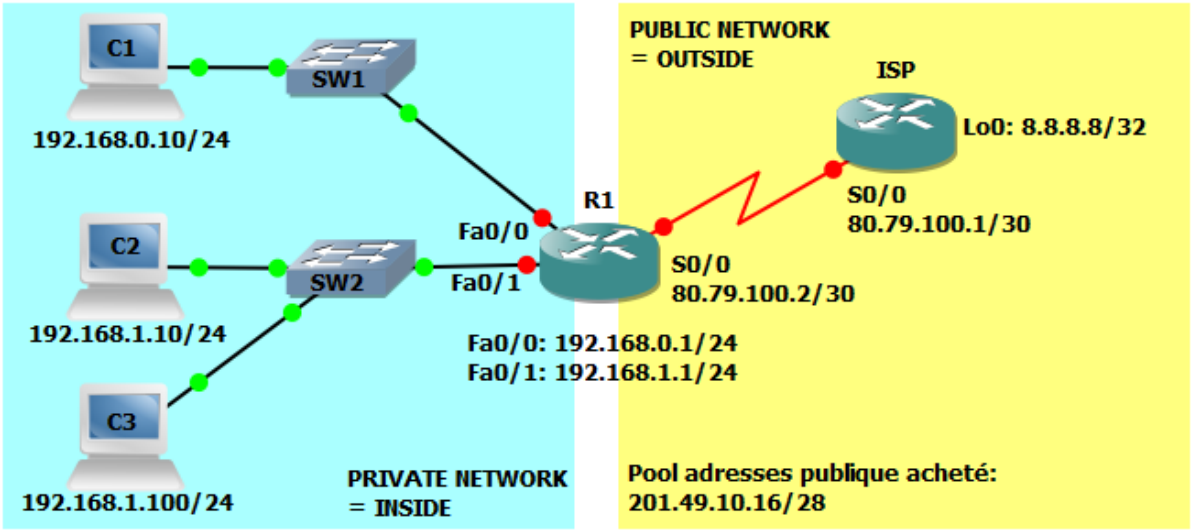
Configuration du NAT sur un routeur Cisco

Quand il s'agit d'interconnecter un réseau privé (que ce soit d'entreprise ou particulier), en IPv4, il est pratiquement impossible de se passer du NAT. Voici donc une configuration qui reprend l'essentiel des trois principaux types de NAT que l'on peut configurer, à savoir:

- Le NAT statique
- Le NAT dynamique avec pool d'adresses
- Le NAT dynamique avec surcharge (NAT overload, aussi connu sous le nom de PAT)

Topologie du labo



Le labo est divisé en deux parties, le côté privé (réseau de l'entreprise) et le côté publique (le FAI et Internet). le routeur ISP (qui représente le FAI), n'a aucune connaissance des réseaux privés de l'entreprise et ne peut donc rien router à destination des réseaux 192.168.x.x. (comme défini dans la RFC1918). Ces adresses sont réservées pour l'utilisation dans les réseaux privés. Il en va de même pour toutes les adresses faisant parties des plages suivantes:

- 10.0.0.0/8 (de 10.0.0.0 à 10.255.255.255)
- 172.16.0.0/12 (de 172.16.0.0 à 172.31.255.255)
- 192.168.0.0/16 (de 192.168.0.0 à 192.168.255.255)

Dés lors, nous allons configurer le NAT afin de permettre un accès à Internet (simulé par l'adresse 8.8.8.8/32 configurée sur une interface loopback de ISP):

- Le réseau 192.168.0.0/24 utilisera du NAT dynamique avec surcharge.
- Le réseau 192.168.1.0/24 utilisera du NAT avec pool d'adresse.
- La machine 192.168.1.100 sera accessible depuis le réseau publique grâce à une configuration de NAT statique.

