3.13 VPN- und Tunnelverbindungen analysieren

Ein Virtual Private Network erweitert die lokale Netzwerkumgebung um einen verschlüsselten oder logisch getrennten Tunnel. Eine angezeigte VPN-Verbindung kann erfolgreich aufgebaut sein, obwohl interne Dienste, DNS-Auflösung oder einzelne Netzbereiche weiterhin nicht funktionieren.

Die zentralen Fragen dieser Seite lauten:

> Wird der Tunnel vollständig aufgebaut, erhält der Client die richtige Konfiguration und wird der betroffene Datenverkehr tatsächlich durch den vorgesehenen Tunnel geleitet?

---

1. Sicherheits- und Aktionskennzeichnungen

Kennzeichnung	Bedeutung
[RO]	Rein lesender Befehl, der normalerweise keine Konfiguration verändert
[TEST]	Aktiver Test, der Netzwerkverkehr erzeugt
[PRIV]	Administrator- oder Root-Rechte können erforderlich sein
[FILE]	Der Befehl schreibt Ausgaben in eine Datei
[SENS]	Die Ausgabe kann sensible Informationen enthalten
[CHANGE]	Der Befehl verändert eine Konfiguration
[DISRUPT]	Der Befehl kann Verbindungen oder Dienste beeinträchtigen

VPN-Ausgaben können Benutzernamen, öffentliche Schlüssel, interne Netze, Serveradressen, Zertifikatsinformationen und Sicherheitsrichtlinien enthalten. Private Schlüssel, Pre-Shared Keys, Passwörter und Sitzungstoken dürfen niemals dokumentiert oder weitergegeben werden.

---

2. VPN- und Tunnelarten unterscheiden

Art	Typische Verwendung
Remote-Access-VPN	Einzelne Clients verbinden sich mit einem Unternehmensnetz

Art	Typische Verwendung
Site-to-Site-VPN	Zwei oder mehr Netzwerke werden miteinander verbunden
Full Tunnel	Grundsätzlich wird der gesamte vorgesehene Datenverkehr durch das VPN geleitet
Split Tunnel	Nur definierte Netze oder Anwendungen verwenden den Tunnel
Device Tunnel	Verbindung wird im Gerätekontext aufgebaut
User Tunnel	Verbindung wird im Benutzerkontext aufgebaut
Always-On-VPN	Tunnel wird automatisch oder dauerhaft aufgebaut
Policy-Based VPN	Auswahl des Tunnelverkehrs anhand definierter Richtlinien
Route-Based VPN	Tunnel wird als routbare virtuelle Schnittstelle behandelt
Application VPN	Nur ausgewählte Anwendungen verwenden den Tunnel
Overlay-Netzwerk	Virtuelles Netz wird über ein bestehendes Netz gelegt
Zero-Trust-Zugang	Zugriff wird zusätzlich anhand von Identität, Gerät und Richtlinien bewertet

Wichtig: Die Anzeige „Verbunden“ bestätigt häufig nur den Tunnelaufbau. Sie beweist nicht, dass die benötigte Anwendung erreichbar ist.

---

3. Häufige VPN-Protokolle und Transportwege

VPN-Technik	Typischer Transport
IKEv2/IPsec	UDP 500 und bei NAT-T UDP 4500
IPsec ESP	IP-Protokollnummer 50, kein TCP- oder UDP-Port
IPsec AH	IP-Protokollnummer 51, kein TCP- oder UDP-Port
L2TP	Typischerweise UDP 1701, häufig zusammen mit IPsec
SSTP	TCP 443
OpenVPN	Häufig UDP oder TCP 1194, aber frei konfigurierbar
WireGuard	UDP, Port frei konfigurierbar; häufig wird 51820 verwendet
GRE	IP-Protokollnummer 47, kein TCP- oder UDP-Port
TLS-basierte Hersteller-VPNs	Häufig TCP 443 oder UDP-basierter Transport
SSH-Tunnel	TCP, häufig Port 22, aber konfigurierbar

Die tatsächlichen Ports und Protokolle müssen immer aus der produktiven VPN-Konfiguration entnommen werden.

Nicht verwechseln:

```
GRE = IP-Protokoll 47
ESP = IP-Protokoll 50
AH  = IP-Protokoll 51
```

Diese Zahlen sind keine Portnummern.

---

4. Die Diagnose in Phasen aufteilen

Phase	Zentrale Frage
1. Basisverbindung	Besitzt der Client eine funktionierende Verbindung zum Internet oder Unternehmensnetz?
2. Serverauflösung	Kann der VPN-Gatewayname korrekt aufgelöst werden?
3. Transport	Ist der benötigte Port beziehungsweise das IP-Protokoll erreichbar?
4. Authentifizierung	Werden Benutzer, Gerät, Zertifikat und MFA akzeptiert?
5. Tunnelerstellung	Wird eine virtuelle Schnittstelle beziehungsweise Security Association aufgebaut?
6. Adressierung	Erhält der Client die vorgesehene Tunneladresse?
7. Routing	Werden die benötigten Zielnetze in den Tunnel geleitet?
8. DNS	Werden interne Namen über die vorgesehenen Resolver aufgelöst?
9. Richtlinie	Darf Benutzer und Gerät auf das Ziel zugreifen?
10. Anwendung	Antwortet der eigentliche Dienst?
11. Stabilität	Bleibt der Tunnel verbunden und leistungsfähig?

Durch diese Trennung wird verhindert, dass ein Anwendungsfehler fälschlich als VPN-Aufbaufehler behandelt wird.

---

5. Störungsumfang zuerst bestimmen

Vor den technischen Prüfungen sind folgende Fragen zu beantworten:

- Ist nur ein Benutzer betroffen?
- Ist nur ein Gerät betroffen?
- Sind alle Benutzer eines Standorts betroffen?
- Funktioniert das VPN über ein anderes Netzwerk?
- Funktioniert es über Mobilfunk, aber nicht über WLAN?
- Ist nur ein internes Ziel betroffen?
- Sind alle internen Namen betroffen?
- Funktionieren interne IP-Adressen, aber keine Hostnamen?
- Funktioniert der Tunnelaufbau, aber kein Datenverkehr?
- Bricht die Verbindung nach einer bestimmten Zeit ab?
- Begann der Fehler nach Passwort-, Zertifikats- oder Clientänderung?
- Betrifft der Fehler IPv4, IPv6 oder beides?
- Wird Full Tunnel oder Split Tunnel erwartet?
- Existiert eine aktuelle allgemeine VPN-Störung?

Wichtiger Vergleichstest:

Gleiches Gerät + anderes Zugangsnetz

Gleiches Benutzerkonto + anderes Gerät

Anderer Benutzer + gleiches Gerät

Funktionierender und fehlerhafter Client im Vergleich

Dabei müssen Datenschutz, Kontosicherheit und betriebliche Berechtigungen eingehalten werden.

---

6. Basisverbindung vor dem VPN prüfen

Ein VPN benötigt zunächst eine funktionierende Verbindung zum VPN-Gateway.

