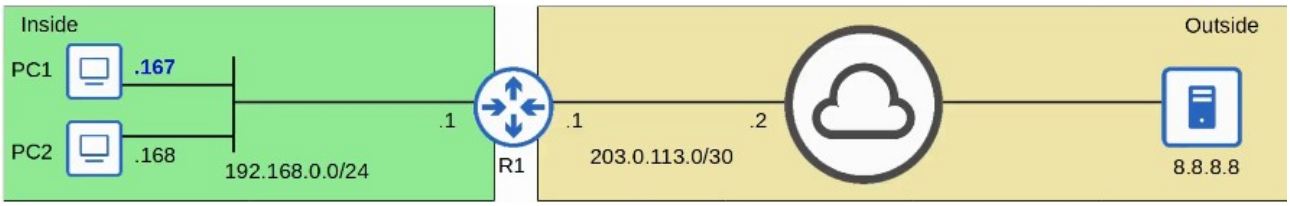
Cours 45 : Network Address Translation (Partie 2)

Dans ce cours nous continuerons la seconde partie du sujet NAT (Network Address Translation). Nous reverrons une nouvelle chose à propos du NAT statique, nous verrons ensuite le NAT dynamique qui permet de cartographier les adresses automatiquement au lieu de le faire manuellement pour chaque adresses. Puis un autre type de NAT qui est le dynamic PAT qui permet de traduire non pas seulement l'adresse IP mais aussi le numéro de port.

Nous verrons tout d'abord dans le réseau suivant le fonctionnement du NAT statique :



Le NAT statique permet de configurer statiquement en cartographiant chaque adresses IP privée une à une par des adresses IP publiques.
Lorsque le trafique depuis l'hôte interne est envoyé vers le réseau externe, le routeur va traduire l'adresse IP source.
Par exemple avec le NAT statique :
192.168.0.167 sera traduit en 100.0.0.1
192.168.0.168 sera traduit en 100.0.0.2

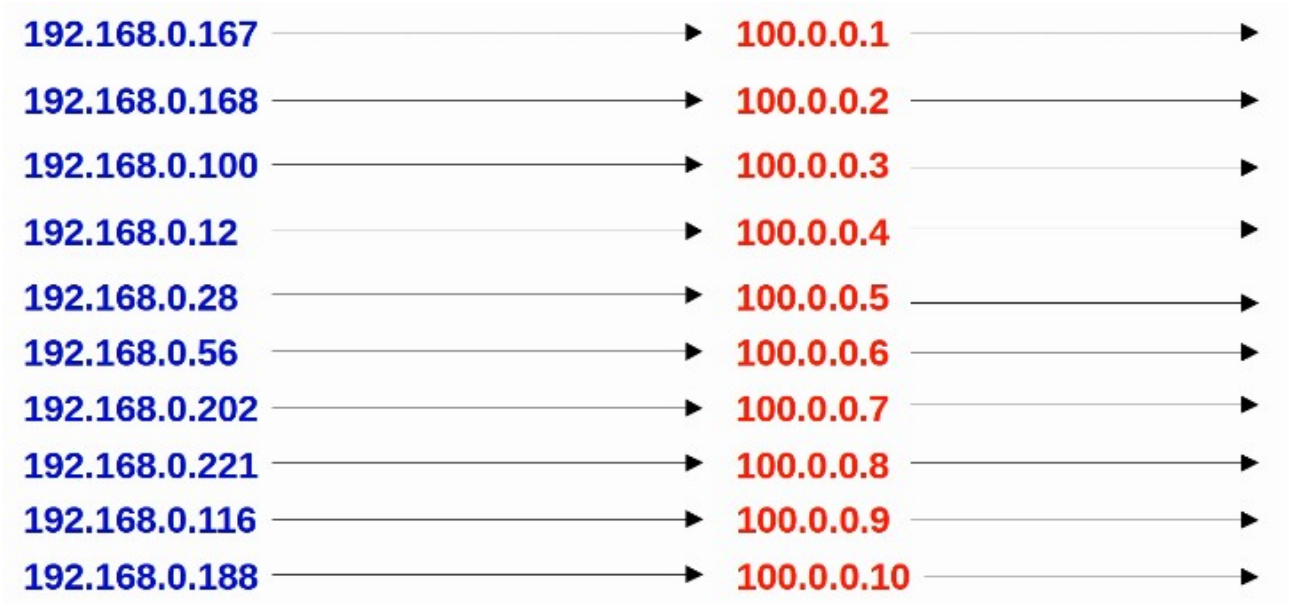
Cette cartographie permet à un hôte externe d'accéder aux hôtes internes par l'adresse IP interne global.

