Projet réseau 2005-2006

- fin thème 1:
  - configuration réseau et routage sous linux
  - travail personnel: (1) mise en place d'un routeur linux.
- theme 2
  - 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 : mardi 6 décembre 2006

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)
-

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, default GW (noté DGW par la suite) : 192.168.10.2
  - 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
    - pas de DGW
  - debian-3: 1 interface réseau
    - adresse IP: 192.168.20.3,sous-réseau R2: 192.168.20/24, DGW: 192.168.20.2
- R1: réseau virtuel vmware: vmnet 3
- R2: réseau virtuel vmware: vmnet 4

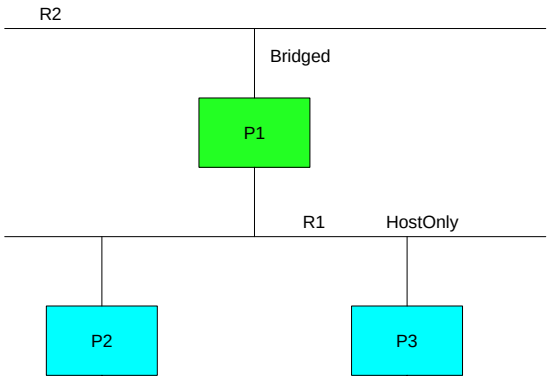
Votre travail (1)

- 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)

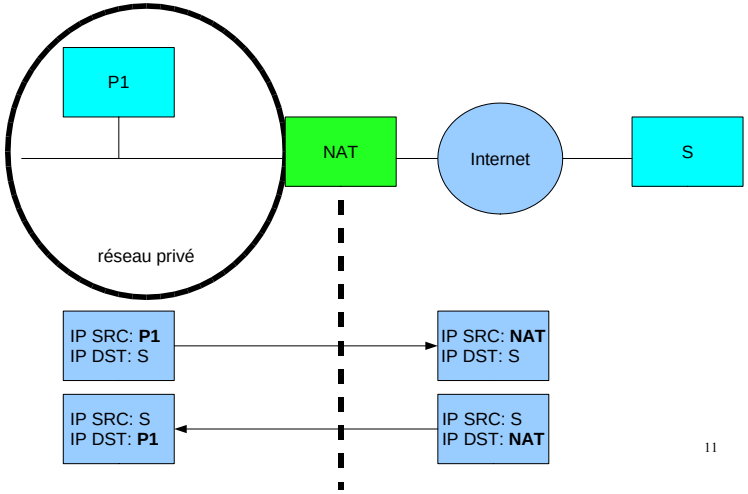
Votre travail (2)

- 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 une ou plusieurs 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

