3.11 NAT- und Portweiterleitungsfehler analysieren

Network Address Translation verändert IP-Adressen und teilweise auch Portnummern während der Übertragung. NAT wird unter anderem an Internetroutern, Firewalls, VPN-Gateways, Virtualisierungshosts und Containerplattformen eingesetzt.

Eine Portweiterleitung kann korrekt eingetragen aussehen und trotzdem nicht funktionieren. Neben der eigentlichen NAT-Regel müssen Routing, Firewall, öffentlicher Anschluss, Rückweg und Zielanwendung geprüft werden.

Die zentralen Fragen dieser Seite lauten:

> Erreicht das Paket das NAT-System, wird es korrekt übersetzt und kann das Zielsystem über einen gültigen Rückweg antworten?

---

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

NAT-Tabellen und Routerkonfigurationen können interne Adressen, öffentliche Adressen und erreichbare Dienste offenlegen. Ausgaben dürfen nicht ungeprüft veröffentlicht werden.

---

2. NAT-Grundbegriffe unterscheiden

Begriff	Bedeutung
NAT	Allgemeiner Begriff für die Übersetzung von Netzwerkadressen
Source NAT – SNAT	Verändert die Quelladresse eines Pakets
Destination NAT – DNAT	Verändert die Zieladresse eines Pakets
PAT	Verändert zusätzlich Portnummern; mehrere Verbindungen können eine öffentliche Adresse teilen
Masquerading	Dynamische Form von SNAT, häufig mit der Adresse der ausgehenden Schnittstelle
Portweiterleitung	Eingehender Port wird an eine interne Adresse und gegebenenfalls einen anderen Port weitergeleitet
Redirect	Verkehr wird auf das NAT-System selbst beziehungsweise einen lokalen Port umgeleitet
Hairpin NAT	Interne Clients greifen über die öffentliche Adresse auf einen ebenfalls internen Dienst zu
Static NAT	Feste Zuordnung zwischen Adressen
Dynamic NAT	Dynamische Zuordnung aus einem Adresspool
Double NAT	Zwei aufeinanderfolgende Systeme führen NAT durch
CGNAT	Provider übersetzt mehrere Kundenanschlüsse hinter gemeinsamen öffentlichen Adressen
NAT-Tabelle	Regel- oder Zustandsspeicher für Übersetzungen
Conntrack	Zustandsverfolgung von Netzwerkverbindungen

Wichtig: Eine Portweiterleitung ist keine automatische Bestätigung, dass der Verkehr auch durch die Firewall erlaubt wird.

---

3. SNAT, DNAT und PAT am Beispiel

Ausgehende Verbindung mit SNAT beziehungsweise PAT:

```
Interner Client
192.168.10.25:53144
        ↓
Internetrouter
        ↓
Öffentliche Darstellung
```

```
203.0.113.10:62001
        ↓
Webserver
198.51.100.20:443
```

Der Webserver sieht als Quelle normalerweise die öffentliche Adresse des NAT-Systems und nicht unmittelbar die private Clientadresse.

Eingehende Portweiterleitung mit DNAT:

```
Externer Client
198.51.100.25:54000
        ↓
203.0.113.10:8443
Öffentliche Routeradresse und externer Port
        ↓ DNAT
192.168.10.50:443
Interner Server und interner Port
```

Die vollständige Zuordnung lautet in diesem Beispiel:

```
TCP 203.0.113.10:8443
→ TCP 192.168.10.50:443
```

Diese Angaben müssen getrennt dokumentiert werden:

- externe IP-Adresse,
- externer Port,
- Transportprotokoll,
- interne Zieladresse,
- interner Zielport,
- erlaubte Quelladressen,
- verwendete Schnittstelle,
- zugehörige Firewallregel.

---

4. Private, öffentliche und gemeinsam genutzte Adressen erkennen

Private IPv4-Adressbereiche nach RFC 1918:

Bereich	CIDR
`10.0.0.0` bis `10.255.255.255`	`10.0.0.0/8`
`172.16.0.0` bis `172.31.255.255`	`172.16.0.0/12`
`192.168.0.0` bis `192.168.255.255`	`192.168.0.0/16`

Shared Address Space für Carrier-Grade NAT:

```
100.64.0.0/10
```

Das entspricht:

```
100.64.0.0 bis 100.127.255.255
```

Weitere nicht öffentlich weiterleitbare Beispiele:

Bereich	Verwendung
`127.0.0.0/8`	IPv4-Loopback
`169.254.0.0/16`	IPv4-Link-Local
`224.0.0.0/4`	IPv4-Multicast
`::1/128`	IPv6-Loopback
`fe80::/10`	IPv6-Link-Local
`fc00::/7`	IPv6 Unique Local Addresses

Eine Adresse aus einem privaten oder CGNAT-Bereich auf der WAN-Seite des eigenen Routers ist ein deutlicher Hinweis darauf, dass noch ein vorgeschaltetes NAT-System existiert.

---

5. Häufige Ursachen einer fehlerhaften Portweiterleitung

Ursache	Typische Wirkung
Falsche öffentliche IP-Adresse	Test erreicht nicht den vorgesehenen Router
Dynamische öffentliche Adresse hat sich geändert	Alter DNS-Eintrag oder alte Dokumentation zeigt auf falsches Ziel
CGNAT beim Provider	Eingehende Verbindungen erreichen den Kundenrouter nicht direkt
Double NAT	Weiterleitung existiert nur auf einem von zwei Routern