Aufgabe	Windows	Linux	macOS
Schnittstellen	`[RO] Get-NetAdapter`	`[RO] ip -br link`	`[RO] ifconfig`
IP-Konfiguration	`[RO] Get-NetIPConfiguration`	`[RO] ip -br address`	`[RO] ifconfig`
Standardroute	`[RO] Get-NetRoute -DestinationPrefix "0.0.0.0/0"`	`[RO] ip route show default`	`[RO] route -n get default`
DNS-Konfiguration	`[RO] Get-DnsClientServerAddress`	`[RO] resolvectl status`	`[RO][SENS] scutil --dns`
Internetziel testen	`[TEST] Test-NetConnection example.com -Port 443`	`[TEST] curl -I https://example.com/`	`[TEST] curl -I https://example.com/`

Vor dem Verbindungsversuch dokumentieren:

- lokale IP-Adresse,
- Standardgateway,
- verwendete Schnittstelle,
- DNS-Server,
- Proxykonfiguration,
- öffentliche Quelladresse,
- Datum und Uhrzeit.

Ein Captive Portal in einem Hotel-, Bahn- oder Gastnetz kann den VPN-Aufbau verhindern, obwohl eine WLAN-Verbindung angezeigt wird.

---

7. VPN-Gateway auflösen und Transportweg prüfen

Namensauflösung:

Windows:

```
[TEST] Resolve-DnsName vpn.example.net
```

Linux:

```
[TEST] getent ahosts vpn.example.net
```

macOS:

```
[TEST] dscacheutil -q host -a name vpn.example.net
```

Route zum Gateway prüfen:

Windows:

```
[RO] Find-NetRoute -RemoteIPAddress VPN_GATEWAY_IP
```

Linux:

```
[RO] ip route get VPN_GATEWAY_IP
```

macOS:

```
[RO] route -n get VPN_GATEWAY_IP
```

TCP-basierten Gatewayport prüfen:

Windows:

```
[TEST] Test-NetConnection vpn.example.net -Port 443 -InformationLevel Detailed
```

Linux und macOS:

```
[TEST] nc -vz -w 5 vpn.example.net 443
```

Ein erfolgreicher TCP-Porttest gilt nur für TCP. Er prüft keine UDP-basierten IKEv2-, OpenVPN- oder WireGuard-Verbindungen.

---

8. Windows-VPN-Profile und Status prüfen

VPN-Profile des aktuellen Benutzers:

```
[RO][SENS] Get-VpnConnection
```

Geräteweite VPN-Profile:

```
[RO][PRIV][SENS] Get-VpnConnection -AllUserConnection
```

Ausführliche Darstellung:

```
[RO][SENS] Get-VpnConnection |
    Format-List *
```

Klassische Anzeige aktiver RAS-Verbindungen:

```
[RO] rasdial
```

Netzwerkadapter einschließlich virtueller Adapter:

```
[RO] Get-NetAdapter -IncludeHidden |
    Sort-Object Status, Name |
    Format-Table Name, InterfaceDescription, Status, LinkSpeed
```

IP-Konfiguration aller Schnittstellen:

```
[RO][SENS] Get-NetIPConfiguration -All
```

Zu kontrollieren sind:

- Name des VPN-Profils,
- Tunneltyp,
- Serveradresse,
- Verbindungsstatus,
- Authentifizierungsverfahren,
- Split-Tunneling-Einstellung,
- Routingrichtlinien,
- Benutzer- oder Geräteprofil,
- verwendeter virtueller Adapter.

`Get-VpnConnection` zeigt nur VPN-Profile, die über die entsprechenden Windows-VPN-Komponenten verwaltet werden. Herstellerclients können eigene Konfigurationen und Diagnosewerkzeuge verwenden.

---

9. Windows-Routen vor und nach dem VPN vergleichen

Routingtabelle:

```
[RO] Get-NetRoute |
    Sort-Object AddressFamily, DestinationPrefix, RouteMetric |
    Format-Table AddressFamily, DestinationPrefix, NextHop,
                 InterfaceAlias, RouteMetric
```

IPv4-Routingtabelle klassisch:

```
[RO] route print -4
```

IPv6-Routingtabelle:

```
[RO] route print -6
```

Route zu einem internen Ziel bestimmen:

```
[RO] Find-NetRoute -RemoteIPAddress 10.20.30.40
```

Besonders relevant:

```
DestinationPrefix
NextHop
InterfaceAlias
RouteMetric
InterfaceMetric
```

Vergleichsablauf:

1. Routingtabelle ohne VPN erfassen.
2. VPN verbinden.
3. Routingtabelle erneut erfassen.
4. Neue und veränderte Routen vergleichen.
5. Route zum betroffenen internen Ziel bestimmen.
6. Prüfen, ob die Route über den VPN-Adapter führt.

Eine vorhandene Route beweist noch nicht, dass die VPN-Gegenstelle den Verkehr erlaubt oder korrekt zurückleitet.

---

10. Windows-VPN-Ereignisprotokolle prüfen

Vorhandene VPN- und RAS-Protokolle ermitteln:

```
[RO][PRIV][SENS] Get-WinEvent -ListLog *Ras* |
    Select-Object LogName, IsEnabled, RecordCount
```

```
[RO][PRIV][SENS] Get-WinEvent -ListLog *VPN* |
    Select-Object LogName, IsEnabled, RecordCount
```

RasClient-Protokoll abfragen, sofern vorhanden:

```
[RO][PRIV][SENS] Get-WinEvent `
    -LogName "Microsoft-Windows-RasClient/Operational" `
    -MaxEvents 100
```

Zeitlich begrenzen:

```
[RO][PRIV][SENS] Get-WinEvent `
    -FilterHashtable @{
        LogName   = "Microsoft-Windows-RasClient/Operational"
        StartTime = (Get-Date).AddMinutes(-30)
    }
```

Zusätzlich können abhängig vom VPN-Typ Protokolle zu folgenden Komponenten relevant sein:

- IKE,
- IPsec,
- EAP,
- Zertifikate,
- NPS,
- Geräteverwaltung,
- Hersteller-VPN-Client.

Nicht jede Umgebung aktiviert alle Diagnoseprotokolle standardmäßig.

---

11. Windows-IPsec-Sicherheitszuordnungen prüfen

Bei IPsec-basierten Verbindungen können Security Associations wichtige Informationen liefern.

Main-Mode-Sicherheitszuordnungen:

```
[RO][PRIV][SENS] Get-NetIPsecMainModeSA
```

Quick-Mode-Sicherheitszuordnungen:

```
[RO][PRIV][SENS] Get-NetIPsecQuickModeSA
```

Main-Mode-Regeln:

```
[RO][PRIV][SENS] Get-NetIPsecMainModeRule
```

Verbindungs-Sicherheitsregeln:

```
[RO][PRIV][SENS] Get-NetIPsecRule
```

Zu prüfen sind:

- lokale und entfernte Adresse,
- Authentifizierungsverfahren,
- Verschlüsselungs- und Integritätsverfahren,
- Status der Security Association,
- Lebensdauer,
- verwendete Richtlinie.

