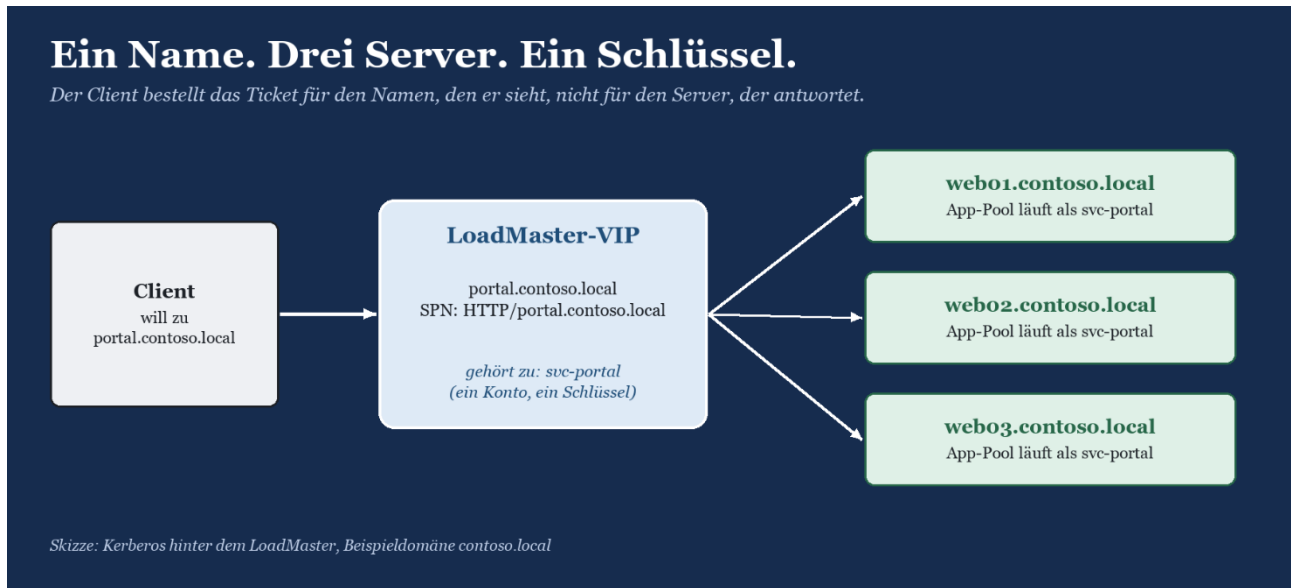


SPN-Probleme hinter dem Load Balancer: ein Name, mehrere Server



WISSEN

Alles zum Kemp LoadMaster mit Microsoft-Brille: Exchange, SharePoint, ADFS, RDS und Kerberos an einem Ort.

› [Kemp LoadMaster](#)

BERATUNG

Kerberos hinter dem Load Balancer hakt, und keiner weiß, warum? Wir gehen SPNs, Konten und Namen gemeinsam durch.

› [Beratung zum Kemp LoadMaster](#)

SCHULUNG

Grundlagen und Praxis für dein Admin-Team: vom ersten Virtual Service bis zur sauberen Kerberos-Konfiguration.

› [Kemp-LoadMaster-Schulung](#)

Es gibt Fehler, die schreien, und Fehler, die flüstern. SPN-Probleme hinter einem Load Balancer gehören zur zweiten Sorte. Niemand bekommt eine rote Fehlerseite. Stattdessen fragt Outlook gelegentlich nach dem Kennwort, das Intranet lädt an manchen Tagen zäh, und die Delegation an den SQL Server im Hintergrund klappt montags, aber nicht dienstags. Das Team zuckt mit den Schultern, startet den App-Pool neu, und weil Persistenz die Clients nach dem Neustart neu verteilt, ist das Problem scheinbar weg. Bis es wiederkommt. Es kommt immer wieder.

Die Ursache ist fast immer dieselbe: Kerberos wurde für einen Server gebaut, nicht für eine Fassade. Sobald ein Progress Kemp LoadMaster einen gemeinsamen Namen wie portal.contoso.local auf mehrere Real Server verteilt, muss jeder dieser Server ein Ticket entschlüsseln können, das für genau diesen gemeinsamen Namen ausgestellt wurde. Das klappt nur, wenn der Service Principal Name am richtigen Konto hängt und alle Server dieses Konto teilen. Klingt banal. Ist es auch. Trotzdem ist es der häufigste Grund, warum Kerberos hinter dem Load Balancer heimlich zu NTLM wird.

In diesem Beitrag geht es um genau dieses Zusammenspiel: warum der SPN zum VIP-Namen gehört und nicht zu den Servernamen, wie Exchange das mit dem Alternate Service Account löst, wie SharePoint und eigene IIS-Anwendungen es mit einem gemeinsamen Dienstkonto lösen, und warum doppelte SPNs und der Fehler KRB_AP_ERR_MODIFIED so oft zusammen auftreten. Am Ende hast du einen Diagnosepfad mit setspn, klist und dem Ereignisprotokoll, der ohne Raten auskommt. Die Grundlagen zu Tickets, KDC und Verschlüsselungstypen findest du im

Wissensbereich [Microsoft Kerberos](#); den Überblick über alle LoadMaster-Themen auf der Seite [Kemp LoadMaster](#).

