Notions de Réseaux TCP/IP
et environnements Microsoft Windows

Michel Cabaré
Novembre 2002
ver 2.0

TABLE DES MATIÈRES

STRUCTURE DE TCP/IP

Modèle TCP/IP :

Par rapport au modèle OSI classique en 7 couches, le modèle présentant **TCP/IP** est composé de 4 couches uniquement :

OSI	TCP/IP
❼Application ❻Présentation ❺Session	❹Application :**SNMP-FTP-SMTP**…
❹Transport	❸Transport : **TCP** ou **UDP**
❸Réseau (routage)	❷Internet : **IP, ARP, ICMP** routage : **RIP, SPF**
❷Liaison ❶Physique	❶Interface Réseau

Couche 1 Interface Réseau :

Elle a en charge la communication physique avec le réseau. par conséquent doit pouvoir accepter les normes **Ethernet**, **Token-Ring**…

Couche 2 Internet :

Elle s'occupe du routage et de la livraison des paquets au travers du protocole **IP (Internet protocol)**.

Tous les protocoles de la couche Transport passent par **IP** pour acheminer leurs données, mais IP est un protocole non connecté, il ne garantit pas donc que les paquets émis ne soient pas perdus, dupliqués ou inutilisables…

C'est au couches supérieures (transport ou application) de vérifier le résultat!

La couche internet contient aussi un **protocole ICMP (Internet Control Messaging Protocol)** permettant de mettre en œuvre des contrôles sur le transport des paquet IP et de rapporter les erreurs…

La couche internet contient aussi un protocole **ARP (Adress resolution Protocol)** permettant de mettre en œuvre des mécanismes de résolution pour trouver une adresse physique avec une adresse IP…

Couche 3 Transport :

Elle a elle le rôle de fournir a la couche application une communication entre 2 machines…

2 protocoles existent selon que l'on souhaite utiliser une communication avec connexion ou sans…

le protocole **TCP (transmission Control Protocol)** est utilisé pour la communication connectée entre deux machines (fiable mais avec un débit relativement faible du fait des contrôles)

le protocole **UDP (user Datagram Protocol)** est utilisé pour la communication non connectée, sans garantie de distribution (moins fiable mais avec un débit plus élevé du fait de l'absence des vérifications)

Couche 4 Application :

Elle prends en charge toutes les activités supérieures du modèle OSI.

Plusieurs protocoles existent dans cette couche selon l'objectif visé :

SNMP (Simple Network management Protocol)	-> gestion de réseau
FTP (File transfer protocol)	-> transfert de fichier
SMTP (Simple Mail transfer Protocol)	-> courrier électronique
HTTP (Hyper Text Transfer Protocol)	-> serveurs web

LES PROTOCOLES DE TCP/IP

TCP (Transmission Control Protocol) :

TCP, on l'a dit, est un protocole utilisé pour la communication connectée entre deux machines (fiable mais avec un débit relativement faible du fait des contrôles)

Port et Socket :

Les **Ports** identifient les processus en cours d'exécution dans la couche application, et par conséquent un n° de port identifie un processus auquel on doit envoyer des données.

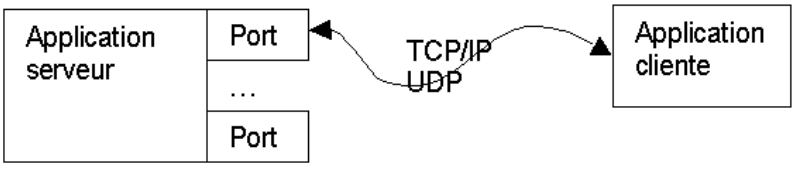
Les numéros de ports sont donnés de manière prédéterminée pour ceux allant de 1 à 1023, mais restent libres pour les autres

Il a été ainsi arbitrairement décidé d'un N° de Port pour chaque usage.

Port n° 21	: File Transfer Protocol
Port n° 22	: SSL connexion à distance sécurisée
Port n° 23	: Telnet
Port n° 25	: SMTP réception de courrier
Port n° 53	: DNS Domain Name Server
Port n° 80	: HTTP pages web
Port n° 88	: Kerberos authentification (NT 2000)
Port n° 110	: POP3 lecture de courrier
Port n° 137 à 139	: NetBios
Port n° 443	: HTTPS pages web sécurisées
Port n° 546	: DHCP

Les ports proposent 65535 point d'accès à un ordinateur à partir d'une seule adresse physique.

L'ensemble d'une adresse IP d'un ordinateur essayant de communiquer et du numéro de ports utilisé crée ce que l'on appelle un "**Socket**"



Communication en mode Connecté :

TCP demande qu'une session soit établie avant de transmettre les données entre les machines connectées.

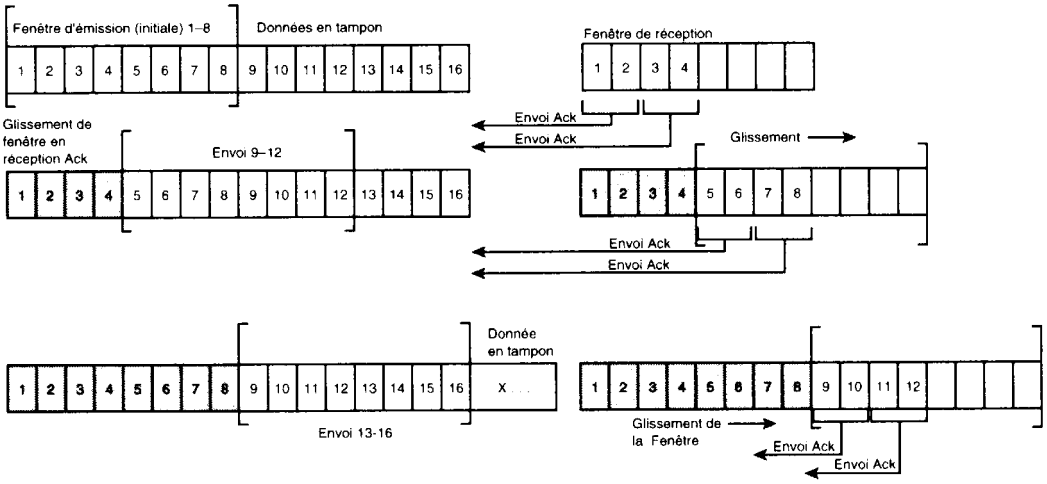
En tant que protocole en mode connecté, TCP suit la transmission et la réception des paquets individuels durant la communication. TCP envoie les

paquets en séquences et demande un accusé de reception de ces paquets avant d'en envoyer d'autres.

Etant donné le mécanisme de vérification effectué par TCP, le format d'un paquet TCP peut être assez complexe…

Fenêtres variables :

Chaque machine dispose d'un **fenêtre d'émission** et **d'une fenêtre de réception** qu'elle utilise comme tampon de donnée pour rendre la communication plus efficace.



Une **fenêtre de réception** permet à une machine de recevoir des paquets en désordre (en effet TCP utilise IP qui ne garantit pas l'ordre d'arrivée, ni même l'arrivée des paquets !) et de les classer pendant qu'elle attends les paquets suivants

Au fur et a mesure que la fenêtre de réception récupère des paquets, elle renvoi des accusés de réception (un accusé tous les 2 paquets reçuts)

Si la fenêtre d'émission ne reçoit pas d'accusé de réception, elle attends puis retransmet les paquet non acquittés .

Dans la **fenêtre d'émission** un temporisateur est positionné pour chaque paquet envoyé, indiquant le temps à attendre avant d'estimer que le paquet n'est pas arrivé. En cas de non acquittement, le paquet est envoyé une nouvelle fois avec le temporisateur doublé, après cette nouvelle attente, s'il n'y a toujours pas d'acquittement, on recommence en doublant encore le temporisateur…avec un maximum de x tentatives…

N.B: Sous WINDOWS NT les fenêtres par défaut ont une taille de 8 kilo-octets, soit 8 trames Ethernet standard

N.B: Sous WINDOWS NT les fenêtres d'émission sont paramétrées par défaut pour tenter d'émettre 5 fois maximum

UDP (User Datagram protocol) :

UDP, on l'a dit, est utilisé pour la communication non connectée, sans garantie de distribution (moins fiable mais avec un débit plus élevé du fait de l'absence des vérifications)

Port et Socket :

Les paquets **UDP** sont transmis comme pour TCP a des **Sockets**, c'est à dire à des couples adresses Ip + N° de Port, mais avec moins de fiabilité (puisque aucun contrôle n'est effectué…)

Port n° 67-68 : SNMP gestion - surveillance réseau

Port n° 520 : RIP routage IP dynamique

Communication en mode non Connecté :

On peut se demander ou réside l'intérêt d'un tel protocole, fondamentalement dans sa faible surcharge (les données qu'il rajoute pour sa gestion sont très faibles par rapport aux données utiles transmises…)

Deux exemples suffiront à se convaincre de l'intérêt de ce protocole

exemple SNMP

SNMP utilise le protocole **UDP** pour véhiculer ses interrogations sur le réseau, et transmettre les messages d'erreurs d'un machine…

Il est normal que lorsque une machine soit défaillante, elle ne puisse réussir à mettre en place une session **TCP** pour transmettre son … malaise !

une diffusion **UDP** est beaucoup plus raisonnable ne terme "espérance de vie" de la part de cette machine

exemple Vidéo et Son en ligne

Dans ce cas de figure il faut privilégié a tout prix le débit, ce que **UDP** fait, au détriment du paquet perdu, qu'il est bon d'ailleurs de ne pas tenter de ré-émettre…

En effet si on écoute un morceau de musique, et qu'un segment manque, notre oreille s'en rends à peine compte, et notre "cerveau" corrige ! Imaginons l'effet auditif du lecteur de CD qui bloque l'émission pour attendre la réception acquittée du fragment retardataire…

IP (Internet Protocol) :

Tous les protocoles de la couche Transport passent par **IP** pour acheminer leurs données, mais IP est un protocole non connecté, il ne garantit pas donc que les paquets émis ne soient pas perdus, dupliqués ou inutilisables…

C'est au couches supérieures (soit via TCP dans le transport ou application si on utilise UDP dans le transport) de vérifier le résultat!

Adresse IP

Ce protocole repose en partie sur la notion d'adresse IP (Internet Protocol) décernée de façon unique pour chaque élément matériel faisant partie d'une réseau

on verra cette notion en détail dans le chapitre "Adresse IP" (page 17)

Datagramme

IP reçoit des information des protocoles **TCP** ou **UDP** et les renvoi dans ce que l'on appelle un **Datagramme**, c'est à dire un bloc de donnée dans lequel IP à rajouté ses informations (type de protocole utilisé : udp ou tcp, adresse ip de la machine d'origine, adresse ip de la machine destinataire, durée de vie…) aux données utiles.

Les données qui circulent sur Internet sous forme de datagrammes (on parle aussi de paquets) sont des données encapsulées, c'est-à-dire des données auxquelles on a ajouté des en-têtes correspondant à des informations sur leur transport (telles que l'adresse IP de destination, ...).

Les données contenues dans les datagrammes sont analysées (et éventuellement modifiées) par les routeurs permettant leur transit.

Voici ce à quoi ressemble un datagramme:

<-------------------------- 32 bits ---------------------------->

<table>
<tr><td>Version</td><td>Taille d'en-tête</td><td>type de service</td><td colspan="2">Longueur totale</td></tr>
<tr><td colspan="3">Identification</td><td>Drapeau</td><td>Décalage fragment</td></tr>
<tr><td colspan="2">Durée de vie</td><td>Protocole</td><td colspan="2">Somme de contrôle en-tête</td></tr>
<tr><td colspan="5">Adresse IP source</td></tr>
<tr><td colspan="5">Adresse IP destination</td></tr>
<tr><td colspan="5">Données</td></tr>
</table>

Voici la signification des différents champs:

- **Version**: il s'agit de la version du protocole IP que l'on utilise (actuellement on utilise la version 4 *IPv4*) afin de vérifier la validité du datagramme. Elle est codée sur 4 bits
- **Taille d'en-tête**: il s'agit du nombre de mots de 32 bits sur lesquels sont répartis l'en-tête
- **Type de service**: il indique la façon de laquelle le datagramme doit être traité
- **Longueur totale**: il indique la taille totale du datagramme en octets. La taille de ce champ étant de 2 octets, la taille totale du datagramme ne peut dépasser 65536 octets. Utilisé conjointement avec la taille de l'en-tête, ce champ permet de déterminer où sont situées les données
- **Identification, drapeaux (flags) et déplacement de fragment** sont des champs qui permettent la fragmentation des datagrammes, il sont expliqués plus loin dans l'assemblage.

- **Durée de vie: (appelée aussi TTL: Time To Live)** indique le nombre maximal de routeurs à travers lesquels le datagramme peut passer. Ainsi ce champ est décrémenté à chaque passage dans un routeur, lorsque celui-ci atteint la valeur critique de 0, le routeur détruit le datagramme. Cela évite l'encombrement du réseau par les datagrammes perdus
- **Protocole:** ce champ permet de savoir de quel protocole est issu le datagramme avec par exemple

  ICMP: 1
  IGMP: 2
  TCP: 6
  UDP: 17
- **Somme de contrôle de l'en-tête (header checksum):** ce champ contient une valeur codée sur 16 bits qui permet de contrôler l'intégrité de l'en-tête afin de déterminé si celui-ci n'a pas été altéré pendant la transmission. Pour ce faire, on considère l'en-tête comme une suite d'entiers, on fait la somme de ces entiers en complément à 1, puis on complémente le résultat à 1, on obtient alors le total de contrôle. Celui-ci est en fait tel que lorsque l'on fait la somme des champs de l'en-tête (somme de contrôle inclue) donne un nombre avec tous les bits positionnés à 1
- **Adresse IP Source:** Ce champ représente l'adresse IP de la machine émettrice, il permet au destinataire de répondre
- **Adresse IP destination:** Adresse IP du destinataire du message

Fragmentation

La taille d'un Datagramme dépendant du type de réseau utilisé, Ethernet, Token-Ring…IP doit alors éventuellement découper les données qu'il reçoit de TCP ou de UDP en morceau pour être émises dans plusieurs Datagrammes de taille adéquate. Ce découpage, avec repérage et étiquetage des morceau s'appelle la Fragmentation.

la taille d'un datagramme maximale est de 65535 octets. Toutefois cette valeur n'est jamais atteinte car les réseaux n'ont pas une capacité suffisante pour envoyer de si gros paquets. De plus, les réseaux sur Internet utilisent différentes technologies, si bien que la taille maximale d'un datagramme varie suivant le type de réseau.

La taille maximale d'une trame est appelée *MTU* (Maximum Transfer Unit), elle entraînera la fragmentation du datagramme si celui-ci a une taille plus importante que le MTU du réseau.

Type de réseau	MTU (en octets)
Arpanet	1000
Ethernet	1500
FDDI	4470

La fragmentation d'un datagramme se fait au niveau des routeurs, c'est-à-dire lors de la transition d'un réseau dont le MTU est important à un réseau dont le MTU est plus faible. Si le datagramme est trop grand pour passer sur le réseau, le routeur va le fragmenter, c'est-à-dire le découper en fragments

de tailles inférieures au MTU du réseau et de telle façon que la taille du fragment soit un multiple de 8 octets.

Assemblage

bien sûr à l'arrivée **IP** doit récupérer tous les morceau et reconstruire les données d'origine, cela s'appelle l'**Assemblage**.

Le routeur va donc ensuite envoyer ces fragments de manière indépendante et réencapsulé (il ajoute un en-tête à chaque fragment) de telle façon à tenir compte de la nouvelle taille du fragment, et en ajoutant des informations afin que la machine de destination puisse réassembler les fragments dans le bon ordre (rien ne dit que les fragments vont arriver dans le bon ordre étant donné qu'ils sont acheminés indépendamment les uns des autres...).

Pour tenir compte de la fragmentation, chaque datagramme possède plusieurs champs permettant leur réassemblage:

- **Champ déplacement de fragment:** champ permettant de connaître la position du début du fragment dans le datagramme initial
- **Champ identification:** numéro attribué à chaque fragment afin de permettre leur réassemblage dans le bon ordre
- **Champ longueur total:** il est recalculé pour chaque fragments
- **Champ drapeau:** il est composé de trois bits**:**
    1. Le premier n'est pas utilisé
    2. Le second (appelé **DF**: *Don't Fragment*) indique si le datagramme peut être fragmenté ou non. Si jamais un datagramme a ce bit positionné à un et que le routeur ne peut pas l'acheminer sans le fragmenter, alors le datagramme est rejeté avec un message d'erreur
    3. Le dernier (appelé **MF**: *More Fragments*, en français *Fragments à suivre*) indique si le datagramme est un fragment de donnée (1). Si l'indicateur est à zéro, cela indique que le fragment est le dernier (donc que le routeur devrait être en possession de tous les fragments précédents) ou bien que le datagramme n'a pas fait l'objet d'une fragmentation

Routage

Le protocole **IP** doit **router** les datagrammes d'un réseau à l'autre. Toutes les machines d'un réseau ne sont pas des routeurs, mais un **routeur** est une machine qui lorsqu'elle recoit un datagramme qui ne lui est pas adressé, doit renvoyer ce paquet sur le réseau dans la bonne direction pour qu'il atteigne sa destination…

Les principes du routage IP peuvent être résumés au moyen de l'**algorithme** suivant (un chapitre sera consacré au routage plus loin):

1. **Extraire** l'adresse IP de destination du datagramme.
2. Appliquer à cette adresse le masque de sous-réseau éventuel (ET logique entre l'adresse et le masque).
3. Extraire la partie "**Identificateur Réseau**" de l'adresse ainsi obtenue.
4. S'agit-il du réseau local ?
    - Si oui, procéder à l'encapsulation et au **routage direct**.
    - Sinon, existe-t-il une entrée dans la **table de routage** pour ce réseau de destination ?
        a. Si oui, envoyer le datagramme vers la passerelle spécifiée dans la table.
        b. Sinon, existe-t-il une **route par défaut** ?
            - Si oui, envoyer le datagramme à la passerelle spécifiée par la route par défaut.
            - Sinon, déclarer une **erreur** de routage. (Protocole ICMP)

Durée de Vie TTL

La durée de vie ou **TTL (Time To Live)** correspond à l'idée suivante. Chaque fois qu'un **Datagramme** prends le départ d'une machine sur le réseau vers une destination connue, il a une espérance de vie exprimée en seconde.

A chaque passage dans un **routeur**, celui-ci décrémente de 1 seconde son compteur **TTL** de vie, de sorte que si le datagramme tarde trop à parvenir à la machine destinataire, à un moment donné un routeur le "détruira" en réduisant son compteur **TTL** de 1 à 0…

N.B: Sous WINDOWS NT les Datagrammes ont une valeur de Vie par défaut de 128 secondes (soit environ 2 minutes…)

ICMP (Internet Control Message Protocol) :

ICMP permet de mettre en œuvre des contrôles sur le transport des paquet IP et de rapporter les erreurs…

Les messages **ICMP** servent principalement à rapporter des erreurs et envoyer des requêtes.

Nous utilisons le protocole **ICMP** essentiellement pour envoyer des requêtes d'Echo request et pour attendre des réponses d'Echo reply, et encore ceci à travers un utilitaire **PING (Personnal Internet Groper)**

on verra cette notion en détail dans le chapitre "Tester IP" (page 40)

ARP (Address Resolution Protocol) :

A part dans le cas ou on émet en diffusion, lorsque IP souhaite émettre, il doit connaître l'adresse physique, ou adresse mac, ou adresse Ethernet du poste destinataire

ARP permet de mettre en œuvre des mécanismes de résolution pour trouver l'adresse physique correspondant à une adresse IP locale…

Si **ARP** ne connaît pas l'adresse physique de l'adresse IP locale demandée, il fonctionne par diffusion locale et, une fois trouvée, stocke cette correspondance dans sa mémoire (pendant un certain temps)

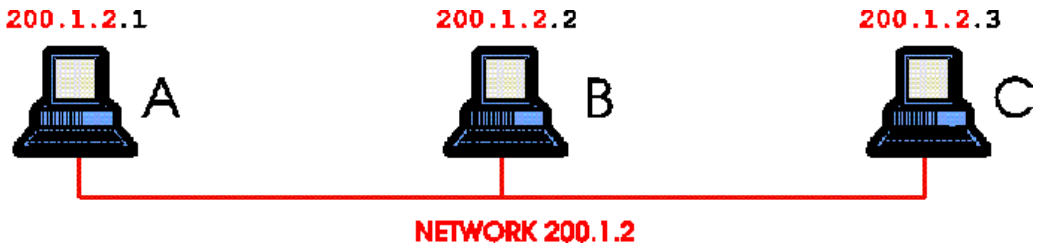
Si **ARP** connaît l'adresse physique dans son cache, il ne diffuse rien sur le réseau et cela fonctionne très bien

N.B : Mais **ARP** ne peut trouver que des adresses physiques locales, et il ne retourne jamais à IP une adresse physique qui se trouve sur un réseau distant ! Dans ce Cas **IP** (qui peut via l'adresse ip se rendre compte que la machine demandée n'est pas une machine locale) ne demande pas à **ARP** de trouver l'adresse physique de la machine distante, mais il lui demande de trouver l'adresse pysique du **routeur** !

on verra cette notion en détail dans le chapitre "Tester IP" (page 40)

Exemple de fonctionnement de ARP en local

Soit un réseau interne TCP/IP comprenant un segment Ethernet et trois machines. Le numéro de réseau IP de ce segment est 200.1.2. Les numéro d'hôte pour A, B et C sont 1, 2 et 3 respectivement. Ce sont des adresses de classe C, ce qui permet d'avoir 254 machines sur ce segment.



Supposons que A veuille envoyer un paquet à C pour la première fois, et qu'il connait l'adresse IP de C. Pour envoyer ce paquet sur ce brin Ethernet, A aura besoin de connaître l'adresse MAC (ou adresse Ethernet) de C. Le protocole **ARP** (Address Resolution Protocol) est utilisé pour trouver dynamiquement cette adresse.