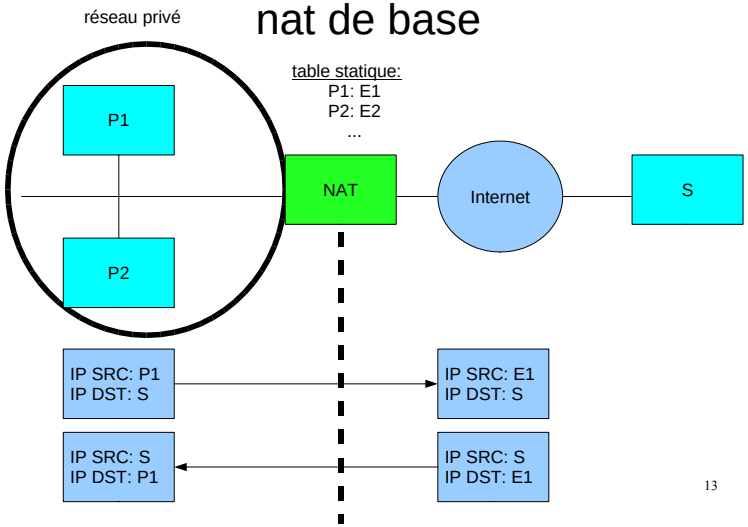
translation d'adresse



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



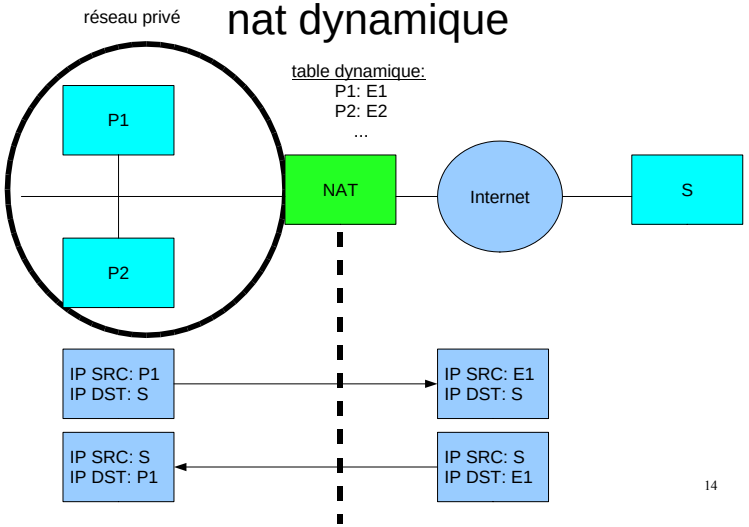
Le nat de base s'appuie sur une table de correspondance statique entre adresses IP du réseau privé et adresse**s** IP publiques du routeur NAT. Si le routeur NAT a n adresses IP publique, au plus n machines internes peuvent accéder à l'extérieur.

Il n'y a pas de translations de ports.

Le fait d'utiliser une table statique de correspondance permet un fonctionnement sans mémorisation d'informations liées à la connexion.

Cette remarque ne s'applique évidemment pas aux protocoles qui nécessitent un ALG (voir plus loin).

nat dynamique



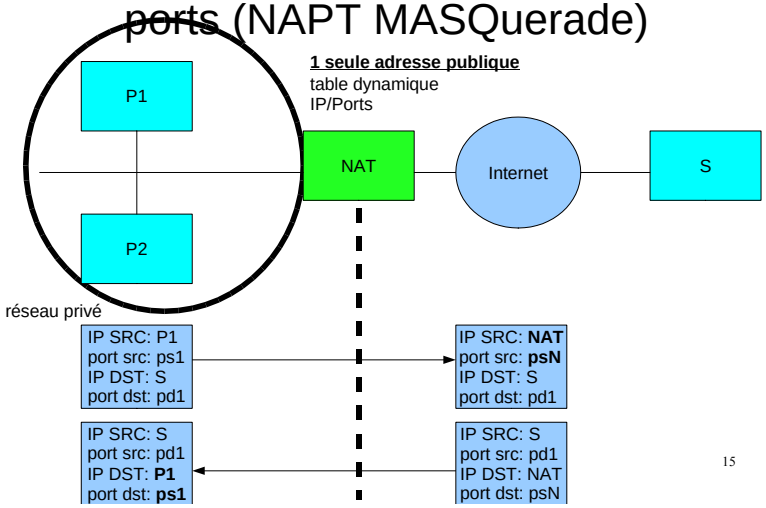
Le nat dynamique est une variante du nat de base où la table donnant la correspondance entre adresses ip internes et externe est construite dynamquement en fonction des besoins.

Avantages par rapport au nat de base: toutes les machines internes peuvent accèder à l'extérieur tant qu'il y en a au plus n (nombre d'adresses ip publiques) qui le font à la fois.

Défauts:

- il est nécessaire de suivre les connexions de façon à faire correspondre à chaque paquet entrant à la machine interne ad hoc.
- si on a n machines internes qui communiquent avec l'extérieur, les autres machines sont bloquées.

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



Le NAPT est utilisé avec une seule adresse publique. 2 problèmes sont posées :

- faire en sorte que des connexions venant de machines internes différentes apparaissent comme des connexions externes différentes. Quid du cas où les adresses/ports destinations et port source sont les mêmes pour deux machines sources internes différentes ? Solution: on change le routeur NAT change au vol le port d'émission des paquets d'une des connexions.
- savoir identifier à quelle machine interne transmettre les paquets entrants.

Solutions: éléments permettant de distinguer les « connexions » les unes des autres :

- adresse source du paquet entrant
- protocole de niveau 4 (tcp/udp)
- ports sources/destination
- en cas d'identité, le port source peut être changé sur les paquets sortant de façon à avoir des ports destinations différents sur les paquets entrants (le P de NAPT)

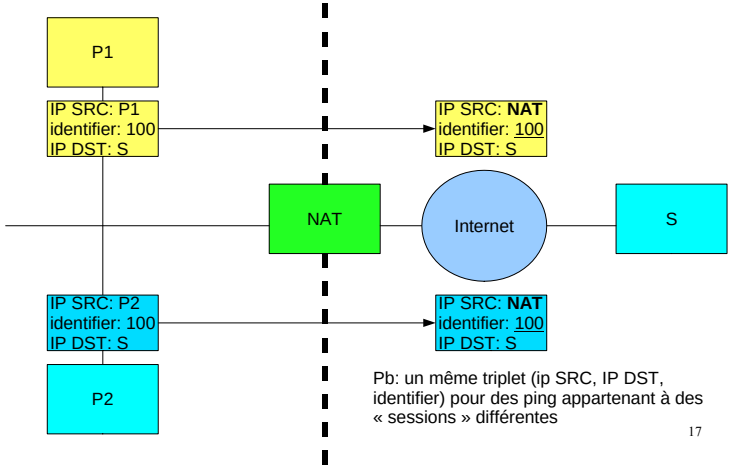
la modification est transparente pour les machines sources et destination.