Ursache	Typische Wirkung
Falsches Transportprotokoll	TCP wurde weitergeleitet, Dienst benötigt aber UDP oder umgekehrt
Falscher externer Port	Client spricht einen anderen Port an
Falscher interner Port	Weiterleitung endet nicht am Listener der Anwendung
Falsche interne Zieladresse	Serveradresse wurde geändert oder per DHCP neu vergeben
Dienst lauscht nur auf Loopback	NAT erreicht den Server, aber nicht den Dienst
Dienst läuft nicht	Zielport ist geschlossen
Host-Firewall blockiert	Paket erreicht das Zielsystem, wird dort aber verworfen
Router-Firewall blockiert	DNAT-Regel allein reicht nicht aus
Fehlender Rückweg	Antwort nimmt einen anderen oder ungültigen Weg
Hairpin NAT fehlt	Interner Test über öffentliche Adresse schlägt fehl, externer Zugriff kann trotzdem funktionieren
Quelladressbeschränkung	Nur bestimmte externe Netze dürfen zugreifen
Falsche Schnittstelle	Regel gilt nicht für die tatsächliche WAN-Schnittstelle
IPv4 und IPv6 verwechselt	IPv4-Portweiterleitung beeinflusst eine direkte IPv6-Verbindung nicht
Mehrfach belegter externer Port	Regelkonflikt oder falsches Ziel
VPN verändert den Pfad	Anfrage oder Antwort läuft über einen Tunnel
Containerport nicht veröffentlicht	NAT endet am Host, aber nicht am Container

6. Portweiterleitung vollständig dokumentieren

Eine Beschreibung wie „Port 443 ist freigegeben“ ist nicht ausreichend.

Vollständige Beschreibung:

```
Bezeichnung:       Externer Webzugriff
Adressfamilie:     IPv4
WAN-Schnittstelle: WAN
Quelladresse:      beliebig oder definierter Bereich
Öffentliche IP:    203.0.113.10
Transport:         TCP
```

```
Externer Port:     8443
Internes Ziel:     192.168.10.50
Interner Port:     443
Firewallaktion:    Erlauben
Protokollierung:   Aktiviert
Zeitplan:          Dauerhaft
```

Zusätzlich sollte dokumentiert werden:

- Hersteller und Modell des NAT-Systems,
- Firmware- oder Betriebssystemversion,
- zuständige Konfigurationsoberfläche,
- Zeitpunkt der letzten Änderung,
- genehmigender Verantwortlicher,
- zugehörige DNS-Namen,
- Rückfallplan,
- Testergebnis von intern und extern.

---

7. Zuerst den internen Zieldienst prüfen

Bevor NAT untersucht wird, muss der Dienst ohne Portweiterleitung direkt im internen Netz funktionieren.

Listener auf dem Zielsystem prüfen:

Aufgabe	Windows	Linux	macOS
TCP-Listener	`[RO] Get-NetTCPConnection -State Listen`	`[RO] ss -ltn`	`[RO] lsof -nP -iTCP -sTCP:LISTEN`
UDP-Endpunkte	`[RO] Get-NetUDPEndpoint`	`[RO] ss -lun`	`[RO] lsof -nP -iUDP`
TCP-Port 443	`[RO] Get-NetTCPConnection -State Listen -LocalPort 443`	`[RO] ss -ltn 'sport = :443'`	`[RO] lsof -nP -iTCP:443 -sTCP:LISTEN`

Dienst direkt über seine interne Adresse testen:

Windows:

```
[TEST] Test-NetConnection 192.168.10.50 -Port 443 -InformationLevel Detailed
```

Linux und macOS:

```
[TEST] nc -vz -w 5 192.168.10.50 443
```

Anwendungstest:

```
[TEST] curl -v https://192.168.10.50/
```

Bei HTTPS kann ein Zertifikatsfehler auftreten, wenn das Zertifikat für einen Hostnamen und nicht für die IP-Adresse ausgestellt wurde. Für eine korrekte Anwendungsprüfung sollte möglichst der vorgesehene Hostname verwendet werden.

Entscheidung:

```
Interner Zugriff funktioniert nicht
↓
Zuerst Dienst, Bind-Adresse, lokale Firewall und internes Routing prüfen
↓
NAT noch nicht als Hauptursache behandeln
```

---

8. Statische interne Zieladresse sicherstellen

Eine Portweiterleitung verweist normalerweise auf eine konkrete interne IP-Adresse. Ändert sich diese Adresse durch DHCP, zeigt die Regel anschließend möglicherweise auf das falsche System.

Windows:

```
[RO] Get-NetIPConfiguration
```

```
[RO] Get-NetIPAddress -AddressFamily IPv4 |
    Format-Table InterfaceAlias, IPAddress, PrefixLength, PrefixOrigin
```

Linux:

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

```
[RO] ip address show
```

macOS:

```
[RO] ifconfig
```

```
[RO] networksetup -listallhardwareports
```

Zu prüfen sind:

- Stimmt die aktuelle Serveradresse mit dem NAT-Ziel überein?
- Ist die Adresse statisch konfiguriert?
- Existiert eine DHCP-Reservierung?
- Liegt eine doppelte IP-Adresse vor?
- Wurde der Server in ein anderes VLAN verschoben?
- Stimmt das Präfix beziehungsweise die Subnetzmaske?
- Ist das Standardgateway korrekt?

Eine statische Adresse außerhalb des verwalteten Adressplans kann ebenfalls zu Konflikten führen. Bevorzugt wird eine dokumentierte statische Zuweisung oder DHCP-Reservierung entsprechend der Unternehmensvorgaben.

---

9. Öffentliche Adresse des Anschlusses bestimmen

Zuerst wird die WAN-Adresse direkt in der Verwaltungsoberfläche des Routers oder der Firewall abgelesen. Anschließend kann sie mit der von einem externen Dienst erkannten Adresse verglichen werden.

Lokale Adressen anzeigen:

Windows:

```
[RO] Get-NetIPAddress |
    Format-Table InterfaceAlias, AddressFamily, IPAddress, PrefixLength
```

Linux:

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

macOS:

```
[RO] ifconfig
```

Lokale Befehle zeigen auf einem Client hinter NAT normalerweise nicht die öffentliche IPv4-Adresse des Internetanschlusses.

Öffentlich sichtbare IPv4-Adresse über einen externen Dienst abfragen:

Windows:

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

Linux und macOS:

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

Öffentlich sichtbare IPv6-Adresse:

```
[TEST][SENS] curl -6 https://api64.ipify.org
```

Dabei wird die öffentliche Quelladresse an einen externen Dienst übertragen. Die Nutzung muss durch die betrieblichen Richtlinien erlaubt sein.