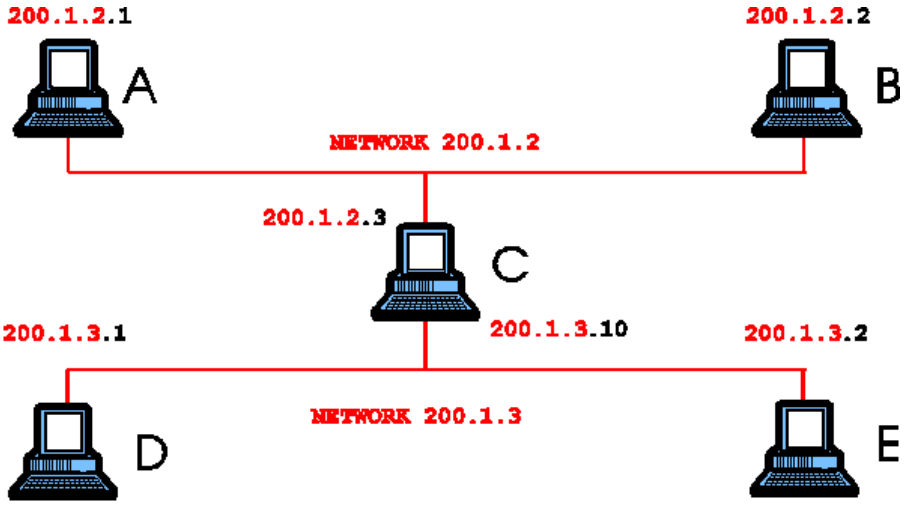
ARP garde une table interne d'adresses IP et d'adresses MAC correspondantes. Quand A essaye d'envoyer un paquet IP à C, le module d'ARP consulte sa table d'adresses IP et ne découvrira aucune entrée pour C. ARP envoie alors un paquet spécial reçu par tous (broadcast),

demandant l'adresse MAC correspondant à l'adresse IP qu'il connait. S'il n'y a pas de "time-out", cela signifie que la machine C a répondu en incluant son adresse MAC dans sa réponse, et le tour est joué. A met à jour sa table d'adresse (ou table d'hôte) et peut envoyer son paquet.

Exemple de fonctionnement de ARP et Routeur

Considérons maintenant 2 réseaux Ethernet séparés et reliés par la machine C, fonctionnant comme un routeur.

La machine C agit comme un routeur entre ces deux réseaux. Un routeur est un élément qui choisit différentes directions pour les paquets en fonction de



l'adresse IP. Comme il y a deux segments Ethernet séparés, chaque réseau a son propre numéro de réseau de classe C. Ceci est indispensable car le routeur ne connaît à des interfaces qui sont associés à un numéro de réseau.

Si A veut envoyer un paquet à E, il doit d'abord l'envoyer à C qui peut faire suivre le paquet à E. Ceci est possible car A utilise l'adresse MAC de C et l'adresse IP de E. C va donc recevoir le paquet destiné à E et va le faire suivre en utilisant l'adresse MAC de E, soit parce qu'il la connaît, soit en faisant une requête ARP comme décrit précédemment.

Si E reçoit le même numéro de réseau que A, soit "200.1.2", A essayera d'atteindre E de la même façon qui atteint C, par exemple, en envoyant une requête ARP et en attendant la réponse. Quoiqu'il en soit, comme E est physiquement sur un fil différent, il ne verra jamais la requête ARP et le paquet ne pourra pas être délivré. En spécifiant que E est sur un réseau différent, le module IP de A saura que E ne peut être atteint sans avoir été fait suivre par un nœud (élément reliant deux réseaux différents comme un routeur) de son réseau.

API NEtBIOS :

L'API **NetBIOS** a été développée par MICROSOFT pour devenir une interface standard des applications qui accèdent aux protocoles de réseau de la couche transport.

Des interface **NetBIOS** ont été écrites pour les protocoles **NetBEUI**, **Nwlink** et **TCP/IP**

NetBios pour identifier de manière unique une machine demande non seulement une adresse IP, mais aussi un **nom NetBIOS**

Lorsqu'elle utilise des noms dans une session **NetBIOS**, la machine émétrice doit pouvoir résoudre le nom **NetBIOS** en une adresse **IP**, (et ensuite cette adresse IP sera résolue en adresse physique …)

API Windows Sockets:

c'est une autre API standard du marché supportée également par Microsoft

ADRESSE IP

Adresse IP :

La version actuelle de ce protocole désormais quasi universel repose en partie sur la notion d'adresse **IP** (Internet Protocol) décernée de façon unique pour chaque élément matériel faisant partie d'une réseau

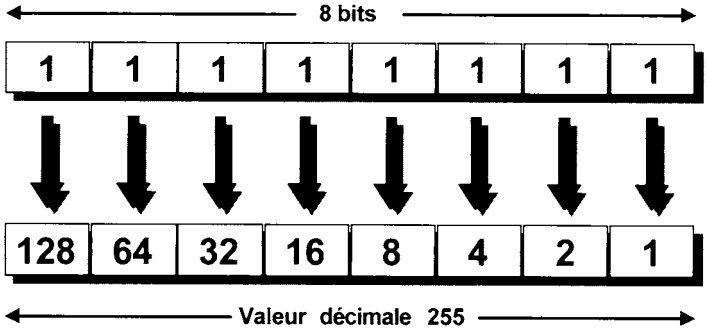
Ces adresses sont codées sur 32 bits, est sont représentées sous la forme de 4 nombre compris entre 0 et 255 (valeur d'un octet) et séparés par un point, soit (par exemple)

255.0.127.2 (ADRESSE DECIMALE)

On parle de bits de poids fort pour les nombres les plus à gauche

On parle de bits de poids faible pour les nombres les plus à droite

Chaque nombre décimal est la représentation d'un nombre binaire de 8 chiffres



On peut alors avoir aussi en notation binaire

11111111.00000000.01111111.00000010 (ADRESSE BINAIRE)

On parle de bits de poids fort pour les nombres les plus à gauche

On parle de bits de poids faible pour les nombres les plus à droite

On pourrait ainsi dire que les adresses IP varient de la plus petite 0.0.0.0 à la plus grande255.255.255.255

En fait toutes les combinaisons ne sont pas disponibles, et elles reflètent une certaine logique

ID réseau et ID hôte :

Les bits de poids fort définissent l'adresse du réseau, on parle de **ID réseau** et Les bits de poids faible définissent l'adresse d'un équipement dans le réseau on parle de **ID hôte**.

L' **ID réseau** identifie toutes les machines qui se trouvent sur le même réseau physique , encore appelé domaine de collision. Il s'agit d'un identifiant pour un réseau local , toutes les machines se trouvant "du même coté d'un routeur…

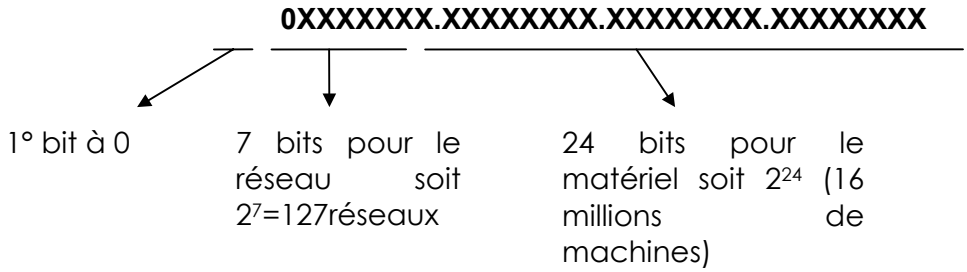
L' **ID hôte** identifie tout poste ou périphérique du réseau, il est unique à l'intérieur de tout **ID réseau**

Classes d'Adresse :

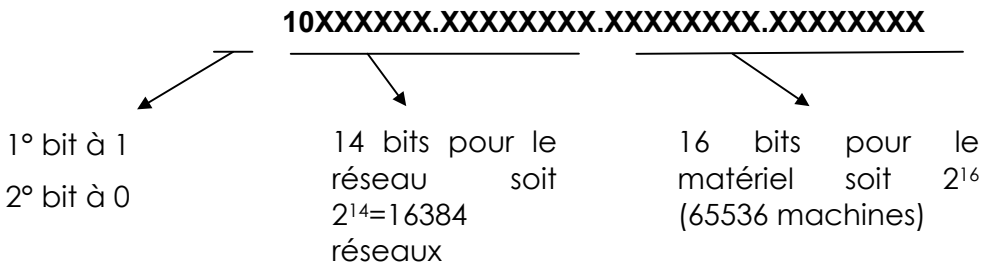
La limite entre poids fort et poids faible n'est pas toujours la même, c'est la notion de "**classe d'adresse**"

- plus les poids fort sont petits, et plus le nombre de machines dans un même réseau sera important, même si on aura peut de réseau
- plus les poids fort sont nombreux, on aura alors peut de machines connectable pour chacun de ces réseau, même s'il sont plus nombreux

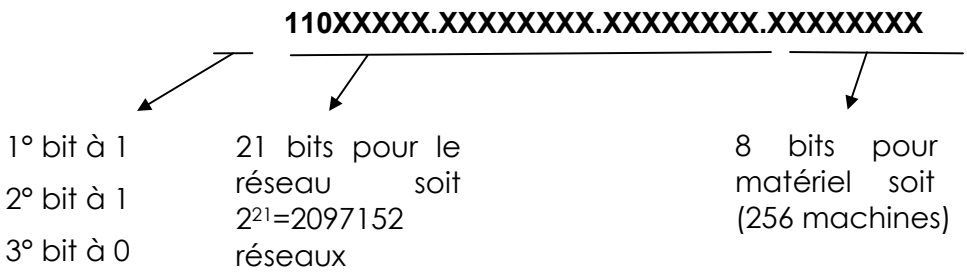
Réseau de **Classe A** : (commence par **1** à **127**)



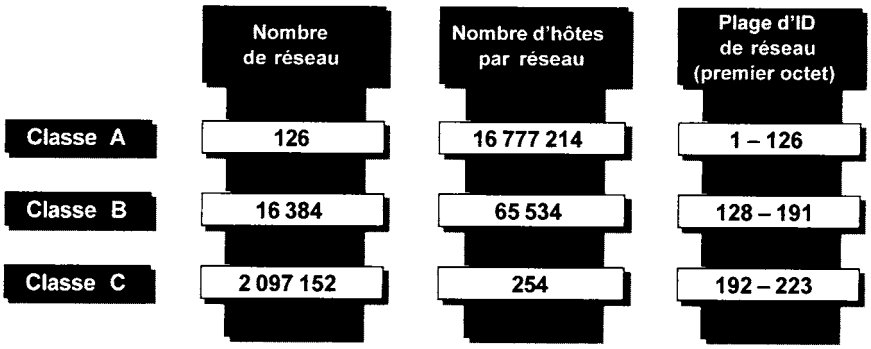
Réseau de **Classe B** : (commence par **128** à **191**)



Réseau de **Classe C** : (commence par **192** à **223**)



Soit en résumant



Avec quelques règles supplémentaires :

- l'**ID réseau 127**, est réservée pour les tests
- Un **ID réseau** composé exclusivement de 1 ou de 0 n'est jamais attribué
- Un **ID hôte** composé exclusivement de 1 ou de 0 n'est jamais attribué
- La valeur **255.255.255.255** correspond à une diffusion générale (**Broadcast**)

Adresses IP Privées :

Il est normal d'assigner des adresses globalement uniques à toutes les machines qui utilisent TCP/IP. Pour pouvoir étendre la durée de vie de l'adressage IPv4, les organismes d'enregistrement demandent plus de justifications qu'auparavant, rendant la tâche plus difficile à des organisations pour acquérir des adrresses supplémentaire [RFC1466].

Les machines de l'entreprise qui utilisent TCP/IP peuvent être divisées en 3 catégories:

- **Catégorie 1 :** les machines qui n'ont pas besoin d'accéder à des machines d'autres entreprises ou à l'Internet dans son ensemble. Les machines de cette catégorie peuvent utiliser des adresses IP qui sont uniques dans l'entreprise, mais qui peuvent être ambigues entre différentes entreprises.
- **Catégorie 2 :** les machines qui ont besoin d'accéder à un nombre limité de services extérieurs (ex: E-Mail, WWW, FTP) qui peuvent êtres servis par des passerelles applicatives. Pour beaucoup de machines dans cette catégorie, un accès non restreint (fourni par la connectivité IP) n'est pas forcément nécessaire et même quelque fois non désiré pour des raisons de sécurité. Pour les mêmes raisons que pour les machines de la première catégorie, de telles machines peuvent utiliser des adresses IP uniques dans l'entreprise, mais qui peuvent être ambigues entre différentes entreprises.

- **Catégorie 3 :** les machines qui ont besoin d'un accès réseau à l'extérieur de l'entreprise (fourni par la connectivité IP). Les machines de cette dernière catégorie ont besoin d'une adresse unique sur tout l'Internet.

On parle pour les machines des catégories 1 et 2 comme de machines "privées", et pour les machines de la 3eme catégorie comme des machines "publiques".

L'Autorité d'Affectation de Numéros sur Internet a réservé les 3 bloc suivant dans l'espace d'adressage pour des réseaux internes RFC 1918:

le premier bloc n'est rien d'autre qu'une classe A n° **10.**

10.0.0.0 - 10.255.255.255 (10/8 prefix)

le second, un ensemble de 16 classes B contiguës entre n° **172.16. et 172.31.**

172.16.0.0 - 172.31.255.255 (172.16/12 prefix)

N.B: pour **172.16.0.** le premier hote dispo sera .0.1 (éviter N° à 0 totalement)

et le troisième, un ensemble de 256 classes C de n° **192.168.0. à 192.168.255.**

192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

N.B: pour **192.168.0.** le premier hote dispo sera .1 (éviter N° à 0 totalement)

Les **machines privées** peuvent communiquer avec toutes les autres machines de l'entreprise, à la fois publiques et privées. Néanmoins, elles ne peuvent avoir de connectivité IP avec une machine à l'extérieur de l'entreprise. Même si elles n'ont pas de connectivité IP vers l'extérieur, les machines privées peuvent toutefois avoir accès à des services extérieurs grâce à des passerelles (ex passerelles applicatives).

Pour connecter un réseau utilisant des adresse privées RFC 1918 sur internet, il est nécessaire de prévoir un système de traduction d'adresse (Network Address Translator) ou un système de proxy

Les **machines publiques** peuvent communiquer avec d'autres machines privées ou publiques à l'intérieur de l'entreprise et possèdent une connectivité IP avec les machines publiques extérieures à l'entreprise. Les machines publiques n'ont pas de connectivité avec des machines privées d'autres entreprises.

MASQUE DE SOUS-RESEAU

Subdivision de réseau :

Très fréquemment on constitue un réseau à partir de segments ou brins interconnectées entre eux via des routeurs…

Les avantages à avoir un réseau bien segmenté sont nombreux :

- Différentes techniques de réseau peuvent être mélangées (Ethernet et Token-Ring par exemple…)
- Les collisions sont limitées car les diffusions générales sont limitées au segment local
- Extension à un nombre pratiquement infini d'hôtes

Masque de sous-réseau :

Le **masque de sous-réseau** permet de définir le découpage entre les bits de l'adresse qui servent à définir l'adresse de réseau, et ceux servant à définir l'adresse de la machine

En effet via un système de **ET bit à bit**, le **masque de sous-réseau** permet de distinguer l'**ID réseau** à partir de l'**Id hôte**, et par conséquent permet à **TCP/IP** de savoir si une **adresse IP** donnée se trouve sur le **réseau local** ou sur un **réseau distant**

Masque par défaut :

Ainsi dans des masques standards, tous les bits correspondants à l'**ID réseau** sont à 1, tous les bits correspondants à l'**ID hôte** sont à 0

Classe d'adresse	Bits utilisés pour le masque de sous-réseau				Notation décimale à points
Classe A	11111111	00000000	00000000	00000000	255.0.0.0
Classe B	11111111	11111111	00000000	00000000	255.255.0.0
Classe C	11111111	11111111	11111111	00000000	255.255.255.0

Masque personnalisé :

L'objectif est ici d'obtenir des adresses d'**ID réseau** et d'**Id hôte** groupées de manière un peu différente par rapport aux classes standardisées A-B-C qui servent de cadre

Pour définir des sous-réseaux personnalisé, il est nécessaire de bien définir deux points :

- Combien de réseau veut on gérer à l'intérieurs de la plage d'adresse attribuée
  N.B: en prévoyant un évolution future raisonnable !
- Combien d'hôtes maximum veut on gérer a l'intérieur d'un sous-réseau
  N.B: en prévoyant un évolution future raisonnable !

Puis travailler de la manière suivante :

- Définir le masque de sous-réseau qui donne le nombre de sous-réseau et d'hôte par sous-réseau voulu
- Déterminer les **ID réseaux** possibles à utiliser
  N.B: (cf tables page 25 pour savoir combien il y en a)
- Déterminer les **ID hôtes** possibles à utiliser
  N.B: (cf tables page 25 pour savoir combien il y en a)

Définir un masque de sous-réseau

On l'a dit, l'**ID réseau** se calcule en regardant le nombre de 1 du masque de sous-réseau.

Pour augmenter le nombre d'**ID réseau**, il faut ajouter des bits au masque de sous-réseau (Bien sûr si on augment le nombre d'**ID réseau**, on diminue le nombre d'**ID hôte**…)

De combien de bit faut-il augmenter le masque de sous-réseau ?

comme on travaille avec les puissances de 2, on augmente les combinaisons de 2 ^ nb bits ajoutés

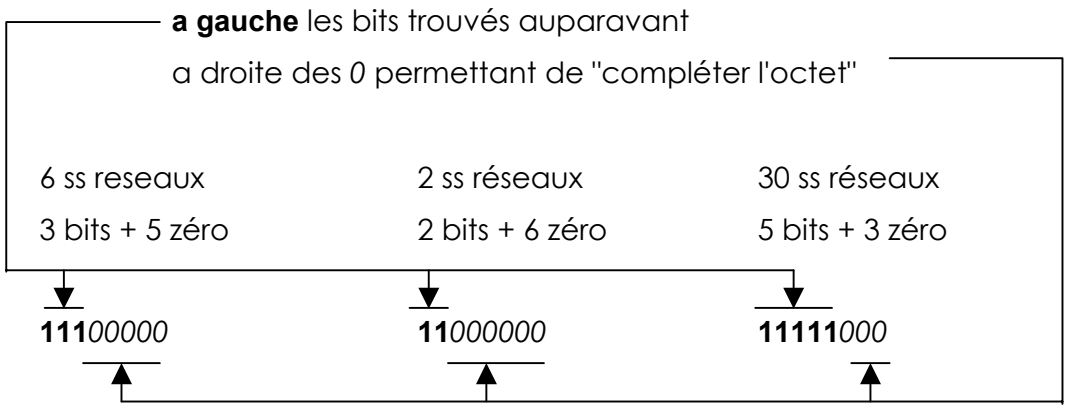
soit	1 bit	2 sous-réseaux
	2 bit	4 sous réseaux
	3 bits	8 sous réseaux
	4 bits	16 sous réseaux
	5 bits	32 sous réseaux
	x bits	2^x. sous-réseaux

mais rappelez vous, les adresse ne contenant que des 0 ou que des 1 ne sont pas autorisées, par conséquent il faut enlever les 2 adresses extrêmes possibles…ce qui nous donne

soit	**1 bit**	**impossible**	(2-2=0)
	2 bit	**2 sous réseaux**	(4-2)
	3 bits	**6 sous réseaux**	(8-2)
	4 bits	**14 sous réseaux**	(16-2)
	5 bits	**30 sous réseaux**	(32-2)
	x bits	(2^x)-2 sous-réseaux	

Comment calculer le nouveau masque de sous-réseau de mes réseaux?

1. Un foi trouvé le nombre de bits me permettant d'obtenir le nombre de sous-réseaux voulu, je dois créer un octet avec :



2. puis le convertir en décimal

111*00000*	**11***000000*	**11111***000*
=128+64+32	=128+64	=128+64+32+16+8
224	192	248

3. et remplacer dans la masque par défaut de ma classe d'adresse, le premier 0 par ce nombre…

6 ss reseaux	2 ss réseaux	30 ss réseaux
224	192	248

si l'adresse est de classe **A** cela donne par rapport au masque **255.0.0.0**

255.224.0.0	**255**.192.0.0	**255**.248.0.0

si l'adresse est de classe **B** cela donne par rapport au masque **255.255.0.0**

255.255.224.0	**255.255**.192.0	**255.255**.248.0

si l'adresse est de classe **C** cela donne rapport au masque **255.255.255.0**

255.255.255.224	**255.255.255**.192	**255.255.255**.248

Comment calculer les ID réseau de mes réseaux?

1. Recenser toutes les combinaisons possibles (en excluant donc celles n'ayant que des 1 ou des 0) de bits ajoutées au masque de sous-réseau précédemment et les convertir en décimal:

6 ss réseaux	2 ss réseaux	30 ss réseaux
(111)*00000*	**(11)***000000*	trop long !
110*00000*	**10***000000*	
101*00000*	**01***000000*	
011*00000*	**(00)***000000*	
100*00000*		
010*00000*		
001*00000*		
(000)*00000*		

2. Les convertir en décimal:

6 ss réseaux	2 ss réseaux
192	64
160	32
128	
96	
64	
32	

3. Ajouter ces valeurs a l'**ID réseau** d'origine:

Comment calculer les ID hôtes disponibles dans mes réseaux?