Configuration de la topologie de base

Sur ISP:

Configuration de l'interface loopback

```
ISP#conf t
ISP(config)#int l0
ISP(config-if)#ip address 8.8.8.8 255.255.255.255
ISP(config-if)#exit
```

Configuration de la liaison sérielle vers R1

```
ISP(config)#int s0/0
ISP(config-if)#no shut
ISP(config-if)#ip address 80.79.100.1 255.255.255.252
ISP(config-if)#exit
```

Configuration de la route vers le pool d'adresses publique

```
ISP(config)#ip route 201.49.10.16 255.255.255.240 serial 0/0
```

Sur R1

Configuration de l'interface sérielle vers ISP

```
R1#conf t
R1(config)#int s0/0
R1(config-if)#ip address 80.79.100.2 255.255.255.252
R1(config-if)#no shut
R1(config-if)#exit
```

Configuration de l'interface du LAN1

```
R1(config)#int fa0/0
R1(config-if)#ip address 192.168.0.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#exit
```

Configuration de l'interface du LAN2

```
R1(config)#int fa0/1
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#exit
```

Configuration de la route par défaut

```
R1(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0
```

Pour le moment il est possible d'effectuer les tests suivants:

- Effectuer un ping depuis chaque PC vers R1

- Effectuer un ping entre les PCs de LAN différent
- Effectuer un ping depuis R1 vers 8.8.8.8

Test de C1 à R1

```
VPCS[1]> ping 192.168.0.1
192.168.0.1 icmp_seq=1 ttl=255 time=70.000 ms
192.168.0.1 icmp_seq=2 ttl=255 time=54.000 ms
192.168.0.1 icmp_seq=3 ttl=255 time=67.000 ms
192.168.0.1 icmp_seq=4 ttl=255 time=65.000 ms
192.168.0.1 icmp_seq=5 ttl=255 time=63.000 ms
```

Test de C1 à C2

```
VPCS[1]> ping 192.168.1.10
192.168.1.10 icmp_seq=1 ttl=63 time=31.000 ms
192.168.1.10 icmp_seq=2 ttl=63 time=16.000 ms
192.168.1.10 icmp_seq=3 ttl=63 time=32.000 ms
192.168.1.10 icmp_seq=4 ttl=63 time=32.000 ms
192.168.1.10 icmp_seq=5 ttl=63 time=32.000 ms
```

Test de R1 à 8.8.8.8

```
R1#ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/29/112 ms
```

Par contre impossible par exemple pour C1 de communiquer avec 8.8.8.8

```
VPCS[1]> ping 8.8.8.8
 8.8.8.8 icmp_seq=1 timeout
 8.8.8.8 icmp_seq=2 timeout
 8.8.8.8 icmp_seq=3 timeout
 8.8.8.8 icmp_seq=4 timeout
 8.8.8.8 icmp_seq=5 timeout
```

Configuration commune à tout type de NAT

La première chose à faire lorsque l'on configure du NAT, quel qu'en soit le type, c'est d'indiquer au routeur où se situe le réseau privé et où se situe le réseau publique.

Le NAT ne prend effet que lorsque qu'un paquet est routé d'une interface « inside » (côté privé) vers une interface « outside » (côté publique) et vice-versa.

Dans notre cas, les interfaces Fa0/0 et Fa0/1 sont du côté privé et seront déclarées comme « inside », l'interface S0/0 par contre, étant du côté publique, sera configurée comme « outside ».

```
R1(config)#int fa0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#int fa0/1
```

```
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#int s0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

Configuration du NAT statique pour C3

Ce que nous allons configurer ici c'est une translation statique dans la table de translation NAT, ce qu'on appelle vulgairement sur du matériel domestique « ouvrir un port ». Nous allons explicitement indiquer au routeur que ce qui arrive sur son interface publique (S0/0) et dont l'adresse destination est 201.49.10.30 (une des adresse du pool publique) doit être redirigé vers 192.168.1.100.

Du point de vue du routeur cela revient à modifier l'adresse IP destionation dans l'en-tête IPv4 avant de router le paquet. Cela signifie aussi que si C3 envoi un paquet vers internet, à la sortie de S0/0 de R1 l'adresse source (192.168.1.100) sera remplacée par l'adresse indiquée dans la translation, soit 201.49.10.30.