Auswertung:

Vergleich	Mögliche Bedeutung
Router-WAN-Adresse entspricht externer IPv4-Adresse	Router besitzt wahrscheinlich direkt diese öffentliche IPv4-Adresse
Router-WAN-Adresse ist privat	Vorgeschalteter Router oder Provider-NAT vorhanden
Router-WAN-Adresse liegt in `100.64.0.0/10`	CGNAT ist wahrscheinlich
Router-WAN-Adresse und erkannte öffentliche Adresse unterscheiden sich	Weiteres NAT-System oder Proxy vorhanden
Keine öffentliche IPv4-Adresse vorhanden	Anschluss kann IPv6, DS-Lite oder andere Providertechnik verwenden

Die genaue Anschlussart muss anhand der Router- und Providerinformationen bestätigt werden.

---

10. CGNAT, DS-Lite und Double NAT erkennen

Double NAT:

```
Internet
  ↓
Router oder Modemrouter 1
  ↓ privates Netz
Router 2
  ↓ privates Netz
Interner Server
```

Für eine klassische eingehende IPv4-Verbindung kann eine Weiterleitung auf beiden NAT-Systemen erforderlich sein.

Carrier-Grade NAT:

```
Internet
  ↓
Öffentliche Provideradresse
  ↓
Provider-NAT
  ↓
Kundenrouter mit Adresse aus 100.64.0.0/10
```

Eine Portweiterleitung auf dem Kundenrouter kann das vorgeschaltete Provider-NAT nicht automatisch konfigurieren.

DS-Lite:

Bei DS-Lite wird IPv4-Verkehr über eine Providerinfrastruktur transportiert und typischerweise auf Providerseite übersetzt. Eine frei erreichbare öffentliche IPv4-Adresse steht dem eigenen Router dabei häufig nicht direkt zur Verfügung.

Prüfschritte:

1. WAN-Adresse des Routers ablesen.
2. Extern erkannte IPv4-Adresse abfragen.
3. Adressen miteinander vergleichen.
4. Prüfen, ob die WAN-Adresse privat oder aus `100.64.0.0/10` ist.
5. Anschlussinformationen des Providers kontrollieren.
6. Prüfen, ob eine öffentliche IPv4-Adresse gebucht oder verfügbar ist.
7. Alternativen wie IPv6, VPN, Reverse Tunnel oder Providerfreigabe nur nach Sicherheitsprüfung planen.

---

11. Portweiterleitung wirklich von außen testen

Ein Test aus demselben LAN über die öffentliche Adresse ist kein zuverlässiger Ersatz für einen externen Test. Er hängt von der Unterstützung für Hairpin NAT ab.

Geeignete externe Testquellen sind beispielsweise:

- freigegebener Administrationsanschluss an einem anderen Standort,
- Mobilfunkverbindung ohne Verbindung zum Firmen-WLAN,
- freigegebener externer Monitoring-Server,
- autorisiertes System in einem anderen Netz,
- definierter Prüfserver in einer Cloudumgebung.

Externer TCP-Test unter Windows:

```
[TEST] Test-NetConnection 203.0.113.10 -Port 8443 -InformationLevel Detailed
```

Externer TCP-Test unter Linux oder macOS:

```
[TEST] nc -vz -w 5 203.0.113.10 8443
```

Anwendungstest mit Hostname:

```
[TEST] curl -v https://service.example.net:8443/
```

Gezielter Nmap-Test:

```
[TEST] nmap -sT -p 8443 203.0.113.10
```

Tests dürfen nur gegen die eigene beziehungsweise ausdrücklich freigegebene Infrastruktur ausgeführt werden.

---

12. Hairpin NAT richtig beurteilen

Hairpin NAT wird benötigt, wenn ein interner Client über die öffentliche Adresse oder den öffentlichen DNS-Namen auf einen internen Server zugreift.

```
Interner Client
    ↓
öffentliche Adresse des eigenen Routers
    ↓
Portweiterleitung
    ↓
interner Server im gleichen Standort
```

Typisches Fehlerbild:

```
Zugriff über Mobilfunk funktioniert
Zugriff aus dem internen LAN über öffentliche Adresse funktioniert nicht
Direkter Zugriff über interne Adresse funktioniert
```

Das deutet auf fehlendes oder fehlerhaftes Hairpin NAT hin. Es beweist nicht, dass die allgemeine externe Portweiterleitung defekt ist.

Mögliche Lösungen abhängig von der Infrastruktur:

- Hairpin NAT korrekt konfigurieren,
- internes und externes DNS unterschiedlich beantworten,
- internen Clients die interne Serveradresse liefern,
- Reverse Proxy mit passender interner Erreichbarkeit verwenden.

Änderungen an DNS oder NAT müssen zum vorgesehenen Netzdesign passen.

---

13. Split DNS als Alternative zu Hairpin NAT erkennen

Bei Split DNS kann derselbe Hostname intern und extern unterschiedliche Antworten liefern.

Extern:

```
service.example.net → 203.0.113.10
```

Intern:

```
service.example.net → 192.168.10.50
```

DNS-Antwort prüfen:

Windows:

```
[TEST] Resolve-DnsName service.example.net -Type A
```

Linux und macOS:

```
[TEST] dig service.example.net A
```

Bestimmten internen DNS-Server abfragen:

```
[TEST] dig @192.168.10.53 service.example.net A
```

Bestimmten externen DNS-Server abfragen:

```
[TEST] dig @1.1.1.1 service.example.net A
```

Die Verwendung öffentlicher DNS-Server muss durch die Netz- und Datenschutzrichtlinien erlaubt sein. Unternehmensinterne Namen dürfen nicht unnötig an externe Resolver übermittelt werden.

Zu prüfen sind:

- Welche Antwort erhält ein interner Client?
- Welche Antwort erhält ein externer Client?
- Sind TTL und Cache berücksichtigt?
- Verweist der interne Datensatz auf den richtigen Server?
- Passt das TLS-Zertifikat weiterhin zum Hostnamen?

14. Windows-NAT-Konfiguration prüfen

Die folgenden Befehle betreffen Windows-NAT-Konfigurationen, die über das NetNat-Modul verwaltet werden, beispielsweise in bestimmten Hyper-V- oder Windows-Container-Szenarien. Sie zeigen nicht automatisch die NAT-Regeln eines externen Routers an.

