9.8 NAT, Portweiterleitung und Hairpin NAT im Datenpfad

Network Address Translation verändert IP-Adressen und gegebenenfalls Transportports innerhalb eines Datenflusses. Eine Verbindung kann deshalb an verschiedenen Stellen des Netzwerkpfads mit unterschiedlichen Adressen und Ports sichtbar sein.

Beispiel einer eingehenden Portweiterleitung:

```
Externer Client:
198.51.100.40:53124

Öffentliche Adresse:
203.0.113.10:8443

Interner Server:
192.0.2.25:443
```

Vor der Zielübersetzung:

```
198.51.100.40:53124 -> 203.0.113.10:8443/TCP
```

Nach der Zielübersetzung:

```
198.51.100.40:53124 -> 192.0.2.25:443/TCP
```

Eine scheinbar richtige Firewallregel kann wirkungslos bleiben, wenn sie die falsche Adressphase, das falsche Protokoll, die falsche Schnittstelle oder den falschen Übersetzungsport verwendet.

Ziele

Nach der Bearbeitung dieser Seite soll nachvollziehbar geprüft werden können:

- welche Adressen und Ports vor und nach NAT verwendet werden;
- ob Source NAT, Destination NAT oder Port Address Translation greift;
- welche NAT-Regel tatsächlich angewendet wird;
- ob eine Portweiterleitung auf das richtige interne Ziel zeigt;
- ob Firewallregel, Routing und NAT zusammenpassen;
- ob der Rückverkehr dieselbe NAT-Komponente durchläuft;

- ob ein vorgeschaltetes NAT oder Carrier-Grade NAT eingehende Verbindungen verhindert;
- ob Hairpin NAT beziehungsweise NAT Loopback unterstützt wird;
- warum eine Verbindung intern, extern oder nur über die öffentliche Adresse fehlschlägt;
- wie Container-, Cloud- und Load-Balancer-NAT berücksichtigt werden;
- wie eine NAT-Änderung kontrolliert verifiziert wird.

Sicherheits- und Änderungsgrundsätze

Eine Portweiterleitung veröffentlicht einen internen Dienst über einen zusätzlichen Netzwerkpfad. Vor der Einrichtung sind mindestens zu prüfen:

- geschäftlicher Zweck;
- verantwortlicher Diensteigentümer;
- erforderliches Protokoll;
- erforderlicher Port;
- erlaubte Quelladressen;
- Authentifizierung;
- Verschlüsselung;
- Patchstand;
- Protokollierung;
- Rate Limiting;
- Monitoring;
- Rückweg;
- Rücknahmezeitpunkt;
- Risiko;
- Freigabe.

Besonders schützenswerte Verwaltungs- und Dateidienste sollten nicht ohne zusätzliche Schutzarchitektur direkt aus dem Internet veröffentlicht werden.

Dazu gehören beispielsweise:

- RDP;
- SMB;
- SSH mit schwacher Authentifizierung;
- Datenbankports;
- Hypervisorverwaltung;
- NAS-Verwaltung;
- Router- oder Firewallverwaltung;
- unverschlüsselte Weboberflächen;
- interne Verzeichnisdienste.

Geeignetere Zugriffsmöglichkeiten können sein:

- VPN;
- Zero-Trust-Zugriff;

- Bastion Host;
- Jump Host;
- Reverse Proxy mit starker Authentifizierung;
- Application Gateway;
- Web Application Firewall;
- privater Cloudendpunkt.

Nicht als erste Diagnosemaßnahme geeignet sind:

- alle Ports weiterleiten;
- DMZ-Host-Funktion eines Routers aktivieren;
- Exposed Host einrichten;
- Firewall vollständig deaktivieren;
- allgemeine `any`-zu-`any`-Regel erstellen;
- NAT-Tabelle vollständig leeren;
- alle Verbindungstabellen löschen;
- UPnP pauschal aktivieren;
- Server direkt mit öffentlicher Adresse verbinden;
- Managementport testweise veröffentlichen;
- Router oder Firewall vorsorglich neu starten;
- mehrere NAT-Regeln gleichzeitig verändern.

NAT ist keine Firewall

NAT verändert Adressen oder Ports. Eine Firewall entscheidet anhand einer Sicherheitsrichtlinie, ob Datenverkehr erlaubt oder verworfen wird.

Beide Funktionen können sich im selben Gerät befinden, bleiben aber logisch getrennt.

Beispiel:

```
NAT-Regel:
203.0.113.10:8443 -> 192.0.2.25:443

Firewallregel:
TCP von 198.51.100.0/24 zu 192.0.2.25:443 erlauben
```

Mögliche Kombinationen:

NAT	Firewall	Ergebnis
vorhanden	erlaubt	Verbindung kann weiterverarbeitet werden

NAT	Firewall	Ergebnis
vorhanden	blockiert	Übersetzung kann passen, Verkehr wird trotzdem verworfen
fehlt	erlaubt	Freigabe besitzt möglicherweise kein erreichbares Ziel
falsch	erlaubt	Verkehr erreicht falsches System oder falschen Port
vorhanden	falsche Richtung	Verbindungsaufbau bleibt blockiert
automatisch erstellt	unbekannt	Sicherheitswirkung muss ausdrücklich geprüft werden

Einige Router erzeugen zusammen mit einer Portweiterleitung automatisch eine Firewallfreigabe. Andere Produkte verlangen eine getrennte Regel. Dieses Verhalten ist hersteller- und versionsabhängig.

NAT-Begriffe

Begriff	Bedeutung
NAT	Oberbegriff für die Übersetzung von Netzwerkadressen
SNAT	Änderung der Quelladresse
DNAT	Änderung der Zieladresse
PAT	Übersetzung von Transportports, häufig zusammen mit Adressübersetzung
NAPT	Übersetzung von Netzwerkadresse und Transportport
Masquerading	SNAT auf die aktuelle Adresse einer Ausgangsschnittstelle
statisches NAT	fest definierte Zuordnung
dynamisches NAT	Zuordnung aus einem Adresspool
Portweiterleitung	statische eingehende Zieladress- und/oder Portübersetzung
1:1-NAT	feste Zuordnung einer internen zu einer externen Adresse
Twice NAT	Quelle und Ziel werden abhängig voneinander in einer Regel verarbeitet
NAT Exemption	bestimmter Datenverkehr wird ausdrücklich nicht übersetzt
Hairpin NAT	interner Client erreicht internes Ziel über dessen externe NAT-Adresse
NAT Loopback	gebräuchliche Bezeichnung für Hairpin NAT
NAT Reflection	herstellerabhängige Bezeichnung für Hairpin-Verarbeitung

Begriff	Bedeutung
U-Turn NAT	weitere gebräuchliche Bezeichnung für Hairpin NAT
CGNAT	NAT im Netz des Internetzugangsanbieters
NAT64	Übersetzung zwischen IPv6 und IPv4
NPTv6	zustandslose Übersetzung von IPv6-Präfixen

Die Begriffe werden von Herstellern nicht immer einheitlich verwendet.

Source NAT

Source NAT verändert die Quelladresse eines ausgehenden Datenflusses.

Vor SNAT:

```
192.0.2.100:53124 -> 198.51.100.25:443
```

Nach SNAT:

```
203.0.113.10:61001 -> 198.51.100.25:443
```

Das Zielsystem sieht als Quelle:

```
203.0.113.10:61001
```

und normalerweise nicht:

```
192.0.2.100:53124
```

Typische Anwendungsfälle:

- private IPv4-Clients greifen auf das Internet zu;
- mehrere interne Systeme teilen sich eine öffentliche Adresse;
- Rückverkehr soll über eine bestimmte Firewall geführt werden;
- überlappende Netze werden verbunden;
- ein Ziel erlaubt nur definierte Quelladressen;
- Hairpin-Verkehr wird symmetrisch über das NAT-Gerät geführt.

Auswirkungen:

- ursprüngliche Quelladresse ist am Ziel nicht direkt sichtbar;

- Protokolle zeigen gegebenenfalls nur die NAT-Adresse;
- Portkapazität der NAT-Adresse wird benötigt;
- Rückverkehr ist an die NAT-Sitzung gebunden;
- Quell-IP-basierte Sicherheitsentscheidungen verändern sich.

Destination NAT

Destination NAT verändert die Zieladresse eines eingehenden Datenflusses.

Vor DNAT:

```
198.51.100.40:53124 -> 203.0.113.10:8443
```

Nach DNAT:

```
198.51.100.40:53124 -> 192.0.2.25:443
```

Typische Anwendungsfälle:

- Portweiterleitung;
- Veröffentlichung eines internen Servers;
- Weiterleitung an Reverse Proxy;
- Load Balancing;
- transparente Proxyfunktion;
- Umleitung auf einen lokalen Dienst.

Zu prüfen sind:

- externe Zieladresse;
- externer Zielport;
- internes Ziel;
- interner Zielport;
- Protokoll;
- Eingangsschnittstelle;
- Quellbereich;
- Firewallregel;
- Rückroute;
- Zustandstabelle.

Port Address Translation

PAT ermöglicht die Unterscheidung vieler Datenflüsse über dieselbe externe IP-Adresse.

Beispiel:

```
192.0.2.100:53124
  ->
203.0.113.10:61001

192.0.2.101:53124
  ->
203.0.113.10:61002
```

Beide Clients können dasselbe Ziel verwenden:

```
198.51.100.25:443
```

Das NAT-Gerät hält dafür getrennte Zuordnungen:

```
203.0.113.10:61001 -> 192.0.2.100:53124
203.0.113.10:61002 -> 192.0.2.101:53124
```

PAT benötigt:

- freie Übersetzungsports;
- eine Sitzungstabelle;
- passende Timeouts;
- einen symmetrischen Rückweg;
- ausreichende NAT-Kapazität.