Leere Ausgaben können bedeuten, dass keine entsprechende aktive IPsec-Sicherheitszuordnung existiert oder dass der eingesetzte VPN-Client nicht über diese Windows-Komponenten arbeitet.

---

12. Linux: Tunnel, Verbindungen und Schnittstellen prüfen

Aktive NetworkManager-Verbindungen:

```
[RO][SENS] nmcli connection show --active
```

Gerätestatus:

```
[RO] nmcli device status
```

Alle Schnittstellen:

```
[RO] ip -br link
```

IP-Adressen:

```
[RO][SENS] ip -br address
```

Routingtabelle:

```
[RO] ip route show
```

IPv6-Routingtabelle:

```
[RO] ip -6 route show
```

Policy-Routingregeln:

```
[RO] ip rule show
```

Alle Routingtabellen:

```
[RO][SENS] ip route show table all
```

Mögliche Tunnelschnittstellen tragen abhängig von Technik und Hersteller Namen wie:

```
tun0
tap0
wg0
ppp0
ipsec0
Herstellerspezifischer Name
```

Der Name allein bestätigt nicht, dass der Tunnel vollständig funktionsfähig ist.

---

13. Linux-Protokolle prüfen

NetworkManager-Protokolle:

```
[RO][PRIV][SENS] sudo journalctl -u NetworkManager --since "-30 minutes"
```

Aktuelle Kernelmeldungen zu Netzwerkproblemen:

```
[RO][PRIV][SENS] sudo journalctl -k --since "-30 minutes"
```

Vorhandene VPN-Dienste suchen:

```
[RO] systemctl list-units --type=service |
    grep -Ei 'openvpn|wireguard|strongswan|ipsec|vpn'
```

Protokoll eines konkret ermittelten Dienstes:

```
[RO][PRIV][SENS] sudo journalctl -u TATSAECHLICHER_DIENSTNAME \
    --since "-30 minutes"
```

Der Unit-Name muss vorher ermittelt werden. Namen wie `openvpn.service` oder `openvpn-client@NAME.service` dürfen nicht ohne Prüfung angenommen werden.

---

14. macOS: VPN-Status und virtuelle Schnittstellen prüfen

VPN-Konfigurationen anzeigen:

```
[RO][SENS] scutil --nc list
```

Status einer bekannten VPN-Konfiguration:

```
[RO][SENS] scutil --nc status "VPN-DIENSTNAME"
```

Alle Schnittstellen anzeigen:

```
[RO][SENS] ifconfig
```

Routingtabelle:

```
[RO] netstat -rn
```

Standardroute:

```
[RO] route -n get default
```

Route zu einem internen Ziel:

```
[RO] route -n get 10.20.30.40
```

DNS-Konfiguration:

```
[RO][SENS] scutil --dns
```

Proxykonfiguration:

```
[RO][SENS] scutil --proxy
```

Viele Hersteller-VPN-Clients verwalten Tunnel und Filter über System Extensions oder Network Extensions. Ihre vollständige Konfiguration ist nicht zwingend über `networksetup` oder `scutil --nc` sichtbar.

---

15. WireGuard-Verbindungen prüfen

Die folgenden Befehle gelten nur, wenn WireGuard eingesetzt wird und die Werkzeuge installiert sind.

WireGuard-Status:

```
[RO][PRIV][SENS] sudo wg show
```

Typische Angaben:

```
interface
public key
listening port
peer
endpoint
allowed ips
latest handshake
transfer
persistent keepalive
```

Bestimmte Schnittstelle:

```
[RO][PRIV][SENS] sudo wg show wg0
```

Routing und Adresse der Schnittstelle:

```
[RO] ip address show dev wg0
```

```
[RO] ip route show
```

Besonders relevant:

Feld	Diagnose
`latest handshake`	Fand kürzlich ein Handshake statt?
`transfer`	Steigen gesendete und empfangene Datenmengen?
`endpoint`	Wird die erwartete Gegenstellenadresse verwendet?
`allowed ips`	Welche Zielnetze werden diesem Peer zugeordnet?
`persistent keepalive`	Ist bei NAT-Szenarien ein Keepalive konfiguriert?

Typische Muster:

Beobachtung	Mögliche Bedeutung
Kein aktueller Handshake	Endpoint, UDP-Port, Schlüssel, Firewall oder Gegenstelle prüfen
Gesendet steigt, empfangen nicht	Rückweg, Peer-Konfiguration oder Firewall prüfen
Handshake aktuell, internes Ziel nicht erreichbar	Routen, AllowedIPs, Forwarding und Zielnetz prüfen

Beobachtung	Mögliche Bedeutung
Nur nach Datenverkehr Handshake sichtbar	Kann je nach Nutzung normal sein
Tunnel funktioniert nur kurz	NAT-Zustand, Keepalive oder Netzwechsel prüfen

Konfigurationsausgaben, die private oder vorab geteilte Schlüssel enthalten könnten, dürfen nicht veröffentlicht werden.

---

16. OpenVPN-Verbindungen prüfen

Die folgenden Prüfungen gelten nur, wenn OpenVPN eingesetzt wird.

Installierte Version:

```
[RO] openvpn --version
```

Laufenden Prozess suchen:

```
[RO][SENS] ps aux | grep '[o]penvpn'
```

Ermittelte systemd-Dienste prüfen:

```
[RO] systemctl list-units --type=service |
    grep -i openvpn
```

Protokoll eines konkret ermittelten Dienstes:

```
[RO][PRIV][SENS] sudo journalctl -u TATSAECHLICHER_OPENVPN_DIENST \
    --since "-30 minutes"
```

Typische Meldungsgruppen:

Meldung	Mögliche Bedeutung
TLS handshake failed	Zertifikat, Uhrzeit, Transport oder Gegenstelle prüfen
AUTH_FAILED	Benutzeranmeldung, MFA oder Kontostatus prüfen
Cannot resolve host address	DNS-Auflösung des Gateways fehlgeschlagen

Meldung	Mögliche Bedeutung
Connection timed out	Transportweg, Firewall oder Gateway prüfen
Inactivity timeout	Kein ausreichender Datenverkehr oder Gegenstelle nicht erreichbar
Route addition failed	Rechte, Routenkonflikt oder Plattformproblem
TUN/TAP device error	Virtueller Adapter oder Berechtigung fehlerhaft
Initialization Sequence Completed	Tunnelinitialisierung wurde abgeschlossen

Die exakte Formulierung hängt von OpenVPN-Version, Plattform und Verwaltungslösung ab.

---

17. VPN-Adressierung prüfen

Nach erfolgreichem Aufbau erhält der Client abhängig vom VPN eine Adresse auf einer virtuellen Schnittstelle.

Windows:

```
[RO][SENS] Get-NetIPAddress |
    Sort-Object InterfaceAlias, AddressFamily |
    Format-Table InterfaceAlias, AddressFamily, IPAddress,
                 PrefixLength, AddressState
```

Linux:

```
[RO][SENS] ip -br address
```

macOS:

```
[RO][SENS] ifconfig
```

Zu kontrollieren sind:

- Existiert eine Tunneladresse?
- Stammt sie aus dem erwarteten Adressbereich?
- Ist die Präfixlänge korrekt?
- Existiert ein Adresskonflikt?
- Überschneidet sich das VPN-Netz mit dem lokalen Netz?