Vorhandene Windows-NAT-Objekte anzeigen:

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

Ausführliche Anzeige:

```
[RO][PRIV][SENS] Get-NetNat |
    Format-List *
```

Statische Zuordnungen anzeigen:

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

NAT-Sitzungen anzeigen:

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

Interne Hyper-V-Switches prüfen, sofern Hyper-V eingesetzt wird:

```
[RO][PRIV] Get-VMSwitch
```

IP-Adressen der virtuellen Schnittstellen:

```
[RO] Get-NetIPAddress |
    Where-Object InterfaceAlias -Like "vEthernet*" |
    Format-Table InterfaceAlias, AddressFamily, IPAddress, PrefixLength
```

Zu prüfen sind:

- Existiert das erwartete NAT-Objekt?
- Stimmt das interne Adresspräfix?
- Existiert die erwartete statische Portzuordnung?

- Stimmen externes und internes Protokoll?
- Stimmen externe und interne Ports?
- Stimmt die interne Zieladresse?
- Existiert eine passende Firewallregel?
- Besteht ein Konflikt mit Container- oder Virtualisierungssoftware?

Keine NAT-Objekte löschen oder neu erstellen, solange der Sollzustand und die Auswirkungen nicht vollständig geklärt sind.

---

15. Linux-NAT mit nftables prüfen

Gesamtes nftables-Regelwerk anzeigen:

```
[RO][PRIV][SENS] sudo nft list ruleset
```

Tabellen anzeigen:

```
[RO][PRIV] sudo nft list tables
```

Regelwerk einschließlich Handles anzeigen:

```
[RO][PRIV][SENS] sudo nft -a list ruleset
```

Bei NAT-Regeln sind insbesondere folgende Hooks relevant:

Hook	Typische Verwendung
`prerouting`	DNAT eingehender Pakete vor der Routingentscheidung
`output`	DNAT lokal erzeugter Pakete
`postrouting`	SNAT oder Masquerading nach der Routingentscheidung
`input`	Bestimmte SNAT-Sonderfälle für lokal zugestellte Pakete

Typische nftables-Ausdrücke erkennen:

```
dnat to 192.168.10.50:443
snat to 203.0.113.10
masquerade
```

```
redirect to :8080
```

Regeln mit Zählern untersuchen:

```
[RO][PRIV][SENS] sudo nft -a list ruleset
```

Zu prüfen ist, ob die Paket- und Bytezähler der erwarteten NAT- und Filterregeln während eines einzelnen Tests steigen.

Wichtig: NAT-Anweisungen sind an NAT-Chains und die dafür vorgesehenen Hooks gebunden. Eine sichtbare Regel beweist nicht, dass das Paket den dazugehörigen Pfad tatsächlich durchläuft.

---

16. Linux-NAT mit iptables prüfen

NAT-Tabelle mit Zählern und Zeilennummern:

```
[RO][PRIV][SENS] sudo iptables -t nat -L -n -v --line-numbers
```

NAT-Regeln in Befehlsschreibweise:

```
[RO][PRIV][SENS] sudo iptables -t nat -S
```

IPv6-NAT-Regeln, sofern in der Umgebung verwendet:

```
[RO][PRIV][SENS] sudo ip6tables -t nat -L -n -v --line-numbers
```

Filterregeln für weitergeleiteten Verkehr:

```
[RO][PRIV][SENS] sudo iptables -L FORWARD -n -v --line-numbers
```

Typische Ziele:

Ziel	Bedeutung
DNAT	Zieladresse oder Zielport verändern
SNAT	Quelladresse oder Quellport verändern

Ziel	Bedeutung
`MASQUERADE`	Dynamische Quelladressübersetzung
`REDIRECT`	Verkehr an das lokale System umleiten
`ACCEPT`	Verkehr zulassen
`DROP`	Verkehr ohne Antwort verwerfen
`REJECT`	Verkehr aktiv ablehnen

Eine passende DNAT-Regel allein reicht nicht aus. Weitergeleiteter Verkehr muss zusätzlich durch die Filterregeln zugelassen und korrekt geroutet werden.

---

17. IP-Forwarding prüfen

Ein Linux-System, das Pakete zwischen Schnittstellen weiterleiten soll, benötigt eine passende Forwarding-Konfiguration.

IPv4-Forwarding:

```
[RO] sysctl net.ipv4.ip_forward
```

Alternative Anzeige:

```
[RO] cat /proc/sys/net/ipv4/ip_forward
```

Typische Werte:

Wert	Bedeutung
`0`	IPv4-Weiterleitung deaktiviert
`1`	IPv4-Weiterleitung aktiviert

IPv6-Forwarding:

```
[RO] sysctl net.ipv6.conf.all.forwarding
```

Die Aktivierung von Forwarding ist eine Konfigurationsänderung und darf nicht automatisch vorgenommen werden. Zuerst muss geklärt werden, ob das System laut Netzdesign als Router arbeiten soll.

Zusätzlich prüfen:

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

NAT ersetzt keine gültige Routingkonfiguration.

---

18. macOS-NAT- und pf-Regeln prüfen

macOS kann NAT und Weiterleitung über den Packet Filter `pf` beziehungsweise durch Anwendungen und Systemdienste umsetzen.

pf-Status:

```
[RO][PRIV] sudo pfctl -s info
```

NAT-Regeln anzeigen:

```
[RO][PRIV][SENS] sudo pfctl -s nat
```

Filterregeln anzeigen:

```
[RO][PRIV][SENS] sudo pfctl -s rules
```

Zustände anzeigen:

```
[RO][PRIV][SENS] sudo pfctl -s states
```

Regeln mit zusätzlichen Informationen und Zählern:

```
[RO][PRIV][SENS] sudo pfctl -vvs nat
```

```
[RO][PRIV][SENS] sudo pfctl -vvs rules
```

Routingtabelle prüfen:

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

IPv4-Forwardingstatus:

```
[RO] sysctl net.inet.ip.forwarding
```

Zu beachten ist, dass VPN-, Virtualisierungs-, Container- und Sicherheitssoftware eigene `pf`-Anchors oder Regeln verwenden kann. Das Regelwerk darf nicht ohne Kenntnis dieser Abhängigkeiten neu geladen oder geleert werden.