FAKTEN · Kerberos hinter dem Load Balancer in drei Sätzen

Der Client bestellt beim KDC ein Ticket für den Namen, den er in der URL sieht, also für den VIP-Namen. Der KDC verschlüsselt das Ticket mit dem Schlüssel des Kontos, an dem dieser SPN registriert ist. Jeder Real Server, der die Anfrage bekommt, muss genau diesen Schlüssel kennen, sonst scheitert die Anmeldung mit KRB_AP_ERR_MODIFIED.

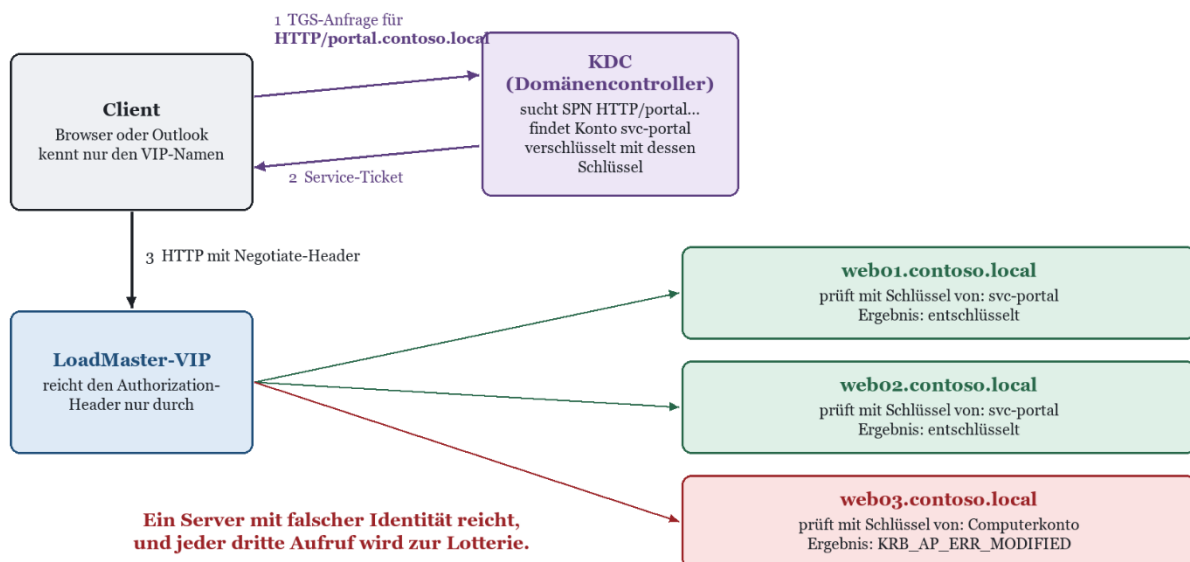
Warum der SPN zum VIP-Namen gehört und nicht zu den Servern

Ein Service Principal Name ist im Kern ein Adressticket: Dienstklasse, Hostname und optional ein Port, also etwa HTTP/portal.contoso.local. Wenn der Browser mit Negotiate anmelden will, baut er sich diesen Namen aus der URL zusammen und fragt den KDC nach einem Ticket. Der KDC schaut im Active Directory nach, an welchem Konto der SPN hängt, und verschlüsselt das Ticket mit dessen Langzeitschlüssel. Er fragt dabei nicht, welcher Server am Ende antworten wird. Er weiß es schlicht nicht, und der LoadMaster erzählt es ihm auch nicht.

Damit ist die Grundregel klar: Der SPN muss zum Namen passen, den der Client benutzt. Bei einem Virtual Service ist das der Name, der im DNS auf die VIP zeigt. Die Servernamen web01 bis web03 tauchen im Ticketgeschäft gar nicht auf. Ihre Computerkonten haben zwar automatisch HOST-SPNs für den eigenen Namen, aber die helfen dir bei der VIP nicht, denn kein Client fragt jemals nach HTTP/web02.contoso.local, solange alles über den LoadMaster läuft.

Wer bestellt was, und wer muss es entschlüsseln?

Der KDC verschlüsselt das Ticket mit dem Schlüssel des Kontos, an dem der SPN hängt



Skizze: Ticketablauf mit drei Real Servern, Namen beispielhaft

Der LoadMaster selbst spielt bei klassischem Kerberos hinter einem Layer-7-Service die Rolle des Briefträgers. Er reicht den Authorization-Header mit dem Ticket durch, egal ob er TLS terminiert oder nicht. Kerberos hängt nicht an IP-Adressen, also ist es auch egal, ob der Real Server die Client-IP sieht oder nur die Adresse des LoadMasters. Entscheidend ist allein der Name. Das unterscheidet Kerberos angenehm von NTLM, das an der TCP-Verbindung klebt und deshalb mit Verbindungswiederverwendung und Server-Wechseln seine eigenen Dramen hat. Wie Virtual

Service, Real Server und SubVS zusammenspielen, erklärt [Virtual Service, Real Server, SubVS: das Grundvokabular des LoadMasters](#).

Die Folge: Alle Server brauchen denselben Schlüssel

Wenn das Ticket mit dem Schlüssel eines Kontos verschlüsselt ist, muss jeder Real Server dieses Konto kennen. Läuft der Dienst auf jedem Server unter seinem eigenen Computerkonto, kann höchstens einer der Server das Ticket lesen, nämlich der, dessen Computerkonto zufällig den SPN trägt. Die anderen bekommen ein Ticket, das für sie wie Datenmüll aussieht. Das ist kein Bug, das ist Kryptografie, die genau das tut, wofür sie gebaut wurde.