```
R1(config)#ip nat inside source static 192.168.1.100 201.49.10.30
```

La table de translations NAT doit mainentant ressembler à ceci:

```
R1#show ip nat translations
Pro Inside global Inside local Outside local Outside global
--- 201.49.10.30  192.168.1.100 ---          ---
R1#
```

A présent C3 doit pouvoir communiquer avec le réseau publique

```
VPCS[3]> ping 8.8.8.8
8.8.8.8 icmp_seq=1 ttl=254 time=119.000 ms
8.8.8.8 icmp_seq=2 ttl=254 time=102.000 ms
8.8.8.8 icmp_seq=3 ttl=254 time=75.000 ms
8.8.8.8 icmp_seq=4 ttl=254 time=117.000 ms
8.8.8.8 icmp_seq=5 ttl=254 time=116.000 ms
```

Chaque paquet a donc été translaté, preuve en est la table de translations juste après l'émission de ces pings:

```
R1#show ip nat translations
Pro Inside global Inside local Outside local Outside global
icmp 201.49.10.30:33598 192.168.1.100:33598 8.8.8.8:33598 8.8.8.8:33598
icmp 201.49.10.30:33854 192.168.1.100:33854 8.8.8.8:33854 8.8.8.8:33854
icmp 201.49.10.30:34366 192.168.1.100:34366 8.8.8.8:34366 8.8.8.8:34366
icmp 201.49.10.30:34622 192.168.1.100:34622 8.8.8.8:34622 8.8.8.8:34622
icmp 201.49.10.30:34878 192.168.1.100:34878 8.8.8.8:34878 8.8.8.8:34878
--- 201.49.10.30 192.168.1.100 --- ---
R1#
```

On peut observer le résultat de la translation du côté de ISP aussi à l'aide de la commande « debug ip packet » qui va afficher le détail de chaque paquet IP traité par le routeur (Attention, dans un environnement réel cette commande peut gravement saturer le routeur).

```
ISP#debug ip packet

*Mar 1 02:22:49.491: IP: tableid=0, s=201.49.10.30 (Serial0/0), d=8.8.8.8
(Loopback0), routed via RIB

*Mar 1 02:22:49.491: IP: s=201.49.10.30 (Serial0/0), d=8.8.8.8, len 92,
rcvd 4

*Mar 1 02:22:49.495: IP: tableid=0, s=8.8.8.8 (local), d=201.49.10.30
(Serial0/0), routed via FIB

*Mar 1 02:22:49.495: IP: s=8.8.8.8 (local), d=201.49.10.30 (Serial0/0), len
92, sending
```

Configuration du NAT avec pool d'adresses

Pour l'instant seul C3 a accès au réseau publique, nous allons maintenant configurer un autre type de NAT pour le réseau 192.168.1.0/24 (à l'exception de C3).

Ici, au lieu de configurer une translation statique, nous allons donner au routeur une plage d'adresses publiques (un pool d'adresse) dans laquelle il peut piocher pour créer dynamiquement les translations.

Tout d'abord créons le pool d'adresses

```
R1(config)#ip nat pool POOL-NAT-LAN2 201.49.10.17 201.49.10.30 netmask
255.255.255.240
```

Ici on crée donc une plage d'adresse nommée POOL-NAT-LAN2 allant de 201.49.10.17 à 201.49.10.30.

Il nous faut ensuite définir quelles adresses IP sources seront susceptibles d'êtres translatées … pour cela il faut créer une ACL.

```
R1(config)#access-list 1 deny 192.168.1.100
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
```

On autorise donc à être translatées les adresses ip du réseau 192.168.1.0/24 sauf 192.168.1.100 (pour laquelle on a déjà une translation statique).

Il ne reste plus qu'à configurer le NAT en lui même

```
R1(config)#ip nat inside source list 1 pool POOL-NAT-LAN2
```

On instruit donc ici le routeur de créer dynamiquement une translation pour les paquets arrivant sur une interface « inside » routés par une interface « outside » dont l'adresse IP source correspond à l'ACL 1 et de remplacer l'IP source par une de celles comprises dans le pool POOL-NAT-LAN2.

Attention, si il y a plus de machine dans le réseau privé que d'adresses publiques disponibles, il faut alors rajouter le mot clé « overload » à la commande:

```
R1(config)#ip nat inside source list 1 pool POOL-NAT-LAN2 overload
```

Ceci permet de « partager » les adresses publiques en translatant également les numéros de ports dans l'entête de la couche transport (méthode communément appelée PAT).

A présent C2 (et les autres machines qui seraient dans le réseau 192.168.1.0/24) peuvent communiquer avec l'extérieur.