---

19. Connection Tracking und NAT-Zustände prüfen

NAT ist normalerweise zustandsorientiert. Das System merkt sich, wie eine Verbindung übersetzt wurde, damit Antwortpakete korrekt zurückübersetzt werden können.

Linux – prüfen, ob conntrack vorhanden ist:

```
[RO] command -v conntrack
```

Alle Verbindungseinträge anzeigen:

```
[RO][PRIV][SENS] sudo conntrack -L
```

TCP-Verbindungen zu einem Zielport filtern:

```
[RO][PRIV][SENS] sudo conntrack -L -p tcp --dport 443
```

UDP-Verbindungen zu einem Zielport filtern:

```
[RO][PRIV][SENS] sudo conntrack -L -p udp --dport 53
```

Je nach Übersetzungsrichtung können in einem Eintrag ursprüngliche und übersetzte Adressen erscheinen.

Zu kontrollieren sind:

- Wird beim Test ein Zustandseintrag erzeugt?
- Stimmen ursprüngliches und übersetztes Ziel?
- Ist Antwortverkehr sichtbar?
- Läuft der Eintrag frühzeitig ab?
- Gibt es sehr viele Verbindungseinträge?
- Ist nur eine Richtung sichtbar?

Zustandstabellen dürfen nicht unkontrolliert geleert werden. Dadurch können bestehende Verbindungen unterbrochen werden.

---

20. Rückweg und asymmetrisches Routing prüfen

Nach DNAT erreicht ein Paket möglicherweise den internen Server. Der Server muss jedoch über einen gültigen Weg antworten.

Sollzustand:

```
Externer Client
      ↓
NAT-Gateway
      ↓
Interner Server
      ↓
NAT-Gateway
      ↓
Externer Client
```

Problematischer asymmetrischer Rückweg:

```
Anfrage:
Client → NAT-Gateway A → Server

Antwort:
Server → Gateway B → Client
```

Das NAT-Gateway A sieht die Antwort nicht und kann die notwendige Rückübersetzung nicht durchführen.

Standardgateway des Zielservers prüfen:

Windows:

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

Linux:

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

macOS:

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

Route zu einer externen Testadresse prüfen:

Windows:

```
[RO] Find-NetRoute -RemoteIPAddress 198.51.100.25
```

Linux:

```
[RO] ip route get 198.51.100.25
```

macOS:

```
[RO] route -n get 198.51.100.25
```

Die Beispieladresse muss durch die tatsächliche Adresse des autorisierten Testsystems ersetzt werden.

---

21. Paketmitschnitt vor und nach der Übersetzung erstellen

Idealerweise wird auf beiden Seiten des NAT-Systems gleichzeitig mitgeschnitten.

```
WAN-Mitschnitt:
Ziel 203.0.113.10:8443

LAN-Mitschnitt:
Ziel 192.168.10.50:443
```

Linux – WAN-Schnittstelle:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i WAN_INTERFACE -nn \
    'tcp port 8443' \
    -w nat-wan.pcap
```

Linux – LAN-Schnittstelle:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i LAN_INTERFACE -nn \
    'host 192.168.10.50 and tcp port 443' \
    -w nat-lan.pcap
```

macOS:

```
[TEST][PRIV][FILE][SENS] sudo tcpdump -i INTERFACE -nn \
    'tcp port 8443 or tcp port 443' \
    -w nat-test.pcap
```

Windows mit Dumpcap – Schnittstellen ermitteln:

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

Mitschnitt auf einer bestimmten Schnittstelle:

```
[TEST][PRIV][FILE][SENS] dumpcap -i 1 `
    -f "tcp port 8443 or tcp port 443" `
    -w nat-test.pcapng
```

`WAN_INTERFACE`, `LAN_INTERFACE`, `INTERFACE` und die Dumpcap-Schnittstellennummer müssen vorher ermittelt werden.

22. Paketmuster bei NAT-Fehlern interpretieren

WAN-Seite	LAN-Seite	Mögliche Interpretation
Kein Paket sichtbar	Kein Paket sichtbar	Falsche öffentliche Adresse, CGNAT, vorgeschalteter Filter oder Clientproblem
SYN sichtbar	Kein weitergeleitetes SYN	DNAT-Regel, Firewall oder Routing auf dem NAT-System prüfen
SYN an externen Port sichtbar	SYN an richtiges internes Ziel sichtbar	Übersetzung funktioniert grundsätzlich
SYN erreicht internen Server	Kein SYN-ACK vom Server	Dienst, Bind-Adresse, Host-Firewall oder Serverrouting prüfen
SYN-ACK verlässt Server	Keine Antwort auf WAN-Seite	NAT-Zustand, Forwarding, Filter oder Rückweg prüfen
Vollständiger Handshake auf beiden Seiten	Anwendung schlägt dennoch fehl	Fehler wahrscheinlich oberhalb der Transportschicht
RST kommt vom internen Server	Ziel erreichbar, aber Port wird nicht angenommen	
ICMP-Fehler sichtbar	Abhängig vom Fehlercode	Routing-, Filter- oder Erreichbarkeitsproblem weiter untersuchen

Wichtig: Ein Mitschnitt auf nur einer Schnittstelle zeigt nicht den gesamten Übersetzungsvorgang.

---

23. Docker-Portweiterleitungen prüfen

Docker verwendet bei Bridge-Netzwerken Port Publishing, NAT beziehungsweise Firewallregeln, um Hostports an Containerports weiterzuleiten.

