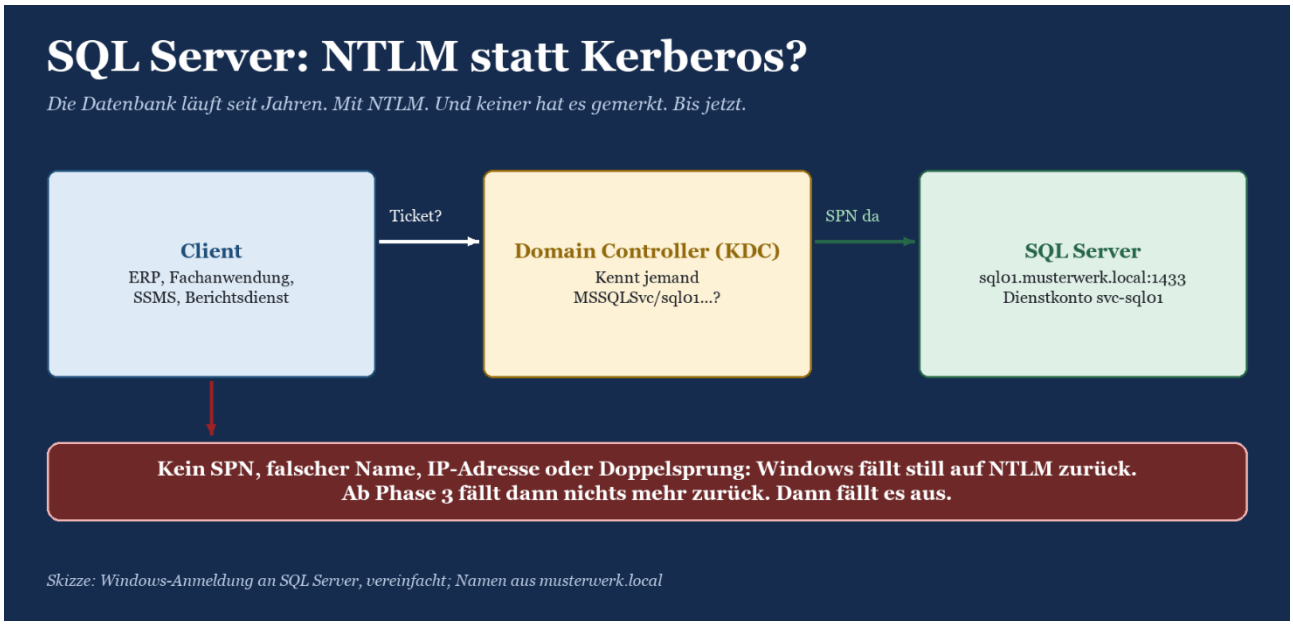
SQL Server und NTLM: Windows-Anmeldung auf Kerberos umstellen



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

BERATUNG
Zwanzig Instanzen, drei Dienstkonten, ein Verbindungsserver-Geflecht? Wir sortieren mit.
› **Beratung zur NTLM-Ablösung**

SCHULUNG
NTLM-Ereignisse lesen, SPNs prüfen, Abschaltung testen: praxisnah für dein Team.
› **Schulung zum NTLM Diagnostiker**

SQL Server ist der stille Riese in jeder NTLM-Auswertung. Die Warenwirtschaft, das Zeiterfassungssystem, das Laborinformationssystem, die Berichte der Controllerin: alles meldet sich per Windows-Anmeldung an der Datenbank an, alles funktioniert, und niemand fragt, womit. Die ehrliche Antwort lautet in vielen Häusern: mit NTLM. Nicht, weil jemand das so wollte, sondern weil der Service Principal Name fehlt, weil in der Verbindungszeichenfolge seit 2014 eine IP-Adresse steht oder weil die benannte Instanz bei jedem Neustart einen anderen Port würfelt.

Bis jetzt war das egal, denn Windows fällt bei Kerberos-Problemen geräuschlos auf NTLM zurück. Mit der nächsten Windows-Server-Version ist NTLM aber standardmäßig abgeschaltet, und dann wird aus dem geräuschlosen Rückfall ein sehr geräuschvoller Ausfall: Anmeldung fehlgeschlagen, mitten im Monatsabschluss. Wer nach sql server ntlm statt kerberos sucht, will genau das vorher wissen und beheben. In diesem Beitrag prüfst du mit sys.dm_exec_connections, welche Verbindungen NTLM nutzen, setzt die MSSQLSvc-SPNs für Standard- und benannte Instanzen richtig, kümmerst dich um Dienstkonto, gMSA und Port und räumst Verbindungszeichenfolgen, Verbindungsserver und Doppelsprünge auf.

Die Phasen der Ablösung und den Stichtag erklärt Der Fahrplan zur NTLM-Abschaltung. Warum ein fehlender SPN grundsätzlich zu NTLM führt, steht in Fehlender SPN: wenn Kerberos den Dienst nicht findet. Hier geht es nur um SQL Server.

FAKTEN · SQL Server und Kerberos in 30 Sekunden

In einer Active-Directory-Umgebung versucht die Windows-Anmeldung an SQL Server immer zuerst Kerberos. Bekommt der Client kein Dienstticket, nimmt er NTLM, ohne Fehlermeldung.

Das Ticket hängt am SPN MSSQLSvc/<FQDN>:<Port> für TCP, MSSQLSvc/<FQDN> für die Standardinstanz über andere Protokolle und MSSQLSvc/<FQDN>:<Instanzname> für benannte Instanzen über Named Pipes oder Shared Memory.

Der SPN muss genau dem Konto zugeordnet sein, unter dem der SQL-Server-Dienst läuft. Fehlt er, gibt es NTLM. Zeigt er auf das falsche Konto, scheitert die Anmeldung ganz.

SQL Server versucht beim Start, seinen SPN selbst zu registrieren, und beim Beenden, ihn wieder zu entfernen. Fehlt dem Dienstkonto das Recht dazu, steht eine Warnung im Fehlerprotokoll, und der Dienst startet trotzdem.

