



## **Configuration du PAT sur routeur Cisco**

**>>> PAT appelé aussi NAT Overload**

### **Description :**

**Cet article a pour but de vous apprendre à configurer le PAT sur un routeur Cisco. La fonctionnalité PAT permet de traduire X adresses IP en une seule adresse IP.**

# Configuration du PAT sur routeur Cisco

## >>> PAT appelé aussi NAT Overload

### Sommaire :

#### I) Introduction

- 1) Qu'est-ce que le NAT ?
- 2) Pourquoi as-tu créé le NAT ?
- 3) Schéma réseau

#### II) Configuration de Base

- 1) Configuration du routeur R1
- 2) Configuration du routeur R0-1
- 3) Configuration du routeur R0-2
- 4) Configuration du routeur R2

#### III) NAT Overload

- 1) Configuration du NAT
  - 2) Vérifications
- #### IV) Téléchargement

---

## I) Introduction

### 1) Qu'est-ce que le NAT ?

Commençons par la signification du NAT, Network Address Translation (en Français Translation d'Adresse Réseau).

On peut définir le NAT en trois types :

- Le NAT Statique
- Le Nat Dynamique
- Le NAT Overload appelé PAT

#### a) Le NAT Statique

Le NAT Statique permet de traduire/transformer une adresse IP par une autre adresse IP.

Exemple :

Sur mon réseau local, je possède un serveur web que je souhaite rendre accessible depuis internet, mon serveur est configuré avec l'adresse IP 192.168.1.1.

Sur ma connexion internet, je possède plusieurs adresses IP public 80.22.33.45, 80.22.33.46, 80.22.33.47.

Le nom de mon site web hébergé sur mon serveur correspondra à l'adresse publique 80.22.33.47.

Je vais donc configurer une règle NAT Statique sur mon routeur pour traduire toutes les requêtes arrivant sur 80.22.33.47 vers 192.168.1.1 de mon réseau local.

Plus techniquement, lorsque mon routeur va recevoir une requête arrivant d'internet ayant pour adresse de destination 80.22.33.47, celui-ci va substituer l'adresse de destination par l'adresse IP 192.168.1.1.

Pour résumer : Une adresse IP locale = Une adresse IP public.

## b) Le NAT Dynamique

Le NAT Dynamique permet de traduire plusieurs adresses IP par un pool d'adresses.

Exemple :

Sur mon réseau local, je possède 10 PC devant naviguer sur internet.

Sur ma connexion internet, je possède un pool d'adresses IP public 80.22.33.45 à 80.22.33.55 (soit 10 adresses public)

Je vais donc configurer une règle NAT dynamique sur mon routeur, pour traduire automatiquement les requêtes qui partent de mes PC vers Internet.

Plus techniquement, lorsque mon routeur va recevoir une requête d'un PC pour naviguer sur internet, il va regarder la table de NAT et prendre la première adresse publique du pool disponible. Il va associer les deux adresses durant l'échange HTTP. Une fois l'échange http terminé, l'adresse sera libérée.

Pour résumer : X adresses IP locales = X adresses IP publiques

On pourrait configurer un réseau ayant par exemple 20 PC pour 10 adresses IP publiques. Mais si les 20 PC veulent naviguer sur internet en même temps, uniquement les 10 premiers pc pourront naviguer.

## c) Le NAT Overload ou PAT

Le NAT Overload permet de traduire plusieurs adresses IP par une adresse IP.

Exemple :

Sur mon réseau, je possède X PC devant naviguer sur Internet.

Sur ma connexion internet, je possède seulement une adresse IP Public 80.22.33.45.

Si j'utilise le NAT Statique, un seul de mes PC pourra naviguer sur Internet.

Si j'utilise le NAT Dynamique, mes X PC pourront naviguer sur Internet, mais seulement un seul à la fois.

Avec le NAT Overload, tous mes PC peuvent naviguer sur Internet en même temps.

Plus techniquement, lorsque mon routeur reçoit une requête, il crée une sorte de session en ajoutant un port sur la requête (port compris entre 0 et 65535).

- PC1 - 192.168.1.1 envoie la requête au routeur.

- Le routeur crée la session 192.168.1.1:1025 - 80.22.33.45:1025

- La requête est envoyée sur internet.

- Le serveur répond à l'adresse 80.22.33.45:1025

- Le routeur regarde dans sa table NAT et cherche la correspondance avec 80.22.33.45:1025

- Il trouve 192.168.1.1:1025

- Il envoie la réponse au PC1.