Container und Portzuordnungen anzeigen:

```
[RO] docker ps --format 'table {{.Names}}\t{{.Ports}}'
```

Portzuordnung eines bestimmten Containers:

```
[RO] docker port CONTAINERNAME
```

Containerkonfiguration prüfen:

```
[RO][SENS] docker inspect CONTAINERNAME
```

Docker-Netzwerke anzeigen:

```
[RO] docker network ls
```

Netzwerk untersuchen:

```
[RO][SENS] docker network inspect NETZWERKNAME
```

Beispiele:

```
0.0.0.0:8080->80/tcp
```

Der Hostport `8080/TCP` wurde an den Containerport `80/TCP` veröffentlicht und ist an den passenden IPv4-Adressen des Hosts gebunden.

```
127.0.0.1:8080->80/tcp
```

Der Hostport ist an die Loopback-Adresse gebunden und normalerweise nur vom Docker-Host erreichbar.

```
8080->80/tcp
```

Die genaue Bindung muss über die vollständige Docker-Ausgabe beziehungsweise `docker inspect` geprüft werden.

Wichtig:

```
EXPOSE 80
```

`EXPOSE` dokumentiert den vorgesehenen Containerport, veröffentlicht ihn aber nicht automatisch auf dem Host.

---

24. Docker-NAT systematisch testen

Ebene 1 – Anwendung im Container

```
[RO] docker ps
```

```
[RO][SENS] docker logs --tail 100 CONTAINERNAME
```

Ebene 2 – Portzuordnung

```
[RO] docker port CONTAINERNAME
```

Ebene 3 – Zugriff vom Docker-Host

```
[TEST] curl -v http://127.0.0.1:8080/
```

Ebene 4 – Zugriff über die LAN-Adresse des Hosts

```
[TEST] curl -v http://HOST_LAN_IP:8080/
```

Ebene 5 – Zugriff aus einem anderen LAN-System

```
[TEST] nc -vz -w 5 HOST_LAN_IP 8080
```

Ebene 6 – Zugriff über eine externe Routerweiterleitung

```
[TEST] nc -vz -w 5 OEFFENTLICHE_IP EXTERNER_PORT
```

Der letzte Test muss wirklich aus einem externen Netz erfolgen, sofern Hairpin NAT nicht ausdrücklich mitgeprüft werden soll.

---

25. IPv4-NAT und IPv6-Zugriff nicht verwechseln

Bei IPv4 wird NAT häufig eingesetzt, weil private Adressen nicht direkt aus dem Internet erreichbar sind.

Bei IPv6 besitzt ein System häufig eine global routbare Adresse. Eine klassische IPv4-Portweiterleitung ist dann nicht automatisch beteiligt. Der Zugriff wird stattdessen insbesondere durch folgende Komponenten bestimmt:

- IPv6-Adressierung,
- IPv6-Routing,
- IPv6-Firewall,
- DNS-AAAA-Eintrag,
- Dienstbindung an IPv6,
- Provider- und Routerrichtlinien.

DNS-Einträge prüfen:

Windows:

```
[TEST] Resolve-DnsName service.example.net -Type A
[TEST] Resolve-DnsName service.example.net -Type AAAA
```

Linux und macOS:

```
[TEST] dig service.example.net A
[TEST] dig service.example.net AAAA
```

HTTPS getrennt testen:

```
[TEST] curl -4 -v https://service.example.net/
[TEST] curl -6 -v https://service.example.net/
```

Typischer Fehler:

```
IPv4-Portweiterleitung ist korrekt
↓
Client bevorzugt jedoch den vorhandenen AAAA-Eintrag
↓
Verbindung erfolgt über IPv6
↓
IPv6-Firewall oder IPv6-Dienstbindung ist fehlerhaft
```

26. Dynamisches DNS prüfen

Bei wechselnden öffentlichen Adressen wird häufig Dynamic DNS verwendet.

Zu prüfen sind:

- Welche öffentliche Adresse besitzt der Anschluss aktuell?
- Welche Adresse liefert der DNS-Eintrag?
- Wann wurde der Eintrag zuletzt aktualisiert?
- Wie hoch ist die TTL?
- Funktioniert der Update-Client?
- Wird IPv4, IPv6 oder beides aktualisiert?
- Existiert ein alter A- oder AAAA-Eintrag?

DNS-Eintrag prüfen:

Windows:

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

Linux und macOS:

```
[TEST] dig service.example.net A
[TEST] dig service.example.net AAAA
```

Autoritativen Nameserver ermitteln:

```
[TEST] dig service.example.net NS
```

Antwort eines bestimmten autoritativen Servers abfragen:

```
[TEST] dig @AUTORITATIVER_DNS_SERVER service.example.net A
```

Ein veralteter lokaler Cache kann sich von der bereits aktualisierten autoritativen Antwort unterscheiden.

---

27. NAT-Loopback, Proxy und Reverse Proxy auseinanderhalten

Ein öffentlich erreichbarer Dienst kann über mehrere Weiterleitungsstufen verfügen:

```
Internet
  ↓
Router-DNAT
  ↓
Reverse Proxy
  ↓
Container-Port
  ↓
Anwendung
```

Jede Ebene benötigt eine eigene Prüfung.

Ebene	Prüffrage
Router-DNAT	Wird der externe Port an das richtige interne System weitergeleitet?
Host-Firewall	Darf der Verkehr den Reverse Proxy erreichen?
Reverse Proxy	Existiert die richtige Host- und Zielkonfiguration?
Container Publishing	Ist der Hostport an den Containerport gebunden?
Anwendung	Lauscht die Anwendung und verarbeitet sie die Anfrage?
DNS	Zeigt der Hostname auf die erwartete öffentliche Adresse?
TLS	Stimmt Zertifikat, Hostname und SNI?

Ein funktionierender Router-DNAT beweist nicht, dass der Reverse Proxy oder das Backend funktioniert.

---

28. Keine unkontrollierte Exposition interner Dienste

Eine Portweiterleitung macht einen Dienst potenziell aus externen Netzen erreichbar. Vor einer Freigabe müssen mindestens geprüft werden:

- Ist die externe Erreichbarkeit wirklich erforderlich?
- Unterstützt der Dienst eine sichere Authentifizierung?
- Wird eine verschlüsselte Verbindung verwendet?
- Ist die Software aktuell?
- Ist der Zugriff auf erforderliche Quelladressen beschränkbar?