1:1-NAT

Bei 1:1-NAT wird eine interne Adresse fest einer externen Adresse zugeordnet.

Beispiel:

```
Öffentlich:
203.0.113.25

Intern:
192.0.2.25
```

Mögliche Übersetzung:

```
203.0.113.25 <-> 192.0.2.25
```

1:1-NAT bedeutet nicht automatisch, dass alle Ports erlaubt sind. Die Firewallrichtlinie muss weiterhin festlegen:

- zulässige Quellen;
- zulässige Protokolle;
- zulässige Ports;
- Richtung;
- Protokollierung;
- Sicherheitsprofile.

Portweiterleitung

Eine Portweiterleitung ordnet eine externe Kombination einer internen Kombination zu.

Beispiel:

```
Externe Adresse:
203.0.113.10

Externer Port:
8443

Protokoll:
TCP

Internes Ziel:
192.0.2.25

Interner Port:
443
```

Vereinfachte Regel:

```
TCP 203.0.113.10:8443
   ->
TCP 192.0.2.25:443
```

Eine vollständige Portweiterleitung benötigt mindestens:

- richtige öffentliche Adresse;
- richtige Eingangsschnittstelle;
- richtiges Protokoll;
- richtigen externen Port;
- richtige interne Zieladresse;
- richtigen internen Port;
- aktiven Serverdienst;
- lokale Serverfirewall;
- zentrale Firewallfreigabe;
- korrekte Rückroute;
- funktionierenden NAT-Zustand.

Externen und internen Port unterscheiden

Die Portnummern müssen nicht identisch sein.

Beispiel:

```
Extern:
TCP 8443

Intern:
TCP 443
```

Mögliche Diagnosefehler:

- extern Port `443` testen, obwohl `8443` veröffentlicht ist;
- Firewall für internen Port `8443` öffnen, obwohl Server auf `443` hört;
- Server auf `8443` prüfen, obwohl DNAT auf `443` übersetzt;
- TCP-Regel anlegen, obwohl Anwendung UDP verwendet;
- IPv4-NAT testen, obwohl DNS eine IPv6-Adresse liefert.

TCP und UDP getrennt weiterleiten

Eine Portnummer bezeichnet ohne Transportprotokoll keinen vollständigen Dienstendpunkt.

Diese beiden Regeln sind verschieden:

```
TCP 203.0.113.10:8443 -> 192.0.2.25:443
```

```
UDP 203.0.113.10:8443 -> 192.0.2.25:443
```

Zu prüfen sind:

- TCP;
- UDP;
- gegebenenfalls beide Protokolle;
- ICMP-Verhalten;
- QUIC über UDP;
- anwendungsspezifische Zusatzports;
- dynamisch ausgehandelte Ports.

Ein erfolgreicher TCP-Test beweist keine funktionierende UDP-Weiterleitung.

Eingangsschnittstelle und Zieladresse

Eine NAT-Regel kann an eine bestimmte Eingangsschnittstelle oder öffentliche Zieladresse gebunden sein.

Beispiele:

- WAN1;
- WAN2;
- VPN;
- Mobilfunk;
- öffentliche Adresse A;
- öffentliche Adresse B;
- virtuelle IP-Adresse;
- Load-Balancer-Adresse.

Typischer Fehler:

```
NAT-Regel:
gilt nur für WAN1

Tatsächlicher Datenverkehr:
trifft über WAN2 ein
```

Die Regel ist vorhanden, verarbeitet diesen Datenfluss aber nicht.

Vollständiger eingehender Datenpfad

Ein eingehender Zugriff kann folgende Ebenen durchlaufen:

1. öffentlicher DNS-Eintrag;
2. Internetzugangsanbieter;

3. vorgeschaltetes Provider-NAT;
4. Modem oder Router;
5. Edge-Firewall;
6. öffentliche IP-Adresse;
7. DNAT- oder Portweiterleitungsregel;
8. Sicherheitsregel;
9. interne Route;
10. VLAN- oder Segmentfirewall;
11. Load Balancer;
12. Reverse Proxy;
13. Containerhost;
14. veröffentlichter Hostport;
15. Containerport;
16. lokale Host-Firewall;
17. Serverprozess;
18. Anwendung.

Jede Ebene muss mit ihrer vor und nach der Verarbeitung sichtbaren Adresse geprüft werden.

Vollständiger Rückweg

Der Rückverkehr einer eingehenden Verbindung muss die zustandsbehaftete NAT-Komponente wieder erreichen.

Beispiel:

```
Hinweg:
198.51.100.40
  ->
203.0.113.10
  ->
192.0.2.25

Rückweg:
192.0.2.25
  ->
203.0.113.10
  ->
198.51.100.40
```

Wenn der Server über ein anderes Gateway antwortet:

```
192.0.2.25
  ->
anderer Router
  ->
198.51.100.40
```

fehlt dort möglicherweise die Rückübersetzung.

Mögliche Folgen:

- Client erhält eine Antwort mit unerwarteter Quelladresse;
- Antwort wird durch eine Firewall verworfen;
- TCP-Handshake bleibt unvollständig;
- NAT-Sitzung bleibt ohne Rückverkehr;
- Server sendet wiederholt `SYN, ACK`;
- Client sendet wiederholt `SYN`.

NAT-Reihenfolge ist produktspezifisch

Die Verarbeitung kann unter anderem umfassen:

- Eingangsschnittstelle bestimmen;
- bestehende Sitzung suchen;
- DNAT anwenden;
- Route bestimmen;
- Zielzone bestimmen;
- Sicherheitsregel prüfen;
- SNAT anwenden;
- Ausgangsschnittstelle bestimmen;
- Paket weiterleiten.

Die genaue Reihenfolge unterscheidet sich zwischen:

- Linux Netfilter;
- Cisco ASA;
- Palo Alto Networks;
- Juniper SRX;
- Fortinet;
- Cloud-Firewalls;
- Heimroutern;
- Load Balancern;
- Containerplattformen.

Es darf nicht pauschal angenommen werden, dass eine Firewallregel immer:

- die öffentliche Adresse;
- die private Adresse;
- den externen Port;
- den internen Port;
- die Vor-NAT-Zone;
- die Nach-NAT-Zone

verwenden muss.

Maßgeblich ist die Dokumentation des eingesetzten Produkts und der Softwareversion.

NAT-Regelreihenfolge

NAT-Regeln können priorisiert oder der Reihe nach ausgewertet werden.

Mögliche Probleme:

- allgemeinere Regel steht vor spezifischer Regel;
- bestehende Masquerade-Regel erfasst den Datenfluss;
- NAT-Ausnahme überschattet die gewünschte Übersetzung;
- doppelte Portweiterleitung verwendet denselben externen Port;
- falsche Schnittstelle besitzt höhere Priorität;
- Twice-NAT-Regel beendet die weitere Auswertung;
- alte Regel ist noch aktiv;
- zentrale Policy überschreibt lokale Regel;
- IPv6-Regel wird mit IPv4-Regel verwechselt.

Zu dokumentieren sind:

- Regel-ID;
- Regelname;
- Reihenfolge;
- Quellbereich;
- Zielbereich;
- Protokoll;
- Ports;
- Schnittstellen;
- Trefferzähler;
- letzte Verwendung;
- aktive Konfigurationsversion.

Statische interne Zieladresse

Das Ziel einer Portweiterleitung muss dauerhaft erreichbar bleiben.

Geeignete Möglichkeiten:

- statische Serveradresse;
- DHCP-Reservierung;
- stabile virtuelle IP-Adresse;
- Load-Balancer-Adresse;
- Serviceadresse.

Typischer Fehler:

```
Portweiterleitung:
192.0.2.25

Aktuelle Serveradresse:
192.0.2.37
```

Die NAT-Regel ist syntaktisch korrekt, zeigt aber auf ein nicht mehr verwendetes Ziel.

Zu prüfen sind:

- aktuelle IP-Adresse;
- DHCP-Lease;
- Adressreservierung;
- DNS;
- ARP- oder Neighbor-Tabelle;
- Serverroute;
- Mehrfachadressierung;
- Cluster- oder Failoveradresse.

Öffentliche Erreichbarkeit prüfen

Eine eingehende Portweiterleitung benötigt eine von außen erreichbare Zieladresse.

Zu prüfen sind:

- WAN-Adresse des eigenen Routers;
- öffentliche DNS-Adresse;
- von einem autorisierten externen System beobachtete Quelladresse;
- vorgeschaltetes Modem;
- weiterer Router;
- Provider-NAT;
- Carrier-Grade NAT;
- DS-Lite;
- Mobilfunk-NAT;

- VPN oder Tunnel;
- dynamische öffentliche Adresse;
- IPv4 oder IPv6.

Eine private Adresse auf der WAN-Schnittstelle ist nicht direkt aus dem öffentlichen Internet erreichbar.

Private IPv4-Bereiche

RFC 1918 definiert:

```
10.0.0.0/8
172.16.0.0/12
192.168.0.0/16
```

Das bedeutet:

```
10.0.0.0 bis 10.255.255.255
172.16.0.0 bis 172.31.255.255
192.168.0.0 bis 192.168.255.255
```

Befindet sich die WAN-Adresse des eigenen Routers in einem dieser Bereiche, existiert mindestens ein weiteres vorgeschaltetes Routing- oder NAT-System.

