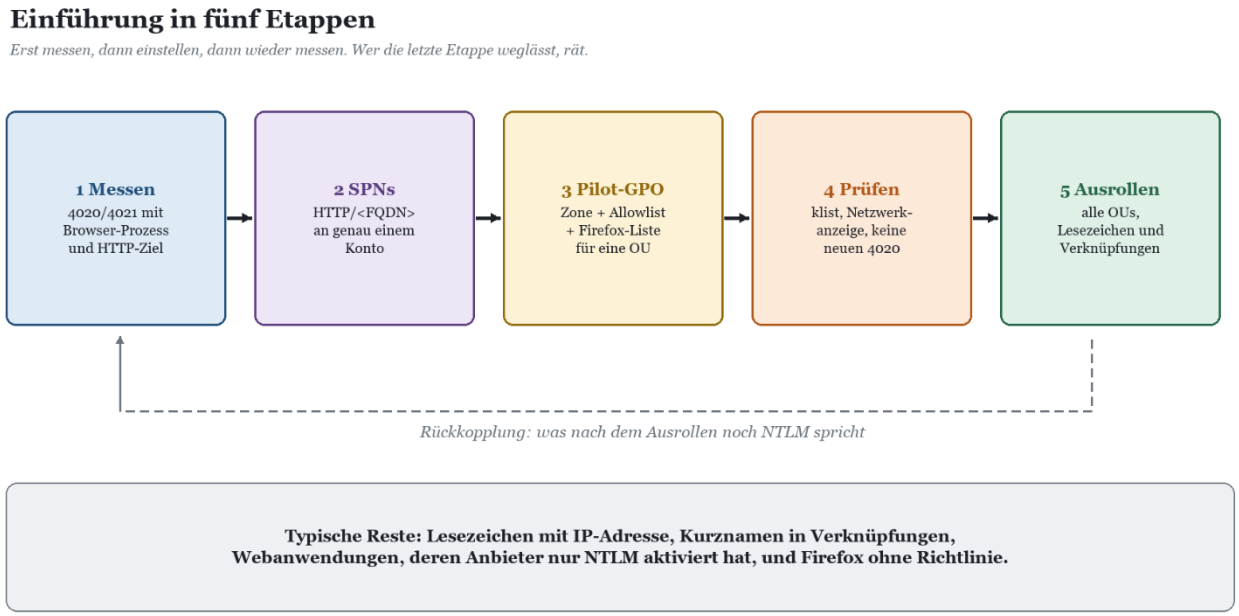
Werte grundsätzlich über die EventData-Felder aus, nicht über den Meldungstext. Bei weitergeleiteten Ereignissen kann der Meldungstext leer sein, die Felder sind trotzdem da. Leere

Felder stehen als „Null“ oder „(NULL)“ in den Daten. Weitere Rezepte findest du in NTLM-Ereignisse mit PowerShell auswerten: Get-WinEvent-Rezepte.

Prüfen, ob es wirklich Kerberos ist

Nach der Umstellung reicht „keine Kennwortabfrage mehr“ als Beweis nicht. Ruf die Seite auf und schau mit klist nach, ob ein Ticket für den erwarteten SPN im Cache liegt. In den Entwicklerwerkzeugen des Browsers siehst du in der Netzwerkansicht den Authorization-Kopf: Beginnt der Wert nach Negotiate mit YII, ist es Kerberos in einer SPNEGO-Hülle; beginnt er mit TlRMTVNTUA, ist es NTLM. Mehr dazu in Mit klist prüfen, ob Kerberos wirklich greift.

```
klist purge                       # Ticket-Cache leeren (nur für den Test)
start msedge http://intranet.musterwerk.local/
klist | findstr /i HTTP/          # erwartet: Server: HTTP/intranet.musterwerk.local @
MUSTERWERK.LOCAL
```



Skizze: Vorgehen bei der Browser-Umstellung auf Kerberos, beispielhaft für die Musterwerk GmbH

