Projet réseau 2006-200

- routage sous Linux:
  - configuration réseau et routage sous linux
  - travail personnel: (1) mise en place d'un routeur linux.
- traduction d'adresse
  - sujet: translation d'adresse (NAPT)
  - travail personnel:
    - (2) : mise en évidence du besoin
    - (3): fonctionnement d'un routeur NATP
    - (4): NAPT et protocole FTP
- date limite de rendu de cette seconde livraison : vendredi 12 janvier 2007

Configuration IP sous Debian Gnu/Linux

- /etc/network/interfaces: configuration des interfaces réseau (cf « man interfaces » pour plus d'info). En particulier: configuration IP des interfaces réseau
- /etc/network/options: options réseau (routage principalement)
- /etc/resolv.conf: dns à utiliser

/etc/network/interfaces

- cf « man interfaces »
- un résumé partiel ne parlant que de config IP
  - ligne commençant pas auto: indique les interfaces ethernet devant être activées automatiquement. ex: « auto eth0 »
  - bloc commençant par iface: « iface nomIface famille méthode ». Exemple: « iface eth0 inet static »
    - familles usuelles: inet: ipv4, ipx: ipx, inet6: ipv6
    - méthodes usuelles pour la famille inet: loopback, static, dhcp, ppp, wvdial, ...
    - options des méthodes :
      - static: address, netmask, gateway, mtu, media type
      - dhcp: hostname, ... (depend du client dhcp utilisé)
-

routage sous linux

- activation du routage :
  - echo 1 > /proc/sys/net/ipv4/ip_forward (actif instantanément mais ne survit pas au reboot)
  - ou via le fichier /etc/network/options (debian Gnu/Linux) :
    - ip_forward=yes
  - ou via /etc/sysconfig/network (mandriva)
- désactivation du routage:
  - echo 0 > /proc/sys/net/ipv4/ip_forward (actif instantanément mais ne survit pas au reboot)
  - ou via le fichier /etc/network/options (debian Gnu/Linux) :
    - ip_forward=no

Plateforme 1

- 3 machines virtuelles Linux
  - debian-1: 1 interface réseau
    - adresse IP: 192.168.10.1,sous-réseau R1: 192.168.10/24
  - debian-2: 2 interfaces réseau
    - adresse IP1: 192.168.10.2,sous-réseau R1
    - adresse IP2: 192.168.20.2, sous-réseau R2: 192.168.20/24
  - debian-3: 1 interface réseau
    - adresse IP: 192.168.20.3,sous-réseau R2: 192.168.20/24
- R1: réseau virtuel vmware: vmnet 3
- R2: réseau virtuel vmware: vmnet 4

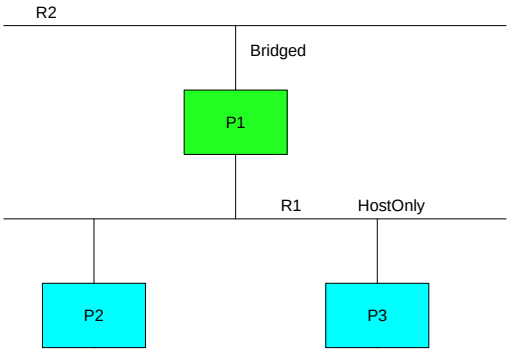
Votre travail (1)

- testez la connectivité IP entre vos trois machines à l'aide la commande ping :
  - vous ferez un tableau indiquant quelles liaisons sont opérationnelles et lesquelles ne le sont pas.
  - vous comparerez dans chaque cas les machines désignées par les adresses ip et celles désignées par les adresses MAC destination. Expliquez.
  - Après avoir expliqué pourquoi certaines liaisons sont opérationnelles et d'autres pas, vous ferez en sorte que toutes les liaisons soient opérationnelles.
- Vous pourrez illustrer votre propos à l'aide de capture ethereal

Bibliographie

- « GNU/Linux Debian » de . Hertzog, Editions Eyrolles
- www.debian.org
- formation Debian Gnu/Linux :
  http://people.via.ecp.fr/~alexis/formation-linux/formation-linux.html
-

maquette de test 1



Couleurs:
- vert: routage activé
- bleu: hôtes non routeur

R1: 192.168.10/24
R2: 192.168.195/24 (réseau de la salle)