```
VPCS[2]> ping 8.8.8.8
8.8.8.8 icmp_seq=1 ttl=254 time=111.000 ms
8.8.8.8 icmp_seq=2 ttl=254 time=97.000 ms
8.8.8.8 icmp_seq=3 ttl=254 time=143.000 ms
8.8.8.8 icmp_seq=4 ttl=254 time=131.000 ms
8.8.8.8 icmp_seq=5 ttl=254 time=99.000 ms
```

La table de translation de R1 a maintenant une nouvelle entrée crée dynamiquement, mais qui réserve l'adresse publique pour C2 (tant que l'on ne purge pas la table NAT).

```
R1#sh ip nat trans
Pro Inside global Inside local Outside local Outside global
--- 201.49.10.17 192.168.1.10 --- ---
--- 201.49.10.30 192.168.1.100 --- ---
R1#
```

Configuration du NAT dynamique avec surcharge (sans pool)

Il reste encore à configurer R2 pour que le réseau 192.168.0.0/24 puisse accéder à l'extérieur. Pour cela nous allons configurer le troisième type de NAT, à savoir du NAT dynamique avec surcharge (overload) en utilisant l'adresse publique configurée sur l'interface S0/0 de R1.

Notez que c'est la configuration la plus courante dans un réseau modeste (par exemple dans un réseau domestique). Cette méthode ne requiert pas d'obtenir de nouvelles adresses publiques auprès du provider.

Nous devons cette fois aussi identifier les adresses sources à faire passer par le NAT, donc nous créons une nouvelle ACL.

```
R1(config)#access-list 2 permit 192.168.0.0 0.0.0.255
```

Il ne reste plus qu'à configurer le NAT.

```
R1(config)#ip nat inside source list 2 interface serial 0/0 overload
```

Nous disons ici au routeur de translater les paquets provenant des adresses décrites dans l'ACL 2 (192.168.0.0/24) et de remplacer l'adresse IP source par celle configurée sur l'interface Serial 0/0 en la surchargeant pour permettre à plus d'une machine de communiquer avec l'extérieur (PAT).

C1 (ainsi que toute machine de ce réseau) peut communiquer avec l'extérieur désormais

```
VPCS[1]> ping 8.8.8.8
8.8.8.8 icmp_seq=1 ttl=254 time=132.000 ms
8.8.8.8 icmp_seq=2 ttl=254 time=130.000 ms
```

```
8.8.8.8 icmp_seq=3 ttl=254 time=127.000 ms
8.8.8.8 icmp_seq=4 ttl=254 time=112.000 ms
8.8.8.8 icmp_seq=5 ttl=254 time=125.000 ms
```

Le « debug ip packets » sur ISP donne le résultat suivant

```
ISP#
*Mar 1 03:11:46.195: IP: tableid=0, s=80.79.100.2 (Serial0/0), d=8.8.8.8
(Loopback0), routed via RIB
*Mar 1 03:11:46.195: IP: s=80.79.100.2 (Serial0/0), d=8.8.8.8, len 92, rcvd
4
*Mar 1 03:11:46.199: IP: tableid=0, s=8.8.8.8 (local), d=80.79.100.2
(Serial0/0), routed via FIB
*Mar 1 03:11:46.199: IP: s=8.8.8.8 (local), d=80.79.100.2 (Serial0/0), len
92, sending
```

On voit là que c'est bien l'adresse de S0/0 qui est utilise pour remplacer l'IP source du paquet.