Les **ID hôte** commencent par la valeur .001 dans le dernier octet et augmentent 1 par 1 jusqu'à atteindre la valeur ID de sous-réseau du réseau suivant, -1

Bien sûr le dernier octet lui aussi ne peut pas être égal à 0 ou 255.

voilà le nombre de sous-réseau utilisables, avec le nombre d'hôte possible pour un masque de sous-réseau donné, et ce pour les

Adresses de classe A:

Bits supplémentaires (n)	*Nombre maximum de sous-réseaux (2^n-2)*	*Nombre maximum d'hôtes par sous-réseau (2^(24-n)-2)*	*Masque de sous-réseau*
0	0	16 777 214	255.0.0.0
1	invalide	invalide	invalide
2	2	4 194 302	255.192.0.0
3	6	2 097 150	255.224.0.0
4	14	1 048 574	255.240.0.0
5	30	524 286	255.248.0.0
6	62	262 142	255.252.0.0
7	126	131 070	255.254.0.0
8	254	65 534	255.255.0.0

Adresses de classe B:

Bits supplémentaires (n)	*Nombre maximum de sous-réseaux (2^n-2)*	*Nombre maximum d'hôtes par sous-réseau (2^(16-n)-2)*	*Masque de sous-réseau*
0	0	65 534	255.255.0.0
1	invalide	invalide	invalide
2	2	16 382	255.255.192.0
3	6	8 190	255.255.224.0
4	14	4 094	255.255.240.0
5	30	2 046	255.255.248.0
6	62	1 022	255.255.252.0
7	126	510	255.255.254.0
8	254	254	255.255.255.0

Adresses de classe C:

Bits supplémentaires (n)	*Nombre maximum de sous-réseaux (2^n-2)*	*Nombre maximum d'hôtes par sous-réseau (2^(8-n)-2)*	*Masque de sous-réseau*
0	0	254	255.255.255.0
1	invalide	invalide	invalide
2	2	62	255.255.255.192
3	6	30	255.255.255.224
4	14	14	255.255.255.240
5	30	6	255.255.255.248
6	62	2	255.255.255.252
7	invalide	invalide	255.255.255.254
8	invalide	invalide	255.255.255.255

Exemple 6 sous réseaux de 30 postes :

Si on veut **6 sous réseaux** comportant chacun 30 machines maximum, on pourra prendre alors comme masque de sous réseau **255.255.255.224**

- **Id réseau**
  pour trouver les Id réseau je dois trouver toutes les combinaisons de **3 bits** de 111 à 000 en laissant tomber les valeurs n'ayant que des 0 ou que des 1 (non autorisée).J'obtient 110-101-011-100-010-001 soit en décimal 192-160-128-96-64-32.

  que je rajoute à mon Id réseau d'origine 192.168.1.xx soit donc les Id réseau suivantes :

  192.168.1.**192** 192.168.1.**160** 192.168.1.**128** 192.168.1.**96**

  192.168.1.**64** 192.168.1.**32**

- Id hôte valide
  un petit calcul nous donne :

sous-réseau	**1° adresse IP**	**dernière adresse IP**
192.168.1.**32**	192.168.1.33	192.168.1.63
192.168.1.**64**	192.168.1.65	192.168.1.95
192.168.1.**96**	192.168.1.97	192.168.1.127
192.168.1.**128**	192.168.1.129	192.168.1.159
192.168.1.**160**	192.168.1.161	192.168.1.191
192.168.1.**192**	192.168.1.193	192.168.1.223

MASQUE DE SUR-RESEAU

Objectif du sur-réseau :

La question ici n'est pas de délimiter des sous-réseaux (donc des sous-ensemble de moins de 255 machines pour une classe C par exemple), **mais plutôt de faire en sorte que l'on puisse adresser "ensemble" plus de 255 machines, mais en restant avec des adresses de classe C ! (par exemple)**

Ainsi imaginons un réseau constitué au départ d'une centaine de machines dont les adresses IP privées ont étés définies en classe C, par exemple sur les adresses de base suivantes: 192.168.25.1 à 192.168.25.100. Ce réseau grandit, et voit le nombre des machines dépasser les 255 postes, que faire ?

classiquement on peut agir de différentes manières :

- Fractionner le réseau en plusieurs zones distinctes, et les relier par un (des) routeurs...
- Passer à des adresse de type Classe B, par exemple 172.16.0.1 à 172.16.1.xxx avec un masque par défaut de 255.255.0.0
- Augmenter la taille du masque par défaut, de 255.255.255 à .... c'est du sur-réseau !

Principe :

l'agrégation de plage d'adresse, ou "**super-netting**" s'effectue en modifiant le masque de sous-réseau. La modification, dépends du nombre (puissance de 2)de classe que l'on souhaite "agréger" :

Nombre Classes à agréger	Masque sous-réseau	nombre de Hosts maximum disponibles
1	255.255.255.0	256
2	255.255.254.0	512
4	255.255.252.0	1024
8	255.255.248.0	2048
16	255.255.240.0	4096
32	255.255.224.0	8192
64	255.255.192.0	16384
128	255.255.128.0	32768
256	255.255.0.0	65536

Dans notre cas pour adresser un maximum de 1024 machines, il faut agréger 4 classes par exemple, et comme masque prendre la valeur 255.255.252.0,

Ce qui permet d'avoir en fait 256/4 plages adressables de 1024 machines chacune, suivant le tableau ci-dessous :

N° plage	Adresse Début	Adresse Fin	Masque	nb Hosts maxi
1	192.168.0.0	192.168.3.255	255.255.252.0	1024
2	192.168.4.0	192.168.7.255	255.255.252.0	1024
3	192.168.8.0	192.168.11.255	255.255.252.0	1024
4	192.168.12.0	192.168.15.255	255.255.252.0	1024
5	192.168.16.0	192.168.19.255	255.255.252.0	1024
...x...	192.168. (x*4)-4 .0	192.168. (x*4)-1 .255	255.255.252.0	1024
64	192.168.252.0	192.168.255.255	255.255.252.0	1024

N.B: les adresses faisant partie du même N° plage sont vues comme faisant partie d'une même réseau, donc ne nécessitent pas de routage entre elles

N.B: les adresses ne faisant pas partie du même N° plage sont vues comme faisant partie de réseaux différents, donc nécessitent un routage entre elles

LE ROUTAGE TCP/IP

Notion de routeur :

De manière générale, une machine peut communiquer uniquement par défaut avec une autre machine de son réseau local, c'est à dire une autre machine faisant partie de son sous-réseau, encore appelé domaine de collision.

Que ce sous-réseau soit obtenu par l'application d'un masque de sous-réseau par défaut isolant des classes A, B ou C complète, ou qu'il soit obtenu par l'application d'un masque de sous-réseau personnalisé modifiant l'étendue par défaut des ID réseau et des ID hôtes, l'idée est la même :

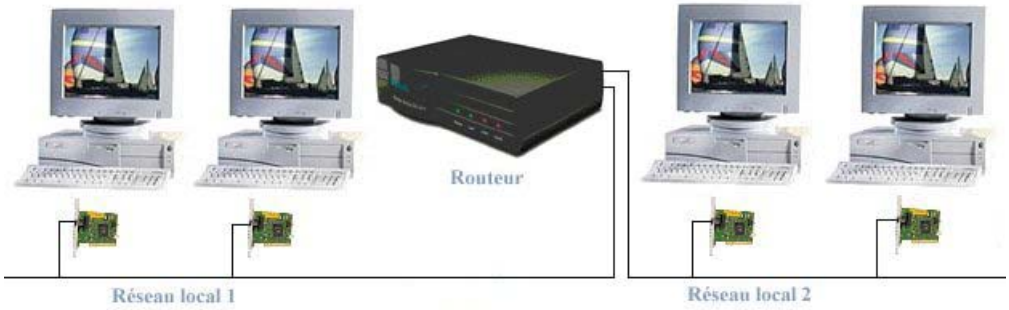
IP compare l'ID de sous-réseau de l'adresse IP que l'on cherche à joindre à l'ID de sous-réseau du réseau local dans lequel il se trouve :

- **Si les deux ID correspondent** : IP peut chercher localement la machine
- **Si les deux ID ne correspondent pas** : IP envoi la trame vers un équipement ou il peut être routé

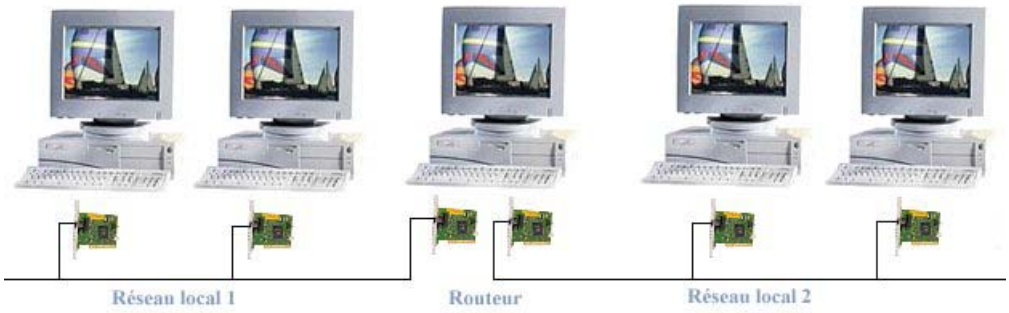
Lorsque des machines sont interconnectées en plusieurs sous-réseaux, elles doivent toutes avoir comme paramétrage l'adresse IP d'une passerelle - **routeur** par défaut

Une adresse IP différente est assignée a chaque carte sur chaque sous-réseau, permettant à ce **routeur** de faire partie de plusieurs réseaux différents. On parle alors aussi **d'hôte multi-résident**.

Un routeur peut être soit un matériel spécifique,



Soit une fonction assurée par une station de travail possédant au moins deux interfaces réseaux, et une application pour le routage.

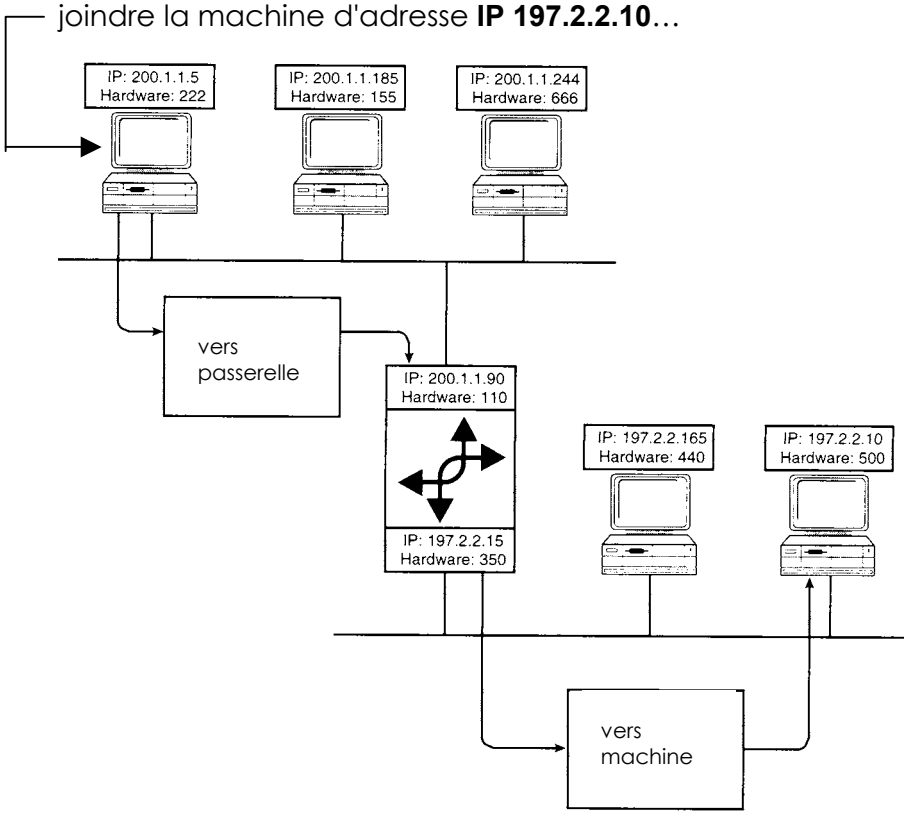


N.B: Toute machine windows NT peut faire office de routeur à partir du moment ou elle dispose d'autant de cartes réseaux que de sous-réseaux auxquels elle souhaite être rattachée.

Routage de base :

Dans la situation la plus simple, on relie **deux sous-réseaux** par **un routeur** ayant donc **deux cartes réseaux** et **deux adresses IP** dans chaque sous-réseau auxquels il appartient :

Dans l'exemple ci-dessous, la machine d'adresse **IP 200.1.1.5** essaye de joindre la machine d'adresse **IP 197.2.2.10**…



Le fonctionnement est le suivant :

1. IP se rends compte que l'adresse de destination n'est pas une adresse locale (ID réseau cherchée **197.2.2.0** différente de ID réseau locale **200.1.1.0**)
2. IP transmet alors le paquet à la passerelle par défaut
3. IP sur le routeur détermine que l'ID cherchée est **197.2.2.0**, comme le routeur possède une carte paramétrée sur ce réseau il l'utilise pour envoyer ce paquet...
4. IP sur la machine de destination récupère le paquet qui lui est destiné…

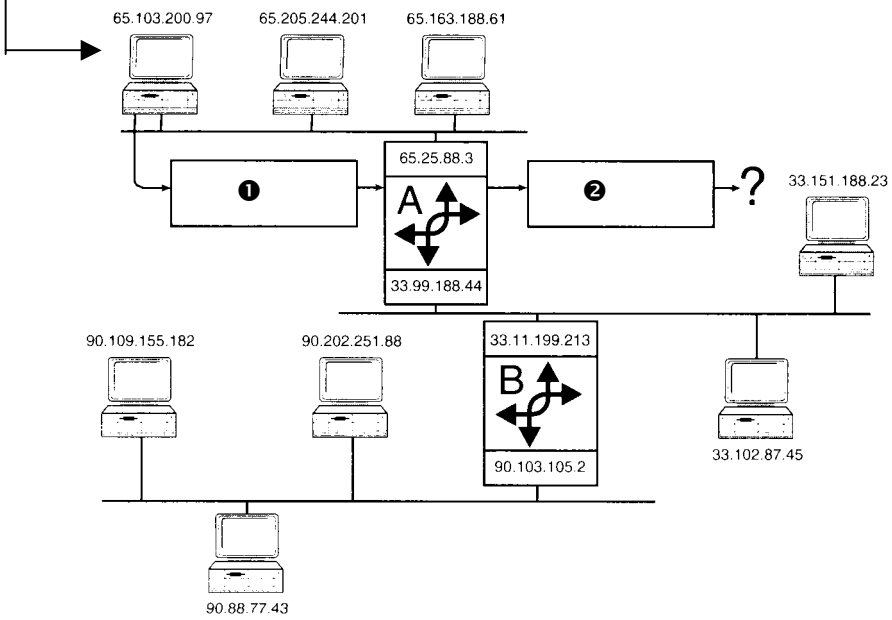
N.B: Par défaut, les tables de routage sur Windows NT ne contiennent que des informations sur les sous-réseaux sur lesquels le routeur est directement connecté.

Ce qui est sans doute un peut limitatif…

Routage complexe :

Dans une situation plus complexe, on relie **trois sous-réseaux par deux routeurs** ayant chacun **deux cartes réseaux** et **deux adresses IP** dans chaque sous-réseaux auxquels ils appartiennent :

Dans l'exemple ci-dessous, la machine d'adresse **IP 65.103.200.97** essaye de joindre la machine d'adresse **IP 90.109.155.182**…



Le fonctionnement est le suivant :

1. IP se rends compte que l'adresse de destination n'est pas une adresse locale (ID réseau cherchée **90.0.0.0** différente de ID réseau locale **65.0.0.0**), IP transmet alors le paquet à la passerelle par défaut
2. IP sur le routeur détermine que l'ID cherchée est **90.0.0.0**, mais comme le routeur **ne possède pas** une carte paramétrée sur ce réseau il ne sait pas où envoyer ce paquet...

N.B: Par défaut, les tables de routage sur Windows NT ne contiennent que des informations sur les sous-réseaux sur lesquels le routeur est directement connecté, ce qui fait que ici le paquet ne saurait être routé vers le réseau 90.0.0.0

Table de Routage :

Dans une situation plus complexe, il est nécessaire de configurer un routeur avec une table de routage qui contient des informations destinées a router des paquets vers d'autres routeurs lorsque l'on ne sait pas directement qui pourrait les prendre en charge.

Dans notre exemple il faudrait indiquer à notre premier routeur que lorsqu'il reçoit des paquets à destination d'un réseau 90 il doit les router vers le réseau 33.0.0.0

Dans notre exemple toujours, le cas n'étant que peux compliqué, on pouurait s'en sortir en paramétrant comme passerelle par défaut de ce routeur, l'adresse du deuxième routeur…

Cette méthode est limité au cas ou l'on a que 2 routeurs …

D'une manière plus générale il va falloir configurer ce que l'on appelle une table de routage

Routage statique :

On appelle **routage statique** un routage qui est mis à jour manuellement sur chaque routeur par l'administrateur

La commande permettant de créer et maintenir une table de routage est la commande

route print

Routage dynamique :

On appelle **routage dynamique** un routage qui est mis à jour automatiquement sur chaque routeur par échange d'information entre les routeurs…

N.B: Windows NT ne dispose pas de cette capacité à travers le protocole RIP

INSTALLER TCP/IP

Installer la carte réseau :

le protocole TCP/IP ne peut être installé que si une carte réseau est présente dans le système

les paramètres éventuellement à savoir pour travailler avec une carte réseau sont les suivants :

IRQ sur Compatibles Intel :

Les IRQ permettent à un périphérique d'interrompre le processeur afin d'effectuer un traitement quelconque. Les XT ne possédaient que 6 lignes d'IRQ (IRQ2 - IRQ7) sur le Bus de donnée, les IRQ 0 et IRQ1 existaient mais se trouvaient réservées. Les AT ont apporté 8 lignes d'IRQ supplémentaires (IRQ8 - IRQ15). Le contrôleur d'interruption supplémentaire est connecté en cascade sur la broche IRQ2 du contrôleur existant, d'ou l'indisponibilité de l'IRQ2. Chaque périphérique utilise une seule IRQ. mais les Bus EISA ou PCI autorisent le partage d'une même IRQ entre deux périphériques.

N° IRQ	Libéllé	Notes
0	Système temps réelr	inutilisable (système)
1	gestion du Clavier	inutilisable (système)
2	branchement IRQ9 en cascade	inutilisable (système)
3	utilisé pour gérer les ports série COM2, COM4	libre
4	port série COM1, COM3	utilisé par défaut
5	utilisé pour gérer le port parallèle LPT2	libre
6	contrôleur de disquette	utilisé par défaut
7	port série LPT1	utilisé par défaut
8	Horloge temps réel	inutilisable (système)
9	gestion écran EGA/VGA	inutilisable (système)
10	-	libre
11	-	libre
12	si PS2 IBM gestion souris	libre
13	gestion coprocesseur mathématique	utilisé par défaut
14	contrôleur de disque dur	utilisé par défaut
15	-	libre

IRQ fréquemment demandée pour carte réseau = 5.

E/S Adresse Entrée/Sortie :

Il s'agit de spécifier le canal par lequel passe l'information entre le périphérique de l'ordinateur (comme la carte réseau) et son unité centrale. L' UC considère le port de base comme une adresse.

Chaque périphérique du système doit avoir une adresse de base différente, deux périphériques ne peuvent absolument pas partager la même adresse.