- Kann stattdessen ein VPN oder Zero-Trust-Zugang verwendet werden?
- Existiert eine Protokollierung und Überwachung?
- Gibt es Schutz vor Brute-Force-Angriffen?
- Ist der Dienst für eine Veröffentlichung vorgesehen?
- Enthält der Dienst eine administrative Oberfläche?
- Ist ein Reverse Proxy mit zusätzlicher Zugriffskontrolle sinnvoll?
- Existiert ein dokumentierter Rückbauplan?

Besonders Datenbanken, interne Verwaltungsoberflächen, Dateifreigaben und unverschlüsselte Protokolle sollten nicht unkontrolliert direkt veröffentlicht werden.

Keine automatische oder pauschale Portfreigabe über UPnP voraussetzen. Dynamisch erstellte Weiterleitungen müssen ebenso geprüft und dokumentiert werden.

---

29. Änderungen nur kontrolliert durchführen

Nicht vorschnell:

- NAT-Tabellen leeren,
- Conntrack-Zustände löschen,
- Router auf Werkseinstellungen zurücksetzen,
- Firewall vollständig deaktivieren,
- alle Quelladressen freigeben,
- DMZ- beziehungsweise Exposed-Host-Funktionen aktivieren,
- mehrere Weiterleitungen für denselben Port erstellen,
- interne Server direkt und ungeschützt veröffentlichen,
- IPv6-Firewall deaktivieren,
- Docker-Regeln manuell überschreiben.

Sicherer Änderungsablauf:

1. Istzustand exportieren oder dokumentieren.
2. Gewünschten Datenfluss vollständig definieren.
3. Bestehende Regeln und Abhängigkeiten prüfen.
4. Änderung genehmigen lassen.
5. Eng begrenzte Regel erstellen.
6. Protokollierung aktivieren, soweit vorgesehen.
7. Von intern und extern testen.
8. Sicherheitsprüfung durchführen.
9. Ergebnis dokumentieren.
10. Bei Misserfolg den vorherigen Zustand wiederherstellen.

---

30. Systematischer Diagnoseablauf

Schritt	Prüfung	Leitfrage
1	Datenfluss dokumentieren	Welche externe Adresse und welcher Port sollen wohin übersetzt werden?
2	Zielanwendung prüfen	Läuft der interne Dienst?
3	Listener prüfen	Lauscht der Dienst auf Adresse, Port und Protokoll?
4	Internen Zugriff testen	Funktioniert der Dienst ohne NAT?
5	Interne Zieladresse prüfen	Stimmt die NAT-Regel noch mit der Serveradresse überein?
6	WAN-Adresse prüfen	Besitzt der Router die erwartete öffentliche Adresse?
7	CGNAT oder Double NAT prüfen	Existiert ein vorgeschaltetes NAT-System?
8	DNS prüfen	Zeigt der Hostname auf die aktuelle öffentliche Adresse?
9	NAT-Regel prüfen	Stimmen Protokoll, externe und interne Ports sowie Zieladresse?
10	Firewallregeln prüfen	Wird der übersetzte Verkehr zugelassen?
11	Forwarding prüfen	Darf das System Pakete zwischen Schnittstellen weiterleiten?
12	Rückweg prüfen	Antwortet der Server über das NAT-Gateway?
13	Extern testen	Erreicht ein Client außerhalb des LANs den Port?
14	Hairpin NAT getrennt testen	Betrifft der Fehler nur interne Clients?
15	Paketmitschnitt auf beiden Seiten	Wird das Paket korrekt übersetzt?
16	NAT- und Regelzähler prüfen	Trifft der Test die erwartete Regel?
17	Container oder Proxy prüfen	Existieren weitere Weiterleitungsstufen?
18	Änderung minimal planen	Welche kleinste genehmigte Korrektur behebt die Ursache?
19	Nachprüfung	Funktioniert der Dienst intern und extern?

Schritt	Prüfung	Leitfrage
20	Dokumentation	Sind Regel, Risiko und Rückfallplan festgehalten?

31. Kompakte Befehlstabelle

Aufgabe	Windows	Linux	macOS
Lokale IP-Adressen	`[RO] Get-NetIPAddress`	`[RO] ip -br address`	`[RO] ifconfig`
Standardgateway	`[RO] Get-NetRoute -DestinationPrefix "0.0.0.0/0"`	`[RO] ip route show default`	`[RO] route -n get default`
Route zu einem Ziel	`[RO] Find-NetRoute -RemoteIPAddress IP`	`[RO] ip route get IP`	`[RO] route -n get IP`
TCP-Listener	`[RO] Get-NetTCPConnection -State Listen`	`[RO] ss -ltn`	`[RO] lsof -nP -iTCP -sTCP:LISTEN`
Internen TCP-Port testen	`[TEST] Test-NetConnection IP -Port PORT`	`[TEST] nc -vz -w 5 IP PORT`	`[TEST] nc -vz -w 5 IP PORT`
Windows-NAT anzeigen	`[RO][PRIV][SENS] Get-NetNat`	Nicht zutreffend	Nicht zutreffend
Statische Windows-NAT-Zuordnungen	`[RO][PRIV][SENS] Get-NetNatStaticMapping`	Nicht zutreffend	Nicht zutreffend
nftables-Regeln	Nicht zutreffend	`[RO][PRIV][SENS] sudo nft -a list ruleset`	Nicht zutreffend
iptables-NAT	Nicht zutreffend	`[RO][PRIV][SENS] sudo iptables -t nat -L -n -v --line-numbers`	Nicht zutreffend
pf-NAT	Nicht zutreffend	Nicht standardmäßig allgemein vorhanden	`[RO][PRIV][SENS] sudo pfctl -vvs nat`
IPv4-Forwarding	Abhängig von Windows-Rolle	`[RO] sysctl net.ipv4.ip_forward`	`[RO] sysctl net.inet.ip.forwarding`
Conntrack	Abhängig von Windows-NAT	`[RO][PRIV][SENS] sudo conntrack -L`	pf: `[RO][PRIV][SENS] sudo pfctl -s states`
Docker-Portzuordnung	`[RO] docker port CONTAINER`	`[RO] docker port CONTAINER`	`[RO] docker port CONTAINER`
Öffentliche IPv4 abfragen	`[TEST][SENS] Invoke-RestMethod https://api.ipify.org`	`[TEST][SENS] curl -4 https://api.ipify.org`	`[TEST][SENS] curl -4 https://api.ipify.org`
DNS-A- und AAAA-Einträge	`[TEST] Resolve-DnsName HOST`	`[TEST] dig HOST A` und `[TEST] dig HOST AAAA`	`[TEST] dig HOST A` und `[TEST] dig HOST AAAA`
Externen TCP-Port testen	`[TEST] Test-NetConnection PUBLIC_IP -Port PORT`	`[TEST] nc -vz -w 5 PUBLIC_IP PORT`	`[TEST] nc -vz -w 5 PUBLIC_IP PORT`

`IP`, `PORT`, `HOST`, `PUBLIC_IP` und `CONTAINER` müssen durch die Werte des konkreten Störungsfalls ersetzt werden.