Ob eine Verbindung Kerberos nutzt, zeigt die Spalte auth_scheme in sys.dm_exec_connections: KERBEROS, NTLM oder SQL für SQL-Server-Anmeldungen.

Prüfen statt raten: sys.dm_exec_connections und auth_scheme

Bevor du an SPNs schraubst, solltest du wissen, wie schlimm es ist. Das geht erfreulich direkt, denn SQL Server führt für jede Verbindung Buch darüber, wie sie sich angemeldet hat. Die dynamische Verwaltungssicht sys.dm_exec_connections hat dafür die Spalte auth_scheme. Darin steht KERBEROS, NTLM oder SQL. Für die eigene Verbindung reicht ein Einzeiler, für die Bestandsaufnahme verknüpfst du die Sicht mit sys.dm_exec_sessions und siehst, welcher Rechner mit welchem Programm und welchem Konto per NTLM kommt.

```
-- Eigene Verbindung: Kerberos oder NTLM?
SELECT net_transport, auth_scheme
FROM sys.dm_exec_connections
WHERE session_id = @@SPID;

-- Bestandsaufnahme: wer kommt per NTLM?
SELECT s.host_name, s.program_name, s.login_name,
       c.client_net_address, c.local_tcp_port, c.net_transport,
       c.auth_scheme, COUNT(*) AS verbindungen
FROM sys.dm_exec_connections AS c
JOIN sys.dm_exec_sessions AS s ON s.session_id = c.session_id
WHERE c.auth_scheme = 'NTLM'
GROUP BY s.host_name, s.program_name, s.login_name,
         c.client_net_address, c.local_tcp_port, c.net_transport, c.auth_scheme
ORDER BY verbindungen DESC;
```

Für die Abfrage brauchst du VIEW SERVER STATE, ab SQL Server 2022 genügt VIEW SERVER PERFORMANCE STATE. Die Sicht ist eine Momentaufnahme: Sie zeigt nur, wer gerade verbunden ist. Die Fachanwendung, die sich einmal pro Nacht für den Datenabgleich anmeldet, siehst du tagsüber nicht. Wer es genauer will, lässt die Abfrage per Agent-Auftrag alle fünfzehn Minuten in eine Tabelle schreiben und schaut nach ein bis zwei Wochen hinein. Das ist kein Hexenwerk, aber eben eine Tabelle pro Instanz, und bei zwanzig Instanzen wird daraus ein Hobby.

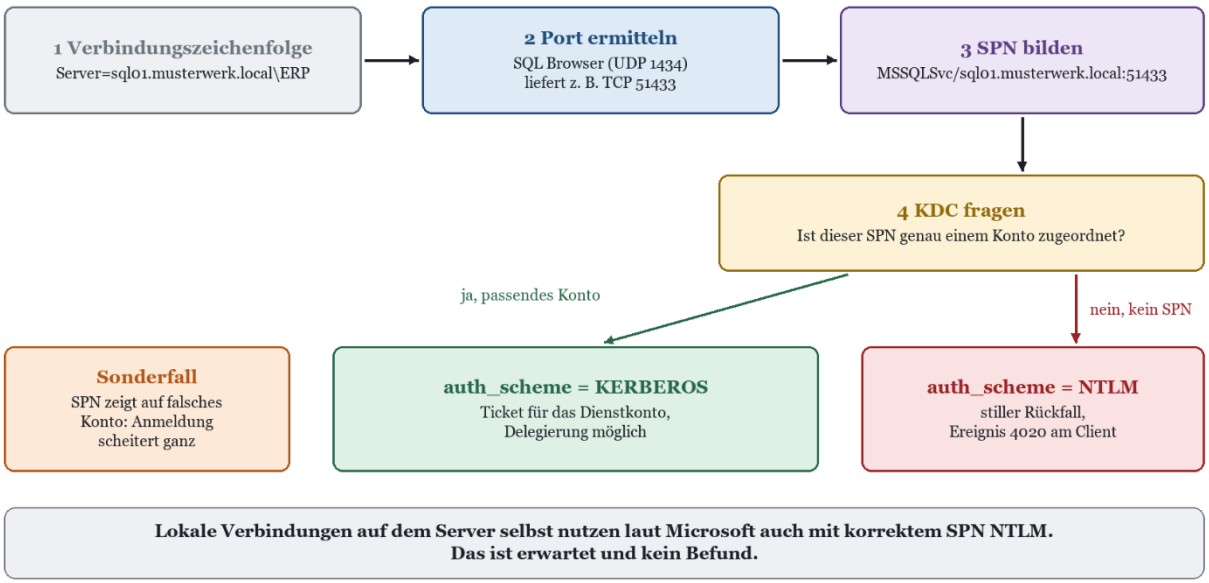
So liest du das Ergebnis

auth_scheme	Bedeutung	Was du tust
KERBEROS	Dienstticket erfolgreich, SPN passt	nichts, freuen
NTLM, Client ist der Server selbst	lokale Verbindung, laut Microsoft auch mit korrektem SPN NTLM	erwartet, kein Befund; aber im Hinterkopf behalten
NTLM, entfernter Client	kein Ticket: SPN fehlt, falscher Name, IP, Alias oder Vertrauensproblem	SPN prüfen, dann Verbindungszeichenfolge
SQL	SQL-Server-Anmeldung mit	nicht NTLM, aber eigene Baustelle

	Benutzername und Kennwort	
Anmeldung scheitert ganz	SPN zeigt auf ein falsches Konto	doppelten oder verwaisten SPN suchen

Wie der Client den SPN baut, und wo er abbiegt

Ohne ausdrücklich angegebenen SPN setzt der Treiber ihn aus Protokoll, Servername und Instanz zusammen



Skizze nach „Register a Service Principal Name for Kerberos Connections" (Microsoft Learn)

TIPP · Die Gegenprobe kommt vom Client

Auf dem Client zeigt dir klist, ob ein Ticket für MSSQLSvc/sql01.musterwerk.local:1433 im Cache liegt. Fehlt es, obwohl die Anwendung gerade verbunden ist, war es NTLM. Auf Windows 11 ab 24H2 und Windows Server 2025 schreibt der Client außerdem Ereignis 4020 oder 4021 ins Protokoll Microsoft-Windows-NTLM/Operational, mit ProcessName, TargetService und dem Begründungstext in NtlmUsageReason. Maßgeblich ist der Text, nicht die Reason-ID-Nummer; die Nummer ist nur ein Hinweis.