Shared Address Space und CGNAT

RFC 6598 reserviert für Provider-NAT:

```
100.64.0.0/10
```

Das umfasst:

```
100.64.0.0 bis 100.127.255.255
```

Dieser Bereich ist kein RFC-1918-Privatnetz, wird aber für Shared Address Space und häufig für Carrier-Grade NAT verwendet.

Hinweise auf vorgeschaltetes NAT:

- WAN-Adresse liegt in einem privaten Bereich;

- WAN-Adresse liegt in `100.64.0.0/10`;
- Router-WAN-Adresse weicht von der extern beobachteten Adresse ab;
- Portweiterleitung funktioniert trotz richtiger lokaler Konfiguration nicht;
- Internetzugang funktioniert ausschließlich ausgehend;
- Anbieter nennt DS-Lite oder CGNAT;
- öffentliche IPv4-Adresse ist nicht Bestandteil des Anschlusses.

Eine Abweichung der Adressen ist ein Hinweis und muss unter Berücksichtigung von VPN, Proxy, mehreren Leitungen und Providerarchitektur bewertet werden.

Doppeltes NAT

Beispiel:

```
Internet
   ->
Providerrouter 192.168.0.1
   ->
eigener Router 192.168.0.2
   ->
Server 192.0.2.25
```

Für eine eingehende Verbindung können zwei Weiterleitungen erforderlich sein:

```
Providerrouter:
203.0.113.10:8443
   ->
192.168.0.2:8443

Eigener Router:
192.168.0.2:8443
   ->
192.0.2.25:443
```

Zusätzlich müssen beide Firewalls und beide Rückwege passen.

Besser kann je nach freigegebener Architektur sein:

- Bridge- oder Modemmodus;
- eigener Router als einziges NAT-Gateway;
- statische Route statt zweitem NAT;

- öffentliche Adresse direkt am Edge-Gerät;
- kontrollierte DMZ-Verbindung zwischen den Geräten.

Ein Exposed Host ist keine geeignete Standardlösung, weil dadurch möglicherweise sehr viele Ports weitergeleitet werden.

Carrier-Grade NAT und Portweiterleitung

Befindet sich die öffentliche IPv4-Adresse auf einem Provider-NAT, kann der eigene Router keine beliebige eingehende Abbildung auf diesem Providergerät erstellen.

Mögliche Lösungen hängen vom Anbieter ab:

- öffentliche IPv4-Adresse buchen;
- statische öffentliche IPv4-Adresse verwenden;
- IPv6 mit geeigneter Firewallfreigabe verwenden;
- VPN mit eingehendem Tunnelendpunkt verwenden;
- Reverse Tunnel;
- Cloud-Reverse-Proxy;
- providerseitige Portzuordnung;
- Port Control Protocol, sofern tatsächlich unterstützt.

Die technische und sicherheitstechnische Eignung muss für den konkreten Dienst bewertet werden.

Hairpin NAT

Hairpin NAT ermöglicht einem internen Client den Zugriff auf einen internen Server über dessen externe NAT-Adresse.

Beispiel:

```
Interner Client:
192.0.2.100

Öffentliche Dienstadresse:
203.0.113.10:443

Interner Server:
192.0.2.25:443
```

Der Client verwendet:

```
192.0.2.100:53124 -> 203.0.113.10:443
```

Das NAT-Gerät übersetzt das Ziel:

```
192.0.2.100:53124 -> 192.0.2.25:443
```

Abhängig von Topologie und Produkt muss zusätzlich die Quelle übersetzt werden, damit der Server über dasselbe NAT-Gerät antwortet.

Hairpin NAT mit Quellübersetzung

Mögliche Verarbeitung:

```
Ursprünglich:
192.0.2.100:53124 -> 203.0.113.10:443

Nach Hairpin DNAT und SNAT:
192.0.2.1:61001 -> 192.0.2.25:443
```

Dabei ist:

```
192.0.2.1
```

eine Adresse des NAT-Gateways.

Der Server antwortet an das Gateway:

```
192.0.2.25:443 -> 192.0.2.1:61001
```

Das Gateway stellt für den Client wieder die erwartete Ansicht her:

```
203.0.113.10:443 -> 192.0.2.100:53124
```

Vorteil:

- Hin- und Rückverkehr durchlaufen dieselbe NAT-Sitzung.

Nachteil:

- der Server sieht möglicherweise nicht die ursprüngliche Clientadresse;
- Protokolle zeigen die Gatewayadresse;
- quelladressbasierte Richtlinien verändern sich.

Direkter Rückweg als Hairpin-Problem

Client und Server befinden sich im selben internen Subnetz:

```
Client:
192.0.2.100

Server:
192.0.2.25
```

Wenn nur das Ziel übersetzt wird, kann der Server direkt antworten:

```
192.0.2.25:443 -> 192.0.2.100:53124
```

Der Client hat die Verbindung jedoch zu folgender Adresse aufgebaut:

```
203.0.113.10:443
```

Eine direkte Antwort von:

```
192.0.2.25:443
```

passt nicht zur erwarteten Gegenstelle.

Mögliche Symptome:

- interner Zugriff über private Adresse funktioniert;
- externer Zugriff über öffentliche Adresse funktioniert;
- interner Zugriff über öffentliche Adresse scheitert;
- Server sieht das eingehende Paket;
- NAT-Gerät sieht keinen Rückverkehr;
- Client verwirft die unerwartete Antwort.

Hairpin NAT ist nicht überall verfügbar

Mögliche Produktbezeichnungen:

- NAT Loopback;
- NAT Reflection;
- Hairpin NAT;
- U-Turn NAT;
- Same-Interface NAT;
- Intra-Interface NAT.

Zu prüfen sind:

- wird Hairpin NAT unterstützt?
- ist es aktiviert?
- gilt es für IPv4, IPv6 oder beide?
- gilt es für dieselbe Schnittstelle?
- benötigt es eine zusätzliche SNAT-Regel?
- wird die ursprüngliche Clientadresse erhalten?
- greift die richtige Firewallregel?
- verwendet es dieselbe öffentliche Adresse?
- funktioniert es bei mehreren WAN-Adressen?
- ist es mit Load Balancer oder Proxy kompatibel?

Das Verhalten darf nicht von einem anderen Routermodell oder Hersteller übertragen werden.

Split DNS als Alternative

Statt Hairpin NAT kann internes DNS den Dienstnamen direkt auf die interne Adresse auflösen.

Extern:

```
app.example.test
  ->
203.0.113.10
```

Intern:

```
app.example.test
  ->
192.0.2.25
```

Vorteile:

- interner Verkehr bleibt intern;
- keine Hairpin-NAT-Sitzung erforderlich;
- ursprüngliche Clientadresse bleibt sichtbar;

- weniger Last auf der Edge-Firewall;
- einfacherer Datenpfad.

Zu prüfen sind:

- identischer FQDN;
- TLS-Zertifikat für den FQDN;
- interne und externe DNS-Zone;
- TTL;
- DNSSEC-Kontext;
- VPN- und Split-DNS-Verhalten;
- Load-Balancer-Abhängigkeit;
- unterschiedliche Anwendungspfade;
- IPv4- und IPv6-Antworten.

Split DNS ist nicht automatisch besser, wenn intern derselbe Reverse Proxy, dieselbe Web Application Firewall oder dieselben Sicherheitsprüfungen benötigt werden.

Internen und externen Test unterscheiden

Mindestens drei Tests sind getrennt zu dokumentieren:

```
Test A:
interner Client -> interne Serveradresse

Test B:
interner Client -> öffentliche Dienstadresse

Test C:
externer Client -> öffentliche Dienstadresse
```

Einordnung:

Test A	Test B	Test C	Mögliche Einordnung
erfolgreich	erfolgreich	erfolgreich	grundlegender Pfad funktioniert
erfolgreich	fehlerhaft	erfolgreich	Hairpin NAT oder internes DNS
erfolgreich	erfolgreich	fehlerhaft	externe Weiterleitung, Provider, WAN oder Firewall
fehlerhaft	fehlerhaft	fehlerhaft	Serverdienst, Host-Firewall oder interne Route

Test A	Test B	Test C	Mögliche Einordnung
fehlerhaft	erfolgreich	erfolgreich	unterschiedliche interne Adresse oder Sicherheitsregel
erfolgreich	fehlerhaft	fehlerhaft	DNAT-Regel oder öffentliche Adresszuordnung
fehlerhaft	fehlerhaft	erfolgreich	externer Proxy oder anderer Backendpfad

Test C muss tatsächlich aus einem externen Netz erfolgen. Ein Mobilgerät im WLAN ist kein externer Test. Auch ein VPN kann den Datenpfad zurück ins interne Netz führen.

DNS und NAT gemeinsam prüfen

Ein FQDN kann liefern:

- öffentliche IPv4-Adresse;
- öffentliche IPv6-Adresse;
- interne IPv4-Adresse;
- interne IPv6-Adresse;
- Load-Balancer-Adresse;
- CDN-Adresse;
- Proxyadresse.

Windows:

```
Resolve-DnsName `
  -Name "app.example.test" `
  -Type A
```

```
Resolve-DnsName `
  -Name "app.example.test" `
  -Type AAAA
```

Linux und macOS:

```
dig app.example.test A
```

```
dig app.example.test AAAA
```

Zu prüfen sind:

- welche Adresse verwendet die Anwendung tatsächlich?
- ist diese Adresse Bestandteil der NAT-Regel?
- greift IPv6 ohne IPv4-NAT?
- existiert Split DNS?
- verwendet der Browser verschlüsseltes DNS?
- ist ein Proxy beteiligt?
- ist der DNS-Eintrag nach Adressänderung aktuell?