## 2) Pourquoi as-tu créé le NAT ?

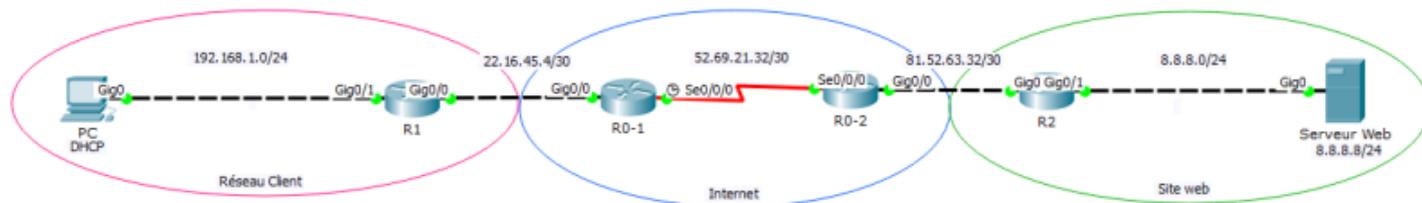
Avant la création du NAT, un PC ou un serveur voulant accéder à internet devait posséder une adresse IP publique.

Ceux qui ont développé Internet n'avaient pas imaginé à l'époque une telle évolution dans le monde. Ils ont donc créé Internet sur IPv4, en pensant que le nombre d'adresses proposées en IPv4 ne serait jamais utilisé à 100%.

Mais le développement d'Internet entre 1980 et 1990 dépassa toutes les attentes et les prévisions. Le nombre d'adresses IP publiques commençait à diminuer. Pour ralentir cette progression, les ingénieurs/chercheurs ont inventé le NAT, permettant ainsi de réduire les besoins en adresses IP publiques.

### 3) Schéma réseau

Voici le réseau que nous allons mettre en place :



## II) Configuration de base

Mise en place de la configuration de base sur les équipements.

Petite informations concernant l'adressage IP :

- Le PC est en DHCP, il récupère une adresse IP dans le réseau 192.168.1.0/24. La gateway 192.168.1.254 et le serveur DNS 8.8.8.8.
- Le serveur Web est statique, il a pour adresse IP 8.8.8.8/24.

### 1) Configuration du routeur R1

Mise en place de la configuration de base :

```
Router# configure terminal
Router(config)# no ip domain-lookup
Router(config)# hostname R1
R1(config)# line con 0
R1(config-line)# logging synchronous
R1(config-line)# exit
```

Configuration des interfaces du routeur :

```
R1(config)# interface gigabitEthernet 0/0
R1(config-if)# description "Interface connecte a Internet"
R1(config-if)# ip address 22.16.45.5 255.255.255.252
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitEthernet 0/1
R1(config-if)# description "Interface Reseau Client"
R1(config-if)# ip address 192.168.1.254 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
```

Configuration du routage :

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

Mise en place d'un serveur DHCP pour le réseau client :

```
R1(config)# ip dhcp pool LAN
R1(dhcp-config)# network 192.168.1.0 255.255.255.0
R1(dhcp-config)# dns-server 8.8.8.8
```

```
R1(dhcp-config)# default-router 192.168.1.254
R1(dhcp-config)# exit
```

Sauvegarde de la configuration :

```
R1(config)# end
R1# write memory
```

## 2) Configuration du routeur R0-1

Mise en place de la configuration de base :

```
Router# configure terminal
Router(config)# no ip domain-lookup
Router(config)# hostname R0-1
R0-1(config)# line con 0
R0-1(config-line)# logging synchronous
R0-1(config-line)# exit
```

Configuration des interfaces du routeur :

```
R0-1(config)# interface gigabitEthernet 0/0
R0-1(config-if)# description "Connecte au routeur R1"
R0-1(config-if)# ip address 22.16.45.6 255.255.255.252
R0-1(config-if)# no shutdown
R0-1(config-if)# exit
R0-1(config)# interface serial 0/0/0
R0-1(config-if)# description "Connecte au routeur R0-2"
R0-1(config-if)# ip address 52.69.21.33 255.255.255.252
R0-1(config-if)# clock rate 2000000
R0-1(config-if)# no shutdown
R0-1(config-if)# exit
```

Configuration du routage :

```
R0-1(config)# ip route 81.52.63.32 255.255.255.252 52.69.21.34
R0-1(config)# ip route 8.8.8.0 255.255.255.0 52.69.21.34
```

Sauvegarde de la configuration :

```
R0-1(config)# end
R0-1# write memory
```

## 3) Configuration du routeur R0-2

Mise en place de la configuration de base :

```
Router# configure terminal
Router(config)# no ip domain-lookup
Router(config)# hostname R0-2
R0-2(config)# line con 0
R0-2(config-line)# logging synchronous
```

```
R0-2(config-line)# exit
```