Dans le NAT dynamique, le routeur cartographie dynamiquement des adresses IP interne local vers des adresses interne globale selon le nombre d'adresses nécessaires.
Une ACL est utilisé pour identifier quelle trafique doit être traduit.
Si l'adresse IP source est permise par l'ACL, l'adresse IP source sera traduite.
Si l'adresse IP source est bloqué par l'ACL, l'adresse IP source ne sera pas traduite.
Un pool NAT est utilisé pour définir les adresses IP interne global.
Par exemple sur le réseau précédent, sur R1,
l'ACL 1 :
permit 192.168.0.0/24
deny any

POOL1 : 100.0.0.1 à 100.0.0.10
Si un paquet avec une adresse IP source permise par l'ACL 1 arrive, l'adresse IP traduira l'adresse IP source vers une adresse du POOL1.
Si une adresse est « denied » par l'ACL cela ne signifie pas que l'adresse IP sera bloqué mais simplement que l'adresse ne sera pas traduite.
Bien que les adresses soient assignés dynamiquement la cartographie est toujours une à une (une adresse ip local interne par adresse ip global interne)
S'il n'y as pas suffisamment d'adresses IP interne global disponible (par exemple que celles disponibles sont utilisés), cela sera appelé « NAT pool exhaustion ».
Si un paquet d'un autre hôte interne arrive et à besoin d'un ANT mais qu'il n'y a pas d'adresses disponibles, le routeur ne gardera pas le paquet.
L'hôte ne sera pas possible d'accès en dehors du réseau jusqu'à ce que l'une des adresses IP global internes ne deviennent disponibles.

Les entrées NAT dynamiques seront expirés automatiquement si non utilisé, il est aussi possible de les supprimer automatiquement.

Voyons comment fonctionne le NAT Pool Exhaustion, l'adresse IP source 192.168.0.167 est traduite en 100.0.0.1, l'adresse IP 192.168.0.168 est aussi traduite en 100.0.0.2, etc...
même chose pour toutes adresses suivantes :



Si une nouvelle adresse veut être traduite par exemple 192.168.0.98, puisque plus aucune adresse n'est disponible le routeur va bloquer le paquet. Pour que cette nouvelle adresse soit joignable, l'adresse IP 192.168.0.167 est supprimé car expiré après un certain temps, l'adresse 192.168.0.98 pourra joindre le trafique en utilisant l'adresse traduite de l'ancienne adresse : 100.0.0.1
Il reste tout de même possible pour les hôtes d'utiliser plusieurs fois la même adresse IP publique par le moyen de PAT (Port Address Translation)

Pour configurer le NAT dynamique on utilise les commandes suivantes :

```
R1(config)#int g0/1
R1(config-if)#ip nat inside

R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#exit

R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255


R1(config)#ip nat pool POOL1 100.0.0.0 100.0.0.255 prefix-length 24




R1(config)#ip nat inside source list 1 pool POOL1
```

On définit l'interface interne connecté au réseau interne avec les commandes :

```
R1(config)#int g0/1
R1(config)#ip nat inside
```

Pour définir l'interface externe connecté au réseau externe on lance les commandes

```
R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

Pour définir le trafique qui devrait être traduit on lance les commandes :

```
R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255
R1(config)#ip nat pool POOL1 100.0.0.0 100.0.0.255 prefic-length 24
R1(config)#ip nat inside source list 1 pool POOL1
```

Pour définir le pool des adresses IP interne global on utilise la commande :