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: chaque série de ping a un champ « identifier » qui permet de l'idenfier et de faire correspondre chaque « réponse echo » à la bonne « requête echo ». Il est garanti que deux sessions ping originaire du même hôte aient des « identifiers » différents.

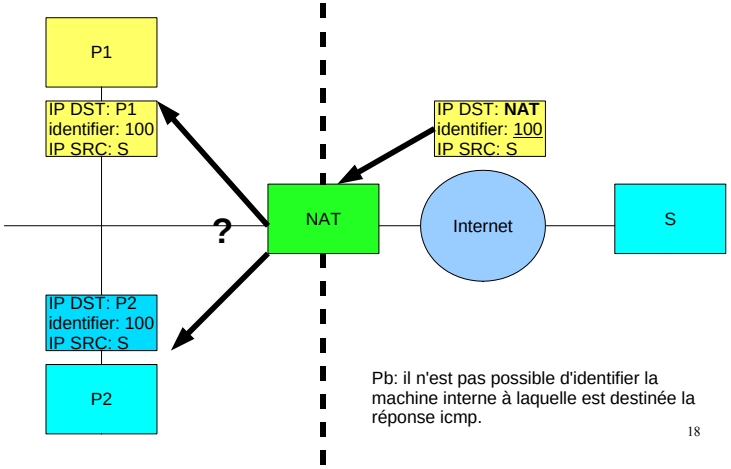
ping: sous unix, ping utilise son numéro de processus comme valeur du champ identifier. Dans le cas où plusieurs instances de la commande ping sont exécutées en même temps, cela garantit de pouvoir transmettre à chaque instance les réponses qui lui sont destinées car les réponses conservent l'identifier de la requête initiale.

NAPT et ping: problème



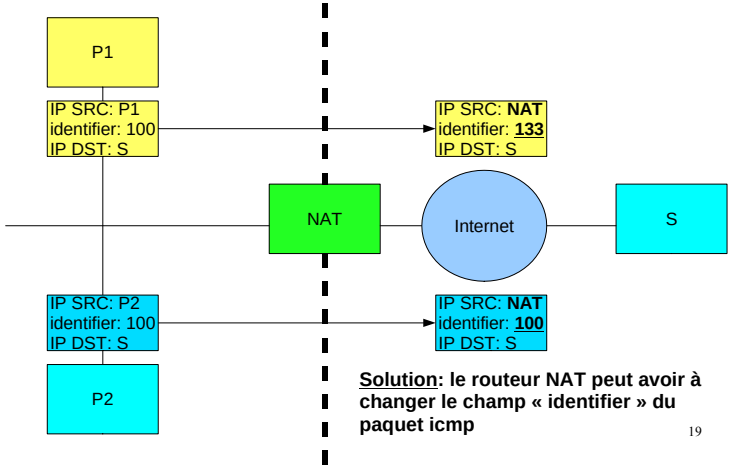
Pb: un même triplet (ip SRC, IP DST, identifier) pour des ping appartenant à des « sessions » différentes

NAPT et ping: problème



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

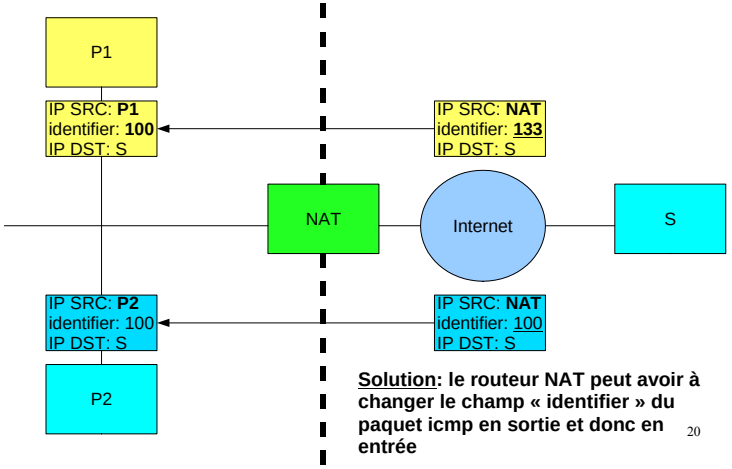
NAPT et ping: solution



Solution: le routeur NAT peut avoir à changer le champ « identifier » du paquet icmp

de façon générale, on appellera l'identifiant icmp ou le port sources tcp/udp : un identifiant de transport (i.-e. de la couche transport).