Wo es mit Bordmitteln mühsam wird

Für einen Pilotrechner und drei Webanwendungen kommst du mit den Bordmitteln gut zurecht: Zonenliste, Allowlist, policies.json, ein Get-WinEvent-Aufruf und klist. In echten Umgebungen sieht die Lage anders aus. Die Ereignisse 4020/4021 entstehen auf jedem einzelnen Client. Ohne Weiterleitung liegen sie verstreut auf Hunderten Rechnern, und mit Weiterleitung hast du Tausende Einträge, in denen sich msedge.exe, chrome.exe und firefox.exe mit Dutzenden Zielen mischen. Dazu kommen die Rechner, die gar nichts liefern, weil die Überwachung dort nie angekommen ist. Wer dort mit einem alten Lesezeichen per IP-Adresse arbeitet, fällt erst am Stichtag auf.

Die eigentliche Frage ist am Ende nicht „Welche Ereignisse gibt es?“, sondern „Welche Webanwendungen werden noch per NTLM angesprochen, von wem, und warum?“ Eine Seite mit fehlendem SPN brauchst du nur einmal zu reparieren, egal ob 300 Clients sie aufrufen. Ein Webserver, der per IP-Adresse angesprochen wird, braucht neue Lesezeichen, keinen SPN. Genau diese Zusammenfassung nach Ursache, Prozess und Ziel ist mit Bordmitteln eine Fleißarbeit in Excel.

Hier setzt der NTLM Diagnostiker an. Er sammelt die NTLM-Ereignisse vom Domain Controller, über einen Windows Event Collector, per Direktabfrage oder als Datei-Import, fasst gleichartige Verbindungen nach Ursache, Prozess und Ziel zusammen und zeigt dir in der Kommunikationsmatrix, welche Clients welche Webziele noch per NTLM ansprechen. Die SPN-Prüfung meldet fehlende und doppelte SPNs einschließlich Aliasnamen, und die Abdeckungskarte zeigt, welche Rechner gar keine Ereignisse liefern. Dabei arbeitet der Diagnostiker ausschließlich lesend: Er ändert weder Gruppenrichtlinien noch Browsereinstellungen noch SPNs. Befehle wie setspn zeigt er dir an und bietet sie zum Kopieren an; ausführen tust du sie selbst.

SO GEHT'S MIT DEM NTLM DIAGNOSTIKER · Webziele mit NTLM auf der Landkarte finden

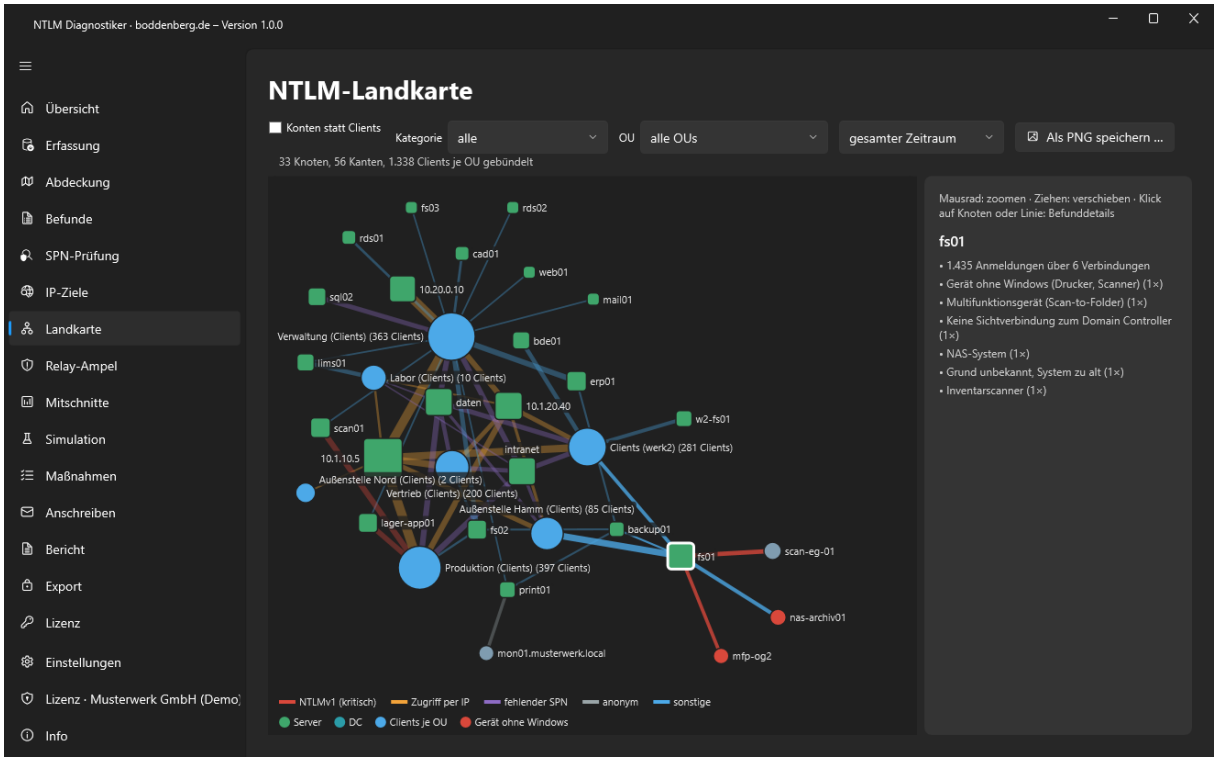
1. Erfass die NTLM-Ereignisse, im einfachsten Fall den Fünf-Minuten-Befund direkt vom Domain Controller, besser zusätzlich die Client-Ereignisse über deinen Event Collector.

