Engineer

NAT

🔀 omskrivning af IP-adresser, når pakker passerer en router

Hvad det er

NAT, en forkortelse af network address translation eller på dansk netværksadresseoversættelse, er omskrivning af IP-adresser i pakker, mens de passerer en router, så værter, der er nummereret i ét adresserum, kan tale med værter, der er nummereret i et andet. RFC 2663, dokumentet fra 1999, der fastlagde ordforrådet, beskriver det som en afbildning af IP-adresser fra ét område til et andet med gennemsigtig routing som mål: værterne på begge sider skal ikke behøve at vide, at det sker.

Den form, næsten alle møder, er det, RFC 3022 fra januar 2001 kalder traditionel NAT. Dens typiske tilfælde er et lille kontor eller et hjem med en enkelt IP-adresse fra udbyderen, mens maskinerne bag routeren bruger private adresser fra RFC 1918-blokkene 10.0.0.0/8, 172.16.0.0/12 og 192.168.0.0/16, som ikke routes mellem organisationer. Routeren skriver kildeadressen i hver udgående pakke om til sin egen offentlige adresse og vender ændringen om på svarene. Basic NAT oversætter kun adresser, så der kan kun være lige så mange interne værter ude på én gang, som der er offentlige adresser i dens pulje. NAPT, network address port translation, omskriver også TCP- eller UDP-porten eller ICMP-identifikatoren, og det lader mange interne værter dele én offentlig adresse. RFC 3022 nævner, at Linux tilbød NAT under navnet IP masquerade, og ordet lever stadig i dets firewallværktøjer.

At ændre en adresse ødelægger alt, der er beregnet ud fra den, så routeren må også rette det. RFC 3022 kræver, at kontrolsummen i IP-headeren justeres, og TCP- og UDP-kontrolsummerne med, fordi de dækker en pseudo-header, der indeholder kilde- og destinationsadressen. For at sende hvert svar til den rigtige interne vært fører routeren en tabel over oversættelser: intern adresse og port på den ene side, ekstern adresse og port på den anden. Den tabel er tilstand, der holdes midt i netværket, og det meste af det, der går galt med NAT, kommer derfra.

Traditionel NAT virker i én retning. RFC 3022 beskriver dens sessioner som udgående fra det private netværk, mens trafik den anden vej kun tillades gennem statiske afbildninger for værter, der er valgt på forhånd. Idéen bruges nu også et niveau højere oppe: RFC 6598 fra april 2012 reserverede 100.64.0.0/10 til carrier-grade NAT, hvor blokken nummererer forbindelserne mellem en udbyders NAT og kundernes udstyr, som selv kan oversætte en gang til.

Brug begrebet i en beslutning

Brug »Hvad er NAT?« i en konkret beslutning. Vi kan knytte begrebet til det system, I bygger eller køber. Wi-Fi, firewalls og fjernadgang sat op, så medarbejdere kan arbejde hvor som helst, og kun de rette kommer ind.

Se vores ydelser Kontakt os →

Undersøg antagelserne bag ordet

Brug forklaringen og det tilknyttede arbejde til at formulere spørgsmål til jeres projekt. Vi kan undersøge antagelserne sammen, før et værktøj vælges. Har administreret 30 Ubuntu Linux VPS-instanser — 30 Linux-servere driftet, med en plan for gendannelse

Har administreret 30 Ubuntu Linux VPS-instanser Kontakt os →

Hvorfor det betyder noget

RFC 3022 er ærlig om byttehandlen. NAT kan installeres uden at ændre værter eller routere, men det fjerner med RFC'ens ord »ende-til-ende-betydningen af en IP-adresse« og opvejer det med mere tilstand i netværket. Næsten alle praktiske følger af NAT udspringer af den sætning.

For en server er kildeadressen på en forbindelse ofte ikke én maskine, men en router med en hel skare bag sig. En blokering eller en rate limit rettet mod én besøgende rammer alle, der deler adressen, og en misbrugsanmeldelse nævner routeren, ikke synderen. RFC 3022 siger det samme set indefra: at skjule private adresser, som den regner for en fordel for privatlivet, gør også fejlfinding sværere, fordi en vært, der angriber andre maskiner eller sender spam, får sin adresse skjult i NAT-routeren.