32. Dokumentationsvorlage für NAT-Fehler

Störung:

Zeitpunkt:

Zeitzone:

Betroffener Dienst:

Betroffener Standort:

Öffentlicher DNS-Name:

Aufgelöste IPv4-Adresse:

Aufgelöste IPv6-Adresse:

Router-WAN-Adresse:

Extern erkannte IPv4-Adresse:

CGNAT oder Double NAT vermutet: Ja / Nein

Anschlussart:

Transportprotokoll:

Externe IP-Adresse:

Externer Port:

Interne Zieladresse:

Interner Zielport:

WAN-Schnittstelle:

LAN-Schnittstelle:

Erlaubte Quelladressen:

Interner Server:

Aktuelle Server-IP:

Server-Standardgateway:

Listener vorhanden:

Bind-Adresse:

Host-Firewall geprüft:

Interner Funktionstest:

NAT-System:

NAT-Regelname:

NAT-Regel aktiviert:

DNAT-Regel:

SNAT- oder Masquerading-Regel:

Forwarding aktiviert:

Filterregel:

Regelzähler vor Test:

Regelzähler nach Test:

Test aus internem Netz:

Test über öffentliche Adresse aus internem Netz:

Test aus externem Netz:

Hairpin NAT erforderlich:

Paket auf WAN-Seite sichtbar:

Paket auf LAN-Seite sichtbar:

Antwort des Servers sichtbar:

Antwort auf WAN-Seite sichtbar:

Container beteiligt:

Reverse Proxy beteiligt:

VPN beteiligt:

Festgestellte Ursache:

Genehmigte Änderung:

Rückfallplan:

Ergebnis der Nachprüfung:

33. Kontrollfragen nach der Diagnose

- Funktioniert der Dienst direkt über seine interne Adresse?
- Lauscht der Dienst auf dem richtigen Port und Transportprotokoll?
- Stimmt die aktuelle interne Serveradresse mit der NAT-Regel überein?
- Ist die Serveradresse dauerhaft reserviert oder statisch verwaltet?
- Stimmt die Router-WAN-Adresse mit der extern erkannten Adresse überein?
- Liegt die WAN-Adresse in einem privaten oder CGNAT-Adressbereich?
- Existiert ein zweiter vorgeschalteter Router?
- Zeigt DNS auf die aktuelle öffentliche Adresse?
- Wurden A- und AAAA-Einträge getrennt geprüft?
- Stimmen externer und interner Port?
- Wurde TCP nicht mit UDP verwechselt?
- Gilt die NAT-Regel für die richtige Schnittstelle?
- Existiert eine passende Firewallregel?
- Ist IP-Forwarding entsprechend dem Sollzustand aktiviert?

- Verwendet der Server das NAT-System als Rückweg?
- Wurde wirklich aus einem externen Netz getestet?
- Wurde Hairpin NAT getrennt von der externen Erreichbarkeit bewertet?
- Wurden Paketmitschnitte vor und nach der Übersetzung verglichen?
- Steigen die Zähler der erwarteten NAT- und Firewallregel?
- Existiert eine weitere Ebene wie Container, Proxy oder VPN?
- Wurde die öffentliche Freigabe auf das erforderliche Minimum begrenzt?
- Wurde der Zustand nach der Änderung erneut intern und extern geprüft?

---

34. Quellen und weiterführende Dokumentation

- Microsoft Learn – NAT-Netzwerk mit Windows und Hyper-V einrichten: https://learn.microsoft.com/windows-server/virtualization/hyper-v/setup-nat-network
- Microsoft Learn – Get-NetNat: https://learn.microsoft.com/powershell/module/netnat/get-netnat
- Microsoft Learn – Get-NetNatStaticMapping: https://learn.microsoft.com/powershell/module/netnat/get-netnatstaticmapping
- Microsoft Learn – Get-NetNatSession: https://learn.microsoft.com/powershell/module/netnat/get-netnatsession
- Netfilter – Network Address Translation mit nftables: https://wiki.nftables.org/wiki-nftables/index.php/Performing_Network_Address_Translation_(NAT)
- Netfilter – Offizielle nftables-Dokumentation: https://netfilter.org/projects/nftables/manpage.html
- Linux-Handbuch – iptables: https://man7.org/linux/man-pages/man8/iptables.8.html
- Linux-Handbuch – conntrack: https://manpages.debian.org/conntrack/conntrack.8.en.html
- Docker Docs – Port publishing and mapping: https://docs.docker.com/engine/network/port-publishing/
- Docker Docs – Container networking: https://docs.docker.com/engine/network/
- Docker Docs – Dockerfile `EXPOSE`: https://docs.docker.com/reference/dockerfile/#expose
- RFC 1918 – Address Allocation for Private Internets: https://www.rfc-editor.org/rfc/rfc1918.html
- RFC 6598 – Shared Address Space für Carrier-Grade NAT: https://www.rfc-editor.org/rfc/rfc6598.html
- RFC 6888 – Common Requirements for Carrier-Grade NAT: https://www.rfc-editor.org/rfc/rfc6888.html
- RFC 4787 – NAT Behavioral Requirements for UDP: https://www.rfc-editor.org/rfc/rfc4787.html

- RFC 5382 – NAT Behavioral Requirements for TCP: https://www.rfc-editor.org/rfc/rfc5382.html