2. Öffne die NTLM-Landkarte. Sie zeigt Server, Clients je OU, Domain Controller und Geräte ohne Windows als Knoten; die farbigen Verbindungen stehen für die Kategorie, etwa fehlender SPN, Zugriff per IP oder NTLMv1.

3. Filtere nach Kategorie, OU und Zeitraum. So siehst du zum Beispiel, welche OU einen Webserver noch per IP-Adresse anspricht, also ein Fall für neue Lesezeichen, und welcher Server an einem fehlenden SPN hängt, also ein Fall für setspn.

4. Klick einen Server an. Rechts stehen Anmeldezahl und Befunde; in der Kommunikationsmatrix siehst du dazu die einzelnen Verbindungen zwischen Clients und Webzielen.

5. Speichere die Ansicht bei Bedarf als PNG für die Abstimmung mit dem Team, das die Webanwendung betreibt, und trag die Maßnahme im Maßnahmen-Board ein. Die Erfolgskontrolle zeigt dir später, ob die NTLM-Anmeldungen zu diesem Ziel wirklich verschwunden sind.



So siehst du auf der NTLM-Landkarte, welche Server und Clients je OU noch NTLM sprechen und aus welchem Grund (Demodaten)

Zum Einstieg reicht die kostenlose Express-Edition: Fünf-Minuten-Befund direkt vom DC, Readiness-Ampel, Abdeckungskarte, Top-10-Verursacher mit Hinweisen aus der Wissensbasis. Landkarte, Matrix und SPN-Prüfung probierst du im siebentägigen Testzeitraum aus, der einmal je Domäne verfügbar ist. Wenn du Webanwendungen, SPNs und Browserrichtlinien in einem Zug umstellen willst und dafür Unterstützung suchst, sprich mit uns über die Beratung zur NTLM-Ablösung. Für SharePoint gelten eigene Regeln, siehe SharePoint Server: Kerberos statt NTLM für Webanwendungen; Java-Webanwendungen mit SPNEGO behandelt Java-Anwendungen und NTLM: Kerberos mit SPNEGO.

FAQ: Browser, Intranet und Kerberos

Reicht es, die Seite in die Intranetzone einzutragen?

Für Edge und Chrome ohne eigene Allowlist reicht es, damit die Anmeldung automatisch erfolgt. Ob Kerberos dabei herauskommt, entscheidet der SPN. Für Firefox reicht es nicht, er nutzt die Windows-Zonen nicht.

Warum klappt es mit http://intranet, aber nicht mit dem vollständigen Namen?