```
R1(config)#ip nat pool POOL1 100.0.0.0 100.0.0.255 prefix-length 24
```

Pour configurer un NAT dynamique avec une cartographie de l'ACL au pool on utilise la commande :

```
R1(config)#ip nat inside source list 1 pool POOL1
```

Pour afficher la table NAT on lance la commande : `show ip nat translations`

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
icmp 100.0.0.1:3       192.168.0.167:3    8.8.8.8:3          8.8.8.8:3
udp 100.0.0.1:58685    192.168.0.167:58685 8.8.8.8:53        8.8.8.8:53
--- 100.0.0.1          192.168.0.167      ---                ---
icmp 100.0.0.2:3       192.168.0.168:3    8.8.8.8:3          8.8.8.8:3
udp 100.0.0.2:49536    192.168.0.168:49536 8.8.8.8:53        8.8.8.8:53
--- 100.0.0.2          192.168.0.168      ---                ---
```

Comme on peut le voir les adresses IP sont retirés lorsque l'on relance la commande après une minute :

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
--- 100.0.0.1          192.168.0.167      ---                ---
--- 100.0.0.2          192.168.0.168      ---                ---
```

Pour afficher les statistiques de NAT on lance la commande : `show ip nat statistics`

```
R1#show ip nat statistics
Total active translations: 6 (0 static, 6 dynamic; 4 extended)
Peak translations: 6, occurred 00:00:30 ago
Outside interfaces:
  GigabitEthernet0/0
Inside interfaces:
  GigabitEthernet0/1
Hits: 32  Misses: 0
CEF Translated packets: 20, CEF Punted packets: 12
Expired translations: 0
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 1 pool POOL1 refcount 6
 pool POOL1: netmask 255.255.255.0
        start 100.0.0.0 end 100.0.0.255
        type generic, total addresses 256, allocated 2 (0%), misses 0
[output omitted]
```

Avec cette commande on peut afficher l'ACL correspondante à la cartographie du Pool.

Voyons à présent le fonctionnement de PAT (NAT Overload).
PAT permet de traduire les adresses IP et le numéro de port d'un paquet si nécessaire.
En utilisant un port unique pour chaque flux de communication, une seule adresse IP publique peut être utilisé par plusieurs différentes adresses internes des hôtes. (Le numéro de port est 16bits ce qui fait un total de 65 000 numéro de port disponible).
Le routeur va suivre quelle adresse interne local est utilisé par quelle adresse interne global et le port.
Par exemple dans le schéma de réseau précédent le PC1 avec l'adresse IP source 192.168.0.167:54321 veut joindre le serveur 8.8.8.8:53, le PC2 avec l'adresse IP source 192.168.0.168:54321 veut lui aussi joindre le même serveur de l'adresse 8.8.8.8:53, pour joindre le serveur R1 va traduire les deux adresses en 100.0.0.1 mais avec des numéros de ports différents qui seront : 100.0.0.1:54321 et 100.0.0.1:54322
Pour répondre le serveur envoie lui aussi pour adresses de destination la même adresse mais avec des numéros de ports différents : 54321 et 54322.
Puisque plusieurs hôtes peuvent partager une seule et même adresse IP publique, PAT est très utile pour préserver les adresses IP publique et est utilisé dans tous les réseau dans le monde.

Voici les commandes qui permettent de configurer PAT :

On commence par définir l'interface interne connecté au réseau interne avec les commandes :
```
R1(config)#int g0/1
R1(config-if)#ip nat inside
```

Pour définir l'interface externe connecté au réseau externe on utilise les commandes :
```
R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

Pour définir le trafique qui doit être traduit on utilise la commande :
```
R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255
```

Pour définir le pool des adresses IP interne global on lance la commande :
```
R1(config)#ip nat pool POOL1 100.0.0.0 100.0.0.3 prefix-length 24
```

Pour configurer le PAT en cartographiant l'ACL au pool et en utilisant l'overload on lance la commande :
```
R1(config)#ip nat inside source list 1 pool POOL1 overload
```

Voyons la configuration utilisé par le routeur R1 :