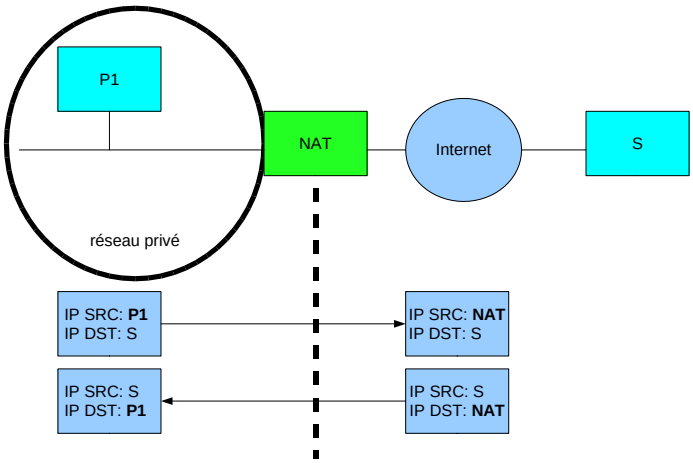
Votre travail (2)

- montez la maquette décrite ci-avant
- la machine P1 a une interface réseau en mode bridged sur R2 et une interface réseau en mode « host only » sur R1.
- les autres ordinateurs ont une seule interface réseau en mode « host only » sur R1.
- testez la connexion IP entre P2 et P3, P2 et P1, P1 et le routeur de la salle (192.168.195.2), P2 et le routeur de la salle.
- Expliquez les comportements constatés en vous appuyant sur des captures de trames

translation d'adresse

- motivations d'origine:
  - palier la pénurie d'adresses IP
  - permettre un accès à internet depuis des adresses privées (RFC 1918)
- Principe:
  - un routeur remplace les adresses IP sources ou destinations des paquets qu'il route de façon à ce que seules des adresses ip publiques apparaissent
  - les ports tcp/udp peuvent aussi être modifiés (selon le type de NAT)
  - la charge utile du paquet peut parfois être modifiée

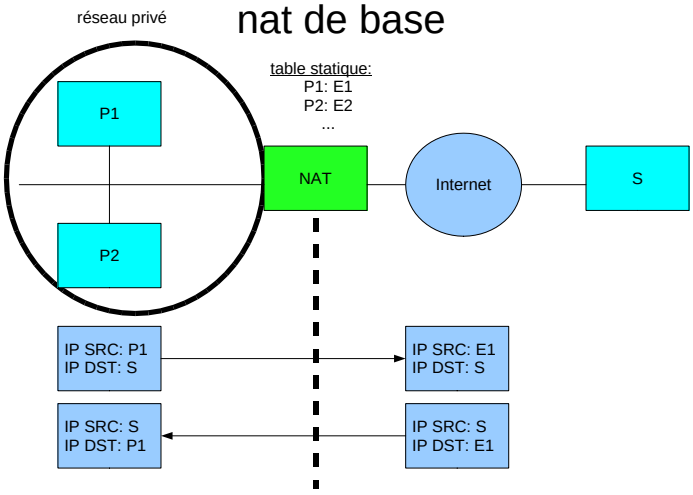
translation d'adresse



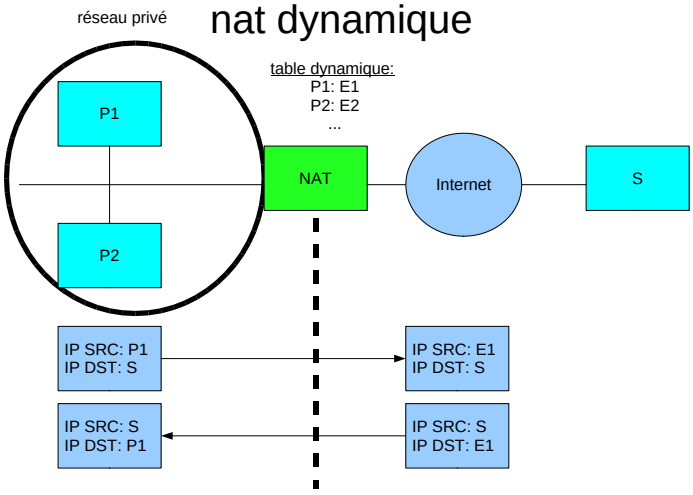
type de NAT:

- nat de base
- nat dynamique
- NAPT: translation d'adresses et de ports (NAPT MASQuerade)
- NAT bi-directionnel
- NAT double (twice NAT)
- NAPT avec redirection de port (port forwarding)

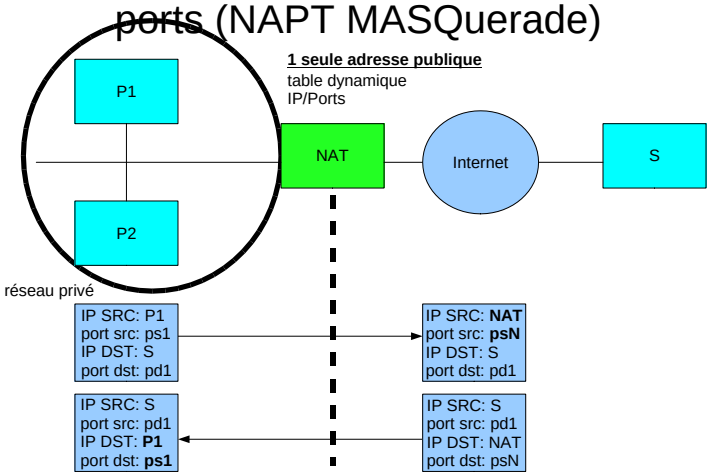
nat de base



nat dynamique



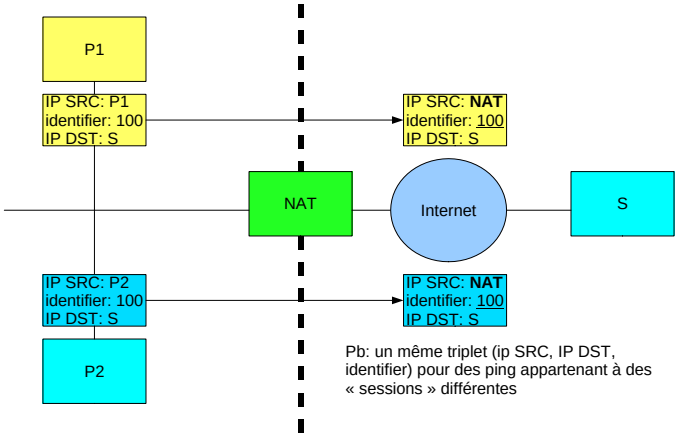
NAPT: translation d'adresses et de ports (NAPT MASQuerade)



identifier des « connexions »venant de la même source

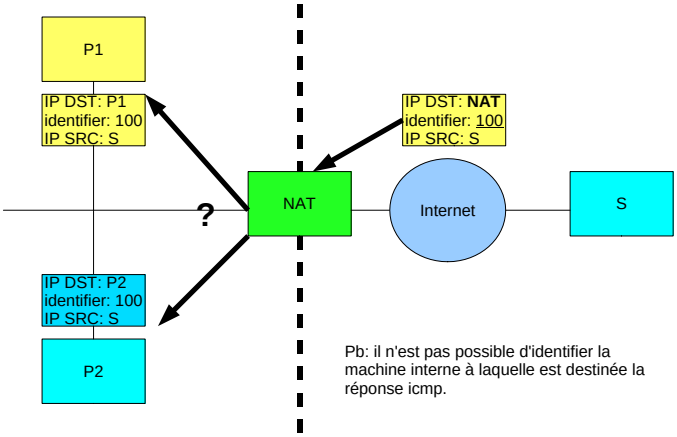
- problème classique sans NAT: gestion des « connexion » venant du même hôte
- Exemples:
  - TCP: 2 connexions ssh ayant même IP SRC et DST.
  - UDP: deux requêtes dns ayant même IP SRC et DST.
    - solution: le port source de chaque connexion est différent
  - deux ping (icmp echo) ayant même IP SRC et DST
    - solution: utilisation des champs identifiers et des numéros de séquence pour associer requêtes et réponses