- Wird IPv4, IPv6 oder beides bereitgestellt?
- Ist die Adresse nur temporär oder instabil?
- Verwendet der Server die richtige Rückroute zum VPN-Adresspool?

---

18. Überschneidende lokale und entfernte Netze erkennen

Ein häufiges Remote-Access-Problem entsteht, wenn das lokale Netz denselben Adressbereich wie das entfernte Unternehmensnetz verwendet.

Beispiel:

```
Lokales Heimnetz:      192.168.1.0/24
Entferntes Firmennetz: 192.168.1.0/24
```

Der Client kann dann nicht allein anhand der Zieladresse eindeutig entscheiden, ob das Ziel lokal oder über das VPN erreichbar ist.

Route zum Ziel prüfen:

Windows:

```
[RO] Find-NetRoute -RemoteIPAddress 192.168.1.50
```

Linux:

```
[RO] ip route get 192.168.1.50
```

macOS:

```
[RO] route -n get 192.168.1.50
```

Mögliche Lösungen im Netzdesign:

- lokale oder entfernte Netze neu adressieren,
- spezifischere VPN-Routen verwenden,
- NAT im VPN-Konzept einsetzen,
- alternative interne Adressen bereitstellen,
- virtualisierte oder anwendungsbasierte Zugänge verwenden.

Eine manuell gesetzte Einzelroute ist höchstens eine kontrollierte Zwischenlösung und kann andere lokale Ziele unerreichbar machen.

---

19. Split Tunnel und Full Tunnel prüfen

Full Tunnel:

```
Interne Ziele → VPN
Internetverkehr → VPN
```

Split Tunnel:

```
Interne Ziele → VPN
Internetverkehr → lokale Verbindung
```

Windows-Profil prüfen:

```
[RO][SENS] Get-VpnConnection |
    Select-Object Name, ConnectionStatus, SplitTunneling,
                  TunnelType, ServerAddress
```

Standardrouten unter Windows:

```
[RO] Get-NetRoute -DestinationPrefix "0.0.0.0/0" |
    Format-Table InterfaceAlias, NextHop, RouteMetric
```

Linux:

```
[RO] ip route show
[RO] ip rule show
```

macOS:

```
[RO] netstat -rn
```

Zu prüfen sind:

- Existiert eine Standardroute über das VPN?
- Werden nur bestimmte interne Präfixe verteilt?
- Fehlt das Netz des betroffenen Dienstes?
- Existieren mehrere gleichwertige Routen?
- Wird Policy Routing verwendet?
- Überschreibt der VPN-Client lokale Routen?
- Bleibt der direkte Internetzugang laut Sollzustand erhalten?

Split Tunneling ist eine Sicherheits- und Architekturentscheidung und darf nicht eigenmächtig aktiviert oder deaktiviert werden.

---

20. Route zu einem betroffenen Ziel eindeutig bestimmen

Windows:

```
[RO] Find-NetRoute -RemoteIPAddress 10.20.30.40
```

Linux:

```
[RO] ip route get 10.20.30.40
```

macOS:

```
[RO] route -n get 10.20.30.40
```

Zu dokumentieren sind:

```
Zieladresse
verwendete Schnittstelle
nächster Hop
Quelladresse
Routenpräfix
Metrik
```

Typische Diagnose:

```
Tunnel ist verbunden
↓
Route zum Ziel zeigt auf WLAN statt VPN
↓
Benötigte VPN-Route fehlt oder wird durch spezifischere Route verdrängt
```

Die längste passende Präfixroute ist grundsätzlich besonders relevant. Eine spezifischere Route kann eine weniger spezifische VPN-Standardroute übersteuern.

---

21. Interne DNS-Auflösung über VPN prüfen

Windows-DNS-Server pro Schnittstelle:

```
[RO][SENS] Get-DnsClientServerAddress |
    Format-Table InterfaceAlias, AddressFamily, ServerAddresses
```

Windows-Suffixe:

```
[RO] Get-DnsClient |
    Select-Object InterfaceAlias, ConnectionSpecificSuffix,
                  RegisterThisConnectionsAddress
```

Linux:

```
[RO][SENS] resolvectl status
```

macOS:

```
[RO][SENS] scutil --dns
```

Internen Namen über den vorgesehenen DNS-Server prüfen:

Windows:

```
[TEST] Resolve-DnsName server.example.internal `
    -Server 10.20.0.53
```

Linux und macOS:

```
[TEST] dig @10.20.0.53 server.example.internal A
```

IP-Adresse und Hostname getrennt testen:

```
Zugriff über interne IP funktioniert
Zugriff über internen Hostnamen funktioniert nicht
```

Das deutet eher auf DNS, Suchsuffixe oder Namensrichtlinien als auf einen allgemeinen Tunnelfehler hin.

---

22. Split DNS und DNS-Leaks beurteilen

Bei Split DNS sollen bestimmte interne Namensräume über interne DNS-Server und andere Namen über lokale oder öffentliche Resolver aufgelöst werden.

Beispiel:

```
*.example.internal → interner DNS-Server über VPN
andere Namen       → lokaler oder definierter externer DNS-Server
```

Mögliche Fehler:

- interne Zone wird an öffentlichen DNS-Server gesendet,
- interner DNS-Server ist nicht über den Tunnel erreichbar,
- Suchsuffix fehlt,
- lokale Schnittstelle besitzt eine höhere DNS-Priorität,
- VPN-Client setzt DNS nur für IPv4,
- Browser verwendet DNS over HTTPS unabhängig vom Betriebssystem,
- ein lokaler Filter oder Proxy verändert die Namensauflösung.

Vergleichstest:

```
[TEST] dig server.example.internal
```

```
[TEST] dig @10.20.0.53 server.example.internal
```

Unterschiedliche Antworten zeigen, dass Resolverauswahl, Cache oder Split-DNS-Regeln weiter untersucht werden müssen.

---

23. Authentifizierung, Zertifikate und Uhrzeit prüfen

Mögliche Authentifizierungsbestandteile:

- Benutzername und Passwort,
- Multifaktor-Authentifizierung,
- Clientzertifikat,
- Gerätezertifikat,
- Pre-Shared Key,
- Smartcard,
- Geräteidentität,
- SAML- oder Browseranmeldung,
- Kerberos oder EAP.