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
udp 100.0.0.1:63925    192.168.0.167:63925 8.8.8.8:53         8.8.8.8:53
udp 100.0.0.1:59549    192.168.0.168:59549 8.8.8.8:53         8.8.8.8:53
```

Comme on peut le voir ci dessus l'adresse IP interne global est identique sur les deux ligne, c'est le numéro de port qui change.

On peut afficher plus de détail avec la commande : `show ip nat statistics`

```
R1#show ip nat statistics
Total active translations: 2 (0 static, 2 dynamic; 2 extended)
Peak translations: 2, occurred 00:00:03 ago
Outside interfaces:
  GigabitEthernet0/0
Inside interfaces:
  GigabitEthernet0/1
Hits: 4  Misses: 0
CEF Translated packets: 0, CEF Punted packets: 4
Expired translations: 0
Dynamic mappings:
-- Inside Source
[Id: 3] access-list 1 pool POOL1 refcount 2
 pool POOL1: netmask 255.255.255.0
        start 100.0.0.0 end 100.0.0.3
        type generic, total addresses 4, allocated 1 (25%), misses 0
```

Une autre manière de configurer PAT, il s'agit même peut être de la manière la plus commune,
Est de configurer le routeur afin qu'il utilise sa propre adresse IP lorsqu'il traduit l'adresse IP source d'autre paquet, pour cela on lance les commandes suivantes :

```
R1(config)#int g0/1
R1(config-if)#ip nat inside

R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#exit


R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255



R1(config)#ip nat inside source list 1 interface g0/0 overload
```

Pour de définir une interface interne connecté au réseau interne on utilise les commandes :

```
R1(config)#int g0/1
R1(config-if)#ip nat inside
```

Pour définir une interface externe connecté au réseau externe, on utilise les commandes :

```
R1(config-if)#int g0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

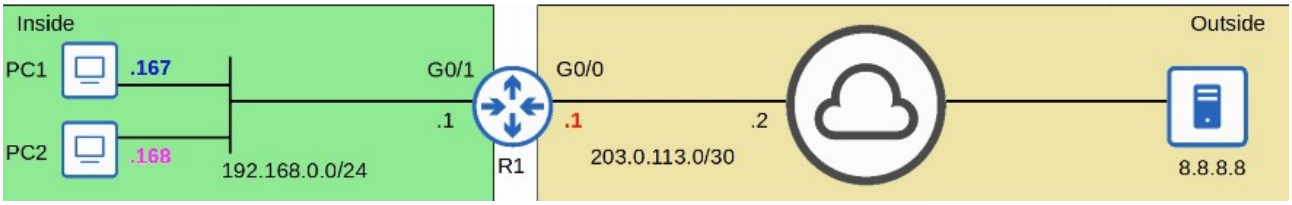
On définit ensuite le trafique qui devra être traduit avec la commande :

```
R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255
```

Pour configurer PAT en cartographiant l'ACL vers l'interface et activer overload on lance la commande :

```
R1(config)#ip nat inside source list 1 interface g0/0 overload
```

On peut démontrer cela avec le réseau suivant :



Lorsque les PC1 et PC2 envoient un paquet au serveur, l'adresse IP traduite par le routeur est la même que celle des PC mais le numéro de leurs ports est différent.

Voici l'affichage de la configuration avec la commande : `show ip nat translations`

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
udp 203.0.113.1:65205  192.168.0.167:65205 8.8.8.8:53         8.8.8.8:53
udp 203.0.113.1:59641  192.168.0.168:59641 8.8.8.8:53         8.8.8.8:53
```

On affiche plus de détails avec la commande : `show ip nat statistics`

```
R1#show ip nat statistics
Total active translations: 2 (0 static, 2 dynamic; 2 extended)
Peak translations: 2, occurred 00:36:30 ago
Outside interfaces:
  GigabitEthernet0/0
Inside interfaces:
  GigabitEthernet0/1
Hits: 12  Misses: 0
CEF Translated packets: 0, CEF Punted packets: 12
Expired translations: 4
Dynamic mappings:
-- Inside Source
[Id: 4] access-list 1 interface GigabitEthernet0/0 refcount 2
```