Wie du das am Client sauber prüfst, steht in Mit klist prüfen, ob Kerberos wirklich greift, die Bedeutung der Begründungstexte in NTLM-Reason-IDs verstehen.

MSSQLSvc richtig registrieren: Instanz, Port und Dienstkonto

Der SPN ist das Klingelschild der Instanz im Active Directory. Der Client baut sich den Namen aus Protokoll, Servername und Instanz zusammen, fragt beim Domain Controller nach einem Ticket für genau diesen Namen und bekommt eins, wenn der SPN genau einem Konto gehört. Steht auf dem Klingelschild etwas anderes, als der Client sucht, klingelt er nicht, sondern ruft NTLM an.

Instanz und Protokoll	SPN-Format	Beispiel musterwerk.local
Standardinstanz, TCP	MSSQLSvc/<FQDN>:<Port>	MSSQLSvc/sql01.musterwerk.local:1433
Standardinstanz, Named Pipes oder Shared Memory	MSSQLSvc/<FQDN>	MSSQLSvc/sql01.musterwerk.local
Benannte Instanz, TCP	MSSQLSvc/<FQDN>:<Port>	MSSQLSvc/sql02.musterwerk.local:51433
Benannte Instanz, Named Pipes oder	MSSQLSvc/<FQDN>:<Instanzname>	MSSQLSvc/sql02.musterwerk.local:ERP

Shared Memory		
Verfügbarkeitsgruppe, Listener	MSSQLSvc/<Listener-FQDN>:<Port>	MSSQLSvc/erp-ag.musterwerk.local:1433

Wichtig ist der Unterschied bei TCP: Standard- und benannte Instanzen werden beide als MSSQLSvc registriert und nur über den Port unterschieden. Der Instanzname taucht im TCP-SPN nicht auf. Damit ein SPN mit Port überhaupt greift, muss TCP an der Instanz aktiviert sein. Die vollständige Formatliste dokumentiert Microsoft in Register a Service Principal Name for Kerberos Connections.

Erst nachsehen, dann setzen

Drei Fragen klärst du vor jeder Änderung: Unter welchem Konto läuft der Dienst? Welche SPNs hat dieses Konto? Gibt es den SPN schon woanders? Das Konto steht im SQL Server Konfigurations-Manager oder in der Diensteverwaltung, den Rest erledigt setspn. Das Fehlerprotokoll der Instanz verrät dir außerdem, ob die automatische Registrierung beim letzten Start geklappt hat.

```
REM Welche SPNs hat das Dienstkonto?
setspn -L MUSTERWERK\svc-sql01

REM Gibt es den SPN irgendwo in der Gesamtstruktur?
setspn -Q MSSQLSvc/sql01.musterwerk.local*

REM Doppelte SPNs suchen
setspn -X

-- Im SQL Server: hat die automatische Registrierung geklappt?
EXEC xp_readerrorlog 0, 1, N'Service Principal Name';
```

Steht im Fehlerprotokoll, dass die Registrierung fehlgeschlagen ist (sinngemäß „could not register the Service Principal Name“), hat das Dienstkonto keine Rechte am eigenen SPN. Steht dort „successfully registered“, ist der SPN zumindest beim Start angelegt worden. Ob er zum Port passt, den der Client gerade anspricht, ist eine zweite Frage.

Dienstkonto, virtuelles Konto, gMSA: wer trägt den SPN?

Der SPN gehört an das Konto, das den Dienst startet. Das klingt banal und ist trotzdem die häufigste Fehlerquelle, denn das Dienstkonto wechselt im Leben einer Instanz öfter, als das Klingelschild umgeschraubt wird.

Dienstkonto	SPN liegt am	Selbstregistrierung	Typischer Stolperstein
Virtuelles Konto NT SERVICE\MSSQLSERVER	Computerkonto SQL01$	ja	Wechsel auf Domänenkonto, alter SPN bleibt am Computerkonto
NETWORK SERVICE	Computerkonto SQL01$	ja	wie oben
Domänenbenutzerkonto svc-sql01	Benutzerkonto svc-sql01	nur mit Rechten auf servicePrincipalName	Recht fehlt, SPN wird nie angelegt
Gruppenverwaltetes Dienstkonto gmsa-sql$	gMSA-Objekt	nur mit Schreibrecht am eigenen SPN	SPN steht noch am alten Domänenkonto, Duplikat
Domänenadministrator	Benutzerkonto	ja	bitte nicht. Wirklich nicht.

Läuft die Instanz unter einem Domänenkonto oder einer gMSA, hast du zwei saubere Wege. Entweder bekommt das Konto die Rechte „Read servicePrincipalName“ und „Write

servicePrincipalName“ beziehungsweise „Validated write to service principal name“, dann registriert SQL Server bei jedem Start selbst. Oder du registrierst den SPN einmal von Hand. Für Instanzen mit festem Port ist der manuelle Weg übersichtlicher, weil nichts mehr kommt und geht, ohne dass du es weißt.

```
REM Standardinstanz, TCP-Port 1433, Domänendienstkonto
setspn -S MSSQLSvc/sql01.musterwerk.local:1433 MUSTERWERK\svc-sql01
setspn -S MSSQLSvc/sql01.musterwerk.local MUSTERWERK\svc-sql01

REM Benannte Instanz ERP, fester Port 51433, gMSA
setspn -S MSSQLSvc/sql02.musterwerk.local:51433 MUSTERWERK\gmsa-sql$
setspn -S MSSQLSvc/sql02.musterwerk.local:ERP MUSTERWERK\gmsa-sql$
```

WARNUNG · Doppelt ist schlimmer als gar nicht

Fehlt der SPN, gibt es NTLM. Hängt derselbe SPN an zwei Konten, kann der Domain Controller kein eindeutiges Ticket ausstellen, und die Anmeldung scheitert. Zeigt der SPN auf ein Konto, unter dem der Dienst gar nicht mehr läuft, scheitert sie ebenfalls, weil der Dienst das Ticket nicht entschlüsseln kann. Nimm deshalb setspn -S statt -A: -S prüft vorher auf Duplikate. Und lösche den alten SPN mit setspn -D, bevor du ihn an ein neues Konto hängst.