Configuration des interfaces du routeur :

```
R0-2(config)# interface gigabitEthernet 0/0
R0-2(config-if)# description "connecte au routeur R2"
R0-2(config-if)# 81.52.63.34 255.255.255.252
R0-2(config-if)# no shutdown
R0-2(config-if)# exit
R0-2(config)# interface serial 0/0/0
R0-2(config-if)# description "Connecte au routeur R0-1"
R0-2(config-if)# ip address 52.69.21.34 255.255.255.252
R0-2(config-if)# no shutdown
R0-2(config-if)# exit
```

Configuration du routage :

```
R0-2(config)# ip route 22.16.45.4 255.255.255.252 52.69.21.33
R0-2(config)# ip route 8.8.8.0 255.255.255.0 81.52.63.33
```

Sauvegarde de la configuration :

```
R0-2(config)# end
R0-2# write memory
```

## 4) Configuration du routeur R2

Mise en place de la configuration de base :

```
Router# configure terminal
Router(config)# no ip domain-lookup
Router(config)# hostname R2
R2(config)# line con 0
R2(config-line)# logging synchronous
R2(config-line)# exit
```

Configuration des interfaces du routeur :

```
R2(config)# interface gigabitEthernet 0/0
R2(config-if)# description "Interface connecte a Internet"
R2(config-if)# ip address 81.52.63.33 255.255.255.252
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# interface gigabitEthernet 0/1
R2(config-if)# description "Interface connecte au site web"
R2(config-if)# ip address 8.8.8.254 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
```

Configuration du routage :

```
R2(config)# ip route 0.0.0.0 0.0.0.0 81.52.63.34
```

Sauvegarde de la configuration :

```
R2(config)# end
R2# write memory
```

**Comme vous pouvez le voir, les routeurs R0-1, R0-2 et R2 ne connaissent pas le réseau 192.168.1.0**

Avant de configurer le NAT, faites un ping du serveur pour observer que le PC n'arrive pas à pinguer le serveur, alors que le routeur R1 ping le serveur.

- Ping depuis le PC :

```
Pinging 8.8.8.8 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Ping depuis le R1 :

```
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 = 3/6/11 ms
```

## III) NAT Overload

### 1) Configuration du NAT

Nous allons commencer par créer une ACL pour sélectionner les réseaux qui utiliseront le NAT.

```
R1(config)# ip access-list standard PAT
R1(config-std-nacl)# permit 192.168.1.0 0.0.0.255
R1(config-std-nacl)# exit
```

Deuxième étape, nous allons créer une relation entre access-list et l'interface WAN.

```
R1(config)# ip nat inside source list PAT interface gi 0/0 overload
```

- List 1 : Correspond à l'ACL 1

Troisième étape, il faut identifier l'interface LAN :

```
R1(config)# interface gigabitEthernet 0/1
R1(config-if)# ip nat inside
```

```
R1(config-if)# exit
```

Dernière étape, il faut identifier l'interface WAN :