Voici la liste de quelques adresses habituelles

Port	***Périphérique***	***Port***	***Périphérique***
200 à 20F	Port jeux	300 à 30F	
210 à 21F		310 à 31F	
220 à 22F		320 à 32F	Contrôleur de disque dur (pour PS/2 modèle 30)
230 à 23F	Souris à bus	330 à 33F	
240 à 24F		340 à 34F	
250 à 25F		350 à 35F	
260 à 26F		360 à 36F	
270 à 27F	LPT3	370 à 37F	LPT2
280 à 28F		380 à 38F	
290 à 29F		390 à 39F	
2A0 à 2AF		3A0 à 3AF	
2B0 à 2BF		3B0 à 3BF	LPT1
2C0 à 2CF		3C0 à 3CF	EGA/VGA
2D0 à 2DF		3D0 à 3DF	CGA/MCGA (également EGA/VGA, en modes vidéo couleur)
2E0 à 2EF		3E0 à 3EF	
2F0 à 2FF	COM2	3F0 à 3FF	Contrôleur de lecteur de disquette; COM1

Adresse de la mémoire de base

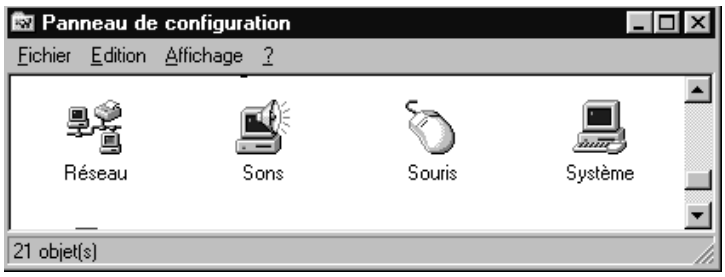
Valeur fréquemment conseillée = 210h ou 280h ou 300h

Installation carte réseau sous Windows 95-98 :

Il faut lancer le panneau de configuration via le menu

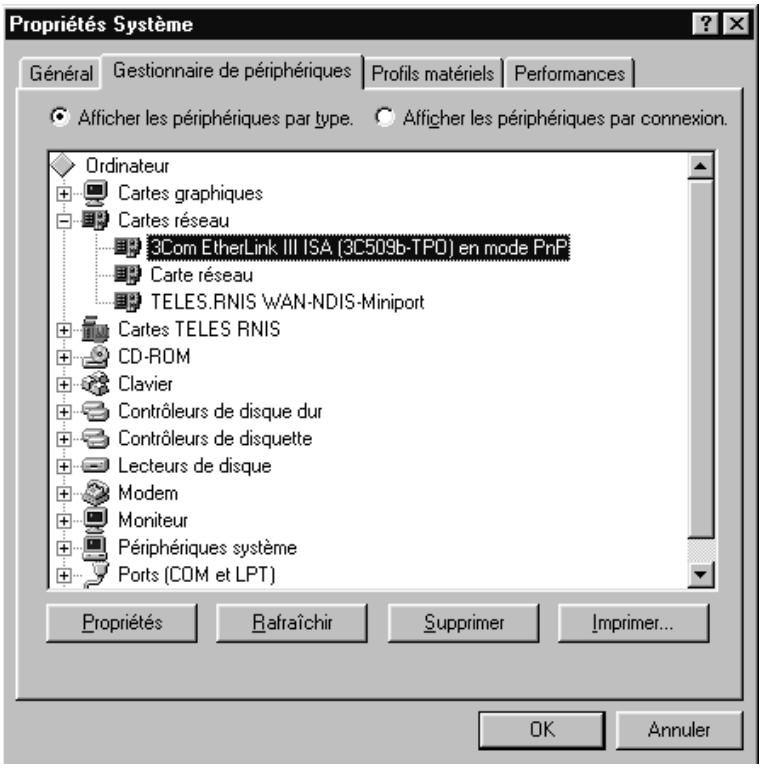
Démarrer / Paramètres / Panneau de Configuration

puis demander système

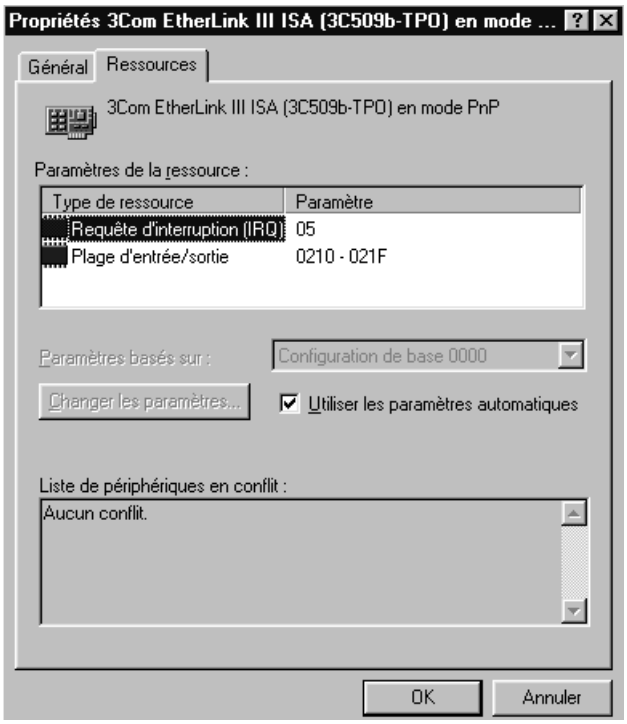


Ou bien faire un clic avec le bouton de droite sur **Poste de travail**

Dans la boite de dialogue qui s'affiche on choisit l'onglet "**Gestionnaire de périphérique**" et on cherche la carte réseau à configurer



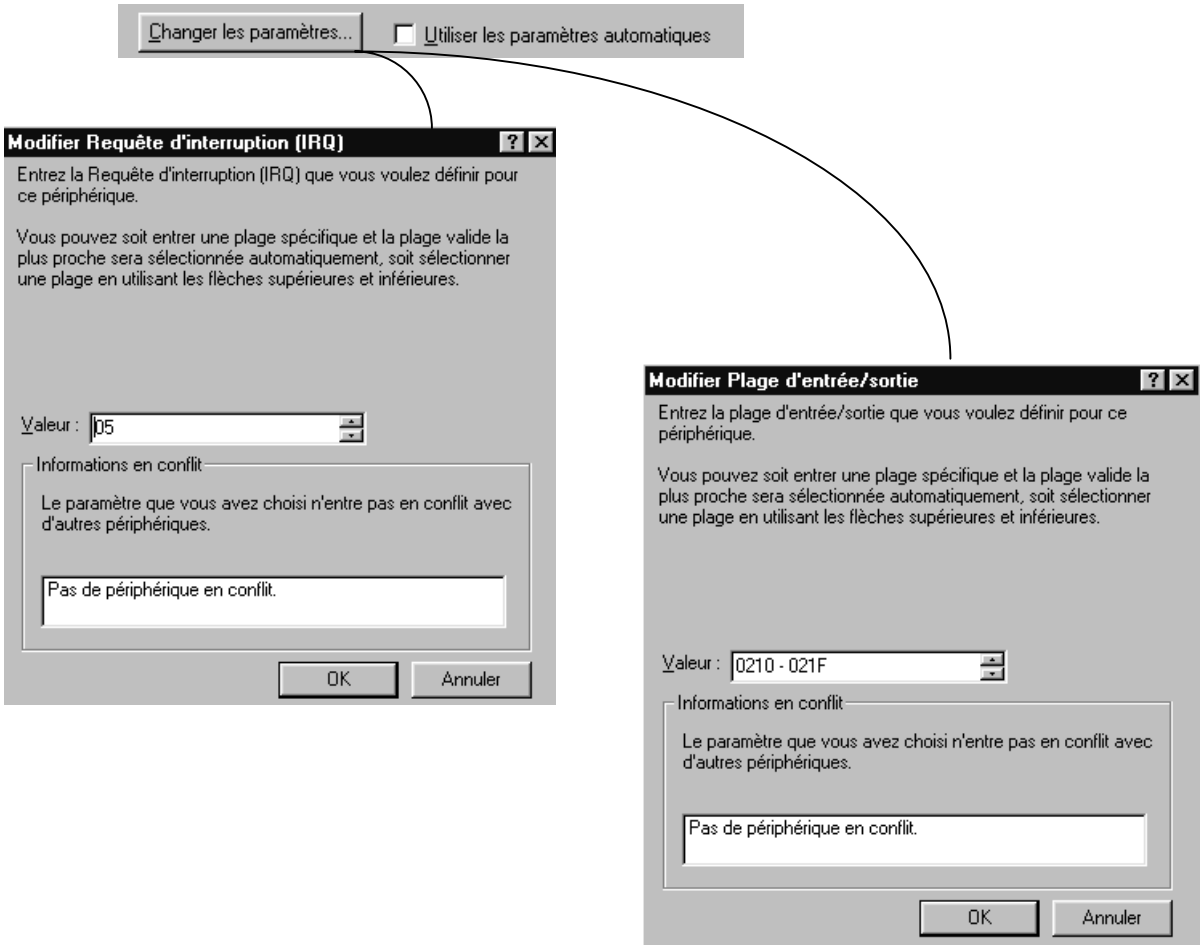
Il suffit ensuite de demande Propriété pour accéder au paramétrage



Windows 95 est "plug and play" c''est à dire qu'il est capable de paramétrer la carte tout seul, mais parfois cela peut poser problème...

On peut donc demander de dévalider le paramétrage automatique et donner les valeurs manuellement

Dévalider le paramétrage automatique



Paramétrage TCP/IP Sous Windows 95-98 :

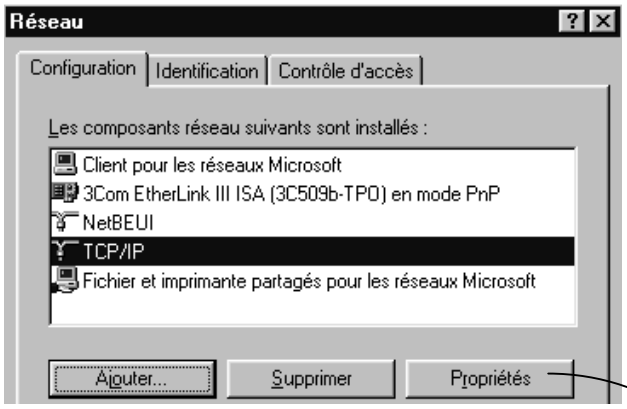
Il faut lancer le panneau de configuration via le menu

Démarrer / Paramètres / Panneau de Configuration

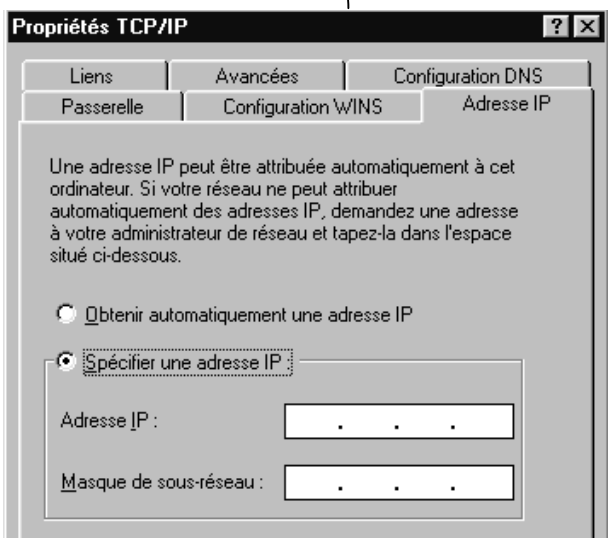
puis demander Réseau

Ou bien faire un clic avec le bouton de droite sur **Voisinage Réseau**

Dans la boite de dialogue qui s'affiche on choisit l'onglet "**Configuration**" et on cherche le Protocole TCP/IP à configurer



En général tout est à désactiver sauf l'onglet Adresse IP dans lequel il faut indiquer si on récupère une adresse via un serveur DHCP ou non



Installation sous Windows NT Station ou Server :

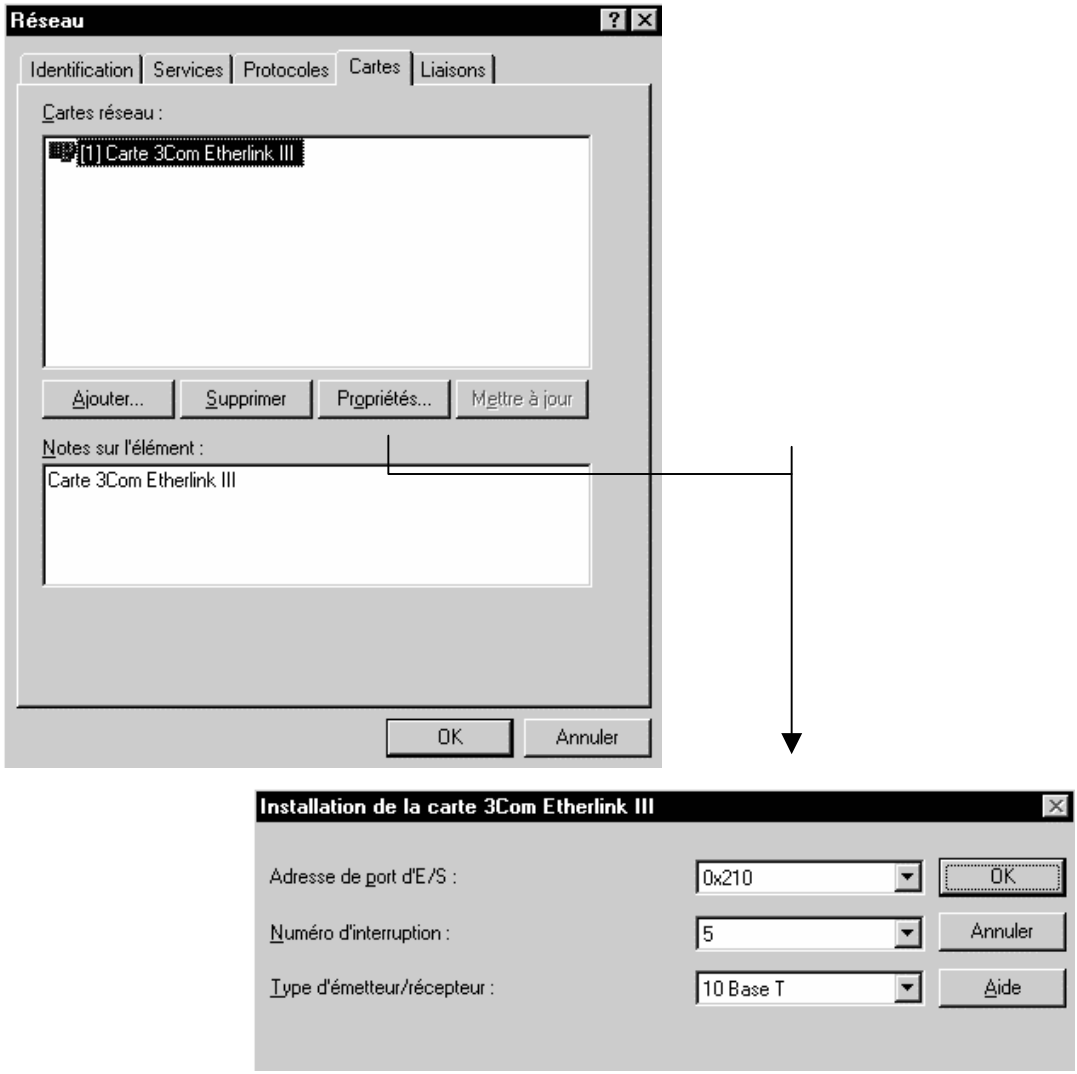
Il faut lancer le panneau de configuration via le menu

Démarrer / Paramètres / Panneau de Configuration

puis demander réseau

Ou bien faire un clic avec le bouton de droite sur **Voisinage réseau**

Dans l'onglet Carte on choisit la carte à paramétrer et on demande **Ajouter**



Paramétrage TCP/IP Sous Windows NT :

Il faut lancer le panneau de configuration via le menu

Démarrer / Paramètres / Panneau de Configuration

puis demander Réseau

Ou bien faire un clic avec le bouton de droite sur **Voisinage Réseau**

Dans la boite de dialogue qui s'affiche on choisit l'onglet "**Protocole**" et on cherche le Protocole TCP/IP à configurer

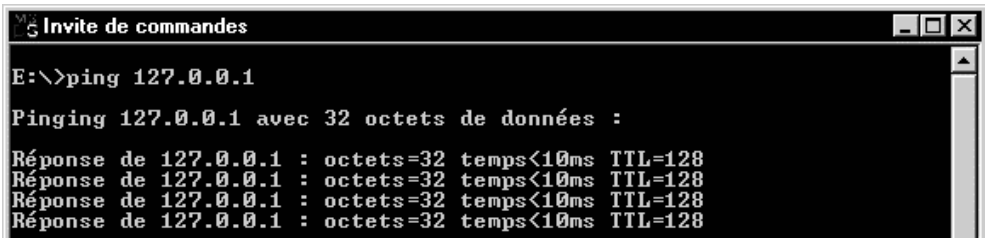
TESTER TCP/IP

ICMP et l'Utilitaire PING :

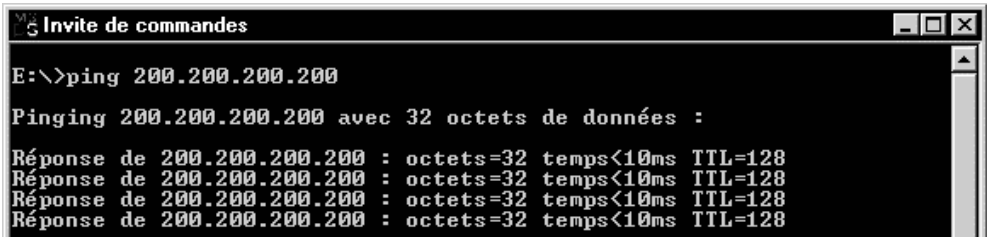
Les essais sur une configuration doivent se faire à bas niveau, au niveau DOS

Permet d'envoyer une trame IP de test vers une machine

En tapant **Ping 127.0.0.1** si on ne reçoit pas les 4 lignes suivantes, cela veut dire que la pile TCP/IP n'est pas installée correctement

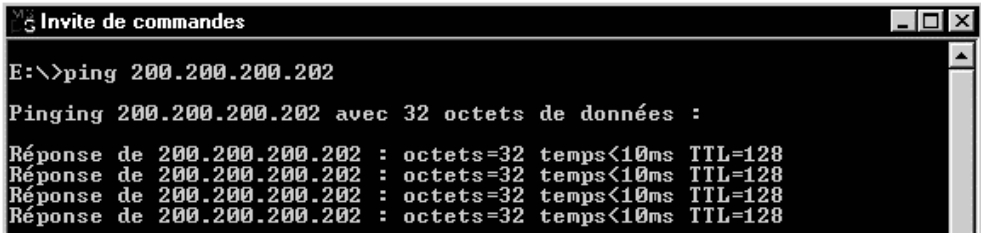


En tapant **Ping XX.XX.XX.XX** avec l'adresse de notre propre station depuis laquelle on « pingue », si on ne reçoit pas les 4 lignes suivantes, cela veut dire que l'adresse de la station est erronée

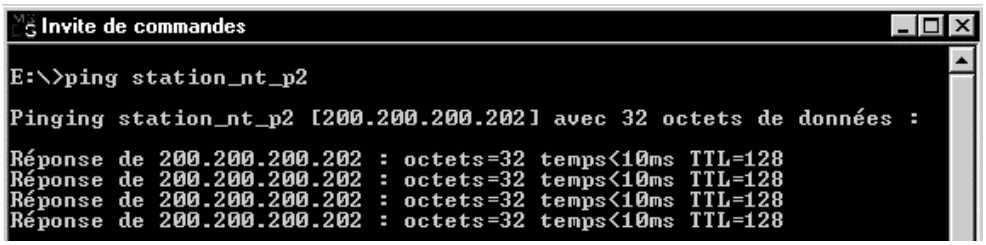


Jusqu'à présent on n'a rien envoyé sur le réseau proprement dit, on peut donc maintenant considérer que notre poste est correctement configuré sous TCP/IP, et on va essayer d'utiliser le réseau

En tapant **Ping XX.XX.XX.XX** avec l'adresse de la station que l'on souhaite atteindre, si on ne reçoit pas les 4 lignes suivantes, cela veut dire soit que l'adresse de la station est erronée soit que la connectique est mauvaise



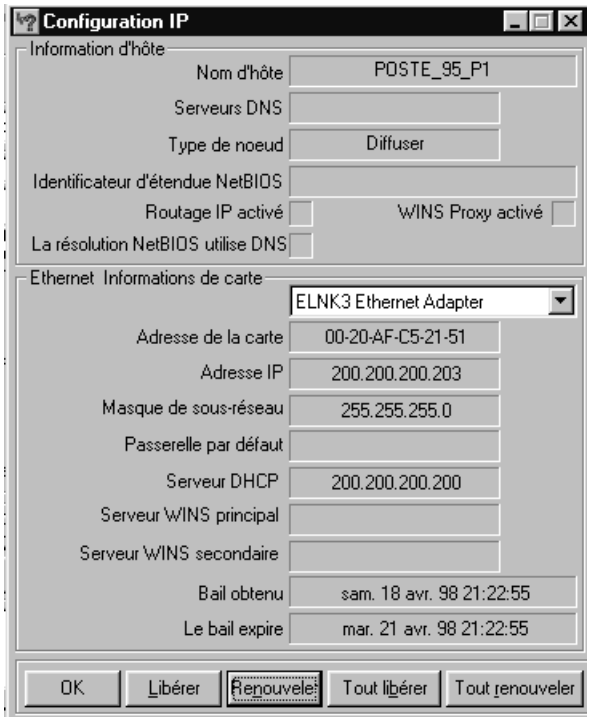
En tapant **Ping NOMSTATION** avec le nom de la station que l'on souhaite atteindre, si on ne reçoit pas les 4 lignes suivantes, cela veut dire que le nom de la station est erroné



On peut aussi taper **Ping –a XX.XX.XX.XX** le nom de la station que l'on souhaite atteindre sera résolu en même temps que le retour de trame, ce qui permet de connaître en cas de problème le nom renvoyé par la machine...

Winipcfg.exe :

Sous Wind95 on fera **Winipcfg.exe** depuis une boite dos



Ipconfig.exe :

Sous Windows NT on fera **Ipconfig.exe** depuis une boite dos

Sous NT2000 les paramètres d'appels sont les suivants

```
ipconfig [/? ¦ /all ¦ /release [carte] ¦ /renew [carte]
         ¦ /flushdns ¦ /registerdns
```

```
carte       Nom complet ou combinaison avec '*' et '?' afin de correspondre,
            * correspond à tout caractère, ? correspond à un caractère.
Options
    /?              Affiche ce message d'aide.
    /all            Affiche toutes les informations de configuration.
    /release        Libère l'adresse IP pour la carte spécifiée.
    /renew          Renouvelle l'adresse IP pour la carte spécifiée.
    /flushdns       Vide le cache de la résolution DNS.
    /registerdns    Actualise tous les baux DHCP et réinscrit les noms DNS.
    /displaydns     Affiche le contenu du cache de la résolution DNS.
```

Sous NT 4.0 les paramètres d'appels sont uniquement

```
/all        Affiche l'ensemble des informations de configuration.
/release    Libère l'adresse IP de la carte spécifiée.
/renew      Renouvelle l'adresse IP de la carte spécifiée.
```

ARP et l'Utilitaire ARP :

Les essais sur une configuration peuvent se faire à bas niveau, directement au niveau d'un boite DOS

Permet de connaître l'adresse physique d'une machine

ARP est un protocole permettant la résolution adresse Ip => adresse physique

ARP est mis en oeuvre automatiquement lors de toute requête IP, et typiquement lors d'un ping....