Dynamische Ports: das bewegliche Klingelschild

Benannte Instanzen starten standardmäßig mit dynamischem Port. Der SQL Browser sagt dem Client, welcher es gerade ist, und der Client baut daraus seinen SPN. Hat SQL Server das Recht zur Selbstregistrierung, zieht er den SPN beim Neustart mit. Hat er es nicht und du hast den SPN einmal von Hand gesetzt, stimmt er genau bis zum nächsten Portwechsel. Danach ist wieder NTLM, und keiner weiß, warum es vorige Woche ging. Gib benannten Instanzen im Konfigurations-Manager einen festen Port, dann passt der SPN dauerhaft, und die Firewall-Freunde freuen sich auch.

Verbindungszeichenfolgen: IP-Adressen, Kurznamen und Aliasse

Ein perfekt registrierter SPN nützt nichts, wenn der Client nach einem anderen Namen sucht. Und das tut er öfter, als dir lieb ist, denn Verbindungszeichenfolgen sind die Zeitkapseln der IT: Irgendwer hat sie bei der Einführung eingetragen, und seitdem hat sie niemand mehr angefasst.

Was in der Verbindungszeichenfolge steht, entscheidet mit

Dieselbe Datenbank, fünf Schreibweisen, sehr unterschiedliche Chancen auf Kerberos

Server=	SPN, den der Client sucht	Erwartung
sql01.musterwerk.local	MSSQLSvc/sql01.musterwerk.local:1433	Kerberos, wenn SPN da
sql01	Treiber ergänzt per DNS zum FQDN	meist Kerberos, DNS prüfen
10.20.0.30	kein sauberer Name, kein SPN	praktisch immer NTLM
erpdb (DNS-CNAME)	je nach Treiber Alias oder Ziel	prüfen, ggf. Alias-SPN
sql01\ERP, Port dynamisch	MSSQLSvc/...:<aktueller Port>	bricht nach Portwechsel
localhost / (local)	lokale Verbindung	NTLM, erwartet

Skizze: typisches Verhalten, Treiber unterscheiden sich; maßgeblich ist auth_scheme

IP-Adresse im Verbindungsstring

Steht Server=10.20.0.30 in der Konfiguration, hat der Client keinen Namen, aus dem er einen MSSQLSvc-SPN mit FQDN bauen könnte. Ob und wie ein Treiber aus der IP-Adresse doch noch einen Namen macht, ist treiberabhängig, und darauf solltest du dich nicht verlassen. In der Praxis landet so eine Verbindung bei NTLM. Die Lösung ist langweilig und wirksam: den FQDN eintragen. Dafür musst du die Stelle aber erst finden, und die steckt gern in einer INI-Datei, einer ODBC-Datenquelle, einer UDL-Datei oder in der Registrierung der Fachanwendung.

DNS-CNAME und SQL-Client-Alias

Aliasnamen sind beliebt, weil man damit Server tauschen kann, ohne hundert Clients anzufassen. Für Kerberos sind sie eine Prüfaufgabe: Je nach Treiber und Auflösung baut der Client den SPN mit dem Aliasnamen oder mit dem kanonischen Namen, auf den der CNAME zeigt. Teste es mit auth_scheme und klist. Sucht der Client MSSQLSvc/erpdb.musterwerk.local:1433, registrierst du diesen SPN zusätzlich am Dienstkonto. Ein Client-Alias aus dem SQL Server Konfigurations-Manager, der auf eine IP-Adresse zeigt, ist dagegen nur eine versteckte IP-Adresse. Die ganze Alias-Problematik jenseits von SQL Server erklärt CNAME, Aliasnamen und hosts-Datei: warum Kerberos den Namen nicht findet.

TIPP · Der Client darf den SPN auch ausdrücklich nennen

Aktuelle Microsoft-Treiber erlauben, den SPN in der Verbindungszeichenfolge anzugeben, etwa mit dem Schlüsselwort ServerSPN beim ODBC- und OLE-DB-Treiber oder Server SPN bei Microsoft.Data.SqlClient. Ein angegebener SPN wird unverändert verwendet. Das ist praktisch für Aliasse und Listener, aber kein Ersatz für einen registrierten SPN: Der muss trotzdem am richtigen Konto hängen. Prüfe die genaue Schreibweise in der Dokumentation deines Treibers.