NAPT et ping: solution



Solution: le routeur NAT peut avoir à changer le champ « identifier » du paquet icmp en sortie et donc en entrée

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)

paquets: un paquet va dans un sens donné
connexion: elle va dans un sens (celui du paquet qui a ouvert la connexion en général) et est composée de mutliples paquets allant dans les deux sens.
sessions: la translation d'adresse travaille sur des sessions constituées d'une ou plusieurs connexions.
Le traitement d'un paquet peut être rarement déterminé indépendamment du contexte(paquet émis précédemment, connexions, ...). La translation d'adresse doit maintenir des tables pour mémoriser les état des connexions constituant une session et appliquer le traitement ad hoc aux paquets qui la compose. Dans le cas le pire, il peut être nécessaire d'analyser/modifier la partie de niveau application du paquet. Un tel travail est réalisé par une passerelle de niveau application (ALG).
Exemples:
- ping: il faut mémoriser les transformations apportées aux adresses sources et identifiants (cf plus tôt)

ftp, h323: une session est constituée de plusieurs connexions dans des sens variés.

L'expression « à état » ou « statefull » est un terme que l'on retrouve quand on parle de firewall. Il désigne le même mécanisme de mémorisation des informations sur les connexions/sessions pour déterminer le traitement ad hoc à appliquer à un paquet individuel.

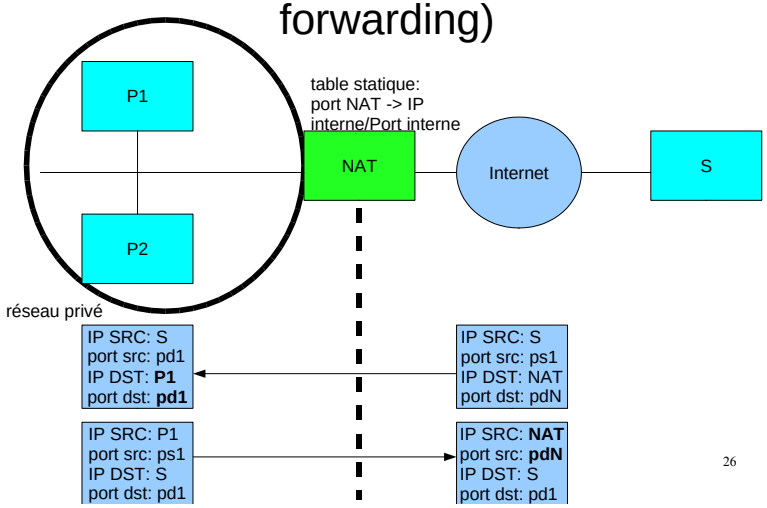
NAT bi-directionnel

- 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)



Le NAPT est utilisé avec une seule adresse publique (ràf: obligation ?). 2 problèmes sont posées :

- faire en sorte que des connexions venant de machines internes différentes apparaissent comme des connexions externes différentes. Quid du cas où les adresses/ports destinations et port source sont les mêmes pour deux machines sources internes différentes ?
- savoir identifier à quelle machine interne transmettre les paquets entrants.

Solutions: éléments permettant de distinguer les « connexions » les unes des autres :

- adresse source du paquet entrant
- protocole de niveau 4 (tcp/udp)
- ports sources/destination
- en cas d'identité, le port source peut être changé sur les paquets sortant de façon à avoir des ports destinations différents sur les paquets entrants (le P de NAPT)

la modification est transparente pour les machines sources et destination.

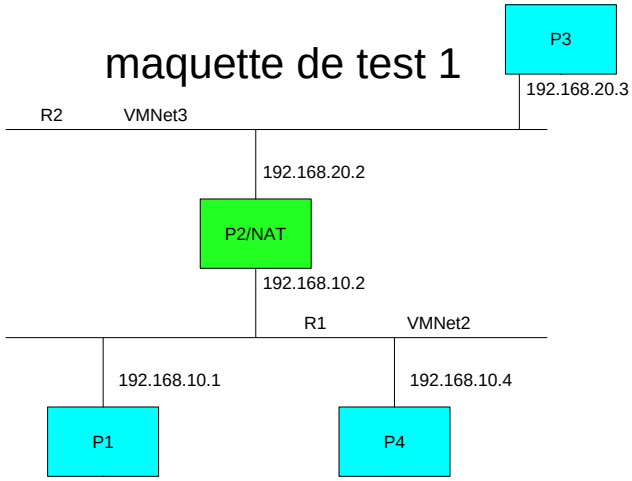
Configuration d'un routeur NAPT sous Linux

- iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source adresseIPPublique avec eth0: interface pour l'accès à internet (à adapter)
- parler aussi de l'accès aux tables NAT

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
- la machine P2 a une interface réseau sur vmnet3 (sur R2) et une interface réseau sur vmnet2 (sur R1.)
- les autres ordinateurs ont une seule interface réseau sur vmnet2 (P1/R1) ou vmnet3 (P3/R2).
- 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 en vous appuyant sur une ou plusieurs captures de trames. On s'intéressera notamment
  - à expliquer les modifications apportées aux paquets
  - à la façon dont le moteur NAT peut identifier les paquets entrants et notamment savoir qu'un paquet est pour P4 et pas pour P1

P1, P4 et P3 seront des machines sous linux.
Pour P2, le choix du système d'exploitation est libre (mais vous pouvez aussi tester les deux).
L'image linux à utiliser est debian-sarge-X-2005-1120 (elle contient un serveur ftp ce qui sera utile par le suite et les vmwares tools).

limitations de la translation d'adresses

- applications transportant les adresses IP/ports dans la charge utile TCP/IP
- applications avec des sessions multiples interdépendantes, négociées dynamiquement
- débogage et flicage
- fragmentation

De base, le mécanisme de translation d'adresses et de ports changent les adresses sources et/ou destination des paquets ainsi que les ports. Cela se révèle insuffisant dans les cas où ces informations sont présentes ou utilisées dans la charge utile du paquet. Le paquet devient alors invalide pour l'application. On peut citer le cas d'IPsec qui ràf. Pour pouvoir gérer ces protocoles, quand c'est possible, il est nécessaire que NAPT aie un module spécifique qui sache décoder ces informations de niveau application (Application Layer Gateway ou ALG) propre à chaque protocole.

sessions multiples: les protocoles comme FTP, H.323, SIP et RTSP qui utilisent plusieurs sessions sont souvent cassés par le NAPT. Les connexionx multiplent seront vues par NAPT comme des connexions indépendantes ce qu'elles ne sont pas. De plus, certaines informations concernant les connexions secondaires pourront avoir circulé dans la charte utile des paquets. Comme dans le cas précédent, pour gérer ces protocoles, il est nécessaire que NAPT aie un module spécifique qui sache décoder ces informations de niveau application (Application Layer Gateway ou ALG) propre à chaque protocole.

Vu de l'extérieur, les paquets semblent venir du routeur-NAT. Il est ainsi difficile d'identifier le poste du réseau privé d'où vient une connexion donnée. Cela pose des problèmes pour le débogage réseau et pour la détection d'abus.

Pour fonctionner dans certains cas, nous venons de voir qu'il était nécessaires d'utiliser les informations présentes dans la charge utile des paquets. C'est extrêmement difficile, dans le cas où les paquets sont fragmentés. Dans ce cas, le routeur NAT doit défragmenter les paquets avant de les traiter.

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
  - pose les mêmes problèmes qu'une firewall (ràf)
  - 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)

L'image linux à utiliser pour la machine qui fait office de serveur ftp est debian-sarge-X-2005-1120: elle contient un serveur ftp ce qui sera utile par le suite et les vmwares tools. A noter que pour des questions d'économie mémoire, le gestionnaire de fenêtre est minimal. Pour avoir une fenêtre de commande « clic gauche puis Xterm »

Vous monterez une maquette avec un routeur Debian Gnu/Linux et une maquette avec un routeur windows 2000 serveur.

l'ALG ftp est présent en standard dans le routeur NAT de windows 2000 server. Pour ce qui est de linux, il est possible qu'il faille charger explicitement les modules netfilter implantant l'ALG ftp:

- modprobe ip_conntrack_ftp
- modprobe ip_nat_ftp

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

le RFC 2663 définit des notions propres à tous les types de NAT. Il définit la terminologie et concepts associés. Il traite aussi des problèmes courants. C'est le premier rfc à lire sur le sujet. C'est un RFC dont la lecture est plutôt aisée.

le RFC 3022 traite uniquement du nat traditionnel (accès à internet de postes ayant des adresses ip privées et redirection de port (accès à certains ports d'un serveur interne). A lire après le RFC2663. C'est un RFC dont la lecture est plutôt aisée.