En tapant **ARP -a** on affiche le contenu du cache dynamique actuellement en vigueur sur notre machine

sur une machine que l'on vient de démarrer, le cache est vide

```
C:\WIN98>arp -a
Aucune entrée ARP n'a été trouvée
```

après un coup de voisinage réseau, le master browse ayant répondu, le cache contient désormais sont adresse IP et son adresse physique

```
C:\WIN98>arp -a

Interface : 192.168.0.4 on Interface 0x2000003
  Adresse Internet      Adresse physique      Type
  192.168.0.1           00-50-04-52-09-14     dynamique
```

si on attend, le cache va finir par se vider et de nouveau on aura

```
C:\WIN98>arp -a
Aucune entrée ARP n'a été trouvée
```

Si on fait un ping sur une machine donnée, alors son "entrée" dans la table est effectuée des que la réponse est obtenue...

```
C:\WIN98>ping 192.168.0.3

Envoi d'une requête 'ping' sur 192.168.0.3 avec 32 octets de donnée

Réponse de 192.168.0.3 : octets=32 temps=1 ms TTL=128
Réponse de 192.168.0.3 : octets=32 temps<10 ms TTL=128
Réponse de 192.168.0.3 : octets=32 temps<10 ms TTL=128
Réponse de 192.168.0.3 : octets=32 temps<10 ms TTL=128

Statistiques Ping pour 192.168.0.3:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en milli-secondes :
    minimum = 0ms, maximum =  1ms, moyenne =  0ms
```

ce qui donne ensuite

```
C:\WIN98>arp -a

Interface : 192.168.0.4 on Interface 0x2000003
  Adresse Internet      Adresse physique      Type
  192.168.0.3           00-20-af-c4-6a-98     dynamique
```

un petit F5 (pour rafraîchir l'écran du voisinage réseau) provoquerait alors une autre entrée dans le cache ARP...etc, etc...

```
C:\WIN98>arp -a

Interface : 192.168.0.4 on Interface 0x2000003
  Adresse Internet      Adresse physique      Type
  192.168.0.1           00-50-04-52-09-14     dynamique
  192.168.0.3           00-20-af-c4-6a-98     dynamique
```

on peut vouloir rentrer une adresse de manière statique

En tapant **ARP –a XX.XX.XX.XX HH-HH-HH-HH-HH-HH**

```
C:\WIN98>arp -s 192.168.0.1 00-50-04-52-09-14
```

ce qui donnerait dans la table l'aspect suivant

```
C:\WIN98>arp -a

Interface : 192.168.0.4 on Interface 0x2000003
  Adresse Internet      Adresse physique      Type
  192.168.0.1           00-50-04-52-09-14     statique
```

cette entrée "statique" ne sera purgée de la table que lors d'un redémarage du poste . Si on souhaite la modifier il suffit de rentrer de nouveau une commande du type **arp -s**

NB : rappelez vous que les trames ARP ne passent pas les routeurs...

Netstat :

Les essais sur une configuration peuvent se faire à bas niveau, directement au niveau d'un boite DOS

Permet de connaître des statistiques sur les protocoles TCP-IP- UDP...

```
NETSTAT [-a] [-e] [-n] [-s] [-p proto] [-r] [intervalle]

  -a            Affiche toutes les connexions et les ports en écoute.
  -e            Affiche les statistiques Ethernet. Cette option peut être
                combiné avec l'option -s.
  -n            Affiche les adresses et numéros de port en format numérique.
  -p proto      Affiche les connexions du protocole spécifié par proto ; proto
                peut être TCP ou UDP. Utilisé avec l'option -s pour afficher
                les statistiques par protocole, proto peut être TCP, UDP ou IP.
  -r            Affiche la table de routage.
  -s            Affiche les statistiques par protocole. Par défaut, les
                statistiques sont affichées pour TCP, UDP et IP ; l'option -p
                peut être utilisée pour spécifier un seul de ces protocoles.
  intervalle    Affiche les statistiques sélectionnées  au délai spécifié par
                intervalle (en secondes). Appuyez sur Ctrl+C pour arrêter
                l'affichage des statistiques. Par défaut, NETSTAT n'affiche les
                informations sur la configuration qu'une seule fois.
```

Par exemple **netstat –a** ou **netstat –an** sont fort intéressant

```
C:\>netstat -a

Connexions actives

  Proto  Adresse locale         Adresse distante       Etat
  TCP    travail:epmap          travail:0              LISTENING
  TCP    travail:microsoft-ds   travail:0              LISTENING
  TCP    travail:1025           travail:0              LISTENING
  TCP    travail:netbios-ssn    travail:0              LISTENING
  TCP    travail:1031           smtp.wanadoo.fr:smtp   TIME_WAIT
  UDP    travail:epmap          *:*
  UDP    travail:microsoft-ds   *:*
  UDP    travail:1026           *:*
  UDP    travail:1027           *:*
  UDP    travail:netbios-ns     *:*
  UDP    travail:netbios-dgm    *:*
  UDP    travail:isakmp         *:*

C:\>netstat -an

Connexions actives

  Proto  Adresse locale         Adresse distante       Etat
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1025           0.0.0.0:0              LISTENING
  TCP    192.168.1.100:139      0.0.0.0:0              LISTENING
  UDP    0.0.0.0:135            *:*
  UDP    0.0.0.0:445            *:*
  UDP    0.0.0.0:1026           *:*
  UDP    0.0.0.0:1027           *:*
  UDP    192.168.1.100:137      *:*
  UDP    192.168.1.100:138      *:*
  UDP    192.168.1.100:500      *:*
```

Nbtstat :

Les essais sur une configuration peuvent se faire à bas niveau, directement au niveau d'un boite DOS

Permet de connaître des statistiques sur NETBIOS SUR TCP/IP

```
Displays protocol statistics and current TCP/IP connections using NBT(NetBIOS ov
er TCP/IP).
NBTSTAT [-a RemoteName] [-A IP address] [-c] [-n]
        [-r] [-R] [-s] [S] [interval] ]
  -a   (adapter status) Lists the remote machine's name table given its name
  -A   (Adapter status) Lists the remote machine's name table given its
                        IP address.
  -c   (cache)          Lists the remote name cache including the IP addresses
  -n   (names)          Lists local NetBIOS names.
  -r   (resolved)       Lists names resolved by broadcast and via WINS
  -R   (Reload)         Purges and reloads the remote cache name table
  -S   (Sessions)       Lists sessions table with the destination IP addresses
  -s   (sessions)       Lists sessions table converting destination IP
                        addresses to host names via the hosts file.

  RemoteName   Remote host machine name.
  IP address   Dotted decimal representation of the IP address.
  interval     Redisplays selected statistics, pausing interval seconds
               between each display. Press Ctrl+C to stop redisplaying
               statistics.
```

SERVICE DHCP

Objectif de DHCP :

Le protocole **DHCP** (Dynamic Host Configuration Protocol) centralise et gère l'attribution des informations de configuration TCP-IP en affectant automatiquement des adresses IP à des ordinateurs configurés pour utiliser DHCP. La mise en œuvre de DHCP élimine certains problèmes de configuration liés à la configuration manuelle de TCP-IP.

A chaque démarrage d'un client DHCP, ce dernier demande des informations d'adressage IP à un serveur DHCP, notamment :

- Une adresse IP
- Un masque de sous-réseau.
- Des valeurs facultatives, telles qu'une adresse de passerelle par défaut, une adresse DNS (Domain Name Server) et l'adresse du serveur de nom NetBios.

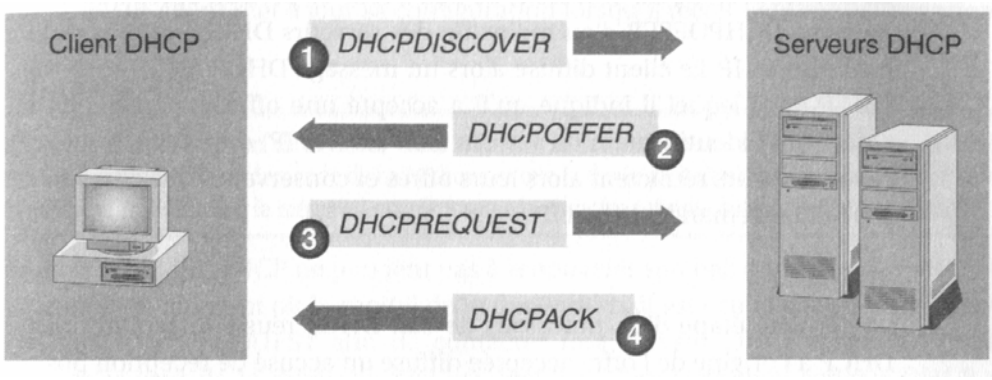
Lorsqu'un serveur DHCP reçoit une requête, il sélectionne des informations d'adressage IP dans une réserve d'adresses définie dans une base de données et les propose au client DHCP. Si le client les accepte, les informations d'adressage IP lui sont cédées sous la forme d'un bail d'une durée spécifique.

Si aucune informations d'adressage IP n'est disponible dans la réserve pour location au client, ce dernier ne peut pas initialiser TCP/IP

Remarque : Le protocole DHCP est défini dans les RFC 1533, 1534, 1541 et 1542.

Fonctionnement de DHCP :

Pour configurer un client DHCP, le protocole DHCP utilise un processus en quatre phases :



DHCPDISCOVER ou **"Demande de bail IP" :**

Le client ne disposant pas d'adresse IP et ne connaissant l'adresse IP d'aucun serveur, il utilise 0.0.0.0 comme adresse de source et 255.255.255.255 comme adresse de destination.

La demande de bail est envoyé au sein d'un message **DHCPDISCOVER**. Ce message contient également l'adresse matérielle et le nom d'ordinateur du client, afin que les serveurs DHCP puissent identifier l'émetteur de la requête.

Le processus de bail IP est utilisé lorsqu'une des situations suivantes se produit :

- TCP/IP est initialisé pour la première fois en tant que client DHCP.
- Le client demande une adresse IP spécifique qui lui est refusée. Il est possible que le serveur DHCP ait supprimé le bail.
- Le client disposait auparavant d'un bail d'adresse IP mais y a mis fin et en demande un nouveau.

DHCPOFFER ou **"Offre de bail IP" :**

Tous les serveurs DHCP qui ont reçu la demande et qui disposent d'une configuration valide vis-à-vis du client diffusent une proposition.

Le client ne disposant pas encore d'une adresse IP, l'envoi de la proposition s'effectue par diffusion sous forme de message **DHCPOFFER.**

Remarque : Lorsque aucun serveur DHCP n'est en ligne, le client DHCP attend une proposition pendant 1 seconde. S'il n'en reçoit aucune, il diffuse à nouveau la requête à trois reprises (selon des intervalles successifs de 9, 13 et 16 secondes). Si aucune proposition n'est reçue après quatre tentatives, le client essaie à nouveau toutes les 5 minutes.

DHCPREQUEST ou **"Selection de bail IP" :**

Après avoir reçu une proposition d'au moins un serveur DHCP, le client informe par diffusion tous les autres serveur DHCP de sa sélection, en acceptant la première proposition reçue.

La diffusion est envoyé dans un message **DHCPREQUEST** et comprend l'identificateur du serveur (AI) dont la proposition a été acceptée. Tous les autres serveurs DHCP retirent leur proposition afin que les adresses IP dont ils disposent restent disponibles pour la requête de bail IP suivante.

DHCPACK / NACK ou **"Accusé de réception de bail IP" :**

Le serveur DHCP dont la proposition est acceptée diffuse au client un accusé de réception stipulant la conclusion du bail, sous la forme d'un message **DHCPACK**. Ce message contient un bail valide pour une adresse IP et éventuellement d'autres informations de configurations.

Si un accusé de réception stipulant la non conclusion du bail (**DHCPNACK**) est diffusé (le client tente de souscrire le bail d'une adresse IP dont il disposait précédemment alors que cette adresse n'est plus disponible par exemle) le client retourne au processus de demande de bail IP.

"Renouvellement de bail IP" :

Tous le clients DHCP tentent de renouveler leur bail lorsqu'il atteint **50 %** de sa durée. Pour renouveler, un client DHCP envoie un message **DHCPREQUEST** directement au serveur DHCP avec qui il a conclu le bail en vigueur.

Si le serveur DHCP est disponible, il renouvelle le bail et envoie au client un accusé de réception stipulant la conclusion du renouvellement (**DHCPACK**) et la nouvelle durée, ainsi que les éventuelles mises à jour des paramètres de configuration.

Lorsque le client reçoit l'accusé de réception, il met à jour sa configuration. Si un client tente de renouveler son bail mais est dans l'impossibilité de contacter le serveur DHCP à l'origine de ce dernier, le client peut encore utiliser l'adresse, puisqu'il lui reste 50 % de la durée du bail.

Lorsqu'un client DHCP redémarre, il tente d'obtenir un bail pour la même adresse avec le serveur DHCP d'origine. Pour ce faire, il diffuse un message **DHCPREQUEST** spécifiant la dernière adresse IP dont il avait le bail. Si la tentative se solde par un échec et qu'il lui reste encore du temps avant l'expiration du bail, le client DHCP continue à utiliser la même adresse IP.

Si un bail, lorsqu'il atteint **50 %** de sa durée, n'a pas pu être renouvelé par le serveur DHCP d'origine, le client tente de contacter les autres serveurs DHCP disponibles lorsque **87,5% du temps s'est écoulé**. Le client diffuse alors un message **DHCPREQUEST**. Tous les serveurs DHCP peuvent répondre par un message **DHCPACK(renouvellement du bail)** ou **DHPCNACK (obligeant le client DHCP à se réinitialiser** et à obtenir le bail d'une adresse IP différente).

Lorsque le bail expire ou qu'un message DHCPNACK est reçu, le client DHCP doit immédiatement cesser d'utiliser l'adresse IP. Il retourne alors au processus de souscription d'un nouveau bail d'adresse IP.

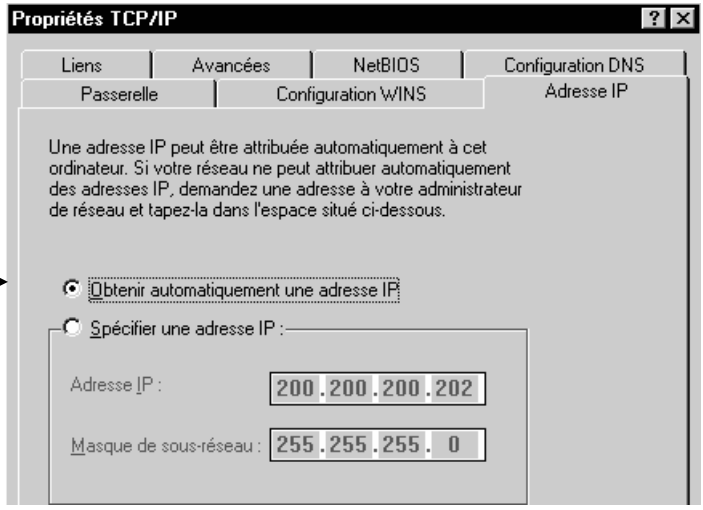
CLIENT DHCP

Un Client Windows 95-98

soit par **propriétés** de **voisinage réseau**,(sur le bureau)

soit par **démarrer / paramètres / panneau de configuration / réseau**

puis propriétés de TCP/IP



Lorsque on demande une adresse automatique, tout le reste du paramétrage IP devient "inactif"

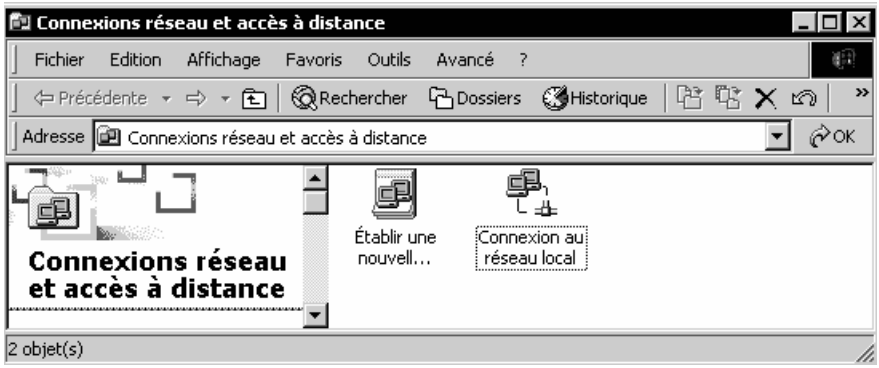
Client DHCP NT 2000:

Un poste devient client DHCP simplement en demandant dans le paramétrage de TCP/IP « **Obtenir automatiquement une adresse IP** »

soit par **propriétés** de **favoris réseau**,(sur le bureau)

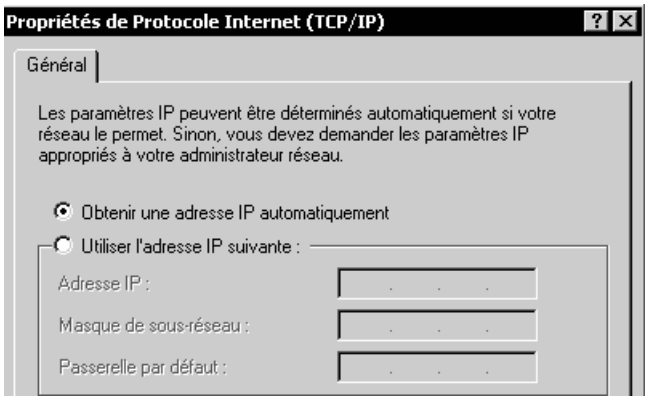
soit par **démarrer / paramètres / connexion réseau** et accès à distance

soit par **démarrer / paramètres / panneau de configuration / connexion réseau et accès à distance**



puis propriétés de connexion au réseau local

puis propriétés de TCP/IP



Lorsque on demande une adresse automatique, tout le reste du paramétrage IP devient "inactif"

Remarques

N.B: si aucun serveur DHCP n'est présent, un mécanisme dit "adresses APIPA" se met en oeuvre, (voir "adresses automatiques APIPA" page 51) uniquement pour des postes **Windows 98** et **Windows NT 2000** les postes **Windows 95** et **Windows NT 4.0** ne gèrent pas les adresses APIPA

N.B: On peut savoir depuis un poste client qui est le serveur DHCP et quelle adresse nous est allouée a un moment donné via des utilitaires :
ipconfig sous Windows NT 4.0 et Windows Nt 2000
winipcfg sous Windows 95-98
(voir "tester TCP/IP Winipcfg Ipconfig" page 41)

ADRESSES IP AUTOMATIQUES (APIPA)

Principe des adresses APIPA:

Dans les réseaux locaux simples, on peut mettre en place un nouveau système d'attribution automatique des adresses IP, donc sans ni attribuer une adresse IP fixe à chaque poste, ni avoir recours à un serveur DHCP...

Le fonctionnement est le suivant :

1. Une machine installée avec un protocole TCP/IP tente de contacter un serveur DHCP pour recevoir une adresse IP de manière dynamique (elle doit être configurée pour...)
2. Si aucun serveur DHCP ne réponds, la fonction APIPA génère une adresse IP au format 169.254.xxx.xxx avec un masque de sous-réseau 255.255.0.0. Si cette adresse est déjà utilisée la fonction APIPA en sélectionne une autre pour un maximum de 10 coups.
3. Une fois une adresse prise, l'ordinateur la diffuse et l'utilise jusqu'à ce qu'un serveur DHCP n'apparaisse opérationnel sur le réseau !

quelques remarques :

- l'IANA (Internet Assigned Number Authority) à réservé les adresses de **169.254.0.0** à **169.254.255.255** à la fonction APIPA, ces adresse n'étant pas routables !
- Par conséquent les machines utilisant des adresse APIPA ne peuvent communiquer qu'avec des machines faisant partie du même sous-réseau, et dotée d'un adresse au format 169.254.xxx.xxx

APIPA et Windows NT 2000:

Pour que NT 2000 gère les adresse APIPA, il est nécessaire d'utiliser TCP/IP comme protocole et de demander le bouton Option "Obtenir une adresse IP automatiquement" dans Propriétés de Protocole Internet (TCP/IP)

Par défaut les adresses APIPA sont actives, il est possible de les inhiber en allant dans la base de registre et en demandant

HKEY_LOCALMACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\GUID_carte_réseau et en lui ajoutant l'entrée

IPAutoconfigurationEnabled avec une valeur de 0

(si cette entrée n'existe pas ou que sa valeur est fixée à 1 APIPA est activée)

N.B: Windows 98 **gère** également APIPA

N.B: Windows NT 4.0 **ne gère pas** les adresses APIPA

NOTION DE DNS

Le DNS:

DNS est au centre de la gestion des domaines dans Windows 2000. il faut en comprendre certaines notions fondamentales

Noms DNS

Selon la définition de la RFC 952 le nom DNS d'un ordinateur est constitué de plusieurs parties séparées par des virgules, par exemple, **www.fnac.presse.fr.**

	NetBIOS	Full computer name
Type	Flat	Hierarchical
Character Restrictions	A-Z, a-z, 0-9, "espace", symbols: ! @ # $ % ^ & ') (. - _ { } ~ Unicode chars,	A-Z, a-z, 0-9, symbols: -_, Unicode chars. Le point '.' est le séparateur
Maximum Length	16 (dont 1 réservé) dont 15 en pratique	63 pour un nom de domaine 255 pour un FQDN
Name Service	NBNS (WINS and broadcast)	DNS

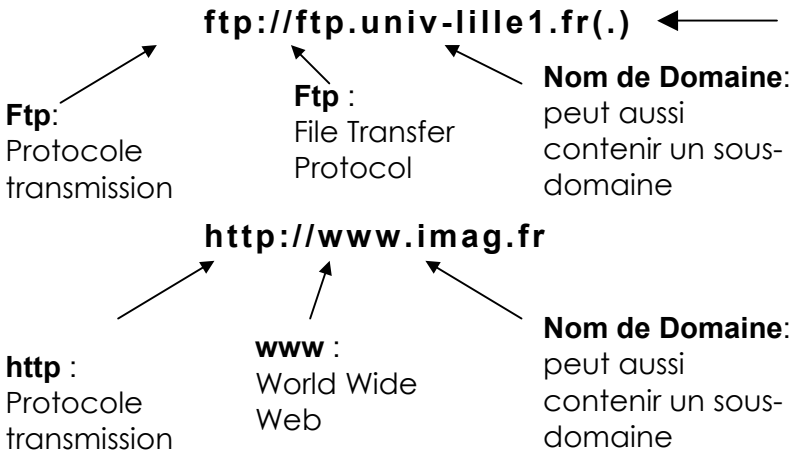
Windows 2000 utilise les noms DNS comportant des caractères soulignés, une fonction qui affectera votre choix de serveur DNS

Nom "Plat" Netbios

Les nom netbios sont crées-enregistrés lors du démarrage de chaque poste, et doivent être uniques sur tous le réseau. Ce simple constat pose les limites d'envergure des noms Netbios gérés par broadcast, d'ou l'apparition de serveur WINS sur les réseaux de taille moyenne-grande. Mais même ainsi, il parait impossible d'assurer l'unicité sur des réseau de grandes envergure...