Systemzeit prüfen:

Betriebssystem	Befehl
Windows	`[RO] Get-Date -Format o`
Linux	`[RO] date --iso-8601=seconds`
macOS	`[RO] date "+%Y-%m-%dT%H:%M:%S%z"`

Zeitsynchronisation:

Windows:

```
[RO] w32tm /query /status
```

Linux:

```
[RO] timedatectl status
```

macOS:

```
[RO] systemsetup -getusingnetworktime
```

Für den macOS-Befehl können erhöhte Rechte erforderlich sein:

```
[RO][PRIV] sudo systemsetup -getusingnetworktime
```

Zu prüfen sind:

- Zertifikat noch gültig?
- Zertifikat bereits gültig?
- Richtiger Aussteller?
- Private-Key-Zugriff vorhanden?
- Servername passt zum Zertifikat?
- Zertifikat wurde widerrufen?
- Benutzerkonto gesperrt?
- Passwort abgelaufen?
- MFA vollständig bestätigt?
- Gerätekonformität erfüllt?
- Systemzeit korrekt?

Private Schlüssel dürfen niemals mit Diagnoseausgaben exportiert werden.

---

24. MTU- und Fragmentierungsprobleme erkennen

VPN-Protokolle fügen zusätzliche Header hinzu. Dadurch sinkt die nutzbare Paketgröße innerhalb des Tunnels.

Typische Symptome:

- Tunnel verbindet sich erfolgreich.
- Ping mit kleinen Paketen funktioniert.
- Webseiten laden nur teilweise.
- Dateiübertragungen bleiben hängen.
- RDP oder SSH friert sporadisch ein.
- Kleine API-Anfragen funktionieren, große nicht.
- TLS-Handshake bleibt stehen.
- Bestimmte Netze oder Provider sind betroffen.

Windows – IPv4-Paket mit gesetztem Don't-Fragment-Bit:

```
[TEST] ping -4 -f -l 1400 10.20.30.40
```

Linux:

```
[TEST] ping -4 -M do -s 1400 10.20.30.40
```

macOS:

```
[TEST] ping -D -s 1400 10.20.30.40
```

Die angegebene Nutzdatenlänge ist nicht identisch mit der gesamten IP-Paketgröße. Bei IPv4 kommen üblicherweise mindestens IP- und ICMP-Header hinzu.

Schnittstellen-MTU anzeigen:

Windows:

```
[RO] Get-NetIPInterface |
    Format-Table InterfaceAlias, AddressFamily, NlMtu, InterfaceMetric
```

Linux:

```
[RO] ip link show
```

macOS:

```
[RO] ifconfig
```

Die MTU darf nicht ohne Messung und Freigabe verändert werden. Eine zu kleine MTU kann Leistung reduzieren; eine zu große MTU kann Fragmentierungs- oder Black-Hole-Probleme verursachen.

---

25. Path-MTU-Discovery und ICMP berücksichtigen

Path-MTU-Discovery benötigt bestimmte ICMP- beziehungsweise ICMPv6-Rückmeldungen.

Wichtige Meldungen:

- IPv4: Fragmentierung erforderlich
- IPv6: Packet Too Big

Werden diese Meldungen blockiert, kann ein Path-MTU-Black-Hole entstehen:

```
Kleine Pakete funktionieren
↓
Großes Paket überschreitet Pfad-MTU
↓
Router kann oder darf nicht fragmentieren
↓
ICMP-Fehlermeldung wird blockiert
↓
Sender erfährt die zulässige Größe nicht
↓
Übertragung bleibt hängen
```

Paketmitschnittfilter:

```
icmp
icmpv6
```

ICMP und ICMPv6 dürfen nicht pauschal als unnötig betrachtet oder vollständig blockiert werden.

---

26. Lokale Firewall und VPN-Kill-Switch prüfen

Ein VPN-Client kann zusätzliche Filterregeln installieren, um Verkehr außerhalb des Tunnels zu verhindern.

Mögliche Funktionen:

- Kill Switch,
- Always-On,
- Lockdown Mode,
- Blockierung lokaler Netze,
- Blockierung unverschlüsselten DNS-Verkehrs,

- anwendungsbasierte Filterung,
- Gerätetunnelrichtlinien,
- Filter über System- oder Network Extensions.

Typische Fehlerbilder:

- Nach VPN-Abbruch funktioniert kein Netzwerk mehr.
- Lokale Drucker sind während des VPNs nicht erreichbar.
- Internet funktioniert nur bei aktivem VPN.
- DNS bleibt nach dem Trennen fehlerhaft.
- Filterregeln bleiben nach Clientabsturz aktiv.
- Herstellerclient zeigt getrennt, Filterkomponente arbeitet aber weiter.

Zu prüfen sind:

- native Host-Firewall,
- Regeln des VPN-Clients,
- Sicherheitssoftware,
- Filtertreiber beziehungsweise System Extensions,
- Proxykonfiguration,
- DNS-Konfiguration,
- Standardrouten.

Kill-Switch- oder Sicherheitsfunktionen dürfen nicht ohne Freigabe deaktiviert werden.

---

27. Lokale Netzwerkzugriffe während des VPNs prüfen

Ein VPN kann Zugriffe auf lokale Netze absichtlich unterbinden.

Betroffene Beispiele:

- lokale Drucker,
- NAS-Systeme,
- Routeroberfläche,
- lokale Entwicklungsserver,
- Smartcard- oder Lizenzserver,
- Geräte im gleichen WLAN.

Route zum lokalen Ziel prüfen:

Windows:

```
[RO] Find-NetRoute -RemoteIPAddress 192.168.1.50
```

Linux:

```
[RO] ip route get 192.168.1.50
```

macOS:

```
[RO] route -n get 192.168.1.50
```

Mögliche Ursachen:

- Full Tunnel erfasst auch lokale Netze,
- Sicherheitsrichtlinie blockiert LAN-Zugriff,
- lokales und entferntes Netz überschneiden sich,
- VPN-Client setzt spezifische Blockierungsrouten,
- Host-Firewall behandelt den VPN-Adapter anders,
- lokales Ziel antwortet nicht auf die VPN-Quelladresse.

Lokalen LAN-Zugriff während des VPNs zu erlauben ist eine Sicherheitsentscheidung und keine rein technische Komforteinstellung.

---

28. VPN über restriktive Netze prüfen

Hotels, Gast-WLANs, Mobilfunkanbieter und Unternehmensnetze können bestimmte VPN-Protokolle einschränken.

Vergleichstest:

```
VPN über Firmennetz fehlerhaft
VPN über Mobilfunk erfolgreich
```

Das deutet auf einen Unterschied im Zugangsnetz hin, beispielsweise:

- UDP wird blockiert,
- IPsec ESP wird blockiert,
- NAT-T wird fehlerhaft behandelt,
- Captive Portal ist noch nicht bestätigt,

- TCP 443 ist nur über einen Proxy erlaubt,
- DNS-Antwort wird verändert,
- IPv6- und IPv4-Pfade unterscheiden sich,
- Provider verwendet problematisches NAT.

Öffentliche Quelladresse vergleichen:

Windows:

```
[TEST][SENS] Invoke-RestMethod -Uri "https://api.ipify.org"
```

Linux und macOS:

```
[TEST][SENS] curl -4 https://api.ipify.org
```

Die Nutzung eines externen Dienstes muss durch die betrieblichen Richtlinien erlaubt sein.