Eine korrekte IPv4-Portweiterleitung hilft nicht, wenn der Client aufgrund eines AAAA-Eintrags direkt IPv6 verwendet.

IPv6 und NAT

IPv6 benötigt für normale öffentliche Erreichbarkeit nicht dieselbe Form von IPv4-Portübersetzung. Ein global adressierter IPv6-Server ist trotzdem nicht automatisch erreichbar.

Zu prüfen sind:

- globale IPv6-Adresse;
- Präfixdelegation;
- IPv6-Route;
- IPv6-Firewall;
- Listener auf IPv6;
- AAAA-Eintrag;
- Privacy Address;
- stabile Serveradresse;
- NDP;
- ICMPv6;
- Providerfilter;
- Reverse Proxy;
- NPTv6 oder NAT64.

NAT darf nicht als Ersatz für eine IPv6-Firewall betrachtet werden.

NPTv6 übersetzt IPv6-Präfixe zustandslos und besitzt ein anderes Verhalten als klassische IPv4-Portübersetzung.

Automatische Portabbildungen

Anwendungen können je nach Umgebung automatische Portabbildungen anfordern, beispielsweise über:

- UPnP Internet Gateway Device;

- NAT-PMP;
- Port Control Protocol;
- herstellerspezifische Verfahren.

Zu prüfen sind:

- ist die Funktion aktiviert?
- welche Anwendung hat die Regel erstellt?
- welcher interne Host ist Ziel?
- welcher externe Port wurde vergeben?
- welches Protokoll wird verwendet?
- wie lange gilt die Abbildung?
- existiert Authentifizierung oder Zugriffskontrolle?
- bleibt die Regel nach Anwendungsende bestehen?
- wird die Regel protokolliert?
- überschneidet sie sich mit administrativen Regeln?

Automatische Portabbildungen können die vorgesehene Sicherheitskontrolle umgehen. Sie dürfen nicht nur zur schnellen Fehlerbehebung aktiviert werden.

Application Level Gateways

Ein NAT-Gerät kann Protokollinhalte analysieren und Adressen oder Ports innerhalb der Nutzdaten verändern.

Mögliche Protokolle:

- FTP;
- SIP;
- H.323;
- TFTP;
- bestimmte VPN-Protokolle;
- ältere RPC-Verfahren.

Zu prüfen sind:

- ALG aktiviert oder deaktiviert;
- verschlüsselte oder unverschlüsselte Nutzdaten;
- dynamisch ausgehandelte Ports;
- NAT-Helper;
- fehlerhafte Protokollumschreibung;
- Herstellerfehler;
- Interaktion mit TLS;
- zusätzliche Firewallregeln.

Ein ALG kann helfen, aber auch Verbindungen beschädigen. Es darf nur nach nachgewiesenem Zusammenhang verändert werden.

NAT und TLS

NAT verändert normalerweise nicht den TLS-Hostnamen oder das Zertifikat.

Beispiel:

```
FQDN:
app.example.test

Öffentliche Adresse:
203.0.113.10

Interner Server:
192.0.2.25
```

Das Zertifikat muss zum verwendeten Namen passen:

```
app.example.test
```

und nicht zwangsläufig zur öffentlichen oder privaten IP-Adresse.

NAT kann jedoch indirekt TLS-Probleme verursachen, wenn:

- falscher Backendserver erreicht wird;
- externer und interner Port unterschiedliche Dienste liefern;
- SNI an falschen Reverse Proxy gelangt;
- Hairpin-Zugriff einen anderen Proxy umgeht;
- Split DNS ein anderes Ziel liefert;
- TLS-Inspektion beteiligt ist;
- Proxy Protocol oder ursprüngliche Clientadresse erwartet wird.

Reverse Proxy und Portweiterleitung

Häufig zeigt eine Portweiterleitung nicht direkt auf den Anwendungsserver, sondern auf einen Reverse Proxy.

Beispiel:

```
Internet:
203.0.113.10:443

DNAT:
192.0.2.20:443

Reverse Proxy:
app.example.test -> 192.0.2.25:8080
```

Zu prüfen sind zwei getrennte Verbindungen:

```
Client -> Reverse Proxy
Reverse Proxy -> Backend
```

Mögliche Fehlerstellen:

- öffentliche DNAT-Regel;
- Firewall;
- TLS-Zertifikat;
- SNI;
- Proxy-VHost;
- Backendadresse;
- Backendport;
- Proxyroute;
- Backendfirewall;
- Health Check;
- Hostheader;
- WebSocket-Weiterleitung;
- Timeout;
- ursprüngliche Clientadresse.

Ein erfolgreicher externer TCP-Handshake bis zum Reverse Proxy beweist nicht, dass das Backend erreichbar ist.

Docker-Portveröffentlichung

Ein Docker-Container in einem Bridge-Netz ist normalerweise nicht allein durch `EXPOSE` von außen erreichbar. Der Port muss veröffentlicht oder über einen anderen Proxy bereitgestellt werden.

Beispiel:

```
docker run \
  -p 8080:80 \
  nginx
```

Bedeutung:

```
Hostport 8080
    ->
Containerport 80
```

Ohne ausdrücklich angegebene Hostadresse kann ein veröffentlichter Port auf allen geeigneten Hostadressen gebunden werden.

Sicherer auf den lokalen Host begrenzt:

```
docker run \
  -p 127.0.0.1:8080:80 \
  nginx
```

Damit ist der Port für einen lokalen Reverse Proxy erreichbar, aber nicht automatisch über jede externe Hostschnittstelle.

Docker-Portzuordnung prüfen

Laufende Container und veröffentlichte Ports:

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

Portzuordnung eines Containers:

```
docker port <Containername>
```

Vollständige Netzwerkinformation:

```
docker inspect <Containername>
```

Hostlistener:

```
ss -lntp
```

Containerlistener:

```
docker exec <Containername> \
  ss -lntp
```

Falls `ss` im Container nicht installiert ist, sind Anwendungsprotokolle, Health Checks oder die im Image verfügbaren Diagnosewerkzeuge zu verwenden.

Zu unterscheiden sind:

```
Routerport
Hostport
Containerport
Anwendungsport
```

Beispiel:

```
203.0.113.10:8443
   ->
192.0.2.20:6875
   ->
Container:80
```

Docker-Netzwerkmodus berücksichtigen

Netzwerkmodus	NAT- oder Portverhalten
Bridge	Portveröffentlichung und NAT üblich
Host	Container verwendet den Netzwerkstack des Hosts; `-p` wird nicht angewendet
Macvlan	Container kann eigene Adresse im Netz besitzen
Ipvlan	direkte adressbasierte Anbindung je nach Betriebsart
Overlay	zusätzlicher Overlay- und Orchestrierungspfad
None	keine normale externe Netzwerkanbindung

Bei `host`-Netzwerkmodus besitzt der Container keinen getrennten Host-zu-Container-Portpfad. Eine konfigurierte `-p`-Option wird nicht als normale Portweiterleitung verwendet.

Docker und Firewallregeln

Docker kann für Bridge-Netze eigene Firewall- und NAT-Regeln erzeugen.

Zu prüfen sind:

- nftables oder iptables;
- Docker-spezifische Chains;
- veröffentlichte Ports;
- Host-Firewall;
- Forwarding;
- Masquerading;
- benutzerdefinierte Regeln;
- Docker-Netzwerk;
- Startreihenfolge;
- Neustart des Docker-Daemons;
- Firewallreload.

Eigene Regeln dürfen nicht ungeprüft vor, nach oder innerhalb von Docker-verwalteten Regelketten eingefügt werden.

Kubernetes-Datenpfad

Mögliche Ebenen:

```
öffentliche Adresse
   ->
Cloud Load Balancer
   ->
Ingress Controller
   ->
Kubernetes Service
   ->
Pod
```

Oder:

```
Node-IP:NodePort
   ->
Service
   ->
Pod
```

Lesende Prüfungen:

```
kubectl get services \
  --all-namespaces \
  -o wide
```

```
kubectl get ingress \
  --all-namespaces
```

```
kubectl get endpointslices \
  --all-namespaces
```

```
kubectl describe service \
  <Servicename> \
  --namespace <Namespace>
```

Zu prüfen sind:

- Service-Typ;
- ClusterIP;
- NodePort;
- LoadBalancer-Adresse;
- TargetPort;
- Selector;
- EndpointSlices;
- Podport;
- Network Policy;
- Ingressregel;
- Hostname;
- TLS;
- External Traffic Policy;
- Source NAT;
- Health Checks.

Windows WinNAT lesend prüfen

NAT-Objekte:

```
Get-NetNat
```

Statische Abbildungen:

```
Get-NetNatStaticMapping
```

Aktive NAT-Sitzungen:

```
Get-NetNatSession
```

Netzwerkschnittstellen:

```
Get-NetIPConfiguration
```

Routing:

```
Get-NetRoute
```

Listener:

```
Get-NetTCPConnection `
  -State Listen
```

WinNAT wird unter anderem für Hyper-V- und Windows-Container-Netze verwendet. Die Cmdlets zeigen keine NAT-Konfiguration eines getrennten physischen Routers.