```
R1(config)# interface gigabitEthernet 0/0  
R1(config-if)# ip nat outside  
R1(config-if)# exit
```

Pour finir, sauvegarde de la configuration :

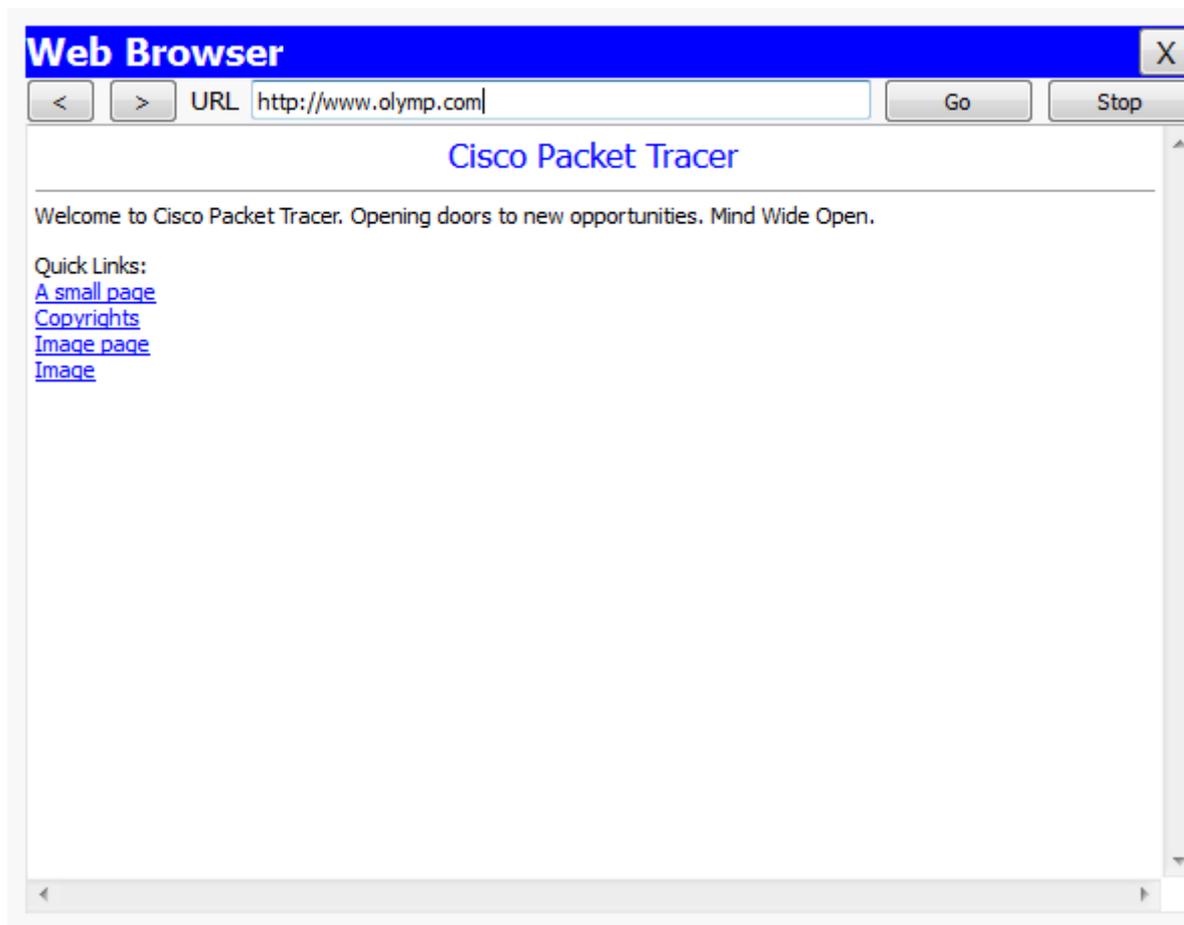
```
R1(config)# end  
R1# write memory
```

Pour finir, nous allons faire quelques tests :

- Faites un ping du serveur web "8.8.8.8" :

```
PC>ping 8.8.8.8  
  
Pinging 8.8.8.8 with 32 bytes of data:  
  
Reply from 8.8.8.8: bytes=32 time=68ms TTL=124  
Reply from 8.8.8.8: bytes=32 time=1ms TTL=124  
Reply from 8.8.8.8: bytes=32 time=1ms TTL=124  
Reply from 8.8.8.8: bytes=32 time=6ms TTL=124  
  
Ping statistics for 8.8.8.8:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
    Minimum = 1ms, Maximum = 68ms, Average = 19ms
```

- Ouvrez un navigateur pour accéder au site web "www.olymp.com" :



## 2) Vérifications

Pour observer les translations, tapez la commande :

```
R1# show ip nat translations
```

exemple :

```
R1#sh ip nat translations
Pro  Inside global      Inside local        Outside local       Outside global
icmp 22.16.45.5:1       192.168.1.1:1      8.8.8.8:1          8.8.8.8:1
icmp 22.16.45.5:2       192.168.1.1:2      8.8.8.8:2          8.8.8.8:2
icmp 22.16.45.5:3       192.168.1.1:3      8.8.8.8:3          8.8.8.8:3
icmp 22.16.45.5:4       192.168.1.1:4      8.8.8.8:4          8.8.8.8:4
udp  22.16.45.5:1025    192.168.1.1:1025   8.8.8.8:53         8.8.8.8:53
udp  22.16.45.5:1026    192.168.1.1:1026   8.8.8.8:53         8.8.8.8:53
udp  22.16.45.5:1027    192.168.1.1:1027   8.8.8.8:53         8.8.8.8:53
udp  22.16.45.5:1028    192.168.1.1:1028   8.8.8.8:53         8.8.8.8:53
udp  22.16.45.5:1029    192.168.1.1:1029   8.8.8.8:53         8.8.8.8:53
tcp  22.16.45.5:1025    192.168.1.1:1025   8.8.8.8:80         8.8.8.8:80
tcp  22.16.45.5:1026    192.168.1.1:1026   8.8.8.8:80         8.8.8.8:80
tcp  22.16.45.5:1027    192.168.1.1:1027   8.8.8.8:80         8.8.8.8:80
tcp  22.16.45.5:1028    192.168.1.1:1028   8.8.8.8:80         8.8.8.8:80
tcp  22.16.45.5:1029    192.168.1.1:1029   8.8.8.8:80         8.8.8.8:80
```

## IV) Téléchargement

Je vous propose le fichier Packet Tracert suivant. Il récapitule la configuration du PAT.



7 novembre 2014 -- N.Salmon -- article\_263.pdf



# Idum