---

29. Stabilität und wiederkehrende Abbrüche untersuchen

Bei sporadischen Abbrüchen dokumentieren:

- exakten Zeitpunkt,
- Dauer bis zum Abbruch,
- Wechsel zwischen WLAN und Mobilfunk,
- Standby oder Bildschirmruhe,
- DHCP-Erneuerung,
- Änderung der öffentlichen Adresse,
- MFA- oder Sitzungslaufzeit,
- Rekey- beziehungsweise SA-Lebensdauer,
- NAT-Timeout,
- Paketverlust,
- Wechsel des Access Points,
- Energiesparzustand,
- parallelen VPN-Client,
- Anzahl gleichzeitiger Sitzungen.

Kontinuierlicher Test zu einem internen Ziel:

Windows:

```
[TEST] ping -t 10.20.30.40
```

Linux:

```
[TEST] ping 10.20.30.40
```

macOS:

```
[TEST] ping 10.20.30.40
```

Zusätzlich sollte ein Ziel außerhalb des VPNs getestet werden, um zwischen Verlust der Basisverbindung und Verlust des Tunnels zu unterscheiden.

Windows – zeitlich begrenzte PowerShell-Ausgabe:

```
[TEST] Test-Connection 10.20.30.40 -Count 20
```

Ping kann durch Richtlinien blockiert sein. Ein fehlender Ping ist deshalb kein alleiniger Nachweis eines Tunnelausfalls.

---

30. Anwendung statt nur Ping testen

Nach Routing- und DNS-Prüfung muss der benötigte Dienst getestet werden.

Dienst	Beispiel
HTTPS	`[TEST] curl -v https://service.example.internal/`
TCP-Port	`[TEST] Test-NetConnection HOST -Port PORT`
SSH	`[TEST] ssh -vvv user@HOST`
DNS	`[TEST] Resolve-DnsName HOST -Server DNS_SERVER`
SMB	Unter Windows: `[TEST] Test-NetConnection HOST -Port 445`
RDP	Unter Windows: `[TEST] Test-NetConnection HOST -Port 3389`

Linux und macOS – TCP-Port:

```
[TEST] nc -vz -w 5 HOST PORT
```

Reihenfolge:

```
Tunnelstatus
→ Route
→ DNS
→ Port
→ Anwendungsprotokoll
→ Authentifizierung
```

Ein erfolgreicher VPN-Aufbau beweist nicht, dass eine Zugriffsrichtlinie den jeweiligen Anwendungsdienst erlaubt.

---

31. Paketmitschnitt für VPN-Fehler erstellen

Es können zwei Datenebenen betrachtet werden:

```
Äußerer Verkehr:
Client ↔ VPN-Gateway

Innerer Verkehr:
Tunneladresse ↔ internes Ziel
```

Linux – äußerer Verkehr zum VPN-Gateway:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i PHYSISCHE_SCHNITTSTELLE -nn \
    'host VPN_GATEWAY_IP' \
    -w vpn-outer.pcap
```

Linux – innerer Verkehr auf einer Tunnelschnittstelle:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i TUNNEL_INTERFACE -nn \
    'host INTERNES_ZIEL' \
    -w vpn-inner.pcap
```

macOS:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i INTERFACE -nn \
    'host VPN_GATEWAY_IP' \
    -w vpn-test.pcap
```

Windows – Schnittstellen mit Dumpcap anzeigen:

```
[RO] dumpcap -D
```

Windows – Mitschnitt:

```
[TEST][PRIV][FILE][SENS] dumpcap -i 1 `
    -f "host VPN_GATEWAY_IP" `
    -w vpn-test.pcapng
```

Die Schnittstellen müssen vorher eindeutig ermittelt werden.

---

32. Wireshark-Filter für VPN-Diagnosen

Aufgabe	Anzeigefilter
IKE	`isakmp`
UDP 500	`udp.port == 500`
NAT-T über UDP 4500	`udp.port == 4500`
ESP	`esp`
AH	`ah`
OpenVPN auf Standardport	`udp.port == 1194
WireGuard auf häufigem Port	`udp.port == 51820`
SSTP-Verkehr	`tcp.port == 443`
DNS	`dns`
ICMP	`icmp`
ICMPv6	`icmpv6`
TCP-Reset	`tcp.flags.reset == 1`
Wiederholte TCP-Pakete	`tcp.analysis.retransmission`

Die tatsächlichen Ports können abweichen und müssen aus der Konfiguration übernommen werden.

Verschlüsselter VPN-Verkehr zeigt im äußeren Mitschnitt normalerweise nicht die ursprünglichen Anwendungsdaten.

---

33. Paketmuster interpretieren

Beobachtung	Mögliche Interpretation
Keine Pakete zum VPN-Gateway	DNS, Route, Anwendung oder lokaler Filter prüfen
Ausgehende Pakete, keine Antwort	Gateway, Firewall, Provider, NAT oder Rückweg prüfen
IKE-Austausch beginnt und endet	Authentifizierung, Richtlinie oder Verfahren prüfen
Tunnelhandshake erfolgreich	Transport und kryptografischer Aufbau funktionieren grundsätzlich
Tunnelverkehr wird gesendet, nichts empfangen	Gegenstelle, Rückroute oder Filter prüfen
Inneres Paket verlässt Tunnel, keine Antwort	Zielnetz, Zielhost oder serverseitige Route prüfen
Antworten kommen im Tunnel an, Anwendung erhält sie nicht	Lokale Firewall, Route oder Anwendung prüfen
Kleine Pakete funktionieren, große nicht	MTU und Path-MTU-Discovery prüfen
Häufige Neuverbindungen	Basisnetz, Rekey, NAT-Timeout oder Clientstabilität prüfen

Ein einseitiger Paketmitschnitt zeigt nur den jeweiligen Messpunkt. Bei komplexen Fehlern sollten Client, VPN-Gateway und internes Ziel zeitgleich betrachtet werden.

---

34. Typische Fehlerbilder systematisch eingrenzen

Fall A – VPN-Gatewayname wird nicht aufgelöst

Prüfen:

- DNS-Server vor dem VPN,
- Gatewayhostname,
- Suchsuffix,
- Captive Portal,
- Proxy,
- IPv4- und IPv6-Antworten.

Fall B – Authentifizierung schlägt fehl

Prüfen:

- Benutzerkonto,
- Passwortablauf,
- MFA,
- Zertifikat,
- Systemzeit,
- Geräteidentität,
- Kontosperre,
- Richtlinie,
- maximal zulässige Sitzungen.

Fall C – VPN verbunden, keine internen Ziele erreichbar

Prüfen:

- Tunneladresse,
- Routen,
- Split Tunnel,
- Netzüberlappung,
- lokale Firewall,
- zentrale VPN-Richtlinie,
- serverseitige Rückroute,
- DNS.

Fall D – Interne IP funktioniert, Hostname nicht

Prüfen:

- interne DNS-Server,
- DNS-Suffix,
- Split DNS,
- DNS-Priorität,
- DNS over HTTPS,
- VPN-DNS-Richtlinie.

Fall E – Nur ein internes Netz ist nicht erreichbar

Prüfen:

- fehlende Route,
- falsches Präfix,
- Access Control,
- Rückroute,
- Firewall zwischen VPN-Pool und Zielnetz,
- überlappende Netze.

Fall F – VPN funktioniert über Mobilfunk, aber nicht über WLAN

Prüfen:

- Captive Portal,
- UDP-Blockierung,
- lokales NAT,
- Router-Firewall,
- DNS-Manipulation,
- IPsec-Passthrough,
- IPv4-/IPv6-Unterschiede.

Fall G – Verbindung bricht bei großen Übertragungen ab

Prüfen:

- MTU,
- ICMP-Blockierung,
- Paketverlust,
- Time-outs,
- Rekey,
- Überlastung,
- instabile Basisverbindung.

Fall H – Nach dem Trennen funktioniert das Internet nicht

Prüfen:

- Standardroute,
- DNS-Server,
- Proxy,
- Kill Switch,
- Firewallfilter,
- virtuellen Adapter,
- fehlerhaft beendeten Herstellerclient.

---

35. VPN-Serverseite mit einbeziehen

Die Clientdiagnose allein reicht nicht immer aus. Auf der Serverseite sind zu prüfen:

- erreicht der Verbindungsversuch das Gateway,
- wird die Authentifizierung akzeptiert,
- wird eine Tunneladresse vergeben,
- wird das richtige Benutzer- oder Geräteprofil angewendet,
- werden die vorgesehenen Routen verteilt,
- werden DNS-Server und Suffixe verteilt,
- greift eine Zugriffsrichtlinie,
- existiert eine Route zum VPN-Clientpool,
- kennt das interne Netz den Rückweg,
- arbeitet NAT wie vorgesehen,
- ist der Adresspool erschöpft,
- wird eine maximale Sitzungszahl erreicht,
- sind Zertifikate gültig,
- sind Zeit und Zeitzone synchron,
- treten Rekey- oder SA-Fehler auf,
- ist der Gateway-Cluster vollständig verfügbar.

Client- und Serverprotokolle müssen anhand von Benutzer, Tunneladresse und Zeitstempel zugeordnet werden.

---

36. Keine vorschnellen VPN-Änderungen durchführen

Nicht unkontrolliert:

- VPN-Profil löschen,
- Zertifikate entfernen,
- private Schlüssel exportieren,
- Split Tunneling aktivieren,
- Kill Switch deaktivieren,
- Firewall ausschalten,
- MTU beliebig verändern,
- manuelle Standardroute hinzufügen,
- DNS-Server dauerhaft überschreiben,
- VPN-Client neu installieren,
- Security Associations oder Zustandstabellen leeren,
- mehrere VPN-Clients gleichzeitig starten.

Sicherer Ablauf:

1. Istzustand dokumentieren.
2. Client- und Profilversion feststellen.
3. Fehler reproduzieren.
4. Basisnetz und Gatewaytransport prüfen.
5. Tunnelstatus, Adresse, Route und DNS prüfen.
6. Client- und Serverprotokolle vergleichen.
7. Ursache nachweisen.
8. Änderung genehmigen lassen.
9. Kleinste erforderliche Korrektur durchführen.
10. Tunnelaufbau und Zielanwendung erneut testen.
11. Rückfallmöglichkeit und Ergebnis dokumentieren.

---

37. Systematischer Diagnoseablauf

Schritt	Prüfung	Leitfrage
1	Störungsumfang bestimmen	Benutzer, Gerät, Standort oder allgemeine Störung?
2	Basisverbindung prüfen	Funktioniert das Netz ohne VPN?
3	Captive Portal ausschließen	Ist der Internetzugang vollständig freigeschaltet?
4	Gateway auflösen	Wird die richtige VPN-Adresse verwendet?
5	Route zum Gateway prüfen	Läuft der äußere Verkehr über die Basisschnittstelle?
6	Transport prüfen	Erreichen benötigte Ports und Protokolle das Gateway?
7	Authentifizierung prüfen	Werden Konto, MFA und Zertifikat akzeptiert?
8	Tunnelstatus prüfen	Wurde der Tunnel tatsächlich aufgebaut?
9	Tunneladresse prüfen	Hat der Client die vorgesehene Adresse erhalten?
10	Routen vergleichen	Werden die Zielnetze durch den Tunnel geleitet?
11	Netzüberlappungen prüfen	Konfligiert das lokale Netz mit dem Zielnetz?
12	DNS prüfen	Werden interne Namen über richtige Resolver aufgelöst?

Schritt	Prüfung	Leitfrage
13	Zielport testen	Ist der benötigte Dienst transportseitig erreichbar?
14	Anwendung testen	Funktioniert das eigentliche Protokoll?
15	MTU prüfen	Scheitern nur größere Pakete oder Übertragungen?
16	Stabilität beobachten	Wann und unter welchen Bedingungen bricht der Tunnel ab?
17	Protokolle vergleichen	Welche Seite beendet oder verweigert die Verbindung?
18	Paketmitschnitt erstellen	Wo endet der erfolgreiche Datenfluss?
19	Minimale Änderung planen	Welche konkrete Korrektur stellt den Sollzustand her?
