Engineer

NAT

🔀 Réécriture des adresses IP des paquets traversant un routeur

Ce que c’est

Le NAT, abréviation de network address translation, ou traduction d’adresses réseau, réécrit les adresses IP des paquets traversant un routeur pour permettre aux hôtes d’un espace d’adressage de communiquer avec ceux d’un autre. La RFC 2663, qui a fixé le vocabulaire en 1999, le décrit comme une correspondance entre adresses IP de domaines d’adressage différents, visant un routage transparent : les hôtes des deux côtés ne devraient pas avoir à savoir qu’elle a lieu.

La forme rencontrée par presque tout le monde est ce que la RFC 3022, publiée en janvier 2001, nomme le NAT traditionnel. Le cas typique est un petit bureau ou un domicile avec une seule adresse IP du fournisseur, tandis que les machines derrière le routeur utilisent les plages privées de la RFC 1918, 10.0.0.0/8, 172.16.0.0/12 et 192.168.0.0/16, non routées entre organisations. Le routeur remplace l’adresse source de chaque paquet sortant par sa propre adresse publique et inverse ce changement pour les réponses. Le NAT de base ne traduit que les adresses : le nombre d’hôtes internes pouvant sortir simultanément est limité au nombre d’adresses publiques du pool. Le NAPT, traduction d’adresses et de ports réseau, réécrit aussi le port TCP ou UDP ou l’identifiant ICMP : de nombreux hôtes internes partagent alors une seule adresse publique. La RFC 3022 note que Linux proposait le NAT sous le nom IP masquerade ; ses outils de pare-feu emploient encore ce mot.

Changer une adresse invalide tout ce qui est calculé à partir d’elle : le routeur doit aussi le réparer. La RFC 3022 exige l’ajustement de la somme de contrôle de l’en-tête IP et des sommes TCP et UDP, car elles couvrent un pseudo-en-tête contenant adresses source et destination. Pour renvoyer chaque réponse au bon hôte interne, le routeur garde une table de correspondances : adresse et port internes d’un côté, adresse et port externes de l’autre. Cette table constitue un état au milieu du réseau, à l’origine de la plupart des problèmes du NAT.

Le NAT traditionnel fonctionne dans un sens. La RFC 3022 décrit des sessions sortant du réseau privé, le trafic inverse n'étant permis que par des correspondances d'adresses statiques pour des hôtes choisis à l'avance. L'idée s'applique aussi un niveau plus haut : la RFC 6598 d'avril 2012 réserve 100.64.0.0/10 au NAT à grande échelle des opérateurs, pour numéroter les liaisons entre le NAT du fournisseur et les équipements clients, qui peuvent traduire à nouveau.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce que le NAT ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Du Wi-Fi, des pare-feu et des accès à distance configurés pour travailler de partout, en laissant entrer uniquement les personnes autorisées.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. 30 instances VPS Ubuntu Linux administrées — 30 serveurs Linux gérés, avec récupération prévue

30 instances VPS Ubuntu Linux administrées Contactez-nous →

Pourquoi c'est important

La RFC 3022 expose franchement le compromis. Le NAT peut être installé sans modifier les hôtes ou routeurs, mais retire, selon ses termes, « la signification de bout en bout d'une adresse IP » et la compense par davantage d'état dans le réseau. Presque toutes ses conséquences pratiques découlent de cette phrase.

Pour un serveur, l'adresse source d'une connexion est souvent celle d'un routeur derrière lequel se trouve une foule, pas celle d'une seule machine. Un blocage ou une limite de débit visant un visiteur touche tous ceux qui partagent l'adresse, et un signalement d'abus désigne le routeur plutôt que le responsable. La RFC 3022 fait le même constat de l'intérieur : masquer les adresses privées, avantage pour la confidentialité selon elle, complique aussi le débogage, car l'hôte qui attaque d'autres machines ou envoie du spam voit son adresse cachée dans le routeur NAT.