Nom "Hierarchique" DNS

une URL se lit de droite à gauche

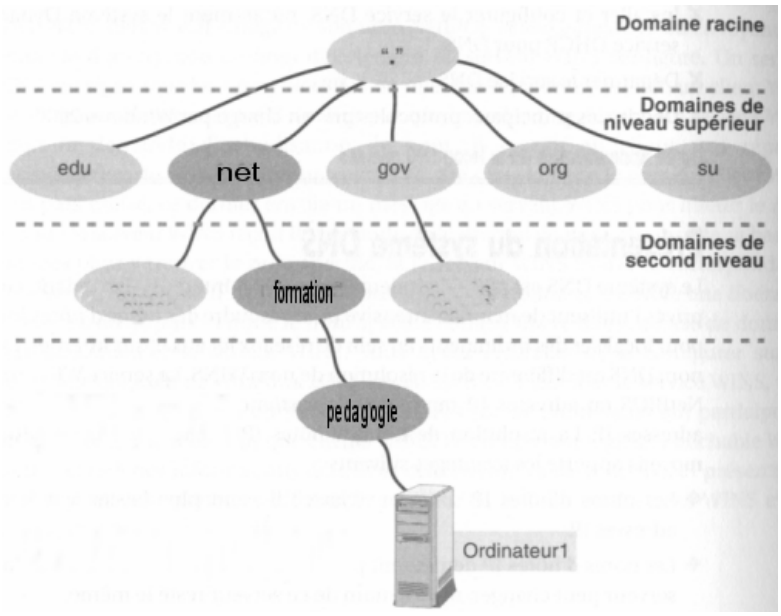


Un nom se termine par un point **.** qu'il n'est plus necessaire d'écrire, mais qui correspond au domaine "racine"

Un nom de domaine (imag.fr) se décompose en

- Un Top Level Domain (exemple : fr)
- Un nom d'organisation (appelé aussi nom de domaine) (ex : imag)

Structure des domaines



le **FQDN** (Fully Qualified Domain Name) de cet ordinateur est **Ordinateur1.pedagogie.formation.net**

Les Top Level Domain les plus courants sont:

Clé	Contenu
.com	Entreprise commerciale
.edu	éducation
.gov	organismes gouvernementaux
.mil	organisations militaires
.net	intervenant d'internet
.org	instance gouvernementale ou institution administrative

Cependant si ces domaines sont a priori internationaux, ils sont à forte dominante américaine. De plus chaque pays possède son nom de domaine (à l'exception des USA qui utilisent les 6 domaines précédents).

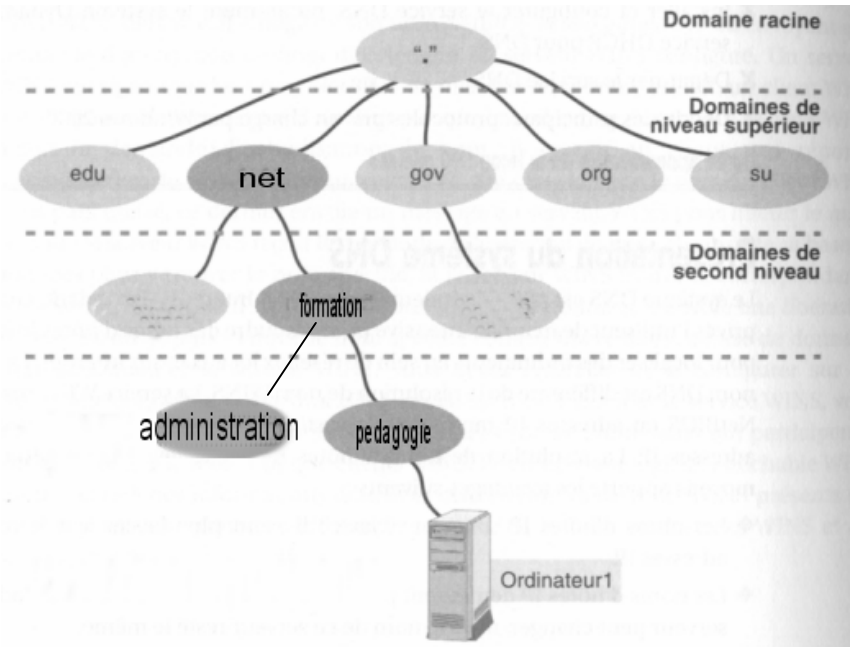
Clé	Contenu
.au	Australie
.ca	Canada
.fr	France
.uk	United Kingdom

L'internic se chargeant de l'attribution des adresses dans les domaines internationaux, c'est le NIC France qui se charge des attributions des noms de domaine en .fr

http://www.nic.fr

Zones DNS

Une **Zone** représente une partie de l'espace de nom de Domaine, a des fins de gestion.



Supposons que vous ayez deux régions, **administration** et **pedagogie**. Chaque région souhaite exploiter un serveur **DNS local**.

Pour répondre aux besoins des deux régions, vous pouvez ajouter un niveau comme par exemple :

administration.formation.net et

pedagogie.formation.net.

Chaque serveur DNS a une sous-section de domaine (une **zone** en jargon DNS).

Le serveur DNS central **formation.net** ne gère alors plus qu'un très petit nombre de noms de hosts. Il stocke en outre les noms et adresses IP des serveurs DNS de ces zones, à savoir **pedagogie.formation.net** et **administration.formation.net**.

Ainsi, si une machine **ordinateur1** se trouve dans la région **pedagogie**, elle se nommera.**ordinateur1**.**pedagogie**.**formation.net**

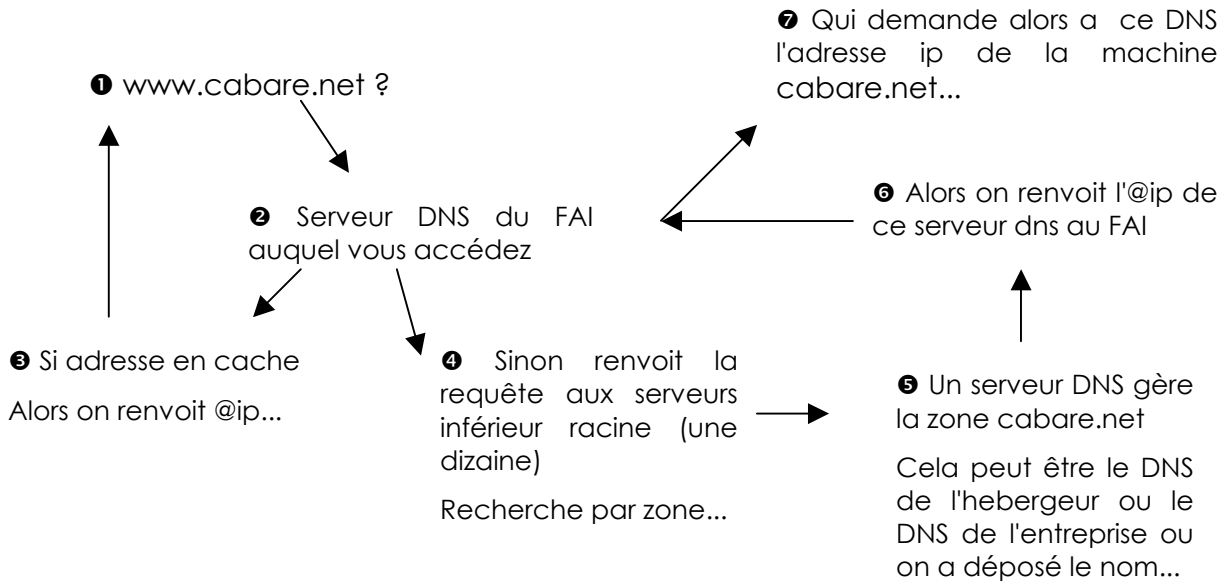
- Si cette machine **ordinateur1** essaye d'atteindre un autre poste du domaine pédagogie, sa requête sera traitée par le serveur DNS de **pedagogie.formation.net**
- Si cette machine **ordinateur1** essaye d'atteindre un poste du domaine administration, sa requête sera traitée par le serveur DNS de **pedagogie.formation.net,** et **redirigée** vers le serveur racine de niveau supérieur**,** à savoir **formation.net.** celui-ci connait le serveur qui gère la zone administration, c'est **administration.formation.net** il renvoit l'adresse de ce serveur DNS au serveur DNS **pedagogie.formation.net** qui peut alors refaire sa demande...

Réquêtes itératives ou récursives

Avec un raisonnement identique a celui précédant pour formation-administration décomposons la requête envoyée à un DNS pour un accès à un site sur Internet.

Vous êtes sur un poste et vous essayez d'atteindre l'URL **www.cabare.net.**

Vous pouvez vous permettre de demander en fait **www.cabare.net**, et cette demande est transmise au serveur DNS de votre FAI.



- Le processus ❶❷ puis ❸ est appellé requête récursive
- Le processus ❶❷puis ❹❺❻❼ est appellé requête iétérative

Résolution de Noms et Résolution inverse

Chaque composant informatique d'Internet a une adresse IP unique sur 32 bit (par exemple **154.23.17.8**). Il est possible de nommer un élément en se référant à son adresse IP. Mais la plupart des utilisateurs préfèrent les noms plus faciles à retenir comme **http://toto.com**. Pour pouvoir utiliser ce type de noms, il faut une base de données capable de convertir les adresses IP en adresses mémorisables. On appelle cela la **résolution de noms**.

la **résolution de nom** (**forward lookup**) permet de trouver une adresse IP à partir d'un nom

la **résolution inverse** (**reverse lookup**) permet de trouver un nom à partir d'une adresse IP

Du fait du faible nombre de systèmes présents sur Internet à ses origines, les machines connectées à Internet prenaient en charge la résolution de noms via une simple table ASCII (**fichier HOSTS**) qui listait les adresses IP et les noms de machines correspondants. (Le code de TCP/IP permet toujours de placer un fichier HOSTS sur un système). Depuis 1984, les systèmes ont recours principalement à **DNS** pour la résolution de noms. Sinon il faudrait maintenir un fichier HOSTS qui contiendrait non seulement des centaines de millions d'ordinateurs, mais qui changerait quotidiennement !

L'implémentation la plus populaire de DNS est **BIND** (Berkeley Internet Name Domain) sous UNIX

DDNS

La méthode utilisée pour ajouter un nouvel enregistrement correspondant à un nouvel ordinateur - un nouveau host en terminologie DNS, dépend de votre logiciel serveur DNS. La plupart utilisent des fichiers ASCII.

Les solutions de serveur DNS les plus récentes n'exigent plus de mises à jour grâce au standard **DDNS** (**Dynamic DNS**) que décrit en détail la RFC 2136. Dans un réseau compatible DDNS, les ordinateurs font d'eux-mêmes les présentations sans qu'un administrateur ne doive intervenir sur le DNS

Enregistrements SRV

Les solutions de serveur DNS les plus récentes gèrent une autre sorte d'enregistrement DNS : les **enregistrements SRV** que décrit en détail la RFC 2052. Ces enregistrements permettent de demander à un serveur DNS si il connaît des machines jouant le rôle de serveur d'un type spécifique

Serveur principal - secondaire

Le serveur DNS peut remplir plusieurs fonctions par rapport à une zone, le serveur chargé de la gestion initiale de la zone est appelé **serveur principal** ou **primary**. mais les informations d'une zone peuvent être répliquées sur d'autres serveurs soit dans un objectif de fiabilité, soit pour un objectif de répartition de charge. Dans ce cas le serveur DNS qui recopie les information depuis le serveur DNS principal s'appelle un **serveur secondaire** ou **backup**. L'édition du fichier de la zone est faite sur le serveur principal qui envoie la version la plus récente du fichier au serveur DNS secondaire. Lorsqu'une machine envoie une requête au serveur secondaire, ce dernier y répond avec sa copie du fichier. Le fichier de zone du serveur secondaire a généralement une durée de vie (généralement de 24 heures). Si le serveur DNS primaire ne met pas à jour le fichier avant la période d'expiration, le serveur secondaire considère l'information comme dépassée. Si votre serveur DNS principal tombe en panne pendant quelques heures, vous n'aurez donc pas de problème. Les serveurs DNS secondaires peuvent être aussi nombreux que l'on le souhaite.

NOM NETBIOS

Protocole NetBeui :

Windows 9x et NT peuvent utiliser le protocole propriétaire Netbeui pour communiquer avec d'autre machine Windows.

D'ailleurs, pour les réseaux de petite taille, une vingtaine de postes, cette solution permet un partage simple des ressources. Cette solution permet aux **applications NETBIOS** d'accéder au réseau en s'appuyant sur le **protocole NETBEUI**.

Quelques définitions :

NetBIOS :

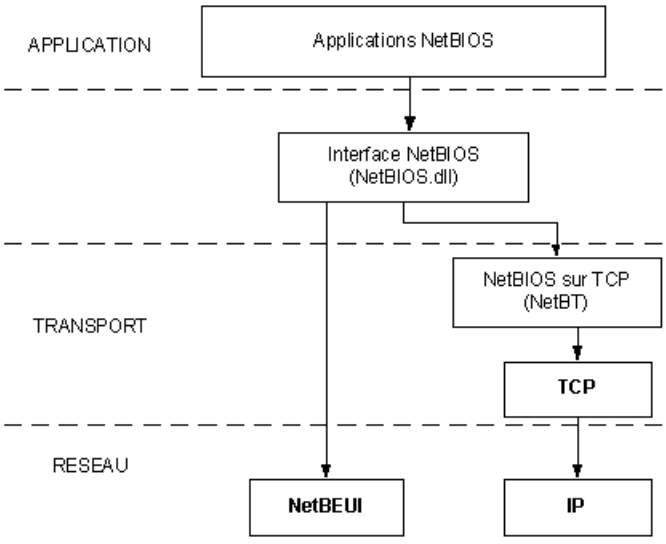
(Network Basic Input/output System) Interface de programmation qui permet aux applications d'accéder au réseau local. NetBIOS utilise un service de noms pour contrôler les échanges de point à point.

NetBEUI :

(NetBIOS Extended User Interface) est le protocole de transport des réseaux Windows. Il ne peut pas être routé et repose principalement sur les diffusions.

NetBT

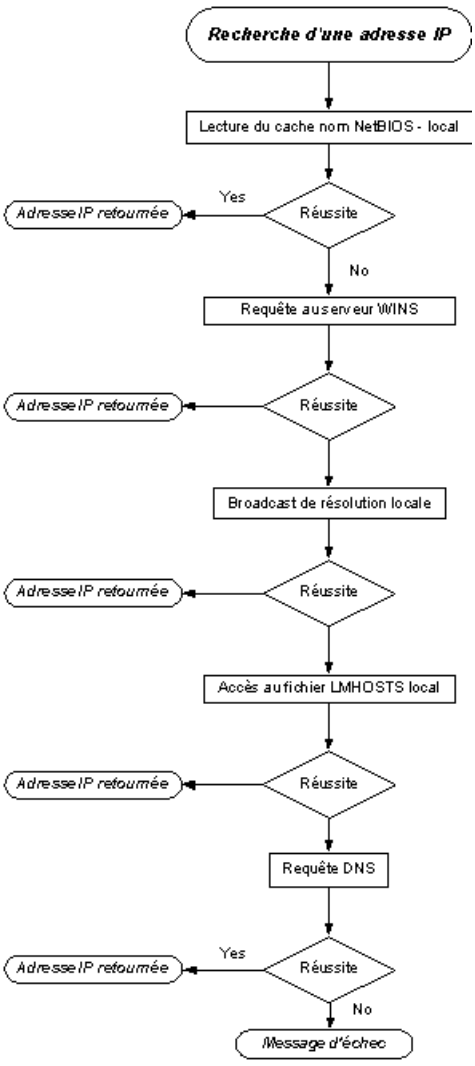
(NetBIOS sur TCP/IP) est le service de résolution de noms NetBIOS pour les réseaux Windows sous TCP/IP.



Résolution de nom NetBIOS

Windows NT peut utiliser différentes méthode pour effectuer la résolution de nom netbios :

- NetBIOS name cache (vérifiable via nbtstat –n)
- NetBIOS name server (WINS Il existe sous NT un serveur de nom NetBIOS connu sous l'appellation serveur WINS.)
- IP subnet broadcasts (limité au sous-réseau)
- Static Lmhosts file. (pour résoudre un nom netbios sur un autre réseau)
- Static Hosts file (**optionnel** pour un nom d'hôte)
- DNS servers (optionnel)



La manière dont NT va résoudre les nom Netbios, dépends du paramétrage du poste, et de la configuration du réseau existant. Les différents modes de résolution suivants sont possibles , on parle de type de noeud:

- **B-node (diffusion) :** utilise des broadcast pour l'enregistrement et la résolution des noms Netbios.
- **P-node** : utilise un serveur de nom NetBios (Wins) pour l'enregistrement et la résolution des noms Netbios.
- **M-node** : utilise des broadcast pour l'enregistrement. Pour la résolution, utilise d'abords des Broadcast, puis en l'absence de réponse passe ne mode P-node (donc utilise un serveur WINS)
- **H-node (hybride)** : utilise un serveur de nom NetBios (Wins) pour l'enregistrement et la résolution des noms Netbios . Si un serveur ne peut pas être trouvé, il passe en b-node. (donc utilise des boradcast) . Il continue à chercher une serveur WINS et repasse en p-node des qu'il en trouve un disponible
- **Microsoft-enhanced** : utilise les fichiers Lmhosts en plus des mode standard.

Par défaut, la plupart des clients sont paramétrés en B-nodes, c'est à dire émettent des broadcast...

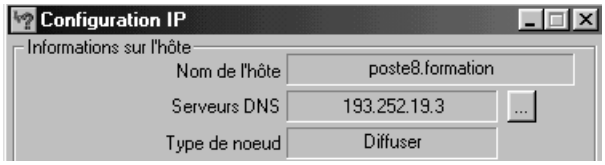
Paramétrer la résolution NetBIOS

il est bien sûr possible de voir le mode de résolution actuellement en cours sur une machine avec **IPCONFIG /ALL** dans la rubrique "**type de noeud**"

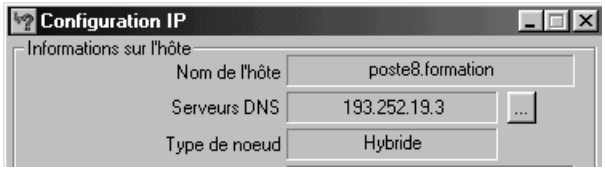
- on peut facilement demander de passer de **B-nodes** à **h-nodes**, et vice-versa.

Il suffit de renseigner ou non l'adresse d'un serveur Wins sur le client...

serveur Wins **non** renseigné

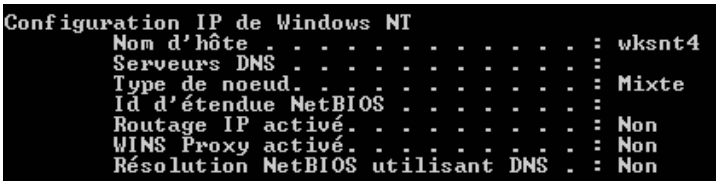


serveur Wins renseigné



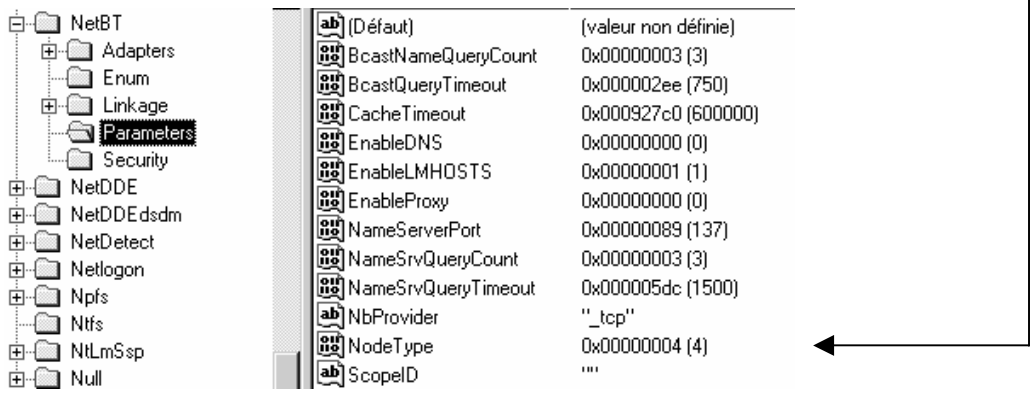
L'accès aux autres modes de résolution n'est possible que sur des machines NT ou 2000:

- Par exemple, l'activation des LmHosts se fait dans les propriétés avancées de TCP-IP, onglets Wins.
- Par exemple le passage en Type de noeud M-Nodes,



ne peut se faire via modification de la base de registre par ajout d'une clé de type Dword dans l'entrée

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NetBt\Parameters



Les valeurs possibles étant :

1	**b-node**
2	**p-node**
4	**m-node**
8	**h-node**

Fichier LMhosts et fichiers Hosts:

Un fichier **HOSTS** permet d'établir un mappage entre une adresse IP et un nom de machine (nom d'hôte), c'est un fichier issu du monde unix. L'alternative au fichier hosts est un serveur DNS.

Un fichier **LMHOSTS** permet également d'établir un mappage entre une adresse IP et un nom de machine (nom d'ordinateur ou nom netbios). L'alternative au fichier LMHOSTS est le service WINS. Le fichier LMHOSTS (Lan Manager HOSTS) concerne essentiellement les réseau Microsoft.

A noter qu'un nom d'hôte et un nom d'ordinateur (nom netbios) sont deux choses différentes. Par défaut, dans un réseau microsoft, le nom d'hôte et le nom d'ordinateur sont les mêmes.