Die Lösung heißt immer: eine gemeinsame Dienstidentität. Ein Konto, ein Kennwort, ein Schlüssel, auf allen Servern hinter derselben VIP. Wie diese Identität heißt und wer ihr Kennwort verwaltet, hängt vom Produkt ab. Das Prinzip nicht.

Ansatz	Wo hängt der SPN?	Ergebnis hinter dem LoadMaster
SPN am Computerkonto von web01	WEB01\$	Nur web01 kann entschlüsseln. Bei Round Robin scheitert jeder zweite oder dritte Aufruf, oder Negotiate fällt auf NTLM zurück.
SPN an allen drei Computerkonten	unmöglich, setspn -S verweigert Duplikate	Wer es mit setspn -A oder per Attribut-Editor erzwingt, baut sich doppelte SPNs.
Kein SPN für den VIP-Namen	nirgends	Kein Kerberos. Der Client fällt still auf NTLM zurück, Delegation ist tot.
SPN am gemeinsamen Dienstkonto	svc-portal oder ASA-Konto	Alle Server entschlüsseln. So ist es gedacht.

A-Record statt CNAME

Ein Detail, das gern unterschätzt wird: Manche Clients lösen einen CNAME auf und bauen den SPN dann aus dem Zielnamen statt aus dem Namen in der URL. Zeigt portal.contoso.local per CNAME auf lm-vip01.contoso.local, fragt der Client unter Umständen nach HTTP/lm-vip01.contoso.local, und dein sauber registrierter SPN liegt ungenutzt im Verzeichnis. Microsoft empfiehlt für die gemeinsamen Exchange-Namespaces deshalb ausdrücklich A-Records, obwohl CNAMEs grundsätzlich funktionieren. Die Empfehlung gilt sinngemäß für jeden Dienst hinter einer VIP.

TIPP · Port im SPN? Meist nicht

Browser hängen bei Standardports keinen Port an den SPN an, und auch bei abweichenden Ports lassen die meisten Clients ihn weg. Registriere also HTTP/portal.contoso.local und nicht HTTP/portal.contoso.local:443. Und die Dienstklasse heißt auch bei HTTPS HTTP, nicht HTTPS. Es gibt kein HTTPS/-Ticket. Wer eines registriert, registriert ein Mahnmal.

WARNUNG · Der HOST-SPN als blinder Passagier

Ein HOST-SPN deckt im Active Directory über eine Zuordnungstabelle auch HTTP und eine ganze Reihe weiterer Dienstklassen mit ab. Hat irgendwann einmal ein Server oder ein Computerkonto denselben Namen getragen wie deine heutige VIP, steht dort womöglich noch HOST/portal.contoso.local. Dann hat dein HTTP-SPN einen Konkurrenten, den ein flüchtiger Blick auf die HTTP-Einträge nicht zeigt. setspn -Q findet ihn, weil es die Zuordnung berücksichtigt.

Ein Konto für alle: ASA, App-Pool-Konto und gMSA

Jetzt wird es produktspezifisch. Exchange, SharePoint und selbst gebaute IIS-Anwendungen lösen das Problem der gemeinsamen Identität unterschiedlich, aber alle drei kommen am Ende beim selben Bild an: Ein Konto trägt den SPN der VIP, und jeder Real Server kennt dessen Kennwort.

Drei Wege zum gemeinsamen Schlüssel

Egal welches Produkt: Alle Real Server müssen das Ticket mit demselben Konto entschlüsseln können



Skizze: gemeinsame Dienstidentität je Produkt, vereinfacht

Exchange: der Alternate Service Account

Exchange Server 2016, 2019 und Subscription Edition authentifizieren Clientzugriffe in IIS standardmäßig im Kontext des Computerkontos. Für einen einzelnen Server ist das in Ordnung, für eine Load-Balancing-Gruppe nicht. Deshalb gibt es den Alternate Service Account, kurz ASA: eine zusätzliche Anmeldeinformation, die alle Exchange-Server mit denselben Namespaces kennen und mit der sie Kerberos-Tickets für die gemeinsamen Namen entschlüsseln.

Microsoft empfiehlt für den ASA ein Computerkonto statt eines Benutzerkontos, weil es keine interaktive Anmeldung erlaubt und das Kennwort nicht abläuft. Besondere Rechte braucht das Konto nicht, Mitgliedschaft in Domänencomputer reicht. Gruppenverwaltete Dienstkonten (gMSA) sind für den ASA ausdrücklich nicht unterstützt. Ein Konto pro Exchange-Version und Gesamtstruktur reicht in der Regel; Exchange 2010 und neuere Versionen können sich aber keinen ASA teilen.

Die Reihenfolge ist wichtig und wird gern vertauscht. Erst wird das Konto angelegt und für AES freigeschaltet, dann wird die Anmeldeinformation per Skript auf den ersten Server verteilt, dann auf alle weiteren, und erst danach bekommen die SPNs ihr neues Zuhause. Wer die SPNs vorher setzt, verteilt Tickets, die noch kein Server lesen kann. Das ist die Kerberos-Variante von „erst den Briefkasten abschrauben, dann umziehen“.

```
New-ADComputer -Name EXASA -SamAccountName EXASA -Enabled $true `
  -AccountPassword (Read-Host 'Kennwort' -AsSecureString) -Description 'Exchange ASA'
Set-ADComputer EXASA -Add @{ 'msDS-SupportedEncryptionTypes'=28}

cd $exscripts
.\RollAlternateServiceAccountPassword.ps1 -ToSpecificServer ex01.contoso.local `
  -GenerateNewPasswordFor 'CONTOSO\EXASA$'
.\RollAlternateServiceAccountPassword.ps1 -ToSpecificServer ex02.contoso.local `
  -CopyFrom ex01.contoso.local
```

```
setspn -F -Q HTTP/mail.contoso.local  
setspn -S HTTP/mail.contoso.local CONTOSO\EXASA$  
setspn -S HTTP/autodiscover.contoso.local CONTOSO\EXASA$
```

```
Get-ClientAccessService -IncludeAlternateServiceAccountCredentialStatus |  
Format-List Name, AlternateServiceAccountConfiguration
```