L'état implique aussi des délais. Une correspondance ne vit que tant que le routeur la conserve. La RFC 4787, bonne pratique courante de janvier 2007, exige qu'une correspondance UDP reste valide au moins deux minutes et recommande cinq minutes ou plus. La RFC 5382, son pendant TCP d'octobre 2008, fixe 2 heures et 4 minutes pour une connexion établie inactive : les messages de maintien partent toutes les deux heures par défaut et les quatre minutes couvrent les paquets encore en transit. Ce sont des minima pour les

équipements respectant ces textes, ce que tous ne font pas. Une connexion silencieuse plus longtemps que le délai propre à l'équipement perd sa correspondance : le prochain paquet extérieur n'a plus de destination.

Deux autres contraintes suivent. La RFC 3022 impose que tous les paquets d'une session traversent le même routeur NAT et, si le réseau a plusieurs sorties, juge essentiel que les équipements NAT partagent une table de traduction identique. Les protocoles transportant des adresses dans leur charge utile ont aussi besoin de l'aide du NAT : elle ne fonctionne que s'il peut lire cette charge.

À quoi cela ressemble en pratique

Sur un routeur Linux, nftables réalise le NAT dans des chaînes de type `nat`. Les règles suivantes illustrent cet article et s'appuient sur les exemples NAT du manuel nft(8). Ici, 198.51.100.0/24 représente le réseau interne et 203.0.113.5 l'adresse publique du routeur ; ces deux valeurs viennent des plages de documentation, tandis qu'un vrai réseau interne utiliserait normalement une plage de la RFC 1918.

```
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
```

La première règle est le cas quotidien : les paquets du réseau interne sortant par eth0 reçoivent 203.0.113.5 comme source. Pour une adresse publique dynamique, le manuel propose `masquerade`, qui utilise l'adresse actuelle de l'interface de sortie. La seconde règle est une correspondance statique inverse : les connexions TCP arrivant sur eth0 au port 443 vont vers 198.51.100.20 à l'intérieur. Un si petit nombre de règles suffit parce que seul le premier paquet d'une connexion traverse une chaîne nat ; la décision est conservée dans l'entrée de suivi de connexion et tous les paquets suivants sont réécrits de la même façon.

L'autre aspect est celui d'un hôte situé derrière un NAT qu'il ne contrôle pas ; WireGuard possède justement un réglage pour ce cas. Dans cette seconde illustration de l'article, un hôte interne maintient un tunnel ouvert vers un pair extérieur :

```
[Peer]
PublicKey = <the outside peer's public key>
Endpoint = 192.0.2.44:51820
AllowedIPs = 192.0.2.128/25
PersistentKeepalive = 25
```

Le manuel wg(8) décrit `PersistentKeepalive` comme un intervalle de 1 à 65 535 secondes pour envoyer un paquet vide authentifié afin de maintenir valide l'état d'un pare-feu ou une correspondance NAT. Il propose 25 secondes pour une interface derrière un NAT qui émet rarement du trafic d'elle-même mais peut en recevoir à tout moment, et désactive le réglage par défaut. Le pair extérieur n'a besoin de rien de particulier : son `Endpoint` pour cet hôte est actualisé avec l'adresse et le port source des paquets correctement authentifiés les plus récents, exactement ceux attribués par le NAT.

C'est souvent sur de tels tunnels que les difficultés de NAT apparaissent. Le portfolio décrit huit connexions WireGuard et IPsec gérées et dépannées de 2022 à 2024 entre Google Cloud et des systèmes Linux. La page indique que les tunnels étaient fragiles et leurs coupures difficiles à diagnostiquer ; les problèmes récurrents de routage et d'échange de clés ont été corrigés à la racine plutôt que par redémarrage, après quoi les interruptions ont cessé. Elle ne précise pas si le NAT jouait un rôle : les deux exemples ci-dessus appartiennent donc à cet article, pas à ce projet.

Où les problèmes surviennent

Des connexions silencieuses cessent de répondre. Un tunnel, une connexion de base de données ou un canal de notifications qui n'envoie rien un moment perd sa correspondance et se rompt sans qu'aucune extrémité l'ait fermé. Le symptôme paraît aléatoire, car il dépend de la durée d'inactivité. Envoyez des messages de maintien depuis l'intérieur à un intervalle inférieur au délai le plus court du trajet, comme `PersistentKeepalive` ; sur un NAT que vous exploitez, maintenez les délais à des valeurs au moins égales aux minima des RFC 4787 et 5382. La RFC 4787 exige que les paquets sortants rafraîchissent la correspondance et avertit qu'un rafraîchissement entrant permet à un attaquant de la maintenir indéfiniment : l'extrémité interne doit donc s'en charger.