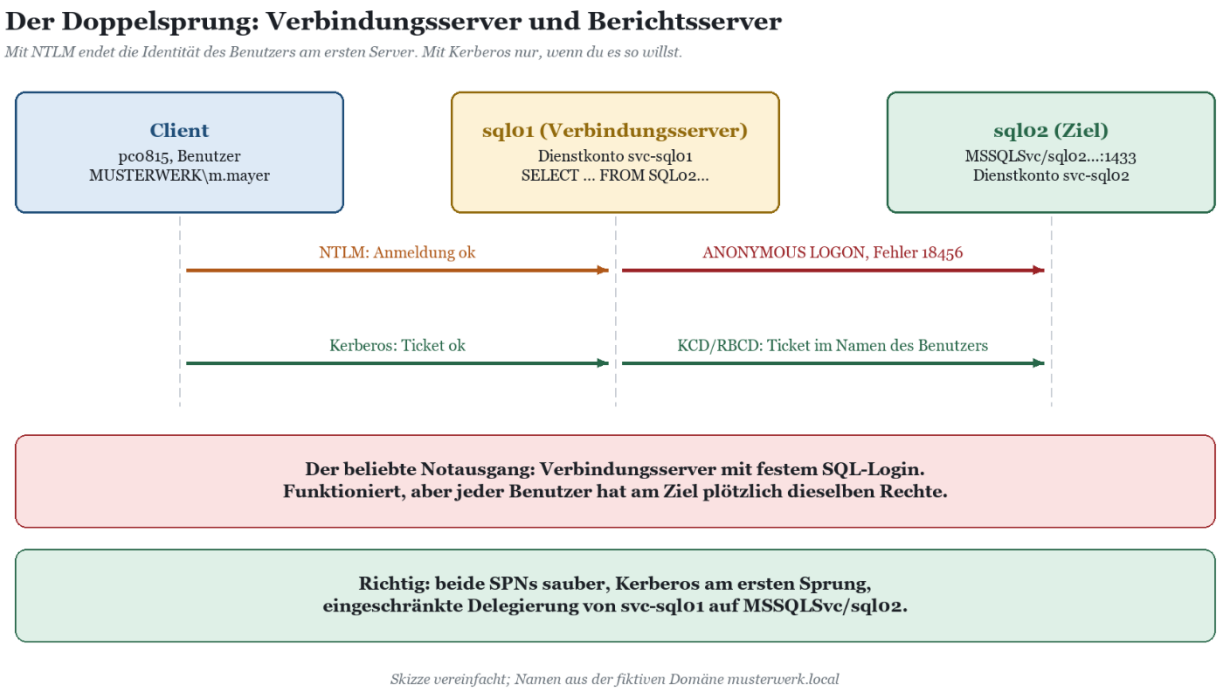
AUS DER PRAXIS · Sparfuchs & Partner und der Server, der umgezogen ist

Die Steuerkanzlei Sparfuchs & Partner hat ihre Kanzleisoftware vor drei Jahren auf einen neuen SQL Server gezogen. Damit niemand umkonfigurieren muss, zeigt seitdem der alte Name als CNAME auf

den neuen. Alles lief. Mit NTLM. Der SPN existierte nur für den neuen Namen, die Clients fragten aber nach dem alten. Ein zusätzlicher SPN am Dienstkonto, ein Neustart der Anwendung, und auth_scheme zeigte KERBEROS. Gesamtaufwand zwanzig Minuten. Gesuchte Zeit vorher: drei Jahre, nur hatte keiner gesucht.

Verbindungsserver und Doppelsprung: wenn NTLM nach der ersten Station endet

Jetzt wird es spannend, denn hier ist NTLM nicht nur unschön, sondern funktional begrenzt. Meldet sich ein Benutzer per NTLM an sql01 an und fragt dort über einen Verbindungsserver Daten von sql02 ab, kann sql01 die Identität des Benutzers nicht weiterreichen. NTLM kennt keine Delegierung. Am Ziel kommt dann NT AUTHORITY\ANONYMOUS LOGON an, quittiert mit Fehler 18456. Dasselbe Muster siehst du bei Reporting Services, bei Webanwendungen vor der Datenbank und bei SharePoint, dazu mehr in SharePoint Server: Kerberos statt NTLM für Webanwendungen.



Weil das schon immer so war, haben viele Umgebungen einen Notausgang gebaut: Der Verbindungsserver meldet sich mit einem festen SQL-Login oder einem Sammelkonto am Ziel an. Das funktioniert, hat aber zwei Nebenwirkungen. Jeder Benutzer hat am Ziel dieselben Rechte, und im Ziel-Audit steht nur noch das Sammelkonto. Der saubere Weg sieht so aus:

- **Erster Sprung mit Kerberos:** Der Client muss sich per Kerberos an sql01 anmelden, sonst gibt es nichts zu delegieren. auth_scheme muss KERBEROS zeigen.
- **SPN am Ziel:** MSSQLSvc/sql02.musterwerk.local:1433 hängt am Dienstkonto von sql02.
- **Eingeschränkte Delegierung:** Das Dienstkonto von sql01 darf an MSSQLSvc/sql02 delegieren, als klassische eingeschränkte Delegierung (KCD) am Konto von sql01 oder ressourcenbasiert (RBCD) am Konto von sql02. Unbeschränkte Delegierung ist keine Option.
- **Verbindungsserver mit Selbstzuordnung:** Der Verbindungsserver nutzt den Sicherheitskontext des angemeldeten Benutzers statt eines festen Logins.
- **Testen mit echten Benutzern:** Nicht als sysadmin, nicht auf dem Server selbst, sondern vom Arbeitsplatz eines Sachbearbeiters aus.

WICHTIG · Konten über Domänengrenzen

Kommt der Client aus einer anderen Domäne oder Gesamtstruktur, entscheidet auch die Vertrauensstellung, ob Kerberos klappt. Delegierung über Gesamtstrukturgrenzen hat eigene Regeln.

Die Grundlagen stehen in NTLM über Vertrauensstellungen: Gesamtstrukturen, Domänen und Kerberos. Wenn Anmeldungen am Ziel mit Statuscodes scheitern, hilft Fehlercodes 0xC000006D und Co.: fehlgeschlagene NTLM-Anmeldungen deuten.

Wo es mühsam wird

Für eine Instanz ist das alles in einer Stunde erledigt. Im echten Mittelstand sind es aber fünfzehn Instanzen auf neun Servern, drei davon benannt, zwei mit dynamischem Port, eine Verfügbarkeitsgruppe und ein Verbindungsserver-Geflecht, das ein Dienstleister 2017 gebaut hat. auth_scheme sagt dir nur, was gerade verbunden ist, und nur auf dem Server, den du gerade abfragst. Die Ereignisse 4020 und 4021 liegen auf den Clients verteilt, Tausende davon, und zeigen dir zwar TargetService, aber nicht, ob der passende SPN fehlt, doppelt ist oder am falschen Konto hängt. Dafür musst du jeden Zielnamen gegen das Active Directory halten. Mit setspn -Q, einzeln.