Der Wert 28 in msDS-SupportedEncryptionTypes ist das, was die Microsoft-Anleitung nennt: RC4 plus AES 128 plus AES 256. Wenn deine Domäne RC4 bereits verbannt hat oder gerade dabei ist, setzt du konsequent auf 24, also nur AES. Warum das gerade bei Keytabs und Delegation am LoadMaster Folgen hat, steht in [Die RC4-Falle: KCD-Keytab trifft AES-only-Domäne](#).

WICHTIG · Das Skript ist der einzige unterstützte Weg

Microsoft unterstützt für das Verteilen des ASA ausschließlich RollAlternateServiceAccountPassword.ps1 aus dem Scripts-Verzeichnis der Exchange-Installation. Der Dienst Microsoft Exchange Service Host verwaltet die Anmeldeinformation auf jedem Server; läuft er nicht, gibt es kein Kerberos. Und wenn du das Kennwort rollierst, dann auf allen Servern, nicht nur auf dem, an dem du gerade sitzt.

AUS DER PRAXIS · Der vierte Server, den keiner eingeladen hat

Eine Verwaltung erweitert ihre Exchange-Gruppe um einen vierten Server und nimmt ihn im LoadMaster als Real Server auf. Health Check grün, OWA läuft, alle zufrieden. Nur der ASA wurde nie auf den neuen Server kopiert. Ergebnis: Jeder vierte Outlook-Verbindungsaufbau fällt auf NTLM zurück, und weil NTLM funktioniert, merkt es niemand außer dem Domänencontroller, der plötzlich mehr zu tun hat. Aufgefallen ist es erst, als jemand NTLM einschränken wollte. Moral: Neuer Real Server heißt neue Checkliste, und der ASA steht drauf.

Was ein Exchange-Virtual-Service am LoadMaster sonst noch braucht, also Namespaces, SubVS und Health Checks, beschreibt [Exchange Server hinter dem LoadMaster: Templates, Namespaces, Health Checks](#). Ein Hinweis noch, weil er gern im falschen Beitrag gesucht wird: Exchange Extended Protection verträgt kein TLS-Offloading mit anschließendem Klartext zum Server. Mit Kerberos und SPNs hat das nichts zu tun, mit dem Design deines Virtual Service aber sehr wohl, siehe [TLS-Offload, Re-Encryption und Zertifikate: PKI trifft LoadMaster](#).

SharePoint: das Konto des Anwendungspools

SharePoint Server macht es auf den ersten Blick einfacher, weil es ohnehin mit Domänenkonten arbeitet. Jede Webanwendung läuft in einem IIS-Anwendungspool, dessen Identität ein verwaltetes Konto ist, und SharePoint sorgt dafür, dass dieser Pool auf allen Servern der Farm mit demselben Konto läuft. Genau das ist die gemeinsame Identität, die Kerberos braucht. Der SPN HTTP/portal.contoso.local gehört also an das App-Pool-Konto der Webanwendung, zum Beispiel CONTOSO\svc-sp-portal, und nicht an die Computerkonten der Server.

Drei Stolpersteine sehen wir regelmäßig. Erstens teilen sich mehrere Webanwendungen denselben Namen über Host-Named Site Collections oder alternative Zugriffszuordnungen, und jemand vergisst einen der Namen. Jeder Name, den Clients aufrufen, braucht einen SPN. Zweitens wechselt die Identität des App-Pools bei einer Umstellung, aber der SPN bleibt am alten Konto. Drittens ist für die Webanwendung zwar Negotiate eingestellt, aber der SPN fehlt ganz, und alle sind glücklich mit NTLM, bis eine Delegation an SQL Server Reporting Services oder einen

Dateiserver gebraucht wird. Mehr zu SharePoint findest du im Bereich [SharePoint](#) und konkret zum LoadMaster in [SharePoint Server Subscription Edition hinter dem LoadMaster](#).

IIS-Eigenentwicklungen: gemeinsames Dienstkonto oder gMSA

Bei eigenen Anwendungen auf IIS bist du selbst SharePoint. Das heißt: Du entscheidest über die App-Pool-Identität, und du musst dafür sorgen, dass sie auf allen Real Servern gleich ist. Für neue Anwendungen ist ein gruppenverwaltetes Dienstkonto die eleganteste Lösung. Alle Server, die das Kennwort abrufen dürfen, bekommen denselben Schlüssel, und das Kennwort wechselt automatisch, ohne dass jemand an fünf Servern gleichzeitig tippen muss.