On attend des connexions entrantes. Le NAT traditionnel est sortant par conception : un service interne est inaccessible tant qu'aucune correspondance statique n'a été écrite. Ajoutez une règle `dnat` pour les seuls ports nécessaires

au service, ou inversez le sens et laissez l'hôte interne se connecter vers l'extérieur, comme dans l'exemple de maintien ci-dessus.

Le rebouclage manque. Deux hôtes internes se joignant par l'adresse externe de l'un d'eux ont besoin que le NAT retourne le paquet. La RFC 4787 exige ce rebouclage, ou hairpinning, et la présentation du paquet retourné avec l'adresse et le port source externes. Testez délibérément ce trajet et, si l'équipement échoue, faites utiliser l'adresse interne aux hôtes internes.

Des adresses figurent dans la charge utile. FTP est le cas classique : la RFC 2663 note que sa commande `PORT` et sa réponse `PASV` contiennent adresse IP et port en texte, laissés incorrects par un NAT ne réécrivant que les en-têtes. Les NAT adaptent ces protocoles avec une passerelle de niveau applicatif, ou ALG ; la RFC 3022 présente celle de FTP comme la plus répandue. Une ALG doit lire la charge utile : le chiffrement la désactive, sauf si la passerelle détient la clé ; la RFC 3022 ajoute qu'on ne peut présumer qu'une communication IPsec de bout en bout fonctionnera à travers un NAT. Préférez les protocoles sans adresses dans leur charge ; si un ancien doit traverser un NAT que vous exploitez, le manuel nft(8) présente un auxiliaire de suivi de connexion FTP.

Les journaux ne nomment que le routeur. Derrière un NAT, le journal d'accès d'un serveur enregistre l'adresse partagée, qui seule n'identifie pas un hôte interne. Consignez le port source et un horodatage précis avec l'adresse, et synchronisez les horloges : un signalement concernant une connexion pourra alors être rapproché de la traduction effectuée par le routeur.

Des fragments passent par le NAPT. Seul le premier fragment d'un paquet porte l'en-tête TCP ou UDP : la RFC 3022 avertit que la traduction de fragments sortants par NAPT est vouée à l'échec et que deux hôtes internes envoyant des fragments de même identifiant à la même destination peuvent corrompre les deux sessions. Gardez des paquets assez petits pour traverser tout le trajet sans fragmentation.

Le NAT est traité comme un pare-feu. Son fonctionnement à sens unique se confond facilement avec une protection. La RFC 4787 reconnaît que les NAT sont souvent déployés pour la sécurité, mais précise que leur mode de correspondance ne change rien à celle-ci : elle dépend entièrement du filtrage, c'est-à-dire des paquets qu'ils laissent entrer. Écrivez des règles explicites sur le trafic entrant et associez à chaque règle `dnat` un filtre n'autorisant que le trafic prévu.

Pour aller plus loin

- RFC 3022 : traduction traditionnelle d'adresses IP — description de 2001 du NAT de base et du NAPT, avec une liste franche de leurs coûts.
- RFC 2663 : terminologie et considérations sur le NAT — vocabulaire du NAT, des domaines d'adressage au routage transparent et aux ALG.
- RFC 4787 : comportement requis du NAT pour UDP unicast — correspondances, délais et rebouclage du trafic UDP.
- RFC 5382 : comportement requis du NAT pour TCP — pendant TCP, dont les délais d'inactivité des connexions établies et en fermeture.
- wg(8) : définir et lire la configuration des interfaces WireGuard — manuel WireGuard, avec `PersistentKeepalive` et les conseils aux hôtes derrière un NAT.
- nft(8) : manuel nftables — référence nftables, dont les chaînes nat et les instructions `snat` , `dnat` et `masquerade` .

Glossaire

Choisissez selon vos contraintes réelles

Présentez le parcours de déploiement, les risques opérationnels et les personnes d'astreinte. Nous pouvons définir un changement avec des vérifications, une procédure de reprise et une responsabilité claire.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce que le NAT ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par **D.**

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

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/nat/