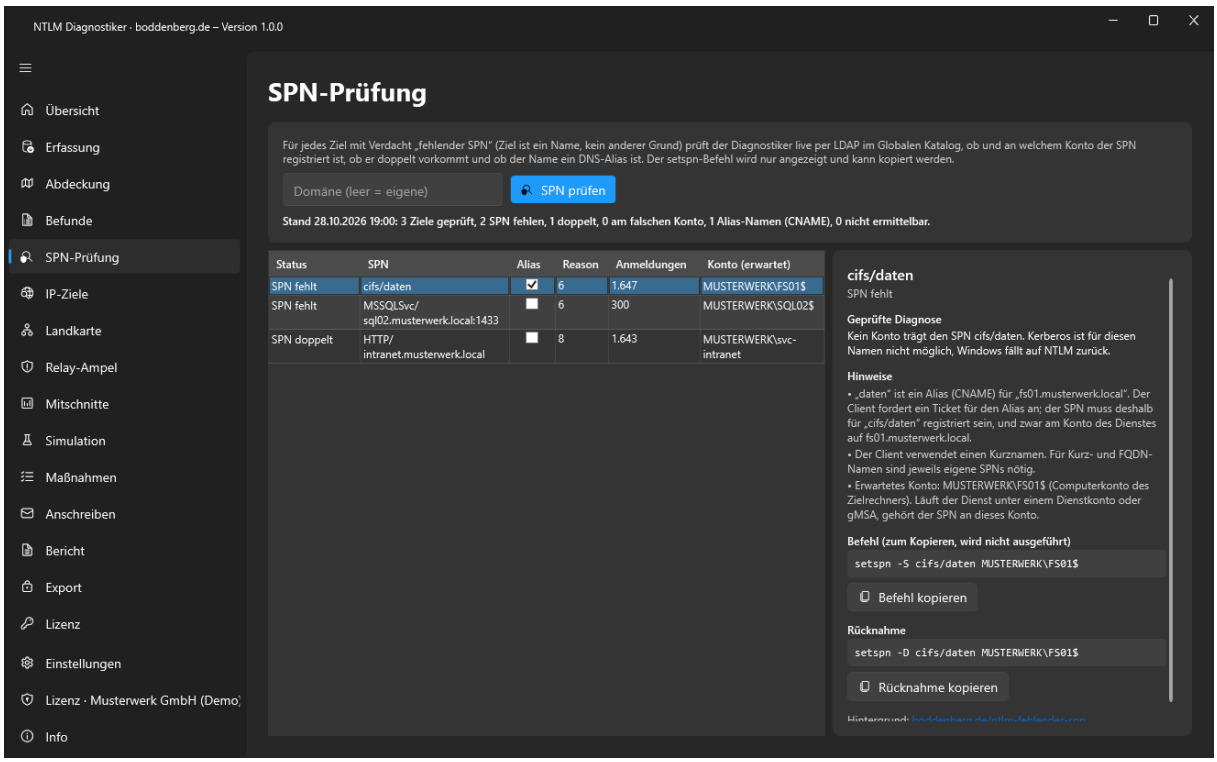
Genau diese Fleißarbeit nimmt dir der NTLM Diagnostiker ab. Er arbeitet ausschließlich lesend, sammelt die NTLM-Ereignisse direkt vom Domain Controller, über einen Windows Event Collector, per Direktabfrage oder aus importierten Dateien und fasst gleichartige Verbindungen nach Ursache, Prozess und Ziel zusammen. Aus 4.000 Einzelereignissen wird so eine Zeile: erp.exe auf 212 Clients, Ziel MSSQLSvc/sql02, Ursache fehlender SPN.

SO GEHT'S MIT DEM NTLM DIAGNOSTIKER · SPN-Prüfung für MSSQLSvc

Die SPN-Prüfung nimmt jedes Ziel, das in den NTLM-Ereignissen auftaucht, und hält es gegen das Active Directory. Für jeden Eintrag siehst du den Status, also SPN fehlt oder SPN doppelt, den SPN, den der Client gesucht hat, die Zahl der NTLM-Anmeldungen dagegen und das Konto, an dem der SPN erwartet wird. Fehlende MSSQLSvc-SPNs fallen damit auf, ohne dass du jede Instanz einzeln abklapperst.

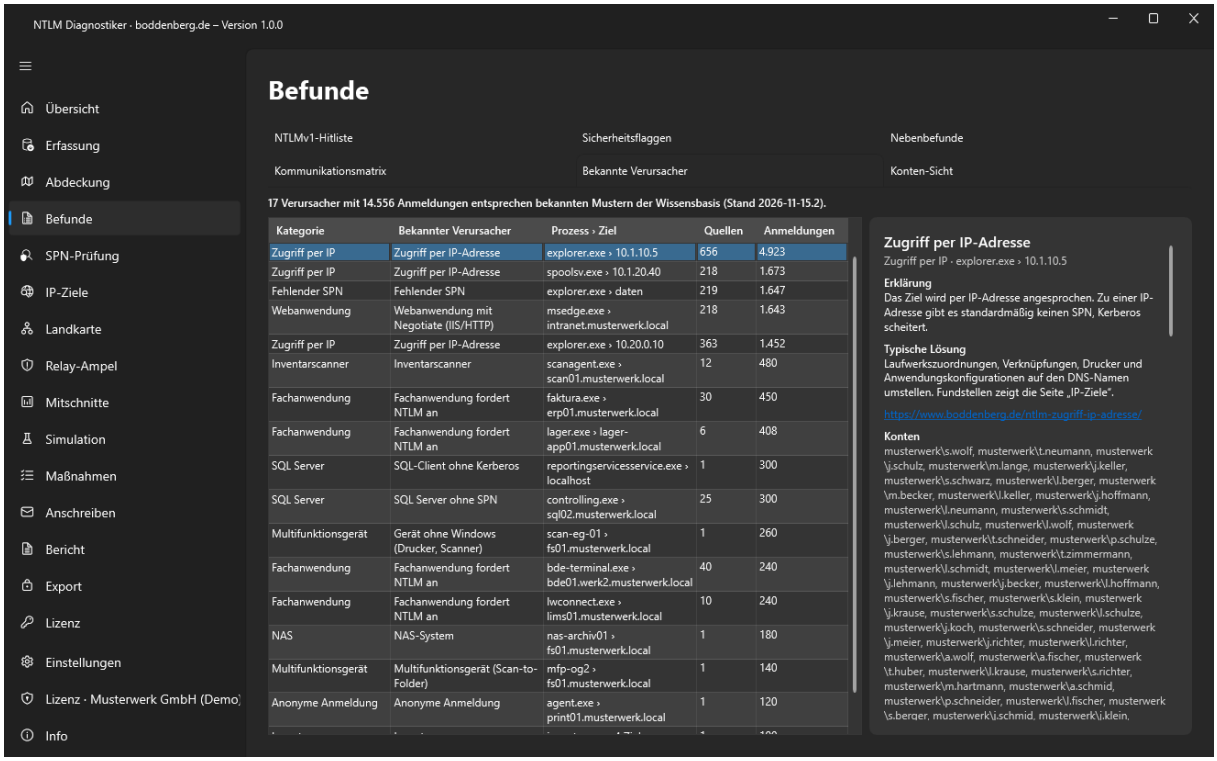
Ist das Ziel ein Aliasname, setzt die Prüfung ein Häkchen in der Spalte Alias und zeigt dir rechts den Hinweis, dass der Alias einen eigenen SPN am Konto des eigentlichen Servers braucht. Genau das Szenario der Kanzlei von oben. Im Bild ist das ein Dateiserver-Alias (cifs/daten mit erwartetem Konto MUSTERWERK\FS01$); für einen SQL-Alias sieht der Eintrag genauso aus, nur mit MSSQLSvc und dem Dienstkonto der Instanz.