NAPT et ping: problème



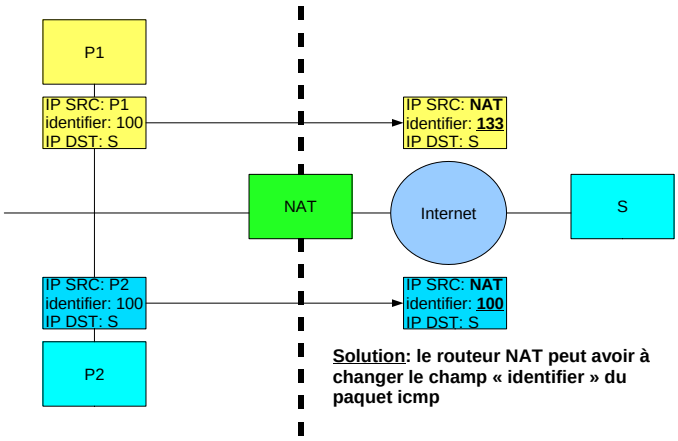
Pb: un même triplet (ip SRC, IP DST, identifier) pour des ping appartenant à des « sessions » différentes

NAPT et ping: problème

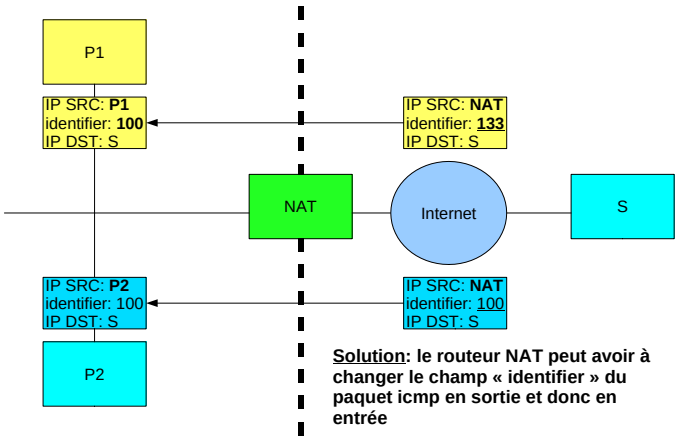


Pb: il n'est pas possible d'identifier la machine interne à laquelle est destinée la réponse icmp.

NAPT et ping: solution



NAPT et ping: solution



Exemple Ping

- Certaines unicités sont cassées par le remplacement des IP src par celle du routeur NAT
- pour les maintenir, il peut être nécessaire de modifier des identifiants de niveau transport :
  - identifiant ICMP
  - ports sources TCP ou UDP

NAPT: identifier les paquets entrant

- Vu de l'extérieur, tous les paquets semblent venir du routeur NAT
- On ne peut plus forcément garantir l'unicité des informations d'identification des paquets des connexions sortantes:
  - TCP/UDP: (IP SRC, port SRC, IP DST, PORT DST) si seule l'IP SRC est remplacé par celle du routeur
  - ICMP: (IP SRC, IP DST, « identifier », No de séquence)
- solution: le routeur NAT modifie aussi l'identifiant de transport source: port tcp/udp, identifiant icmp.

paquets/connexions/sessions

- paquets
- connexions
- sessions
- traitement à état (« statefull »)
- passerelles de niveau application (ALG: Application Layer Gateway)

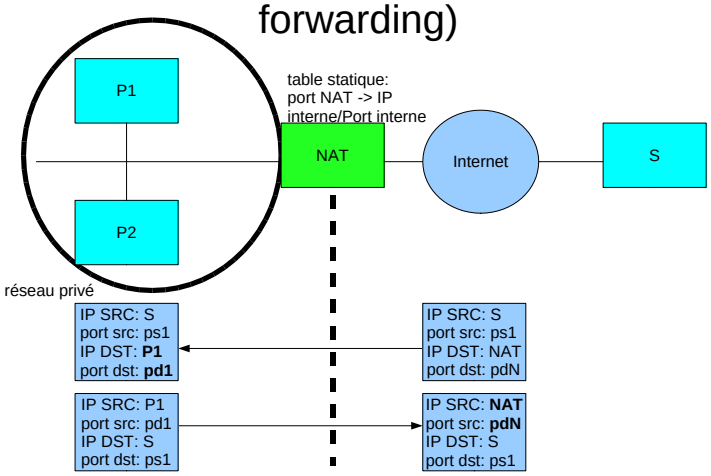
NAT bi-directionnel

- dans une version ultérieure de ce support
- pour permettre à des machines distantes d'accèder directement à des machines internes
- s'appuie sur le dns:
  - le serveur dns (en général la passerelle NAT) permet à la passerelle NAT de noter les association requete dns, ip distante
  - quid en cas de plusieurs requetes depuis la même ip distante ?