Linux nftables lesend prüfen

Vollständiges Regelwerk:

```
sudo nft list ruleset
```

Tabellen:

```
sudo nft list tables
```

Eine bekannte NAT-Tabelle anzeigen:

```
sudo nft list table ip nat
```

Die Tabelle muss nicht `ip nat` heißen. Maßgeblich ist die Ausgabe von:

```
sudo nft list tables
```

Regelwerk mit Handles:

```
sudo nft -a list ruleset
```

Zu suchen sind unter anderem:

```
dnat
snat
masquerade
redirect
prerouting
postrouting
output
input
```

Bei nftables sind:

- `dnat` und `redirect` für geeignete NAT-Chains wie `prerouting` oder `output` vorgesehen;
- `snat` und `masquerade` für geeignete NAT-Chains wie `postrouting` vorgesehen.

Die tatsächliche Chain, Priorität und Familie sind zu prüfen.

Vereinfachtes nftables-DNAT-Beispiel

```
table ip nat {
    chain prerouting {
        type nat hook prerouting priority dstnat;
```

```
        iifname "wan0" \
        ip daddr 203.0.113.10 \
        tcp dport 8443 \
        dnat to 192.0.2.25:443
    }
}
```

Dieses Beispiel enthält nur die Zielübersetzung. Zusätzlich erforderlich sein können:

- Forwarding;
- Filterregel;
- Rückroute;
- SNAT;
- Hairpin-Regel;
- Protokollierung;
- IPv6-Regel;
- dauerhaftes Konfigurationsmanagement.

Es darf nicht ungeprüft in ein produktives Regelwerk übernommen werden.

Vereinfachtes nftables-Hairpin-Prinzip

```
table ip nat {
    chain prerouting {
        type nat hook prerouting priority dstnat;

        ip saddr 192.0.2.0/24 \
        ip daddr 203.0.113.10 \
        tcp dport 443 \
        dnat to 192.0.2.25:443
    }

    chain postrouting {
        type nat hook postrouting priority srcnat;

        ip saddr 192.0.2.0/24 \
        ip daddr 192.0.2.25 \
        tcp dport 443 \
        masquerade
```

```
  }
}
```

Dieses vereinfachte Prinzip erzwingt, dass der Rückverkehr wieder über das NAT-Gateway läuft. Adressen, Schnittstellen, Sicherheitsregeln und Regelprioritäten müssen an die tatsächliche Umgebung angepasst und freigegeben werden.

Linux iptables lesend prüfen

NAT-Regeln:

```
sudo iptables \
  -t nat \
  -L \
  -n \
  -v \
  --line-numbers
```

Regeln in speicherbarer Syntax:

```
sudo iptables \
  -t nat \
  -S
```

IPv6-Regeln, sofern verwendet:

```
sudo ip6tables \
  -t nat \
  -S
```

Filter- und Forwardingregeln:

```
sudo iptables \
  -L FORWARD \
  -n \
  -v \
  --line-numbers
```

Aktive Weiterleitung:

```
sysctl net.ipv4.ip_forward
```

IPv6-Weiterleitung:

```
sysctl net.ipv6.conf.all.forwarding
```

Linux Connection Tracking

Aktive NAT- und Verbindungseinträge:

```
sudo conntrack -L
```

TCP-Einträge zu einem internen Server:

```
sudo conntrack -L \
  -p tcp \
  -d 192.0.2.25 \
  --dport 443
```

Ereignisse während eines kontrollierten Tests:

```
sudo conntrack -E
```

Zu prüfen sind:

- Originalrichtung;
- Antwortrichtung;
- Quell- und Zielports;
- `src-nat`;
- `dst-nat`;
- `[UNREPLIED]`;
- `[ASSURED]`;
- Timeout;
- Paket- und Bytezähler.

Das vollständige Leeren der Conntrack-Tabelle ist keine erste Diagnosemaßnahme.

Cisco Secure Firewall ASA prüfen

NAT-Regeln:

```
show nat
```

Details und Treffer:

```
show nat detail
```

Übersetzungen:

```
show xlate
```

Verbindungen:

```
show conn
```

Nach Adresse filtern:

```
show conn address 192.0.2.25
```

Routing:

```
show route
```

Simulation:

```
packet-tracer input outside tcp 198.51.100.40 53124 203.0.113.10 8443 detailed
```

Zu prüfen sind:

- passende NAT-Regel;
- NAT-Abschnitt und Reihenfolge;
- reale und übersetzte Adresse;
- externer und interner Port;
- ACL;
- Route;
- Eingangs- und Ausgangsschnittstelle;
- Ergebnis jeder Verarbeitungsphase.

Bei aktuellen Cisco-ASA-Regelwerken werden Zugriffsregeln im NAT-Kontext anhand realer Adressen ausgewertet. Dieses Verhalten darf nicht auf andere Produkte übertragen werden.

Cloud-NAT und eingehende Verbindungen

Ein Cloud-NAT-Gateway dient häufig ausschließlich der ausgehenden Kommunikation privater Ressourcen.

Wichtig:

```
Outbound NAT
ist nicht automatisch
Inbound Port Forwarding
```

Für eingehende Veröffentlichung können stattdessen erforderlich sein:

- öffentliche Instanzadresse;
- Load Balancer;
- Network Load Balancer;
- Application Gateway;
- Cloud Firewall DNAT;
- Reverse Proxy;
- API Gateway;
- Ingress Controller;
- private Verbindung;
- VPN.

AWS NAT Gateway

AWS NAT Gateway erlaubt privaten Ressourcen ausgehende Verbindungen. Es nimmt keine unaufgeforderten eingehenden Internetverbindungen für diese Ressourcen an.

Zu prüfen sind:

- public oder private NAT Gateway;
- Subnetz;
- Routingtabelle;
- Internet Gateway;
- Elastic IP;
- Security Groups der Instanz;
- Network ACLs;
- Rückverkehr einer ausgehenden Sitzung;
- CloudWatch-Metriken.

Eine Portweiterleitung auf ein AWS NAT Gateway ist nicht der vorgesehene Weg zur Veröffentlichung eines privaten Servers.

Mögliche Alternativen:

- Application Load Balancer;
- Network Load Balancer;
- öffentliche Instanzadresse mit Security Group;
- AWS Network Firewall beziehungsweise geeignete zentrale Firewall;
- API Gateway;
- VPN oder PrivateLink, abhängig vom Anwendungsfall.

Azure NAT Gateway

Azure NAT Gateway stellt ausgehende Internetkonnektivität für ein Subnetz bereit. Unaufgeforderte eingehende Internetverbindungen werden nicht bereitgestellt. DNAT wird nur für Antwortpakete einer zuvor ausgehend aufgebauten Kommunikation angewendet.

Für eingehende Veröffentlichung können je nach Anforderung verwendet werden:

- Azure Load Balancer;
- Application Gateway;
- Azure Firewall DNAT;
- öffentliche IP-Adresse einer Ressource;
- API Management;
- Front Door;
- privater Endpunkt;
- VPN.

Zu prüfen sind:

- Subnetzanbindung;
- öffentliche NAT-Adressen;
- effektive Route;
- Azure Firewall;
- Load Balancer;
- Network Security Group;
- User Defined Route;
- Zielressource;
- Health Probe.

Paketaufzeichnung entlang des NAT-Pfads

Für einen NAT-Nachweis sind mehrere Beobachtungspunkte erforderlich.

Beispiel:

Beobachtungspunkt	Erwarteter Datenfluss
vor Edge-Firewall	`198.51.100.40:53124 -> 203.0.113.10:8443`
nach DNAT	`198.51.100.40:53124 -> 192.0.2.25:443`
am Server	`198.51.100.40:53124 -> 192.0.2.25:443`
Serverantwort	`192.0.2.25:443 -> 198.51.100.40:53124`
nach Rückübersetzung	`203.0.113.10:8443 -> 198.51.100.40:53124`

Mit `tcpdump` auf dem externen Abschnitt:

```
sudo tcpdump \
  -ni <WAN-Schnittstelle> \
  'host 198.51.100.40 and host 203.0.113.10 and tcp port 8443'
```

Auf dem internen Abschnitt:

```
sudo tcpdump \
  -ni <LAN-Schnittstelle> \
  'host 198.51.100.40 and host 192.0.2.25 and tcp port 443'
```

Am Server:

```
sudo tcpdump \
  -ni any \
  'host 198.51.100.40 and tcp port 443'
```

Die Platzhalter müssen durch autorisierte reale Werte ersetzt werden.

Befund aus Paketaufzeichnungen

WAN-Eingang	LAN-Ausgang	Serverantwort	Einordnung
nein	nein	nein	Problem vor dem NAT-Gerät
ja	nein	nein	NAT-, Firewall- oder Routingproblem
ja	ja	nein	Server, Host-Firewall oder Dienst prüfen
ja	ja	ja, aber nicht am NAT	falscher Rückweg

WAN-Eingang	LAN-Ausgang	Serverantwort	Einordnung
ja	ja	ja und Rückübersetzung	Clientpfad oder Anwendung prüfen
Hairpin am LAN sichtbar	kein Hairpin-Ausgang	Hairpin NAT fehlt oder passt nicht	
DNAT korrekt	falscher interner Port	NAT-Regel korrigieren	
Paket erreicht falschen Server	falsches Zielobjekt oder alte Regel	Regel und Objekt prüfen	

Hardware-Offloading kann verhindern, dass eine allgemeine Softwareaufzeichnung alle Pakete zeigt. Dann sind herstellerspezifische Captures, SPAN oder TAP erforderlich.