20	Nachprüfung	Funktionieren Tunnel, DNS, Routing und Anwendung?

38. Kompakte Befehlstabelle

Aufgabe	Windows	Linux	macOS
VPN-Profile	`[RO][SENS] Get-VpnConnection`	`[RO][SENS] nmcli connection show`	`[RO][SENS] scutil --nc list`
Geräteweite Profile	`[RO][PRIV][SENS] Get-VpnConnection -AllUserConnection`	Abhängig vom VPN-Client	Abhängig von MDM und VPN-Client
Aktiver VPN-Status	`[RO] rasdial`	`[RO][SENS] nmcli connection show --active`	`[RO][SENS] scutil --nc status "VPN-NAME"`
Schnittstellen	`[RO] Get-NetAdapter -IncludeHidden`	`[RO] ip -br link`	`[RO] ifconfig`
IP-Adressen	`[RO][SENS] Get-NetIPAddress`	`[RO][SENS] ip -br address`	`[RO][SENS] ifconfig`
Routingtabelle	`[RO] Get-NetRoute`	`[RO] ip route show`	`[RO] netstat -rn`
Route zum Ziel	`[RO] Find-NetRoute -RemoteIPAddress IP`	`[RO] ip route get IP`	`[RO] route -n get IP`
DNS-Konfiguration	`[RO][SENS] Get-DnsClientServerAddress`	`[RO][SENS] resolvectl status`	`[RO][SENS] scutil --dns`
Gateway auflösen	`[TEST] Resolve-DnsName VPN_HOST`	`[TEST] getent ahosts VPN_HOST`	`[TEST] dscacheutil -q host -a name VPN_HOST`
TCP-Gatewayport	`[TEST] Test-NetConnection VPN_HOST -Port PORT`	`[TEST] nc -vz -w 5 VPN_HOST PORT`	`[TEST] nc -vz -w 5 VPN_HOST PORT`

Aufgabe	Windows	Linux	macOS
Internen Port testen	`[TEST] Test-NetConnection INTERNER_HOST -Port PORT`	`[TEST] nc -vz -w 5 INTERNER_HOST PORT`	`[TEST] nc -vz -w 5 INTERNER_HOST PORT`
MTU anzeigen	`[RO] Get-NetIPInterface`	`[RO] ip link show`	`[RO] ifconfig`
DF-Test	`[TEST] ping -4 -f -l 1400 IP`	`[TEST] ping -4 -M do -s 1400 IP`	`[TEST] ping -D -s 1400 IP`
WireGuard	Falls installiert: `[RO][PRIV][SENS] wg show`	`[RO][PRIV][SENS] sudo wg show`	Falls installiert: `[RO][PRIV][SENS] sudo wg show`
NetworkManager-Protokoll	Nicht zutreffend	`[RO][PRIV][SENS] sudo journalctl -u NetworkManager --since "-30 minutes"`	Nicht zutreffend
RAS-Protokoll	`[RO][PRIV][SENS] Get-WinEvent -LogName "Microsoft-Windows-RasClient/Operational"`	Nicht zutreffend	Nicht zutreffend

`VPN-NAME`, `IP`, `VPN_HOST`, `PORT` und `INTERNER_HOST` müssen durch die Werte des konkreten Störungsfalls ersetzt werden.

---

39. Dokumentationsvorlage für VPN-Fehler

Störung:

Zeitpunkt:

Zeitzone:

Benutzer:

Gerät:

Betriebssystem:

VPN-Client:

VPN-Clientversion:

VPN-Profil:

VPN-Technik:

Zugangsnetz:

Zugangsart: LAN / WLAN / Mobilfunk

Lokale IP-Adresse:

Standardgateway:

Öffentliche IP-Adresse:

Captive Portal ausgeschlossen:

Proxy vorhanden:

VPN-Gatewayname:
Aufgelöste Gatewayadresse:
Verwendete Adressfamilie:
Gatewayport beziehungsweise IP-Protokoll:
Route zum Gateway:
Transporttest:
Authentifizierungsverfahren:
MFA:
Zertifikat geprüft:
Systemzeit geprüft:

Tunnelstatus:
Tunneladresse:
Virtuelle Schnittstelle:
Split Tunnel oder Full Tunnel:
Verteilte IPv4-Routen:
Verteilte IPv6-Routen:
Route zum betroffenen Ziel:
Lokale Netzüberlappung:
VPN-DNS-Server:
DNS-Suffixe:
Interne Namensauflösung:

Betroffenes Ziel:
Ziel-IP:
Zielport:
Transportprotokoll:
Porttest:
Anwendungstest:
MTU-Test:
Paketverlust:
Abbruchzeitpunkt:

Clientprotokoll:
Gatewayprotokoll:
Serverseitige Richtlinie:

Rückroute zum VPN-Adresspool:

Paketmitschnitt vorhanden:

Vergleich über anderes Zugangsnetz:

Vergleich mit anderem Benutzer:

Vergleich mit anderem Gerät:

Festgestellte Ursache:

Genehmigte Änderung:

Rückfallplan:

Ergebnis der Nachprüfung:

---

40. Kontrollfragen nach der Diagnose

- Funktioniert das Netzwerk ohne VPN?
- Wurde ein Captive Portal ausgeschlossen?
- Wird der VPN-Gatewayname richtig aufgelöst?
- Führt die Route zum Gateway über die physische Basisschnittstelle?
- Ist das benötigte Transportprotokoll erreichbar?
- Werden Benutzer, Gerät, MFA und Zertifikat akzeptiert?
- Ist der Tunnel wirklich aufgebaut?
- Hat der Client eine gültige Tunneladresse erhalten?
- Existieren die benötigten Routen?
- Führt die Route zum Ziel tatsächlich über das VPN?
- Überschneiden sich lokales und entferntes Netz?
- Ist Full Tunnel oder Split Tunnel vorgesehen?
- Werden die richtigen DNS-Server und Suffixe verwendet?
- Funktioniert das Ziel über IP-Adresse?
- Funktioniert das Ziel über Hostname?
- Ist nur ein bestimmter Port oder Dienst betroffen?
- Wurden IPv4 und IPv6 getrennt geprüft?
- Gibt es Hinweise auf ein MTU-Problem?
- Funktioniert das VPN über ein anderes Zugangsnetz?
- Wurde ein Kill Switch beziehungsweise lokaler Filter berücksichtigt?
- Wurde der serverseitige Rückweg zum VPN-Adresspool geprüft?
- Wurden Client- und Gatewayprotokolle zeitlich abgeglichen?
- Wurde die Ursache nachgewiesen, bevor Einstellungen verändert wurden?
- Wurde nach der Änderung der gesamte Datenfluss erneut geprüft?

---

41. Quellen und weiterführende Dokumentation

- Microsoft Learn – Get-VpnConnection: https://learn.microsoft.com/powershell/module/vpnclient/get-vpnconnection
- Microsoft Learn – Windows-VPN-Client und VPN-Protokolle: https://learn.microsoft.com/windows-server/remote/remote-access/vpn/vpn-top
- Microsoft Learn – Always-On-VPN: https://learn.microsoft.com/windows-server/remote/remote-access/tutorial-aovpn-deploy-setup
- Microsoft Learn – Windows-VPN-Gerätetunnel: https://learn.microsoft.com/windows-server/remote/remote-access/vpn/vpn-device-tunnel-config
- Microsoft Learn – Get-NetIPsecMainModeSA: https://learn.microsoft.com/powershell/module/netsecurity/get-netipsecmainmodesa
- Microsoft Learn – Get-NetIPsecQuickModeSA: https://learn.microsoft.com/powershell/module/netsecurity/get-netipsecquickmodesa
- Apple – VPN-Verbindung auf dem Mac konfigurieren: https://support.apple.com/de-de/guide/mac-help/mchlp2963/mac
- Apple – VPN-Einstellungen auf dem Mac ändern: https://support.apple.com/de-de/guide/mac-help/mchlp1579/mac
- WireGuard – Offizielle Kurzanleitung: https://www.wireguard.com/quickstart/
- WireGuard – Protokollbeschreibung: https://www.wireguard.com/protocol/
- OpenVPN – Offizielle Dokumentation: https://openvpn.net/community-resources/reference-manual-for-openvpn-2-6/
- NetworkManager – nmcli-Dokumentation: https://networkmanager.dev/docs/api/latest/nmcli.html
- strongSwan – Dokumentation: https://docs.strongswan.org/docs/latest/
- RFC 7296 – Internet Key Exchange Protocol Version 2: https://www.rfc-editor.org/rfc/rfc7296.html
- RFC 3947 – Negotiation of NAT-Traversal in IKE: https://www.rfc-editor.org/rfc/rfc3947.html
- RFC 3948 – UDP Encapsulation of IPsec ESP Packets: https://www.rfc-editor.org/rfc/rfc3948.html
- RFC 4301 – Security Architecture for the Internet Protocol: https://www.rfc-editor.org/rfc/rfc4301.html
- RFC 8201 – Path MTU Discovery for IPv6: https://www.rfc-editor.org/rfc/rfc8201.html