Dazu kommt ein IIS-Detail, das die meisten Kerberos-Probleme mit eigenen Anwendungen erklärt: die Kernelmodus-Authentifizierung. Ist sie aktiv, was der Standard ist, entschlüsselt http.sys das Ticket mit dem Schlüssel des Computerkontos, nicht mit dem des App-Pools. Erst mit `useAppPoolCredentials` auf `true` nimmt IIS den Schlüssel der Pool-Identität. Ohne diese Einstellung hast du den SPN sauber am Dienstkonto, die Tickets werden mit dessen Schlüssel verschlüsselt, und IIS versucht es trotzdem mit dem Computerkonto. Das Resultat ist `KRB_AP_ERR_MODIFIED`, obwohl alles richtig aussieht.

```
New-ADServiceAccount -Name gmsa-portal -DNSHostName gmsa-portal.contoso.local `
-PrincipalsAllowedToRetrieveManagedPassword 'GG-Portal-Webserver'
setspn -S HTTP/portal.contoso.local CONTOSO\gmsa-portal$

%windir%\system32\inetsrv\appcmd.exe set config "Default Web Site/portal" `
-section:system.webServer/security/authentication/windowsAuthentication `
/useAppPoolCredentials:"True" /commit:apphost
```

Alternativ schaltest du die Kernelmodus-Authentifizierung für die Site ab, dann entschlüsselt der Benutzermodus mit der Pool-Identität. Beides funktioniert; wichtig ist nur, dass es auf allen Real Servern identisch konfiguriert ist. Wie du eigene Anwendungen und http.sys-Dienste generell hinter den LoadMaster bringst, beschreibt [IIS-Eigenentwicklungen und http.sys-Dienste veröffentlichen](#).

Produkt	Gemeinsame Identität	SPN-Beispiel	Typischer Fehler
Exchange 2016, 2019, SE	ASA, idealerweise Computerkonto; gMSA nicht unterstützt	HTTP/mail.contoso.local, HTTP/autodiscover.contoso.local	ASA nicht auf alle Server verteilt; SPN vor der Verteilung gesetzt
SharePoint Server SE	App-Pool-Konto der Webanwendung	HTTP/portal.contoso.local	Zusätzliche Namen ohne SPN; SPN am alten Konto
IIS-Eigenbau	Dienstkonto oder gMSA als Pool-Identität	HTTP/app.contoso.local	Kernelmodus entschlüsselt mit Computerkonto, <code>useAppPoolCredentials</code> fehlt
ADFS-Farm	Dienstkonto oder gMSA der Farm	HOST/sts.contoso.local	SPN nach Farmmigration am alten Dienstkonto

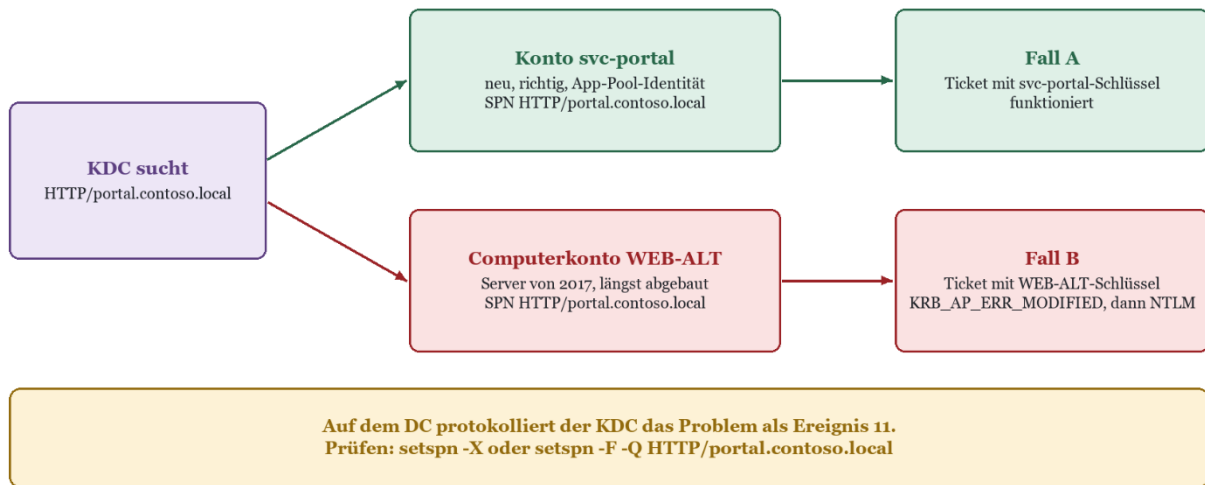
Doppelte SPNs und KRB_AP_ERR_MODIFIED

Ein SPN darf in der gesamten Gesamtstruktur genau einem Konto gehören. Das ist keine Empfehlung, das ist die Geschäftsgrundlage. Findet der KDC denselben SPN an zwei Konten, weiß er nicht, welchen Schlüssel er nehmen soll. Er protokolliert auf dem Domänencontroller ein Ereignis 11 der Quelle KDC und liefert entweder gar kein Ticket oder eines, das mit dem falschen

Schlüssel verschlüsselt ist. Welcher Fall eintritt, entscheidet nicht dein Wille, sondern die Struktur deines Verzeichnisses.

Doppelte SPNs: zwei Besitzer, ein KDC mit Schulterzucken

Der KDC nimmt irgendeinen Treffer. Ob es der richtige ist, erfährst du am Server.