Weil Windows Namen ohne Punkt automatisch der Intranetzone zuordnet, Namen mit Punkt dagegen der Internetzone, solange keine Zuordnung existiert. Ordne *.musterwerk.local der Zone 1 zu oder setz AuthServerAllowlist.

Muss ich in Firefox Kurznamen erlauben?

Nur, wenn noch Kurznamen in Gebrauch sind. Firefox lässt sie standardmäßig nicht zu. Mit AllowNonFQDN in der Richtlinie Authentication kannst du sie freigeben; besser ist der Umstieg auf vollständige Namen.

Gilt die Zonenzuordnung noch, wenn ich AuthServerAllowlist setze?

Für die Entscheidung über die automatische Anmeldung in Edge und Chrome nicht mehr: Dann zählt die Liste. Alles, was dort fehlt, bekommt einen Anmeldedialog, auch Kurznamen.

Kann ich NTLM im Browser einfach über AuthSchemes abschalten?

Technisch kannst du ntlm aus der Liste nehmen, aber das wirkt auf alle Seiten und Proxys und erfasst nicht unbedingt den Rückfall innerhalb von Negotiate. Für eine gezielte Steuerung sind die Restrict-NTLM-Richtlinien das richtige Werkzeug.

Warum bekommt der Browser für eine IP-Adresse kein Kerberos-Ticket?

Weil der SPN aus dem Namen gebildet wird und für eine IP-Adresse standardmäßig kein passender SPN existiert. Die Lösung ist fast immer ein DNS-Name in Lesezeichen und Verknüpfungen.

Woran erkenne ich, welcher Browser noch NTLM verwendet?

An den Ereignissen 4020/4021 auf dem Client: ProcessName nennt msedge.exe, chrome.exe oder firefox.exe, TargetMachine und TargetService das Ziel, NtlmUsageReason die Begründung.

Brauche ich AuthNegotiateDelegateAllowlist?

Nur, wenn eine Webanwendung deine Anmeldung an ein weiteres System durchreichen muss und das nicht per eingeschränkter Delegierung auf Serverseite gelöst ist. In den meisten Intranets bleibt sie leer.

Fazit: Der Browser ist nicht dumm, er hat nur keine Liste

Dass Browser bei internen Webanwendungen mal Kerberos, mal NTLM und mal einen Anmeldedialog liefern, ist kein Zufall, sondern die Folge von Faustregeln: Punkt im Namen oder nicht, Proxy oder nicht, Firefox oder nicht. Wer sich auf die Erkennung verlässt, bekommt genau

dieses Durcheinander. Wer es dem Browser ausdrücklich sagt, mit Zonenzuordnung, AuthServerAllowlist für Edge und Chrome und der Richtlinie Authentication für Firefox, bekommt ein vorhersagbares Verhalten.

Danach bleibt der SPN als zweite Hälfte: Ohne ihn hilft die schönste Allowlist nichts, und der Rückfall auf NTLM passiert lautlos. Miss deshalb vorher und nachher, und räum Lesezeichen mit IP-Adressen und Kurznamen auf, bevor der Stichtag es für dich tut. Wie du NTLM danach Schritt für Schritt einschränkst, steht in NTLM schrittweise abschalten, ohne dass etwas ausfällt, wie du Rückfälle erkennst, in Nach der Abschaltung: Rückfälle erkennen und NTLM-frei bleiben. Wenn dein Team Kerberos und die NTLM-Ereignisse künftig selbst sicher lesen will, schau in die Schulung zum NTLM Diagnostiker.

WEITERLESEN · Weiter in dieser Serie

› NTLM-Nutzung ermitteln: der komplette Leitfaden
› NTLM-Reason-IDs erklärt: warum Windows nicht Kerberos benutzt hat
› NTLM in IIS und Webanwendungen: Negotiate statt NTLM
› Fehlender SPN: wenn Kerberos den Dienst nicht findet
› CNAME, Aliasnamen und hosts-Datei: warum Kerberos den Namen nicht findet
› Mit klist prüfen, ob Kerberos wirklich greift
› SharePoint Server: Kerberos statt NTLM für Webanwendungen
› Java-Anwendungen und NTLM: Kerberos mit SPNEGO
› Microsoft Kerberos