Un nom d'hôte pourra donc prendre la forme soit d'un nom d'ordinateur (nom Netbios) soit d'un nom du type ordinateur.masociété.com (FQDN: Fully Qualified Domain Name).

Pour les routeurs, un nom d'hôte semble plus adapté car un routeur n'est généralement pas une machine Windows (bien que Windows NT puisse faire office de routeur) mais un boitier électronique contenant la plupart du temps un micro noyau de type UNIX.

Fichiers lmhosts (nom netbios)

Un exemple est fournit sur les machines avec le fichier **lmhosts.sam** avec une extension .sam pour sample qu'il faut evidemment enlever pour rendre actif le fichier **lmhosts**. Il permet de solutionner un nom netbios sur un autre sous-réseau. Un fichier lmhost peut contenir une ligne du genre

192.168.1.1 #PRE #DOM:nomdomaine

avec 192.168.1.1 l'adresse ip du controlleur de domaine

avec nomdomaine le nom du domaine géré par le Contrôleur

Après modification du fichier **lmhosts** il faut impérativement redemarrer le poste, ou faire une commande en ligne

Nbtstat –R (avec le R majuscule...)

Pui vérifier la prise ne compte avec un

Nbtstat –c (avec le c minuscule...)

Détails écriture lmhosts

1. **10.0.0.1 PDCName #PRE #DOM:Domain-name**
2. **10.0.0.1 "Domain-name \0x1b" #PRE**

N.B : Le nom de domaine dans cette entrée respecte la casse.

N.B: L'espacement de ces entrées est obligatoire. Remplacez 10.0.0.1 par l'adresse IP de votre contrôleur principal de domaine, PDCName par le nom NetBIOS de votre contrôleur principal de domaine, et Domaine par le nom de domaine de Windows NT. Au total il doit y avoir 20 caractères à l'intérieur des guillemets (le nom de domaine, + le nombre d'espaces

appropriés pour obtenir 15 caractères, + la barre oblique inverse, + la représentation hexadécimale NetBIOS du type de service).

N.B: Pour déterminer l'emplacement du 16e caractère, copiez la ligne suivante dans votre fichier LMHOSTS :

 # Adresse IP "123456789012345*7890"

Alignez les guillemets doubles (") en ajoutant ou supprimant des espaces dans la ligne de commentaire, et placez la barre oblique inverse sur la 16e colonne (marquée d'une astérisque). Vous ne devez pas utiliser de tabulation mais des ESPACES après le nom et avant la barre oblique inverse (\).

NB: Attention, le fichier contient toujours une ligne blanche vide à la fin !

Fichiers hosts (nom d'hôte)

Un exemple est fournit sur les machines avec le fichier **hosts.sam** avec une extension .sam pour sample qu'il faut evidemment enlever pour rendre actif le fichier **lmhosts**. Il permet de solutionner un nom d'hôte. Un fichier hosts peut contenir une ligne du genre

```
# Copyright (c) 1998 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP stack for Windows98
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

127.0.0.1       localhost
```

Attention, le fichier contient toujours une ligne blanche vide à la fin !

Nom NetBios :

N.B: On peut utiliser l'utilitaire **nbtstat** pour voir les noms NetBIOS avec la syntaxe suivante :

nbtstat –n

ou nbtstat –a nommachine

Les **15 premiers caractères** d'un nom peuvent être spécifiés par un utilisateur. En revanche, le **16e caractère** du nom (hexadécimal 00-FF) indique toujours un type de ressource:

Name	**Nb (hexa)**	**Type**	**Usage**
<computername>	00	U	Workstation Service
<computername>	01	U	Messenger Service
-- MSBrowse --	01	G	Domain Master Browser
<computername>	03	U	Messenger Service
<computername>	06	U	RAS Server Service
<computername>	1F	U	NetDDE Service
<computername>	20	U	File Server Service
<computername>	21	U	RAS Client Service
<computername>	22	U	Microsoft Exchange Connector
<computername	23	U	Microsoft Exchange Store
<computername>	24	U	Microsoft Exchange Directory
<computername>	30	U	Modem Sharing Server Service
<computername>	31	U	Modem Sharing Client Service
<computername>	43	U	SMS Clients Remote Control
<computername>	44	U	SMS Admin Remote Control Tool
<computername>	45	U	SMS Clients Remote Chat
<computername>	46	U	SMS Clients Remote Transfer
<computername>	4C	U	DEC TCPIP service on NT
<computername>	42	U	mccaffee anti-virus
<computername>	52	U	DEC TCPIP service on NT
<computername>	87	U	Microsoft Exchange MTA
<computername>	6A	U	Microsoft Exchange IMC
<computername>	BE	U	Network Monitor Agent
<computername>	BF	U	Network Monitor Application
<username>	03	U	Messenger Service
<domain>	0	G	Domain Name
<domain>	1B	U	Domain
<domain>	1C	G	Domain Controllers
<domain>	1D	U	Master Browser
<domain>	1E	G	Browser Service Elections
<INet~Services>	1C	G	IIS
<IS~computer name>	00	U	IIS
<computername>	[2B]	U	Lotus Notes Server Service

Unique (U): Utilisé pour associer l'ordinateur désigné par son nom dans Nom d'ordinateur à une adresse IP unique dans l'adresse IP de cette entrée mappée statique. Lorsque vous sélectionnez ce type de nom, trois types d'enregistrements sont ajoutés statiquement à la base de données WINS pour le nom d'ordinateur spécifié. Les types [00h]**WorkStation**, [03h]**Messenger** et [20h] **Serveur de fichiers**.

Noms uniques NetBIOS

Format	Description
nom_ordinateur [00h]	Inscrit par le service Station de travail sur le client WINS. En général, ce nom est appelé *nom d'ordinateur NetBIOS*.
nom_ordinateur [03h]	Inscrit par le service Affichage des messages sur le client WINS. Ce service est utilisé par le client pour envoyer et recevoir des messages. Ce nom est généralement ajouté au nom d'ordinateur NetBIOS du client WINS et au nom de l'utilisateur actuellement connecté à ce client pour envoyer des messages sur le réseau.
nom_ordinateur [06h]	Inscrit sur le client WINS par le service de routage et d'accès distant (lorsque ce service est démarré).
nom_domaine [1Bh]	Inscrit par chaque contrôleur de domaine Windows NT Server qui s'exécute en tant qu'explorateur principal de domaine. Cet enregistrement de nom est utilisé pour permettre l'exploration à distance des domaines. Lorsque ce nom est demandé à un serveur WINS, ce dernier renvoie l'adresse IP de l'ordinateur qui a inscrit ce nom.
nom_ordinateur [1Fh]	Inscrit par les services NetDDE (Network Dynamic Data Exchange). Ne s'affiche que si les services NetDDE sont démarrés sur l'ordinateur.
nom_ordinateur [20h]	Inscrit par le service Serveur sur le client WINS. Ce service est utilisé pour fournir des points de service au client WINS, qui lui permettent de partager ses fichiers sur le réseau.
nom_ordinateur [21h]	Inscrit sur le client WINS par le service Client RAS (lorsque ce service est démarré).
nom_ordinateur [BEh]	Inscrit par l'Agent de surveillance du réseau et n'apparaissant que si ce service est démarré sur le client WINS. Si le nom d'ordinateur compte moins de 15 caractères, les espaces restants sont remplis par des signes plus (+).
nom_ordinateur [BFh]	Inscrit par l'utilitaire de surveillance du réseau (livré avec Microsoft Systems Management Server). Si le nom d'ordinateur compte moins de 15 caractères, les espaces restants sont remplis par des signes plus (+).
nom_utilisateur [03h]	Les noms des utilisateurs actuellement connectés sont inscrits dans la base de données WINS. Chaque nom d'utilisateur est inscrit par le service Serveur de sorte que les utilisateurs peuvent recevoir toutes les commandes **net send** envoyées au nom d'utilisateur. Si plusieurs utilisateurs se connectent sous le même nom, seul le premier ordinateur connecté avec ce nom enregistre le nom.

Group (G): Appelé aussi groupe ordinaire. Ce type est utilisé pour ajouter une entrée statique pour l'ordinateur, spécifié par un nom dans un mappage statique, dans un groupe de travail utilisé sur votre réseau. Si vous utilisez le type, l'adresse IP de l'ordinateur n'est pas stockée dans WINS, mais résolue par le biais des diffusions du sous-réseau local.

Noms de groupes NetBIOS

Format	Description
nom_domaine [00h]	Inscrit par le service Station de travail de sorte qu'il puisse recevoir les diffusions d'exploration provenant d'ordinateurs LAN Manager.
nom_domaine [1Ch]	Inscrit à l'usage du contrôleur de domaine dans le cadre du domaine. Peut contenir jusqu'à 25 adresses IP.
nom_domaine [1Dh]	Inscrit à l'usage des explorateurs principaux (un seul explorateur principal par sous-réseau). Les explorateurs de sauvegarde utilisent ce nom pour communiquer avec l'explorateur principal, en extrayant la liste des serveurs disponibles de l'explorateur principal. Les serveurs WINS renvoient toujours une réponse positive d'inscription pour *nom_domaine*[1D], même si le serveur WINS n'*inscrit* pas ce nom dans sa base de données. En conséquence, lorsque le *domain_name*[1D] est demandé à un serveur WINS, ce dernier renvoie une réponse négative, ce qui force le client à lancer une diffusion de résolution de noms.
nom_groupe [1Eh]	Un nom de groupe ordinaire. Tout ordinateur configuré en tant qu'explorateur de réseau peut diffuser vers ce nom, et écouter les diffusions vers ce nom, pour choisir un explorateur principal. Un nom de groupe mappé statiquement utilise ce nom pour s'inscrire sur le réseau. Lorsqu'un serveur WINS reçoit une demande de nom se terminant par [1E], il renvoie toujours l'adresse de diffusion du réseau local du client qui a émis la demande. Le client peut ensuite utiliser cette adresse pour diffuser aux membres du groupe. Ces diffusions sont destinées au sous-réseau local et ne doivent pas traverser de routeurs
nom_groupe [20h]	Un nom de groupe spécial appelé *groupe Internet* est inscrit sur les serveurs WINS pour identifier des groupes d'ordinateurs pour des besoins administratifs. Par exemple, "printersg" peut être un nom de groupe inscrit utilisé pour identifier un groupe administratif de serveurs d'impression.
-- __MSBROWSE__ [01h]	Inscrit par l'explorateur principal pour chaque sous-réseau. Lorsqu'un serveur WINS reçoit une demande concernant ce nom, il renvoie toujours l'adresse de diffusion du réseau local du client qui a émis la demande.

enfin, moins important

Multihomed (M): Utilisé pour inscrire un nom unique pour un ordinateur ayant plusieurs adresses IP (plusieurs cartes utilisant chacune une adresse unique ou une seule carte réseau configurée avec plusieurs adresses IP).

Internet **Group (I):** Utilisés pour des groupes administratifs spéciaux définis par l'utilisateur. Vous pouvez utiliser ce type pour regrouper des ressources. Par exemple, vous pouvez indiquer un groupe de fichiers ou de serveurs d'impression pour organiser les ressources partagées visibles lorsque vous parcourez les favoris réseau. Chaque groupe Internet est représenté par un *nom de groupe* partagé de type [20h] dans la base de données WINS

Domain Name (D): Indique une entrée mappée de *nom de domaine* [1C] pour la localisation des contrôleurs de domaine Windows NT

MECANISME DU VOISINAGE RESEAU

Principe de fonctionnement :

Lorsque l'on clique sur voisinage réseau, on a souvent une réponse lors du démarrage de la machine comme quoi le "parcours du réseau est impossible", or **il suffit d'attendre et tout rentre dans l'ordre**...

Mais la signification du message est la suivante : actuellement un **Explorateur Principal** n'est pas encore identifié...

Environ toutes les 12 minutes, les serveurs annoncent leur présence avec des trâmes spéciales au format NetBios. Une élection d' Explorateur Principal peut arriver lorsque

- un ordinateur n'arrive pas à trouver un Explorateur Principal
- Lorsque un Explorateur Principal arrive sur le réseau, ou s'arrête.
- Lorsque un Contrôleur de Domaine démarre:

Lorsque une élection est lancée, un algorithme compliqué basé sur plusieurs variables se déroule (type de OS, version d'OS, configuration, adressage IP, nombre de machines présentes etc) et un seul Explorateur Principal sera déclaré !

A chaque fois qu'un PC démarre, il est configuré par défaut pour tenter de savoir s'il doit devenir Explorateur...

Il peut exister jusqu'à 5 types de machines dans un réseau Windows

Non-Browser / Non Explorateur

Un **non-browser** ou **non Explorateur** est un ordinateur qui a été configuré pour ne pas maintenir une liste des ordinateurs devant apparaître dans le voisinage réseau

Potential Browser / Explorateur Potentiel

Un **Potential-Browser** ou **Explorateur Potentiel** est un ordinateur capable de maintenir une liste des ordinateurs devant apparaître dans le voisinage réseau , et pouvant être promu comme Explorateur principal. Un **Explorateur Potentiel** est aussi capable de jouer le rôle d'un **Explorateur de Secours**, s'il est piloté par un **Explorateur Principal**

Backup Browser / Explorateur de Secours

Un **Backup-Browser** ou **Explorateur de Secours** reçoit une copie des ordinateurs devant apparaître dans le voisinage réseau depuis un **Explorateur Principal** et fournis cette liste à la demande des autres ordinateurs du domaine ou du groupe de travail

N.B: Lorsqu'un poste démarre, c'est l' **Explorateur Principal** qui lui indique s'il doit devenir un **Explorateur de Secours** ou non

Master Browser / Explorateur Principal

Un **Master-Browser** ou **Explorateur Principal** est responsable de la collecte des informations nécessaires à la création et à mise à jour de la liste des ordinateurs figurant dans le voisinage réseau. Cette liste inclus tous les serveurs du domaine de l' **Explorateur Principal** et la liste de tous les domaines sur le réseau. Les machines windows annoncent leur présence à l' **Explorateur Principal** par un datagrame appelé "server annoucement", et celui-ci les ajoute

- Si un Domaine s'étends sur plus d'un sous-réseau, l' **Explorateur Principal** travaille de la manière suivante :
  - ✓ Il gère la liste pour le sous-réseau dont il fait partie
  - ✓ fournit cette liste à chaque Explorateur de Secours de chaque sous-réseau
- Si un sous-réseau comprends plusieurs Domaines, chaque Domaine à son **Explorateur Principal** et éventuellement ses **Explorateurs de Secours**

Domain Master Browser / Explorateur Principal de Domaine

Un **Domain Master-Browser** ou **Explorateur Principal de Domaine** est responsable de la collecte des informations pourla création et la mise à jour de la liste pour tout le domaine, collecte les informations des **Explorateur Principaux** des autres sous-réseaux et fournit les informations aux **Explorateur Principaux** des autres sous-réseaux.

Un **Explorateur Principal de Domaine** est toujours le Contrôleur Principal de Domaine

N.B: Un poste peu jouer plusieurs rôles, par exemple l' **Explorateur Principal** peut aussi être un **Explorateur Principal de Domaine**

Rafraîchissement Tests et vérifications :

Quelles sont les vitesses de raffraîchissement ?

de quelques secondes, à plusieurs minutes, jusqu'à 12 minute pour la prise en compte d'un serveur dans un Domaine, ce qui par rebonds peut aller à 24 minutes entre 2 Domaines...

Pour la suppression d'une machine c'est pire, Microsoft annonçant jusqu'à 45 minutes pour la mise à jour d'une liste "rayant" une machine qui ne se serait pas correctement déconnectée du réseau (arrêt système brutal...)

Peut on éviter l'élections d'un Explorateur ? :

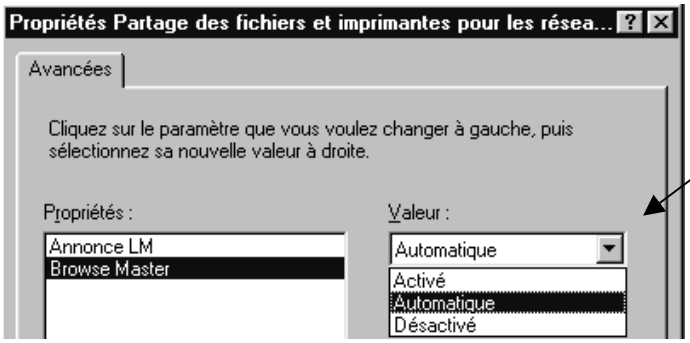
la réponse est non; il doit y en avoir toujours un, mais on peut a la limite accélérer un peut les choses

En implémentant un serveur WINS qui diminuera le trafic réseau pour les résolution de nom Netbios,

En implémentant un serveur DNS qui diminuera le trafic réseau pour les résolution de nom

En modifiant le status d'une machine : si on modifie dans propriété de partage des fichiers et imprimantes le fait qu'une machine soit éligible ou non (on peut éviter les élections et diminuer les trâmes émises...)

Sous Windows 95-98

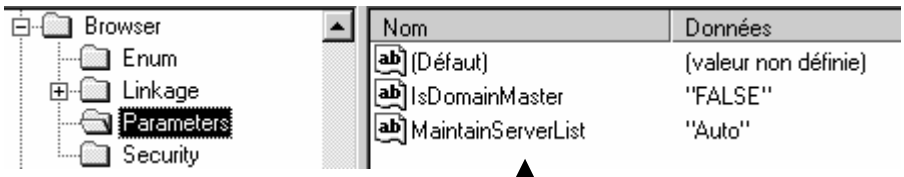


définir qui peut être Browse Master

N.B: Il doit y en avoir toujours 1 seul !

Sous Windows NT ou 2000

Il faut modifier la base de registre NT "ce qui reste délicat"

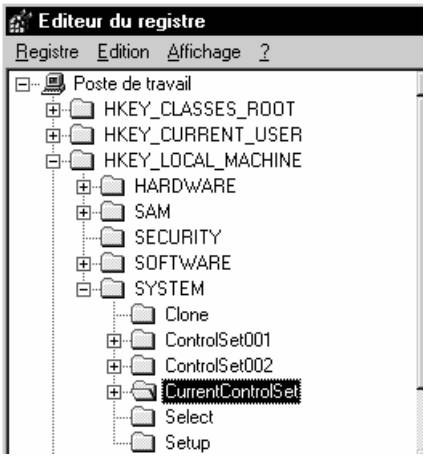


Il faut se positionner sur la clé

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Browser\ Parameters

et y modifier la clé de type DWORD-value nommée **MaintainServer List** les valeurs possibles sont **"Auto" "No" et "Yes"**

En accélérant la vitesse de rafraîchissement...Il faut modifier la base de registre NT "ce qui reste délicat"



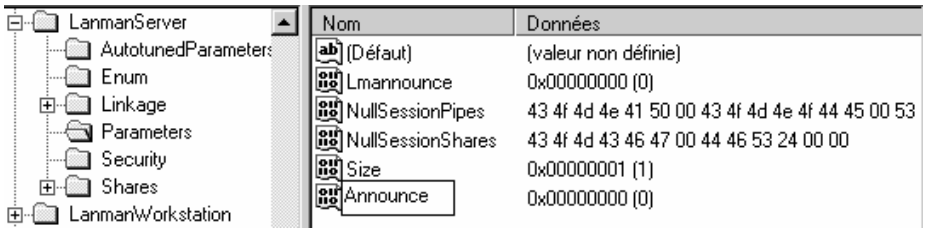
Il faut se positionner sur la clé **HKEY_LOCAL_MACHINE\SYSTEM\ CurrentControlSet\Services\LanmanServer\ Parameters**

et y créer une clé de type DWORD-value

en allant dans le menu

Edition / nouveau / valeur Dword

et y entrer la clé **Announce**



cette valeur Announce il faut ensuite la modifier via le menu

Edition / modifier

une valeur de 60 secondes (3c hexa) semble un bon compromis entre vitesse et nombre de trâmes...