Skizze: doppelte Registrierung eines SPN, Konten beispielhaft

Wie Duplikate entstehen

- **Umzug ohne Auszug:** Der SPN wird für das neue Dienstkonto gesetzt, aber vom alten Konto oder Computerkonto nie entfernt.
- **setspn -A statt -S:** Die alte Option -A fügt ohne Prüfung auf Duplikate hinzu. -S prüft vorher und verweigert, wenn der SPN schon vergeben ist.
- **Mehrere Domänen:** In einer Domäne ist alles sauber, in der Nachbardomäne derselben Gesamtstruktur steht ein Altlasten-SPN. Ohne -F für die Gesamtstruktursuche bleibt er unsichtbar.
- **Recycelte Namen:** Die VIP heißt heute wie ein Server von früher, dessen Computerkonto noch existiert oder dessen HOST-SPN noch irgendwo steht.
- **Produkte, die selbst registrieren:** Manche Dienste tragen SPNs beim Start selbst ein. Läuft der Dienst auf einem Server testweise unter dem Computerkonto, hat das Computerkonto danach einen SPN, von dem niemand weiß.

Was KRB_AP_ERR_MODIFIED eigentlich sagt

Der Fehler ist ehrlicher, als sein Name klingt. Er bedeutet: Der Server hat ein Ticket bekommen, das er mit seinem Schlüssel nicht entschlüsseln konnte. Modified ist dabei im Sinne von „passt nicht zu dem, was ich erwartet habe“ zu verstehen, nicht im Sinne eines Angriffs. Der Client protokolliert das im Systemprotokoll als Ereignis 4 der Quelle Security-Kerberos, inklusive des Zielnamens. Das ist der wertvollste Hinweis, den du bekommst.

Hinter dem Load Balancer gibt es dafür vier übliche Ursachen. Ein doppelter SPN, bei dem der KDC das falsche Konto erwischt. Ein Real Server, der unter einer anderen Identität läuft als die übrigen. IIS im Kernelmodus ohne useAppPoolCredentials. Und ein Kennwortwechsel, der nicht überall angekommen ist, etwa ein ASA, der nur auf einem Teil der Exchange-Server rolliert wurde. Allen gemeinsam: Der Schlüssel, mit dem das Ticket verschlüsselt wurde, ist nicht der Schlüssel, den der Server benutzt.

Symptom	Wahrscheinliche Ursache	Erste Prüfung
Fehler bei jedem Aufruf,	SPN am falschen Konto oder doppelt	setspn -Q HTTP/name, setspn -X

auf allen Servern		
Fehler nur jedes zweite oder dritte Mal	Ein Real Server mit abweichender Identität	Server einzeln testen, App-Pool und ASA-Status vergleichen
Ticket im klist, trotzdem NTLM im IIS-Log	Kernelmodus nutzt Computerkonto	useAppPoolCredentials prüfen
Kein HTTP-Ticket im klist	Kein SPN, CNAME-Effekt oder Site nicht in der Intranetzone	DNS prüfen, klist get HTTP/name
Fehler seit dem letzten Kennwortwechsel	Neues Kennwort nicht überall bekannt	ASA-Status, App-Pool-Anmeldeinformationen, Replikation

WARNUNG · Der Fallback, der nichts sagt

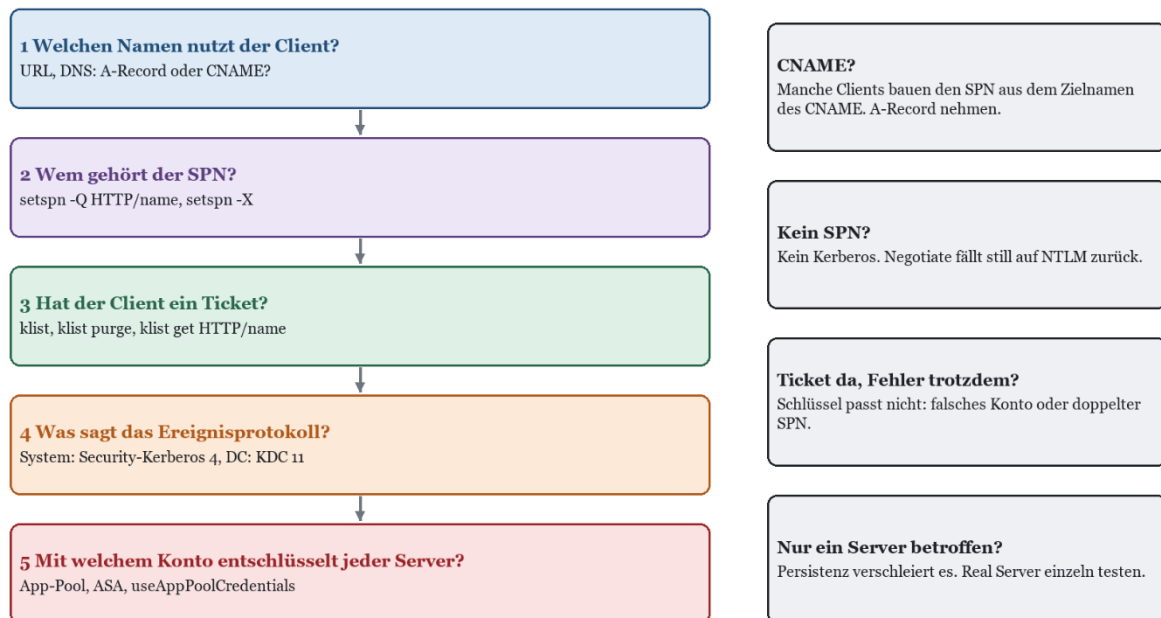
Negotiate ist höflich. Scheitert Kerberos, versucht es der Client mit NTLM, und wenn das klappt, sieht der Benutzer nichts. Deine Delegation ist dann trotzdem kaputt, und du verlässt dich auf ein Protokoll, das Microsoft schrittweise zurückdrängt. Wie du diesen Fallback sichtbar machst, steht in [NTLM, NTLM-Proxy und der Kerberos-Fallback, den keiner bemerkt](#).