Tilstand betyder også udløbstider. En oversættelse lever kun, så længe routeren husker den. RFC 4787, en Best Current Practice fra januar 2007, siger, at en UDP-oversættelse ikke må udløbe på under to minutter, og anbefaler fem minutter eller mere. RFC 5382, modstykket for TCP fra oktober 2008, sætter 2 timer og 4 minutter for en etableret forbindelse, der står stille: keepalives sendes som standard hver anden time, og de fire minutter giver plads til pakker, der stadig er undervejs. Det er bundgrænser for enheder, der følger dokumenterne, og ikke

alle enheder gør det. En forbindelse, der er tavs længere end enhedens egen udløbstid, mister sin oversættelse, og den næste pakke udefra har ingen steder at gå hen.

To yderligere begrænsninger følger. RFC 3022 gør det obligatorisk, at alle pakker i en session passerer den samme NAT-router, og hvor et netværk har mere end én udgang, kalder den det meget vigtigt, at hver NAT har den samme oversættelsestabel. Og protokoller, der bærer adresser inde i deres indhold, har brug for hjælp fra NAT-enheden, og den hjælp virker kun, så længe enheden kan læse indholdet.

Sådan ser det ud i praksis

På en Linux-router udfører nftables NAT i kæder af typen `nat`. Regelsættet herunder er denne artikels illustration, bygget på NAT-eksemplerne i manualen nft(8). Her står 198.51.100.0/24 for det interne netværk og 203.0.113.5 for routerens offentlige adresse; begge kommer fra dokumentationsområderne, og et rigtigt internt netværk ville normalt bruge en af RFC 1918-blokkene.

```
add table nat
add chain nat prerouting { type nat hook prerouting priority dstnat; }
add chain nat postrouting { type nat hook postrouting priority srcnat; }
add rule nat postrouting oif eth0 ip saddr 198.51.100.0/24 snat to 203.0.113.5
add rule nat prerouting iif eth0 tcp dport 443 dnat to 198.51.100.20
```

Den første regel er hverdagstilfældet: pakker fra det interne netværk, der forlader routeren gennem eth0, får 203.0.113.5 som kilde. Til en dynamisk offentlig adresse tilbyder manualen `masquerade`, som bruger den adresse, den udgående grænseflade har i øjeblikket. Den anden regel er en statisk afbildning den anden vej: TCP-forbindelser, der ankommer på eth0 til port 443, sendes videre til 198.51.100.20 indenfor. Så få regler er nok, fordi kun den første pakke i en forbindelse passerer en NAT-kæde; dens beslutning gemmes i forbindelsessporingens post, og hver senere pakke omskrives på samme måde.

Den anden side af billedet er en vært bag en NAT, den ikke selv styrer, og WireGuard har en indstilling til netop det. I denne artikels anden illustration holder en intern vært en tunnel åben til en modpart udenfor:

```
[Peer]
PublicKey = <den eksterne modparts offentlige nøgle>
Endpoint = 192.0.2.44:51820
AllowedIPs = 192.0.2.128/25
PersistentKeepalive = 25
```

Manualen wg(8) beskriver `PersistentKeepalive` som et interval på 1 til 65.535 sekunder for afsendelse af en autentificeret tom pakke, så en tilstandsfuld firewall eller en NAT-oversættelse forbliver gyldig. Den foreslår 25 sekunder for en grænseflade bag NAT, der sjældent selv sender noget og kan modtage trafik når som helst, og lader indstillingen være slået fra som standard. Modparten udenfor behøver intet særligt: dens `Endpoint` for denne vært opdateres til den seneste kildeadresse og port fra korrekt autentificerede pakker, og det er netop dem, NAT'en tildelte.

Tunneler som disse er dér, hvor problemer med NAT plejer at vise sig. Porteføljen beskriver, hvordan otte WireGuard- og IPsec-forbindelser blev administreret og fejlsøgt mellem Google Cloud og Linux-systemer i årene 2022 til 2024. Siden siger, at tunnelerne var skrøbelige og besværlige at diagnosticere, når de faldt ud, og at problemerne med routing og nøgleudveksling blev løst ved roden i stedet for at blive dækket over med en genstart, hvorefter de tilbagevendende forbindelsesproblemer hørte op. Den siger ikke, om NAT spillede en rolle, så de to eksempler ovenfor hører til denne artikel og ikke til det projekt.

Hvor det går galt

Stille forbindelser, der holder op med at svare. En tunnel, en databaseforbindelse eller en push-kanal, der ikke sender noget i et stykke tid, mister sin oversættelse og går i stykker, uden at nogen af enderne har lukket den. Symptomet ser tilfældigt ud, fordi det afhænger af, hvor længe linjen var stille. Send keepalives indefra med kortere mellemrum end den korteste udløbstid på vejen, sådan som `PersistentKeepalive` gør, og hold udløbstiderne på en NAT, du selv driver, på eller over minimummene i RFC 4787 og RFC 5382. RFC 4787 kræver, at udgående pakker fornyer en oversættelse, og advarer om, at fornyelse fra indgående trafik lader en angriber holde den i live for evigt, så det er den interne ende, der bør stå for fornyelsen.