NAT double (twice NAT)

- on change adresses sources et destination.
- utilisé pour cacher les adresses sources aux destinations et lycée de Versailles.
- utile en cas de collission d'adresses entre sources et destination. Exemple: une entreprise qui a utilisé deux sous-réseaux privés identiques.

NAPT avec redirection de port (port forwarding)



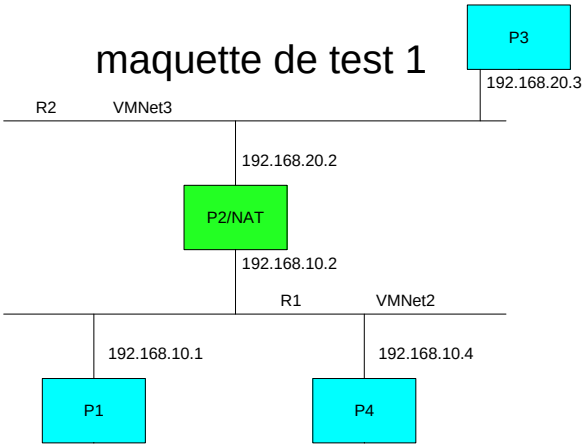
Configuration d'un routeur NAPT sous Linux

- iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source adresseIPPublique avec eth0: interface pour l'accès à internet (à adapter)
- pour effacer les règles correspondantes :
  - iptables -t nat -F
- pour les lister :
  - iptables -t nat -L

Configuration d'un routeur NATP sous windows

- mmc « routage et accès distant »
- puis « nom de votre serveur »/routage IP/general
- clic droit ou Action/nouveau protocole de routage
- « traduction d'adresse réseau (NAT) »
- « nom de votre serveur »/routage IP/NAT puis clic droit/nouvelle interface.Préciser pour chaque interface
  - si elle est du côté public ou privé
  - s'il faut activer la translation de ports (cocher « traduire les entêtes tcp/udp »)

maquette de test 1



Couleurs:
•vert: routage activé
•- bleu: hôtes non routeur

R1: 192.168.10/24
R2: 192.168.20/24

Votre travail (3)

- montez la maquette décrite ci-avant
- configurer la machine P2 en routeur/NAPT
- testez la connexion IP entre P1, P4, P2 et P3 (icmp avec ping, tcp avec ssh et udp avec netcat)
- Expliquez les comportements constatés. Notamment:
  - expliquer les modifications apportées aux paquets
  - comment le moteur NAT peut-il identifier les paquets entrants (savoir qu'un paquet est pour P4 et pas pour P1) ?

limitations de la translation d'adresses

- casse le principe de bout en bout, fondement de tcp/ip:
  - applications transportant les adresses IP/ports dans la charge utile TCP/IP
  - applications avec des sessions multiples interdépendantes, négociées dynamiquement
- débogage et flicage
- fragmentation: il faut défragmenter au vol
- gestion des états : 15 à 20% de charge cpu pour les routeurs/fw

translation d'adresse et sécurité

- du point de vue des machines internes :
  - le réseau interne n'est pas directement joignable
  - si les adresses internes sont affectées par dhcp: augmentation de la difficulté pour un intrus de désigner précisément un hôte
  - le routeur NAT est un point central critique en cas de piratage :
    - syndrome du « renard dans le poulailler »
    - MiM sur tout le trafic sortant
- du point de vue des machines externes:
  - tout est vu comme venant du routeur NAT ce qui ne facilite pas l'identification de la source d'une attaque

votre travail (4)

- expliquez le fonctionnement d'une connexion ftp du point de vue des connexion tcp (mode passif, du mode actif, connexions en jeu, commande PORT)
- mettez en évidence sous windows et sous linux (OS du routeur NAT) les interactions avec NAPT
  - les points posant problèmes
  - la façon dont ils sont résolus (vous illustrerez votre propos à l'aide de capture de trames)

Bibliographie : translation d'adresses :

- résumé en français : http://www.securiteinfo.com/conseils/nat.shtml
- rfc 3022: Traditional IP Network Address Translator (Traditional NAT)
- rfc 2663: IP Network Address Translator (NAT) Terminology and Considerations
- TCP/IP: « TCP/IP illustré: les protocoles »: W. R. Stevens