Werkzeuge: setspn, klist und das Ereignisprotokoll

Für die Diagnose brauchst du keinen Paketmitschnitt am LoadMaster und kein Debug-Log, jedenfalls nicht als ersten Schritt. Die meisten SPN-Probleme lassen sich mit drei Bordwerkzeugen in zehn Minuten eingrenzen. Wichtig ist die Reihenfolge, denn wer bei Schritt fünf anfängt, sucht im falschen Stockwerk.

Diagnosepfad in fünf Schritten

Von „fragt nach Kennwort“ zur Ursache, ohne am LoadMaster herumzuraten



Skizze: Reihenfolge der Prüfungen bei Kerberos-Problemen hinter dem LoadMaster

setspn: wem gehört der Name?

setspn -Q fragt das Verzeichnis, welches Konto einen bestimmten SPN trägt, und berücksichtigt dabei die HOST-Zuordnung. Mit -F suchst du in der gesamten Gesamtstruktur statt nur in der aktuellen Domäne. setspn -X sucht nach Duplikaten, setspn -L listet alle SPNs eines Kontos. Zum Hinzufügen nimmst du immer -S, nie -A.

```
setspn -F -Q HTTP/portal.contoso.local
setspn -X -F
setspn -L CONTOSO\svc-portal
setspn -D HTTP/portal.contoso.local CONTOSO\WEB01$
setspn -S HTTP/portal.contoso.local CONTOSO\svc-portal
```

Liefert setspn -Q genau ein Konto, und zwar das richtige, ist die Verzeichnisseite in Ordnung. Liefert es zwei, hast du dein Problem gefunden. Liefert es keines, hast du es auch gefunden, nur ein anderes. Kleine Warnung am Rande: SPN-Änderungen müssen erst replizieren, bevor alle Domänencontroller sie kennen. Wer in einer Umgebung mit mehreren Standorten sofort testet, testet gelegentlich gegen den Stand von vor fünf Minuten.

klist: was hat der Client wirklich bekommen?

klist zeigt die Tickets der aktuellen Anmeldesitzung. Nach einem Aufruf der Seite muss dort ein Ticket für HTTP/portal.contoso.local stehen. Fehlt es, hat der Client gar nicht erst nach Kerberos gefragt oder keins bekommen. Mit klist get kannst du ein Ticket gezielt anfordern und siehst die Fehlermeldung des KDC direkt, statt sie im Browser zu erraten. Mit klist purge wirfst du alte Tickets weg, was nach jeder SPN-Änderung Pflicht ist, denn der Client hält ein einmal bezogenes Ticket bis zu seinem Ablauf und benutzt fröhlich weiter den alten Schlüssel.

```
klist purge
klist get HTTP/portal.contoso.local
klist
```

TIPP · Die Verschlüsselungsart verrät viel

klist zeigt zu jedem Ticket den KerbTicket-Verschlüsselungstyp. Steht dort RC4-HMAC, obwohl du AES erwartest, hat das Zielkonto vermutlich kein passendes msDS-SupportedEncryptionTypes oder sein Kennwort stammt aus einer Zeit, in der noch keine AES-Schlüssel erzeugt wurden. In einer Domäne, die RC4 abschaltet, wird aus diesem Hinweis schnell ein Ausfall.

Ereignisprotokoll: Client, Server und KDC

Drei Stellen lohnen den Blick. Auf dem Client steht im Systemprotokoll das Ereignis 4 von Security-Kerberos mit KRB_AP_ERR_MODIFIED und dem Zielnamen. Auf den Domänencontrollern meldet die Quelle KDC mit Ereignis 11 doppelte SPNs, und im Sicherheitsprotokoll zeigen die Ereignisse 4769 jede Ticketanforderung samt Dienstname und Verschlüsselungstyp, sofern die Überwachung dafür aktiv ist. Auf den Real Servern verraten die Anmeldeereignisse 4624 im Feld zum Authentifizierungspaket, ob Kerberos oder NTLM verwendet wurde.

Wo	Protokoll	Ereignis	Was es sagt
Client	System	4, Security-Kerberos	KRB_AP_ERR_MODIFIED: Server konnte Ticket nicht entschlüsseln
Domänencontroller	System	11, KDC	SPN mehrfach registriert
Domänencontroller	Sicherheit	4769	Ticketanforderung mit Dienstname und Verschlüsselungstyp
Real Server	Sicherheit	4624	Anmeldung, Authentifizierungspaket Kerberos oder NTLM
Exchange-Server	HttpProxy-Logs	Feld AuthenticationType	Negotiate zeigt erfolgreiche Kerberos-Anmeldung

Und der LoadMaster?

Der LoadMaster hilft bei der Diagnose vor allem dadurch, dass du ihn umgehen kannst. Teste jeden Real Server einzeln mit dem VIP-Namen, etwa über einen Eintrag in der hosts-Datei eines Testclients, der den VIP-Namen direkt auf die Adresse eines Servers zeigt. Weil der Client das Ticket für den Namen bestellt, nicht für die Adresse, bekommst du dasselbe Ticket wie über die VIP, nur ohne Lastverteilung. Funktioniert es auf web01 und web02, aber nicht auf web03, ist die Suche beendet. Persistenz verschleiert solche Fehler übrigens gern, weil ein Client tagelang am guten Server klebt, siehe [Persistence und Scheduling: wer hängt an welchem Server und warum](#). Für die tieferen Werkzeuge am LoadMaster selbst gibt es [Troubleshooting am LoadMaster: Logs, L7-Debug und TCP-Dump](#).