Porttest richtig ausführen

Windows:

```
Test-NetConnection `
  -ComputerName "203.0.113.10" `
  -Port 8443 `
  -InformationLevel Detailed
```

Linux und macOS:

```
nc -vz 203.0.113.10 8443
```

HTTPS mit FQDN:

```
curl \
  -v \
  https://app.example.test:8443/
```

Ein TCP-Porttest beweist nur den TCP-Aufbau bis zu der Komponente, die antwortet. Er beweist nicht:

- richtigen Backendserver;
- korrekte Anwendung;
- gültiges TLS-Zertifikat;
- korrekten Hostheader;
- funktionierende Authentifizierung;
- funktionierende Datenbank;

- korrekte UDP-Weiterleitung.

Ein UDP-Test benötigt eine anwendungsspezifische Antwort, Protokollauswertung oder Paketaufzeichnung.

Externer Test

Ein externer Test muss aus einem tatsächlich unabhängigen Netz erfolgen.

Mögliche Testsysteme:

- autorisierter Server in einem anderen Standort;
- freigegebene Cloud-VM;
- Mobilfunkverbindung ohne WLAN;
- externer Monitoringstandort;
- vereinbarter Testhost eines Dienstleisters.

Zu dokumentieren sind:

- externe Quelladresse;
- Zeitpunkt;
- DNS-Ergebnis;
- Zieladresse;
- Zielport;
- Protokoll;
- Ergebnis;
- Firewalllog;
- NAT-Regeltreffer;
- Serversicht.

Öffentliche Portscan-Webseiten sollten nicht unkontrolliert für interne oder sensible Dienste verwendet werden.

NAT-Protokollierung

Ein geeigneter NAT- oder Firewalllogeintrag enthält möglichst:

- Zeitstempel;
- Eingangs- und Ausgangsschnittstelle;
- Quellzone;
- Zielzone;
- Original-Quelladresse;
- Original-Quellport;
- Original-Zieladresse;
- Original-Zielport;

- übersetzte Quelladresse;
- übersetzter Quellport;
- übersetzte Zieladresse;
- übersetzter Zielport;
- Protokoll;
- NAT-Regel;
- Sicherheitsregel;
- Sitzungs-ID;
- Aktion;
- Pakete;
- Bytes;
- Sitzungsendgrund.

Ein fehlender NAT-Logeintrag kann bedeuten:

- Datenverkehr erreicht das Gerät nicht;
- Logging ist deaktiviert;
- falscher Clusterknoten wird geprüft;
- falscher virtueller Kontext wird geprüft;
- andere NAT-Regel trifft;
- Paket wird vor NAT verworfen;
- Sitzung bestand bereits;
- Hardware-Offload verwendet andere Zähler.

Hypothese und Gegenbeweis formulieren

Beispiel:

```
Hypothese:
Die Portweiterleitung 203.0.113.10:8443 zeigt korrekt auf
192.0.2.25:443. Interne Clients können die öffentliche Adresse
jedoch nicht verwenden, weil die Firewall kein Hairpin NAT für
dieselbe interne Zone ausführt.

Erwarteter Befund:
Der interne Direktzugriff auf 192.0.2.25:443 funktioniert.
Ein externer Zugriff auf 203.0.113.10:8443 funktioniert.
Beim internen Zugriff auf 203.0.113.10:8443 steigt der Trefferzähler
der normalen WAN-DNAT-Regel nicht oder der Rückverkehr umgeht die
NAT-Sitzung.

Gegenbeweis:
```

Der Hairpin-Datenfluss wird nachweislich übersetzt, Hin- und Rückverkehr durchlaufen dieselbe NAT-Sitzung und die Antwort erreicht den Client mit der erwarteten öffentlichen Quelladresse.

Testmethode:
Drei getrennte Zugriffe über interne Adresse, öffentliche Adresse von intern und öffentliche Adresse von extern durchführen. NAT-Zähler, Sitzungstabelle und Paketaufzeichnung korrelieren.

Risiko:
Nur lesende Diagnose und kontrollierte Verbindungstests.

Erfolgskriterium:
Fehler ist eindeutig auf Hairpin-Verarbeitung, Rückweg oder eine nachgelagerte Anwendungsebene begrenzt.

Kontrollierte Maßnahmen

Maßnahme	Voraussetzung	Risiko	Rückweg
DNAT-Ziel korrigieren	falsche interne Adresse nachgewiesen	anderes System wird veröffentlicht	alte Zieladresse wiederherstellen
externen Port korrigieren	falscher Port bestätigt	Clients müssen neuen Port verwenden	alten Port wiederherstellen
internen Port korrigieren	tatsächlicher Listener nachgewiesen	falscher Dienst kann erreicht werden	alten Zielport wiederherstellen
Protokoll korrigieren	TCP-/UDP-Abweichung bestätigt	zusätzliche Erreichbarkeit	alte Protokollregel
Quellbereich einschränken	zulässige Quellen bekannt	legitime Quellen können fehlen	vorherigen Bereich wiederherstellen
Rückroute korrigieren	asymmetrischer Rückweg bestätigt	weitere Ziele betroffen	alte Route wiederherstellen
Hairpin DNAT ergänzen	interner öffentlicher Zugriff erforderlich	zusätzlicher interner Pfad	Regel entfernen
Hairpin SNAT ergänzen	direkter Rückweg verhindert Sitzung	ursprüngliche Client-IP geht verloren	SNAT-Regel entfernen
Split DNS einrichten	interner Direktpfad freigegeben	interner Pfad unterscheidet sich extern	vorherige DNS-Antwort
Host-Firewall korrigieren	Serverdrop bestätigt	zusätzliche Angriffsfläche	alte Regel
Containerport korrigieren	Host-/Containerabweichung bestätigt	Dienst wird anders veröffentlicht	alte Portzuordnung

Maßnahme	Voraussetzung	Risiko	Rückweg
Reverse-Proxy-Ziel korrigieren	falsches Backend bestätigt	Anwendungspfad verändert sich	alte Proxykonfiguration
öffentliche IPv4 bereitstellen	CGNAT bestätigt und fachlich erforderlich	Kosten und größere Angriffsfläche	Vertrag oder Zuweisung zurücknehmen
NAT-Regelreihenfolge korrigieren	Überschattung nachgewiesen	andere Datenflüsse betroffen	alte Reihenfolge

---

Vollständiger Diagnoseablauf

1. Exakte Fehlermeldung erfassen.
2. Datum, Uhrzeit und Zeitzone dokumentieren.
3. Betroffene Anwendung bestimmen.
4. FQDN dokumentieren.
5. DNS-A- und AAAA-Antworten prüfen.
6. tatsächlich verwendete Zieladresse bestimmen.
7. TCP, UDP oder anderes Protokoll bestimmen.
8. externen Zielport bestimmen.
9. internen Zielport bestimmen.
10. externe Quelladresse dokumentieren.
11. öffentliche Zieladresse dokumentieren.
12. internes Ziel dokumentieren.
13. Vor-NAT-Tupel dokumentieren.
14. Nach-NAT-Tupel dokumentieren.
15. Umfang der Störung bestimmen.
16. interne Direktverbindung testen.
17. internen Zugriff über öffentliche Adresse testen.
18. tatsächlich externen Zugriff testen.
19. WAN-Adresse des Edge-Geräts prüfen.
20. vorgeschaltetes NAT oder CGNAT prüfen.
21. doppeltes NAT berücksichtigen.
22. richtige Eingangsschnittstelle bestimmen.
23. richtige öffentliche Zieladresse bestimmen.
24. aktive NAT-Konfiguration prüfen.
25. NAT-Regelreihenfolge prüfen.
26. NAT-Regeltreffer dokumentieren.
27. Firewallregel getrennt prüfen.
28. Routing zum internen Ziel prüfen.
29. Host-Firewall prüfen.
30. Listener des Serverdienstes prüfen.
31. Serverantwort und Rückroute prüfen.
32. NAT-Sitzung oder Translation Table prüfen.
33. Hin- und Rückverkehr vergleichen.
34. Hairpin-Unterstützung prüfen.

35. Hairpin-DNAT und gegebenenfalls SNAT prüfen.
36. Split-DNS-Verhalten berücksichtigen.
37. Reverse Proxy oder Load Balancer prüfen.
38. Container-Hostport und Containerport prüfen.
39. Kubernetes-Servicepfad berücksichtigen.
40. automatische Portabbildungen prüfen.
41. ALG oder NAT-Helper berücksichtigen.
42. IPv4 und IPv6 getrennt prüfen.
43. Cloud-NAT und Inbound-Komponente unterscheiden.
44. bei Bedarf Paketaufzeichnung vor und nach NAT durchführen.
45. Hypothese und Gegenbeweis formulieren.
46. genau eine kontrollierte Maßnahme vorbereiten.
47. Risiko, Rückweg und Erfolgskriterium dokumentieren.
48. Maßnahme freigeben und umsetzen.
49. neue Sitzung aufbauen.
50. internen, Hairpin- und externen Test wiederholen.
51. NAT-, Firewall- und Anwendungslogs verifizieren.
52. weitere repräsentative Quellen testen.
53. temporäre Regeln und Captures zurücknehmen.
54. Ursache, Maßnahme und Prävention dokumentieren.