At forvente, at forbindelser kommer ind. Traditionel NAT er udgående af design, så en tjeneste på en intern vært kan ikke nås, før nogen skriver en statisk afbildning til den. Tilføj en `dnat` -regel for præcis de porte, tjenesten har brug

for, eller vend retningen om og lad den interne vært forbinde udad, sådan som keepalive-eksemplet ovenfor gør.

Manglende hairpinning. To interne værter, der når hinanden gennem den enes eksterne adresse, har brug for, at NAT'en vender trafikken om. RFC 4787 kræver, at NAT-enheder understøtter denne hairpinning og præsenterer den vendte pakke med den eksterne kildeadresse og port. Test den vej med vilje, og hvor en enhed ikke klarer den, så lad de interne værter bruge den interne adresse.

Adresser inde i indholdet. FTP er det klassiske tilfælde: RFC 2663 bemærker, at kommandoen `PORT` og svaret `PASV` bærer en IP-adresse og en port som tekst, som en NAT, der kun omskriver headere, efterlader forkert. NAT-enheder lapper sådanne protokoller med en Application Level Gateway (ALG), og RFC 3022 kalder den til FTP den mest populære. En ALG skal kunne læse indholdet, så kryptering sætter den ud af spil, medmindre gatewayen har nøglen, og RFC 3022 tilføjer, at man ikke kan regne med ende-til-ende-IPsec gennem en NAT. Foretræk protokoller, der holder adresser ude af indholdet; hvor en gammel protokol skal krydse en NAT, du selv driver, viser manualen nft(8) en hjælper til forbindelsessporing af FTP.

Logs, der kun nævner routeren. Bag NAT registrerer en servers adgangslog den delte adresse, som alene ikke kan udpege én intern vært. Log kildeporten og et præcist tidsstempel ved siden af adressen, og hold urene synkroniseret, så en klage over én forbindelse kan holdes op mod oversættelsen på routeren.

Fragmenter under NAPT. Kun det første fragment af en pakke bærer TCP- eller UDP-headeren, så RFC 3022 advarer om, at oversættelse af udgående fragmenter med NAPT er dømt til at mislykkes, og at to interne værter, der sender fragmenter med samme identifikator til samme destination, kan ødelægge begge sessioner. Hold pakker små nok til at krydse hele vejen uden at blive delt.

At behandle NAT som en firewall. At NAT kun virker i én retning, er let at forveksle med beskyttelse. RFC 4787 erkender, at NAT'er ofte sættes op for at opnå sikkerhedsmål, men siger klart, at den måde, en NAT oversætter på, ikke gør nogen forskel for dens sikkerhed, som helt afgøres af dens filtrering: hvilke pakker den lukker ind. Skriv eksplicitte filterregler for, hvad der må komme ind, og par hver `dnat` -regel med en regel, der kun tillader den trafik, den var tiltænkt.

Læs mere

- RFC 3022: Traditional IP Network Address Translator (Traditional NAT) — beskrivelsen fra 2001 af Basic NAT og NAPT med en ærlig liste over, hvad de koster.
- RFC 2663: IP Network Address Translator (NAT) Terminology and Considerations — ordforrådet for NAT, fra adresseområder til gennemsigtig routing og ALG'er.
- RFC 4787: Network Address Translation (NAT) Behavioral Requirements for Unicast UDP — hvordan en NAT bør oversætte, lade udløbe og vende UDP-trafik.
- RFC 5382: NAT Behavioral Requirements for TCP — modstykket for TCP, med udløbstiderne for etablerede forbindelser og forbindelser, der er ved at lukke.
- wg(8): set and retrieve configuration of WireGuard interfaces — WireGuard-manualen med `PersistentKeepalive` og rådet til værter bag NAT.
- nft(8): nftables manual — opslagsværket for nftables, med NAT-kæder og sætningerne `snat`, `dnat` og `masquerade`.

Ordliste

Vælg ud fra jeres faktiske vilkår

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Gør begreberne praktiske

Spørg, hvordan »Hvad er NAT?« gælder i jeres organisation. Et konkret eksempel fra jeres nuværende system er nok til at begynde samtalen.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/glossary/nat/