WICHTIG · Sonderfall: Der LoadMaster holt selbst Tickets

Sobald du am LoadMaster Vorauthentifizierung mit Kerberos Constrained Delegation nutzt, fordert der LoadMaster die Tickets für die Real Server selbst an. Dann gelten zwei SPN-Welten: die des Clients gegenüber dem LoadMaster und die des LoadMasters gegenüber den Servern. Wie du das sauber trennst und welche SPNs dort gebraucht werden, steht in [Kerberos Constrained Delegation am LoadMaster: KCD und RBCD sauber einrichten](#).

FAQ: Kerberos, Load Balancer und SPN

Muss ich am Kemp LoadMaster etwas für Kerberos einstellen?

Für klassisches Kerberos zwischen Client und Real Server nicht. Der LoadMaster reicht den Authorization-Header durch. Entscheidend sind DNS, SPN und die Identität auf den Servern. Anders ist es nur, wenn der LoadMaster per KCD selbst Tickets anfordert.

Darf derselbe SPN an mehreren Computerkonten hängen, damit alle Server ihn haben?

Nein. Ein SPN gehört genau einem Konto in der Gesamtstruktur. Mehrere Besitzer führen zu KDC-Ereignis 11 und zu KRB_AP_ERR_MODIFIED. Die Lösung ist ein gemeinsames Dienstkonto.

Funktioniert Kerberos mit TLS-Offload am LoadMaster?

Ja. Das Ticket steckt im HTTP-Header, nicht in der TLS-Sitzung. Ob der LoadMaster entschlüsselt und neu verschlüsselt, ist für Kerberos egal. Für Exchange Extended Protection ist es das nicht, das ist aber ein anderes Thema.

Kann ich für den Exchange-ASA ein gMSA verwenden?

Nein, Microsoft unterstützt gruppenverwaltete Dienstkonten für den ASA nicht. Empfohlen ist ein Computerkonto, verteilt mit RollAlternateServiceAccountPassword.ps1.

Warum sehe ich nach dem Korrigieren des SPN immer noch den Fehler?

Weil der Client sein altes Ticket behält, bis es abläuft. klist purge auf dem Testclient, Replikation zwischen den Domänencontrollern abwarten, dann erneut testen.

Brauche ich für HTTPS einen eigenen SPN?

Nein. Die Dienstklasse ist für HTTP und HTTPS gleichermaßen HTTP, und einen Port hängt der Browser in aller Regel nicht an.

Woran erkenne ich, dass heimlich NTLM statt Kerberos läuft?

Am Ereignis 4624 auf dem Real Server, am fehlenden HTTP-Ticket im klist des Clients und bei Exchange an den HttpProxy-Logs. Ein dauerhaft hoher NTLM-Anteil ist fast immer ein SPN-Thema.

Fazit: Ein Name, ein Konto, ein Schlüssel

Kerberos hinter dem Load Balancer ist kein Hexenwerk, es ist Buchhaltung. Der Client bestellt sein Ticket für den Namen der VIP, der KDC verschlüsselt es für das Konto, an dem dieser SPN hängt, und jeder Real Server muss den Schlüssel dieses Kontos kennen. Bei Exchange heißt dieses Konto ASA, bei SharePoint App-Pool-Konto, bei deinen eigenen Anwendungen am besten gMSA. Der SPN gehört genau einmal an genau dieses Konto, und zwar für jeden Namen, den Clients tatsächlich aufrufen.

Wenn es trotzdem hakt, gehst du der Reihe nach vor: Name und DNS, setspn, klist, Ereignisprotokoll, Identität auf jedem einzelnen Server. In neun von zehn Fällen ist die Ursache ein vergessener Server, ein vergessener Name oder ein SPN, der beim letzten Umzug im alten Haus geblieben ist. Der zehnte Fall ist IIS im Kernelmodus. Und falls du nach alledem feststellst, dass deine Umgebung seit Jahren fröhlich NTLM spricht: Willkommen im Club. Die gute Nachricht ist, dass man das reparieren kann, bevor Microsoft es für dich entscheidet.

TIPP · Hilfe beim Aufräumen

Wenn du SPNs, Dienstkonten und Delegationen in einer gewachsenen Umgebung einmal gründlich durchgehen willst, unterstützt dich die [Beratung zum Kemp LoadMaster](#). Wer Kerberos hinter dem LoadMaster mit dem eigenen Team im Labor nachstellen möchte, findet das in der [Kemp-LoadMaster-Schulung](#).

WEITERLESEN · Weiter im Thema

- › [Microsoft Kerberos](#)
- › [Kerberos Constrained Delegation am LoadMaster: KCD und RBCD sauber einrichten](#)
- › [NTLM, NTLM-Proxy und der Kerberos-Fallback, den keiner bemerkt](#)
- › [Die RC4-Falle: KCD-Keytab trifft AES-only-Domäne](#)
- › [Exchange Server hinter dem LoadMaster: Templates, Namespaces, Health Checks](#)
- › [SharePoint Server Subscription Edition hinter dem LoadMaster](#)
- › [IIS-Eigenentwicklungen und http.sys-Dienste veröffentlichen](#)
- › [Troubleshooting am LoadMaster: Logs, L7-Debug und TCP-Dump](#)