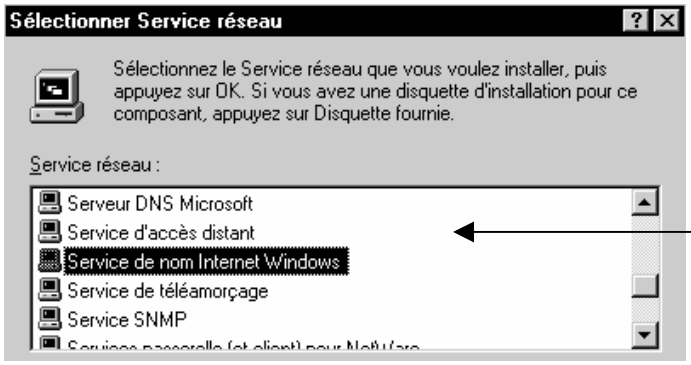
WINS WINDOWS N.T.4.0

Installer le Service Wins sous NT 4.0:

Pour installer un serveur WINS il faut être sur un PDC, un CSD ou un serveur autonome

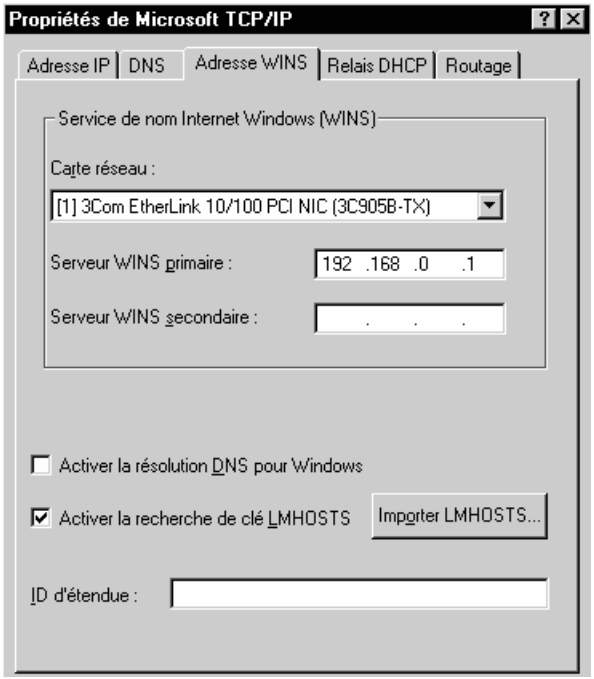
Il faut demander d'ajouter un service via le menu

propriété de Voisinage réseau, onglet service



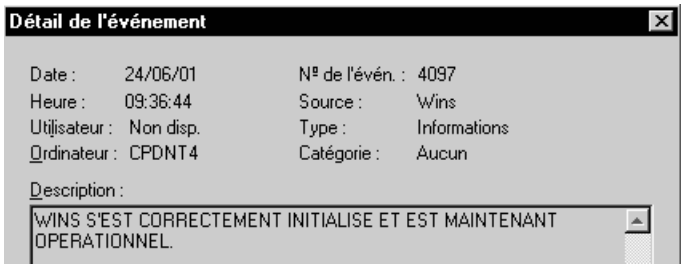
Service de nom Internet Windows

Il faut ensuite aller dans les propriétés du protocole TCP/IP et demander l'onglet Adresse WINS



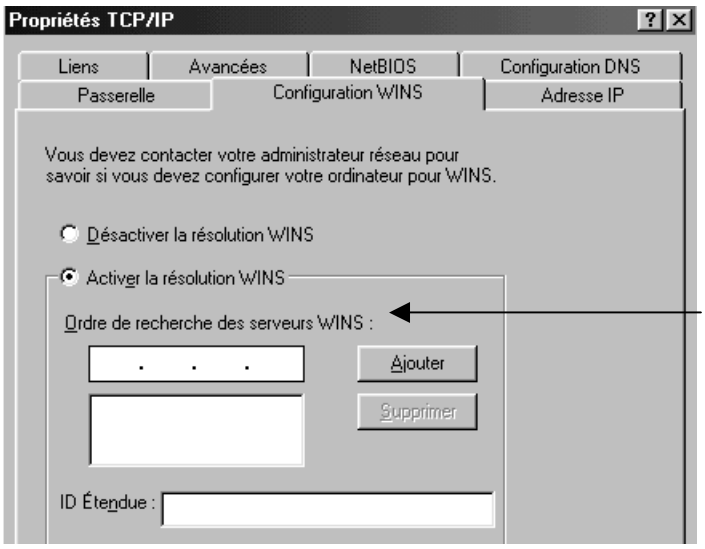
Dans lequel il suffit de donner l'adresse IP du serveur WINS ! (ne pas entrer 127.0.0.1)

On peut vérifier que l'installation s'est correctement effectuée dans le journal d'évènement :



Déclarer un client Wins:

Il suffit simplement depuis une machine NT de demander propriété du protocole TCP/IP, et de se positionner sur l'onglet WINS



Qu'il s'agisse d'un client NT ou d'un client Windows, il suffit de donner l'adresse IP du serveur WINS !

A partir de là on est en Type de Noeud "mode Hybride", et non plus en "diffusion" ...

```
D:\>ipconfig /all

Configuration IP de Windows NT
        Nom d'hôte . . . . . . . . . . . . : wksnt4
        Serveurs DNS . . . . . . . . . . . :
        Type de noeud. . . . . . . . . . . : Hybride
        Id d'étendue NetBIOS . . . . . . . :
        Routage IP activé. . . . . . . . . : Non
        WINS Proxy activé. . . . . . . . . : Non
        Résolution NetBIOS utilisant DNS . : Non

Ethernet carte Elnk31 :

        Description. . . . . . . . . . . . : ELNK3 Ethernet Adapter.
        Adresse physique . . . . . . . . . : 00-20-AF-C4-6A-98
        DHCP activé. . . . . . . . . . . . : Non
        Adresse IP . . . . . . . . . . . . : 192.168.0.3
        Masque de sous-réseau. . . . . . . : 255.255.255.0
        Passerelle par défaut. . . . . . . :
        Serveur WINS primaire. . . . . . . : 198.168.0.1
```

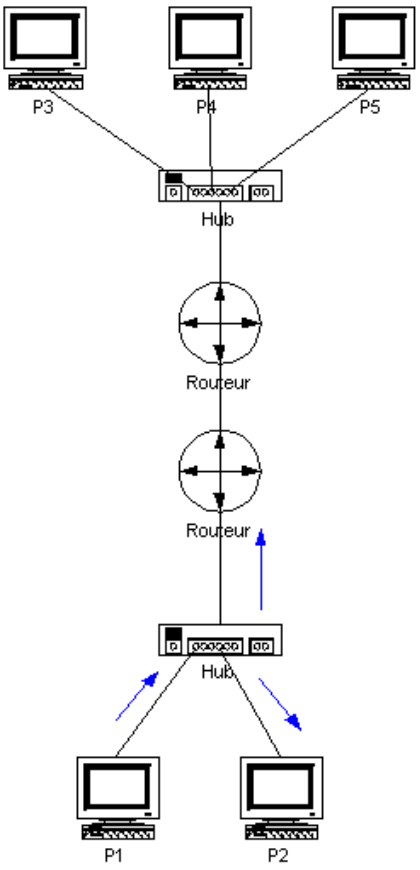
ANNEXE : TRAMES TCP/IP

Broadcast :

Le principe du broadcast est d'envoyer une information à tous les ordinateurs du réseau où l'on est. Au lieu d'envoyer en unicast vers l'adresse IP de la chaque machine (ex. 193.169.1.37 avec un masque 255.255.255.0),

on envoie la trame à tous les ordinateurs du sous-réseau en utilisant l'adresse de broadcast (ici, 193.169.1.255). Cette adresse est réservée à cet usage. Chacun des ordinateurs du sous-réseau regarde et traite la trame comme si elle leur était personnellement adressée.

Les trames de broadcast ont une caractéristique particulière : c'est de ne pas pouvoir passer les routeurs puisqu'il s'adresse uniquement à tous les ordinateurs d'un même sous-réseau.



Broadcast

P1 envoie des informations à tous les éléments de son sous-réseau

Unicast :

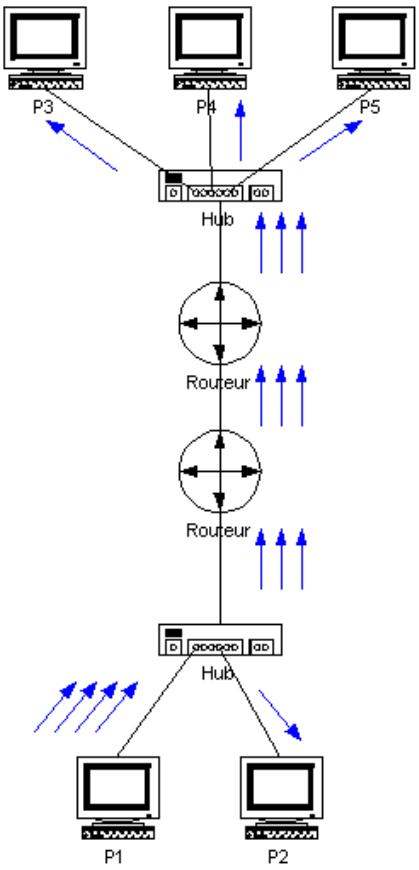
C'est le principe le plus utilisé et le plus simple. Les ordinateurs possédant chacun une adresse IP, on peut envoyer les trames en spécifiant l'adresse IP de l'ordinateur à qui on veut envoyer les informations. Les éléments actifs et passifs du réseau (commutateurs, répéteurs, routeurs, ...) dirigent l'information dans la bonne direction pour que les trames arrivent au bon endroit. Seule la machine ayant l'adresse contenue dans la trame regarde et traite l'information.

Il existe 3 classes d'adresses unicast :

La classe A : Adresses comprises entre 1.0.0.x et 127.255.255.x

La classe B : Adresses comprises entre 128.0.0.x et 191.255.255.x

La classe C : Adresses comprises entre 192.0.0.x et 223.255.255.x



Unicast

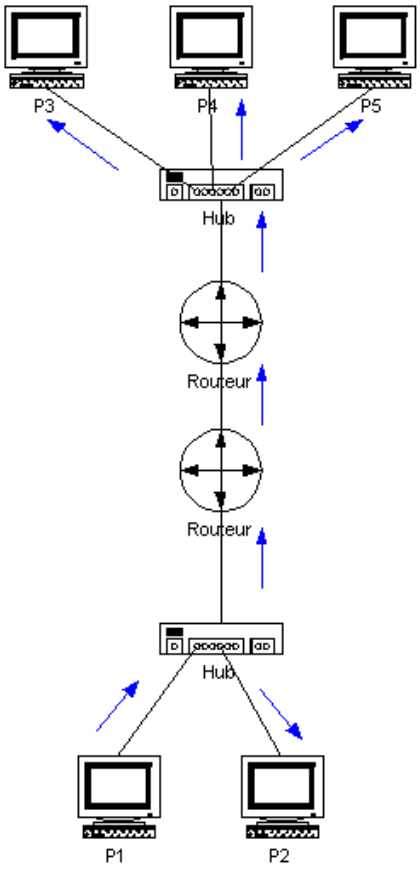
P1 envoie des informations à P2, P3, P4 et P5

Multicast :

Plutôt que d'envoyer les fichiers du serveur vers chacune des machines clientes (unicast) on peut n'envoyer l'information qu'une seule fois et chaque ordinateur client la récupère. En effet, dans un réseau Ethernet par exemple, toutes les trames qui circulent passent par tous les ordinateurs. C'est le principe du multicast : on envoie l'information à une adresse et tous les clients écoutent cette adresse.

Chaque client multicast s'enregistre avec une adresse IP multicast de classe D (entre 224.0.0.0 et 239.255.255.255 sauf 224.0.0.0 non utilisée et 224.0.0.1 qui correspond au "broadcast du multicast"). C'est sur cette adresse que les informations vont être envoyées.

Les clients écoutent ce qui arrive sur cette adresse et suivent la procédure décrite par le protocole multicast implémenté.



Multicast
P1 envoie des informations à P2, P3, P4 et P5

ANNEXE : CALCUL ADRESSES

Questions sur les classes - masques sous-réseaux:

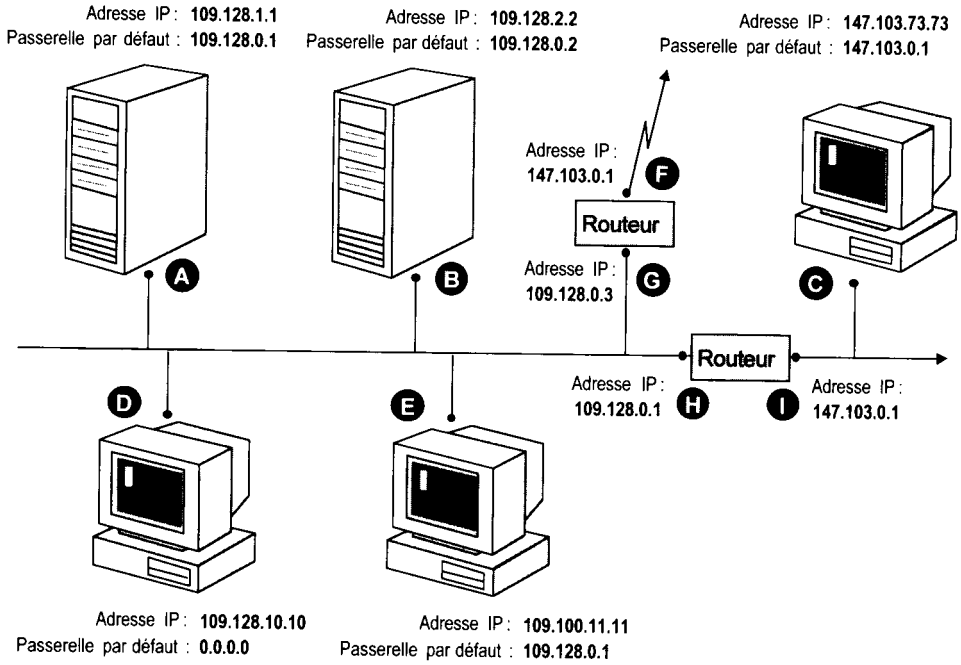
1. Quelle est la classe de l'adresse 127.0.0.1 ?
2. Quelle est la classe de l'adresse 21.34.55.55 ?
3. Quelle est la classe de l'adresse 223.75.234.239 ?
4. Quelle est la classe de l'adresse 223.322.232.127 ?
5. Quelle est la classe de l'adresse 192.192.232.127 ?
6. Quelle est la classe de l'adresse 44.12.256.254 ?
7. Quelle est la classe de l'adresse 126.122.243.34 ?
8. Quelle est l'ID par défaut de réseau de l'adresse 201.102.2.12 ?
9. Quelle est l'ID par défaut de réseau de l'adresse 121.212.112.122 ?
10. Quelle est l'ID par défaut de réseau de l'adresse 198.81.91.119 ?
11. Quelle est l'ID par défaut de réseau de l'adresse 201.44.45.54 ?
12. Quelle est l'ID par défaut d'hôte de l'adresse 179.79.234.234 ?
13. Quelle est l'ID par défaut d'hôte de l'adresse 41.1.6.222 ?
14. Quelle est l'ID par défaut d'hôte de l'adresse 201.44.45.54 ?

Réponses sur les classes - masques sous-réseaux:

1. Adresse test boucle
2. Classe A
3. Classe C
4. Classe illegale
5. Classe C
6. Classe illegale
7. Classe A
8. Classe C donc masque 255.255.255.0 donc ID réseau 201.102.21.0
9. Classe A donc masque 255.0.0.0 donc ID réseau 121.0.0.0
10. Classe C donc masque 255.255.255.0 donc ID réseau 198.81.91.0
11. Classe C donc masque 255.255.255.0 donc ID réseau 201.44.45.0
12. Classe B donc masque 255.255.0.0 donc ID hôte 0.0.234.234
13. Classe A donc masque 255.0.0.0 donc ID hôte 0.1.6.222
14. Classe C donc masque 255.255.255.0 donc ID hôte 0.0.0.54

ANNEXE : 1° DEPANNAGE RESEAU

Détecter les problèmes d'adressage de ce réseau:



hôte B :

l'adresse de la passerelle par défaut est fausse : Cette machine ne pourra que communiquer localement

hôte D :

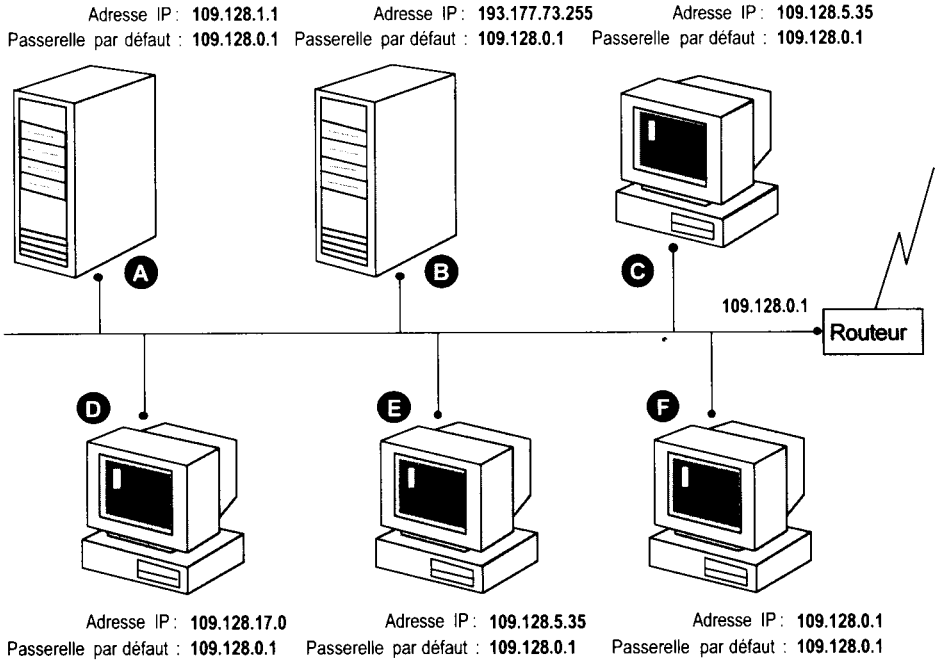
Pas de passerelle par défaut : Cette machine ne pourra que communiquer localement

hôte F - I :

Leur adresse IP est commune, cela risque de poser problème à un moment donné si on renseigne plusieurs passerelles…

ANNEXE : 2° DEPANNAGE RESEAU

Détecter les problèmes d'adressage de ce réseau:



hôte C - E :

Leur adresse étant en double ces deux machines risquent de voir leur interface réseau désactivée

hôte B :

Son adresse IP possède un ID réseau différent de celui des autres postes (193.177.0.0) il ne pourra ni atteindre les autres postes du réseau local, ni les machines distantes car sa passerelle à un ID réseau différent aussi…

hôte F :

Son adresse IP est la même que celle de la passerelle par défaut… il ne pourra ni atteindre les autres postes du réseau local, ni les machines distantes

ANNEXE : 1° CALCUL RESEAU

Données exemple :

Avec une adresse donnée par votre organisation de type 168.20.0.0

On vous demande de créer 5 sous-réseaux :

indiquez combien de machines vous pourrez piloter par sous-réseau, calculez le nouveau masque de sous-réseau ,les ID réseaux disponibles, les plages Id hôtes utilisables !

Solution :

l'adresse 168.20.0.0 est une adresse de classe B, son masque de sous-réseau par défaut est de 255.255.0.0 et il peut y avoir 65000 postes sur le réseau.

l'Id réseau de base est 168.20.0.0

masque de sous-réseau

Pour créer 5 sous réseaux, il faut ajouter 3 bits supplémentaires au masque de sous réseau, ce qui me permettra d'ailleurs à terme de pouvoir créer 6 sous-réseaux (croissance de 20% possible) chacun de ces sous-réseau pourra voir 8000 postes maxi (soit 48000 par rapport a 65000 en un seul sous-réseau)

la conversion de ces 3 bits 111 en un octet donne 11100000 soit en décimal 128+64+32 soit 224 en décimal

le masque de sous réseau par défaut est de 255.255.0.0 il devient pour mon réseau alors 255.255.224.0

Id réseau

pour trouver les Id réseau je dois trouver toutes les combinaisons de 111 à 000 en laissant tomber les valeurs n'ayant que des 0 ou que des 1

j'obtient 110-101-011-100-010-001

soit en décimal 192-160-128-96-64-32

que je rajoute à mon Id réseau d'origine 168.20 soit donc les Id réseau suivantes :

168.20.192.0168.20.160.0 168.20.128.0 168.20.96.0

168.20.64.0 168.20.32.0

toutes ayant comme masque de sous-réseau 255.255.224.0

plages Id hôtes valides

un petit calcul nous donne :

sous-réseau	1° adresse IP	dernière adresse IP
168.20.32.0	168.20.32.1	168.20.63.254
168.20.64.0	168.20.64.1	168.20.95.254
168.20.96.0	168.20.96.1	168.20.127.254
168.20.128.0	168.20.128.1	168.20.159.254
168.20.160.0	168.20.160.1	168.20.191.254
168.20.192.0	168.20.192.1	168.20.223.254

ANNEXE : 2° CALCUL RESEAU

Données exemple :

Vous avez à mettre en place un réseau globalement comportant une trentaine de machines au grand maximum

Il faut soigneusement prévoir de protéger ce réseau d'accès intempestif et différencier le trafic de la salle pédagogique (1 salles d'une dizaine de poste maxi), des administratifs (une douzaine de poste) et des postes de production par ailleurs (une vingtaine de poste environ). Deux postes échangerons un fort volume de données, ceux de la cartographie, mais on voudrait bien qu'ils n'encombrent pas le réseaux à eux deux !

Solution :

vu le nombre de machines, 30 une adresse de classe C suffira amplement. n'ayant absolument aucune ambition d'avoir des adresses publiques, une adresse privée sera parfaite, et on prendra la première de la liste à savoir 192.168.1.0

l'adresse 192.168.1.0 étant une adresse de classe C, son masque de sous-réseau par défaut est de 255.255.255.0 et il peut y avoir 255 postes sur le réseau.

l'Id réseau de base est 192.168.1.0

masque de sous-réseau

je voudrais créer 4 sous réseaux, respectivement

1 pour la salle pédagogique	10 postes
1 pour les administratifs	12 postes
1 pour la production	20 postes
1 pour la cartographie	2 postes

En prévoyant une augmentation moyenne de 50% j'arrive à un besoin de 6 réseaux de 30 postes pour le plus important…

Pour créer 6 sous réseaux, il faut ajouter 3 bits supplémentaires au masque de sous réseau, chacun de ces sous-réseau pourra voir (2^5)-2 postes maxi (soit 6 réseaux de 30 postes c'est à dire 180 postes par rapport a 254 machines en un seul sous-réseau)

la conversion de ces 3 bits 111 en un octet donne 11100000 soit en décimal 128+64+32 soit 224 en décimal

le masque de sous réseau par défaut est de 255.255.255.0 il devient pour mon réseau alors 255.255.255.224

Id réseau

pour trouver les Id réseau je dois trouver toutes les combinaisons de 111 à 000 en laissant tomber les valeurs n'ayant que des 0 ou que des 1

j'obtient 110-101-011-100-010-001

soit en décimal 192-160-128-96-64-32

que je rajoute à mon Id réseau d'origine 192.168.1.0 soit donc les Id réseau suivantes :

192.168.1.192 192.168.1.160 192.168.1.128 192.168.1.96

192.168.1.64 192.168.1.32

toutes ayant comme masque de sous-réseau 255.255.255.224

plages Id hôtes valides

un petit calcul nous donne :

sous-réseau	**1° adresse IP**	**dernière adresse IP**
192.168.1.32	192.168.1.33	192.168.1.63
192.168.1.64	192.168.1.65	192.168.1.95
192.168.1.96	192.168.1.97	192.168.1.127
192.168.1.128	192.168.1.129	192.168.1.159
192.168.1.160	192.168.1.161	192.168.1.191
192.168.1.192	192.168.1.193	192.168.1.223