Rechts bietet dir der Diagnostiker den passenden Befehl setspn -S zum Kopieren an und gleich daneben die Rücknahme mit setspn -D. Ausführen tust du ihn selbst. Der Diagnostiker ändert nichts an deinem Active Directory.



So erkennst du in der SPN-Prüfung, dass ein Aliasname einen eigenen SPN braucht, und kopierst den passenden setspn-Befehl samt Rücknahme (Demodaten).

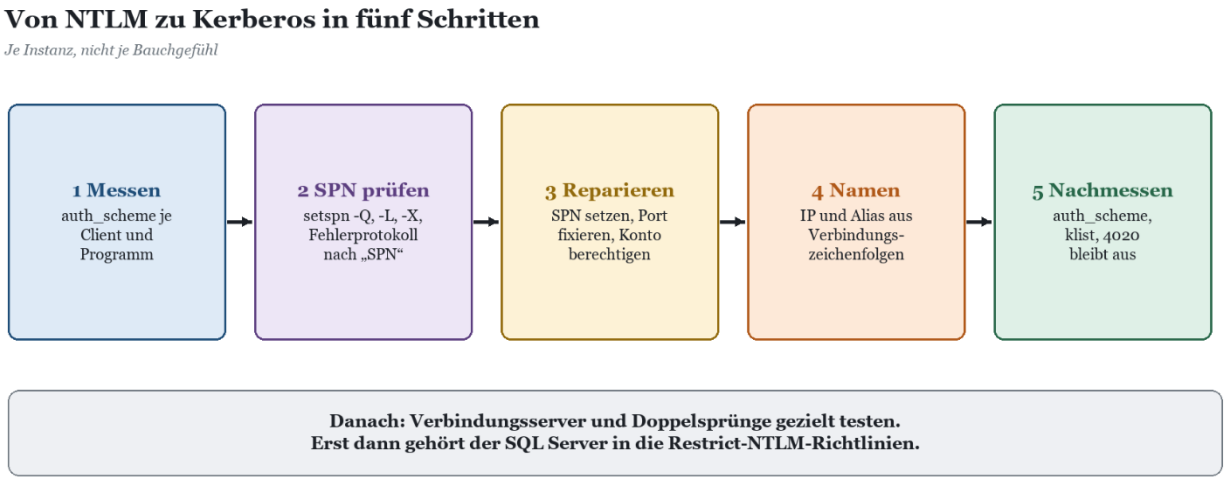
Welche Anwendung hinter den SQL-Verbindungen steckt, zeigen die Befunde: Die Wissensbasis bekannter Verursacher erkennt Muster wie „SQL Server ohne SPN" und liefert Erklärung, typische Lösung und die betroffenen Konten mit. Die Ursachen-Engine wertet dafür den Begründungstext in NtlmUsageReason aus, nicht nur die Nummer.



So findest du in den Befunden Verursacher wie „SQL Server ohne SPN" samt Erklärung, Lösung und betroffenen Konten (Demodaten).

Readiness-Ampel, Abdeckungskarte, Top-10-Verursacher und Hinweise aus der Wissensbasis bekommst du in der kostenlosen Express-Edition als Fünf-Minuten-Befund direkt vom Domain Controller. Die SPN-Prüfung testest du im siebentägigen Testzeitraum, einmal pro Domäne. Wenn du die Umstellung deiner SQL-Landschaft lieber gemeinsam planst, inklusive Delegierung und Verfügbarkeitsgruppen, ist die Beratung zur NTLM-Ablösung der richtige Ansprechpartner.

Umstellen ohne Montagmorgen-Drama: Reihenfolge und Erfolgskontrolle



Skizze: Reihenfolge beispielhaft für die Musterwerk GmbH

Die Reihenfolge ist wichtiger als die Geschwindigkeit. Ein SPN, der fehlt, kostet dich NTLM. Ein SPN, der auf das falsche Konto zeigt, kostet dich die Anmeldung. Deshalb gilt: erst messen, dann jeden SPN gegen das tatsächliche Dienstkonto prüfen, dann ändern, und zwar zu einer Zeit, in der ein Fehler niemanden um seinen Monatsabschluss bringt.