Befundmatrix

Befund	Mögliche Einordnung	Nächster Nachweis
interne Adresse funktioniert, öffentliche intern nicht	Hairpin NAT oder Split DNS	Hairpin-Sitzung und Rückweg prüfen
öffentliche Adresse extern funktioniert	DNAT und externer Pfad grundsätzlich aktiv	internen Hairpin-Pfad getrennt prüfen
öffentliche Adresse extern nicht erreichbar	Provider, CGNAT, WAN, DNAT oder Firewall	WAN-IP und NAT-Regeltreffer
NAT-Regelzähler bleibt null	falsche IP, Schnittstelle, Port oder Pfad	Capture vor dem NAT-Gerät
NAT-Regel trifft, Firewall blockiert	getrennte Sicherheitsrichtlinie	Firewalllog und Adressphase
NAT und Firewall erlauben, Server antwortet nicht	Host-Firewall, Listener oder Dienst	serverseitiges Capture
Server antwortet, NAT sieht Antwort nicht	falsche Rückroute	Serverroute und Gateway
Rückantwort besitzt private Quelladresse	fehlende Rückübersetzung oder Hairpin-Asymmetrie	NAT-Sitzung und Pfad
WAN-Adresse ist privat	vorgeschaltetes NAT	Upstreamgerät prüfen
WAN-Adresse liegt in `100.64.0.0/10`	CGNAT wahrscheinlich	Provider und externe Adresse

Befund	Mögliche Einordnung	Nächster Nachweis
Router-WAN-IP weicht von externer IP ab	vorgeschaltetes NAT, VPN oder Proxy	vollständigen Ausgangspfad prüfen
TCP funktioniert, UDP nicht	Protokollregel oder UDP-State	UDP-Capture und NAT-Timeout
IPv4 funktioniert, FQDN nicht	AAAA-Pfad oder DNS	A-/AAAA-Antwort vergleichen
externer Port offen, falsche Anwendung	falsches DNAT-Ziel oder Reverse Proxy	Backend und Hostheader prüfen
Docker-Hostport lauscht nicht	Port nicht veröffentlicht oder Container gestoppt	`docker ps` und `docker port`
Hostport lauscht, Container antwortet nicht	Containerdienst oder Zielport	Containerlistener und Logs
Container direkt erreichbar, öffentlich nicht	Router-, Host- oder DNAT-Pfad	schrittweise Porttests
Zugriff nach Routerneustart kurz möglich	Zustand, dynamische Adresse oder Konflikt	Translation Table und WAN-IP
nur manche Quellen funktionieren	Quellfilter oder NAT-Filterverhalten	Regeln und Quelladressen vergleichen
Zugriff über IP, nicht über Namen	DNS, SNI oder Zertifikat	FQDN und TLS-Prüfung
intern anderer Server als extern	Split DNS oder Proxyunterschied	DNS-Antworten und Backend
AWS NAT Gateway sendet ausgehend, kein Inbound	erwartetes Produktverhalten	Load Balancer oder andere Inbound-Komponente
Azure NAT Gateway sendet ausgehend, kein Inbound	erwartetes Produktverhalten	Azure Firewall DNAT oder Load Balancer
nach Regeländerung alte Sitzung bleibt	bestehender Translation State	vollständig neue Sitzung testen
extern klappt nur über einen WAN-Anschluss	NAT-Regel nur an einer Schnittstelle	WAN-Bindung und Routing

---

Typische Diagnosefehler

- NAT als Firewall betrachten.
- Vorhandene NAT-Regel als vollständige Freigabe bewerten.
- Öffentliche und private Adresse verwechseln.
- externen und internen Port verwechseln.
- TCP und UDP nicht unterscheiden.
- IPv4 und IPv6 nicht getrennt prüfen.
- FQDN nicht auf A und AAAA prüfen.
- Regel an falscher WAN-Schnittstelle übersehen.
- NAT-Regelreihenfolge ignorieren.
- alte oder überschattete Regel nicht erkennen.
- Rückweg nicht prüfen.
- nur Hinrichtung aufzeichnen.

- NAT-Sitzung nicht auswerten.
- bestehende und neue Sitzung nicht unterscheiden.
- interne Direktverbindung als Nachweis externer Erreichbarkeit verwenden.
- internen Zugriff auf öffentliche Adresse als externen Test bewerten.
- Mobilgerät im WLAN als externes System verwenden.
- VPN-Pfad beim externen Test übersehen.
- CGNAT nicht berücksichtigen.
- `100.64.0.0/10` mit normalem RFC-1918-Netz gleichsetzen.
- doppeltes NAT übersehen.
- DMZ Host oder Exposed Host als schnelle Lösung aktivieren.
- Hairpin NAT als überall vorhandene Funktion annehmen.
- Split DNS ohne Betrachtung von Proxy und Sicherheitskontrollen einsetzen.
- Hairpin-SNAT ohne Auswirkungen auf Clientprotokollierung aktivieren.
- NAT als Schutzmechanismus für IPv6 verwenden.
- UPnP zur Fehlerbehebung pauschal aktivieren.
- automatisch erzeugte Portabbildungen nicht prüfen.
- ALG ohne Nachweis deaktivieren oder aktivieren.
- Docker-`EXPOSE` mit Portveröffentlichung verwechseln.
- Docker-Hostport und Containerport verwechseln.
- `-p` bei Host-Netzwerkmodus als wirksam annehmen.
- Kubernetes Service Port und TargetPort verwechseln.
- Cloud NAT Gateway als Inbound-Portweiterleitung behandeln.
- erfolgreichen TCP-Handshake als Anwendungserfolg bewerten.
- NAT- oder Conntrack-Tabelle vorsorglich leeren.
- mehrere NAT- und Firewallregeln gleichzeitig verändern.
- temporäre Portweiterleitung aktiv lassen.
- Ursache und Sicherheitswirkung nicht dokumentieren.

Verifikation

Nach einer Maßnahme müssen mindestens folgende Punkte geprüft werden:

- DNS liefert die vorgesehenen Adressen;
- Client verwendet die erwartete Adressfamilie;
- öffentliche Zieladresse stimmt;
- WAN-Adresse stimmt;
- vorgeschaltetes NAT ist berücksichtigt;
- externes Protokoll stimmt;
- externer Port stimmt;
- internes Ziel stimmt;
- interner Port stimmt;
- NAT-Regel trifft;
- richtige NAT-Regel trifft;
- Regelreihenfolge ist korrekt;
- Firewallregel erlaubt nur den benötigten Verkehr;

- Quellbereich ist angemessen begrenzt;
- internes Routing funktioniert;
- Server besitzt die erwartete Adresse;
- Serverdienst lauscht;
- Host-Firewall erlaubt den benötigten Verkehr;
- Rückroute führt über die richtige NAT-Komponente;
- NAT-Sitzung enthält korrekte Original- und Antworttupel;
- interner Direktzugriff funktioniert;
- Hairpin-Zugriff funktioniert, sofern vorgesehen;
- Split DNS funktioniert, sofern vorgesehen;
- externer Zugriff aus einem unabhängigen Netz funktioniert;
- TCP und UDP funktionieren, sofern erforderlich;
- IPv4 und IPv6 funktionieren, sofern vorgesehen;
- TLS-Zertifikat und SNI stimmen;
- Reverse Proxy erreicht das richtige Backend;
- Docker-Host- und Containerport stimmen;
- Kubernetes Service und Endpoint stimmen;
- keine unbeabsichtigte Portveröffentlichung besteht;
- keine allgemeine Exposed-Host-Regel aktiv ist;
- Logs zeigen die erwartete Quelle und Übersetzung;
- Monitoring erkennt Ausfälle;
- temporäre Regeln und Captures wurden zurückgenommen;
- Ursache, Maßnahme und Prävention wurden dokumentiert.

Eine erfolgreiche Verbindung aus dem internen Netz reicht nicht als Verifikation einer Internet-Portweiterleitung.

Dokumentationsvorlage

```
Störung:
<exakte Beschreibung>

Datum und Uhrzeit:
<Zeitpunkt mit Zeitzone>

Betroffene Anwendung:
<Anwendung oder Dienst>

FQDN:
<Name>

DNS-Antworten:
```

<A- und AAAA-Einträge>

Externe Quelle:

<IP-Adresse und Quellport>

Öffentliches Ziel:

<IP-Adresse und Zielport>

Protokoll:

<TCP, UDP oder anderes Protokoll>

Internes Ziel:

<IP-Adresse und Zielport>

Vor-NAT-Tupel:

<Quelle:Port -> Ziel:Port>

Nach-NAT-Tupel:

<Quelle:Port -> Ziel:Port>

NAT-Gerät:

<Name, Clusterknoten und Kontext>

Eingangsschnittstelle:

<Name>

Ausgangsschnittstelle:

<Name>

NAT-Regel:

<Name, ID und Position>

Firewallregel:

<Name und ID>

NAT-Sitzung:

<Original- und Antwortdaten>

WAN-Adresse:

<Adresse>

Extern beobachtete Adresse:

<Adresse>

Vorgeschaltetes NAT:

<ja, nein oder ungeklärt>

CGNAT:

<ja, nein oder ungeklärt>

Rückroute:

<Gateway und Pfad>

Serverlistener:

<Adresse und Port>

Host-Firewall:

<Befund>

Containerzuordnung:

<Routerport -> Hostport -> Containerport>

Reverse Proxy:

<Frontend und Backend>

Interner Direktzugriff:

<Ergebnis>

Interner Zugriff über öffentliche Adresse:

<Ergebnis>

Externer Zugriff:

<Ergebnis>

Paketaufzeichnung:

<Befund vor und nach NAT>

Hypothese:

```
<vermutete Ursache>

Erwarteter Nachweis:
<messbarer Befund>

Gegenbeweis:
<widerlegender Befund>

Nachgewiesene Ursache:
<technischer Nachweis>

Durchgeführte Maßnahme:
<genau eine kontrollierte Änderung>

Risiko:
<mögliche Nebenwirkungen>

Rückweg:
<Rollback>

Verifikation:
<interner, Hairpin- und externer Test>

Prävention:
<Monitoring, Dokumentation oder Architekturverbesserung>
```

Checkliste

- [ ] exakte Fehlermeldung dokumentiert
- [ ] Datum, Uhrzeit und Zeitzone erfasst
- [ ] Anwendung bestimmt
- [ ] FQDN dokumentiert
- [ ] A-Eintrag geprüft
- [ ] AAAA-Eintrag geprüft
- [ ] tatsächliche Zieladresse bestimmt
- [ ] IPv4 und IPv6 unterschieden
- [ ] TCP und UDP unterschieden
- [ ] externe Quelladresse erfasst

- [ ] öffentlichen Zielport erfasst
- [ ] internes Ziel erfasst
- [ ] internen Zielport erfasst
- [ ] Vor-NAT-Tupel dokumentiert
- [ ] Nach-NAT-Tupel dokumentiert
- [ ] Fehlerumfang bestimmt
- [ ] interne Direktverbindung getestet
- [ ] internen Zugriff über öffentliche Adresse getestet
- [ ] echten externen Zugriff getestet
- [ ] WAN-Adresse geprüft
- [ ] extern beobachtete Adresse geprüft
- [ ] vorgeschaltetes NAT geprüft
- [ ] CGNAT geprüft
- [ ] doppeltes NAT geprüft
- [ ] öffentliche Adresszuordnung geprüft
- [ ] richtige Eingangsschnittstelle geprüft
- [ ] richtige Ausgangsschnittstelle geprüft
- [ ] NAT-Regel bestimmt
- [ ] NAT-Regelposition geprüft
- [ ] NAT-Regeltreffer geprüft
- [ ] Überschattung geprüft
- [ ] NAT-Ausnahme geprüft
- [ ] Firewallregel getrennt geprüft
- [ ] Quellbereich geprüft
- [ ] interne Route geprüft
- [ ] Rückroute geprüft
- [ ] Serveradresse geprüft
- [ ] DHCP-Reservierung geprüft
- [ ] Serverlistener geprüft
- [ ] Host-Firewall geprüft
- [ ] NAT-Sitzung geprüft
- [ ] Originalrichtung geprüft
- [ ] Antwortrichtung geprüft
- [ ] Pakete und Bytes je Richtung geprüft

- [ ] Hairpin-Unterstützung geprüft
- [ ] Hairpin-DNAT geprüft
- [ ] Hairpin-SNAT geprüft
- [ ] Auswirkung auf Client-IP-Protokollierung geprüft
- [ ] Split DNS geprüft
- [ ] TLS-Zertifikat geprüft
- [ ] SNI geprüft
- [ ] Reverse Proxy geprüft
- [ ] Load Balancer geprüft
- [ ] Backend geprüft
- [ ] Docker-Netzwerkmodus geprüft
- [ ] Docker-Hostport geprüft
- [ ] Containerport geprüft
- [ ] Containerlistener geprüft
- [ ] Kubernetes Service geprüft
- [ ] Kubernetes TargetPort geprüft
- [ ] EndpointSlices geprüft
- [ ] automatische Portabbildungen geprüft
- [ ] UPnP, NAT-PMP oder PCP geprüft
- [ ] ALG und NAT-Helper geprüft
- [ ] Cloud-NAT und Inbound-Komponente unterschieden
- [ ] Paketaufzeichnung vor NAT durchgeführt, sofern erforderlich
- [ ] Paketaufzeichnung nach NAT durchgeführt, sofern erforderlich
- [ ] Hypothese formuliert
- [ ] Gegenbeweis definiert
- [ ] Risiko dokumentiert
- [ ] Rückweg dokumentiert
- [ ] nur eine kontrollierte Maßnahme durchgeführt
- [ ] vollständig neue Sitzung getestet
- [ ] ursprüngliche Anwendung verifiziert
- [ ] weitere repräsentative Quellen geprüft
- [ ] temporäre Weiterleitung entfernt
- [ ] temporäre Captures entfernt
- [ ] Ursache und Prävention dokumentiert

Schnellreferenz

Aufgabe	Befehl
Windows-NAT-Objekte	`Get-NetNat`
Windows-statische NAT-Abbildungen	`Get-NetNatStaticMapping`
Windows-NAT-Sitzungen	`Get-NetNatSession`
Windows-Routing	`Get-NetRoute`
Windows-Listener	`Get-NetTCPConnection -State Listen`
Windows-DNS A	`Resolve-DnsName -Name "app.example.test" -Type A`
Windows-DNS AAAA	`Resolve-DnsName -Name "app.example.test" -Type AAAA`
Windows-TCP-Porttest	`Test-NetConnection -ComputerName "203.0.113.10" -Port 8443`
Linux-nftables-Regelwerk	`sudo nft list ruleset`
Linux-nftables-Tabellen	`sudo nft list tables`
Linux-nftables mit Handles	`sudo nft -a list ruleset`
Linux-iptables-NAT	`sudo iptables -t nat -L -n -v --line-numbers`
Linux-iptables-NAT-Regeln	`sudo iptables -t nat -S`
Linux-Forwardingregeln	`sudo iptables -L FORWARD -n -v --line-numbers`
Linux-IPv4-Forwarding	`sysctl net.ipv4.ip_forward`
Linux-Conntrack	`sudo conntrack -L`
Linux-Conntrack-Ereignisse	`sudo conntrack -E`
Linux-Listener	`ss -lntp`
Linux/macOS-DNS A	`dig app.example.test A`
Linux/macOS-DNS AAAA	`dig app.example.test AAAA`
Linux/macOS-TCP-Porttest	`nc -vz 203.0.113.10 8443`
HTTPS-Test	`curl -v https://app.example.test:8443/`
Docker-Ports	`docker ps --format 'table {{.Names}}\t{{.Ports}}'`
Docker-Portzuordnung	`docker port <Containername>`
Docker-Details	`docker inspect <Containername>`
Kubernetes-Services	`kubectl get services --all-namespaces -o wide`
Kubernetes-Ingress	`kubectl get ingress --all-namespaces`
Kubernetes-Endpunkte	`kubectl get endpointslices --all-namespaces`
Cisco-ASA-NAT	`show nat`
Cisco-ASA-NAT-Details	`show nat detail`

Aufgabe	Befehl
Cisco-ASA-Übersetzungen	`show xlate`
Cisco-ASA-Verbindungen	`show conn`
Cisco-ASA-Routing	`show route`

Verändernde Befehle und Maßnahmen, die nicht als erste Diagnose verwendet werden dürfen:

```
New-NetNat
Add-NetNatStaticMapping
Remove-NetNat
Remove-NetNatStaticMapping
nft add rule ... dnat ...
nft add rule ... snat ...
nft flush ruleset
iptables -t nat -F
conntrack -F
clear xlate
clear conn
Firewall deaktivieren
Exposed Host aktivieren
DMZ Host aktivieren
alle Ports weiterleiten
UPnP pauschal aktivieren
NAT-PMP pauschal aktivieren
PCP pauschal aktivieren
ALG ungeprüft deaktivieren
ALG ungeprüft aktivieren
Router vorsorglich neu starten
Firewall vorsorglich neu starten
öffentliche Verwaltungsports testweise freigeben
allgemeine any-any-Regel erstellen
```

Quellen

Standards

- RFC 2663 – IP Network Address Translator Terminology and Considerations
- RFC 3022 – Traditional IP Network Address Translator

- RFC 4787 – NAT Behavioral Requirements for UDP
- RFC 5382 – NAT Behavioral Requirements for TCP
- RFC 5508 – NAT Behavioral Requirements for ICMP
- RFC 7857 – Updates to NAT Behavioral Requirements
- RFC 1918 – Address Allocation for Private Internets
- RFC 6598 – Shared Address Space Request
- RFC 6886 – NAT Port Mapping Protocol
- RFC 6887 – Port Control Protocol
- RFC 6296 – IPv6-to-IPv6 Network Prefix Translation
- RFC 2993 – Architectural Implications of NAT

Microsoft Windows

- Microsoft Learn – Get-NetNat
- Microsoft Learn – Get-NetNatStaticMapping
- Microsoft Learn – Get-NetNatSession
- Microsoft Learn – Set Up a NAT Network
- Microsoft Learn – Windows Container Network Drivers

Linux Netfilter

- Netfilter – nftables Manual
- Netfilter – iptables Extensions Manual
- Netfilter – Conntrack Tools User Manual
- Linux Kernel Documentation – Netfilter Conntrack Sysfs Variables

Docker und Kubernetes

- Docker Docs – Port Publishing and Mapping
- Docker Docs – Networking Overview
- Docker Docs – Bridge Network Driver
- Docker Docs – Host Network Driver
- Docker Docs – Packet Filtering and Firewalls
- Kubernetes Documentation – Service
- Kubernetes Documentation – Ingress

Cisco

- Cisco Secure Firewall ASA – NAT Basics
- Cisco – Configure NAT and ACLs on ASA
- Cisco Secure Firewall ASA – `packet-tracer` Command Reference

Amazon Web Services

- AWS – NAT Gateways
- AWS – NAT Gateway Basics
- AWS – NAT Gateway Use Cases
- AWS – Internet Gateways

Microsoft Azure

- Microsoft Learn – Azure NAT Gateway Overview
- Microsoft Learn – Azure NAT Gateway Resource
- Microsoft Learn – Azure Load Balancer Outbound Connections
- Microsoft Learn – Azure Firewall DNAT

Für diese Seite wurden keine Community-Beiträge oder Hersteller-Social-Media-Aussagen als fachlicher Nachweis verwendet.