Schritt	Werkzeug	Erfolgskriterium
NTLM-Verbindungen je Instanz erfassen	sys.dm_exec_connections, Ereignisse 4020/4021	Liste aus Client, Programm, Konto, Ziel
Dienstkonto und vorhandene SPNs prüfen	setspn -L, -Q, -X, Fehlerprotokoll	kein Duplikat, SPN am Dienstkonto
Port fixieren, SPN setzen oder Rechte vergeben	Konfigurations-Manager, setspn -S, AD-Rechte	Fehlerprotokoll meldet erfolgreiche Registrierung
IP und Alias in Verbindungszeichenfolgen ersetzen	Konfiguration der Anwendung, ODBC, UDL	FQDN oder registrierter Alias
Delegierung für Verbindungsserver einrichten	KCD oder RBCD	kein ANONYMOUS LOGON am Ziel
Nachmessen	auth_scheme, klist, NTLM-Ereignisse	remote nur noch KERBEROS

WARNUNG · Nicht vergessen: die Dienste rund um die Datenbank

Reporting Services, Analysis Services und Integration Services haben eigene SPNs (HTTP für den Berichtsserver, MSOLAPSvc.3 für Analysis Services) und eigene Dienstkonten. Backup-Software und Überwachungsagenten melden sich ebenfalls an der Datenbank an, oft per IP-Adresse aus einer

Konfiguration, die seit der Erstinstallation niemand angesehen hat. Prüfe sie mit.

Agenten als NTLM-Quelle behandelt Backup-, Monitoring- und Inventaragenten als NTLM-Quelle. Erst wenn auth_scheme für entfernte Clients nur noch KERBEROS zeigt, nimmst du den SQL Server in die Restrict-NTLM-Richtlinien auf, wie in Die Restrict-NTLM-Richtlinien richtig einsetzen beschrieben. Dass danach nichts zurückfällt, überwachst du wie in Nach der Abschaltung: Rückfälle erkennen und NTLM-frei bleiben.

FAQ zu SQL Server, NTLM und Kerberos

Wie sehe ich, ob SQL Server NTLM statt Kerberos nutzt?

Mit SELECT auth_scheme FROM sys.dm_exec_connections. Für entfernte Windows-Anmeldungen sollte dort KERBEROS stehen. NTLM bei einem entfernten Client heißt: Der Client hat kein Dienstticket bekommen. Verknüpft mit sys.dm_exec_sessions siehst du auch Rechner, Programm und Konto.

Warum zeigt meine Verbindung auf dem Server selbst NTLM?

Weil lokale Verbindungen laut Microsoft auch bei korrekt registriertem SPN NTLM verwenden. Teste deshalb immer von einem anderen Rechner aus.

Braucht eine benannte Instanz einen anderen SPN als die Standardinstanz?

Für TCP nicht im Format: Beide nutzen MSSQLSvc/<FQDN>:<Port>, die Instanz wird über den Port unterschieden. Für Named Pipes und Shared Memory bekommt die benannte Instanz MSSQLSvc/<FQDN>:<Instanzname>, die Standardinstanz MSSQLSvc/<FQDN>.

Registriert SQL Server seinen SPN nicht selbst?

Doch, beim Start, wenn das Dienstkonto es darf. Virtuelle Konten, NETWORK SERVICE und verwaltete Dienstkonten können das. Ein normales Domänenkonto braucht die Rechte zum Lesen und Schreiben von servicePrincipalName. Scheitert die Registrierung, steht eine Warnung im Fehlerprotokoll, und der Dienst startet trotzdem, mit NTLM für alle.

Funktioniert Kerberos mit einer gMSA als Dienstkonto?

Ja. Der SPN gehört dann an das gMSA-Objekt. Entweder gibst du der gMSA das Recht, ihren SPN selbst zu schreiben, oder du registrierst ihn von Hand. Achte beim Wechsel auf die gMSA darauf, den SPN vom alten Konto zu entfernen, sonst ist er doppelt.

Was passiert mit Verbindungen per IP-Adresse, wenn NTLM abgeschaltet ist?

Sie scheitern, sofern der Treiber keinen passenden Namen findet. Ersetze die IP-Adresse durch den FQDN. Allgemeine Strategien für Dienstkonten findest du in Dienstkonten und NTLM.

Mein Verbindungsserver liefert ANONYMOUS LOGON. Ist das NTLM?

In aller Regel ja, genauer: Der erste Sprung lief per NTLM, oder es fehlt die Delegierung. Prüfe auth_scheme am ersten Server, den SPN am Ziel und die eingeschränkte Delegierung des Dienstkontos.

Hilft das auch bei Java- oder Linux-Clients?

Das Prinzip ist dasselbe, die Konfiguration nicht: JDBC-Treiber brauchen Kerberos-Einstellungen und oft eine krb5-Konfiguration. Mehr dazu in Java-Anwendungen und NTLM: Kerberos mit SPNEGO.

Fazit: Der SQL Server kann Kerberos längst. Man muss ihn nur lassen.

SQL Server ist kein NTLM-Problem, sondern ein Namensproblem mit NTLM als Symptom. Die Datenbank kann Kerberos seit Ewigkeiten, sie braucht nur ein Klingelschild, das zum Dienstkonto passt, einen Port, der still hält, und Clients, die nach dem richtigen Namen fragen. Wer auth_scheme abfragt, setspn ernst nimmt, IP-Adressen aus den Verbindungszeichenfolgen wirft und den Verbindungsservern eine echte Delegierung gönnt, hat den größten NTLM-Brocken im Haus in wenigen Tagen erledigt.

Und wenn die nächste Windows-Server-Version NTLM standardmäßig abdreht, steht deine Warenwirtschaft nicht am Abgrund, sondern einfach weiter im Rechenzentrum. Langweilig. Genau so soll es sein. Wer tiefer in die Kerberos-Seite einsteigen will, findet die Grundlagen unter Microsoft Kerberos.

WEITERLESEN · Weiter in dieser Serie

› 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
› NTLM-Ereignisse mit PowerShell auswerten: Get-WinEvent-Rezepte
› Dienstkonten und NTLM
› Checkliste vor dem NTLM-Stichtag: 25 Prüfpunkte