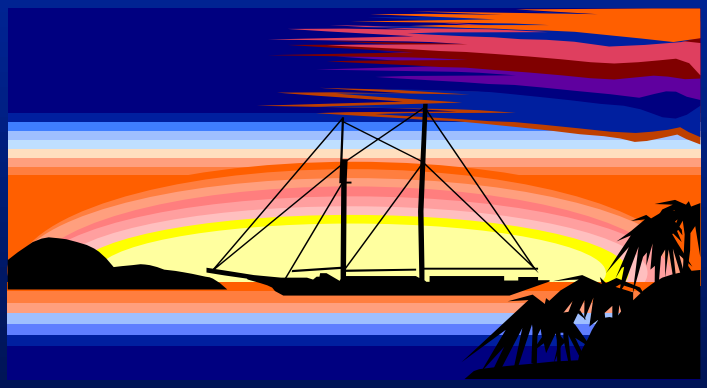
TCP/IP
Protocoles de l'Internet



cours@urec.cnrs.fr

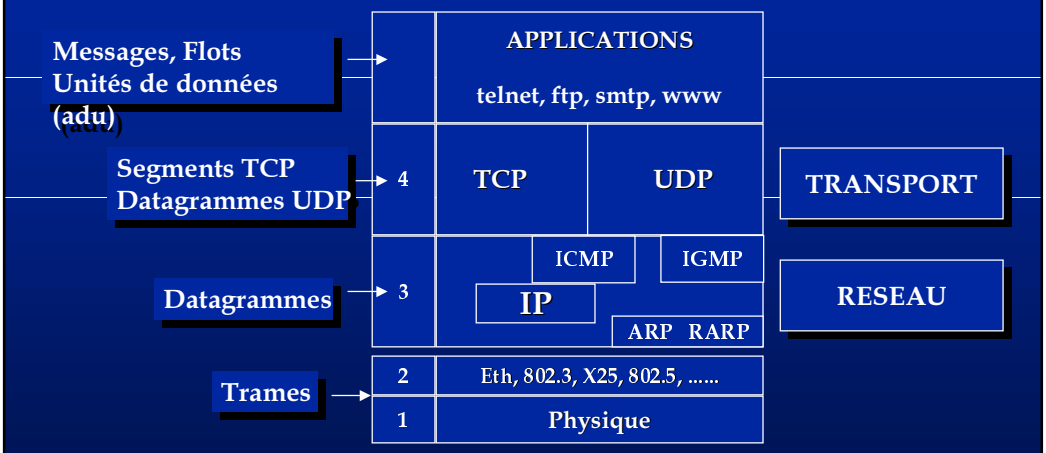
Protocoles de l'Internet

- 1992 : Jean-Luc Archimbaud
- modifications
  - 1993-1998 : Jean-Paul Gautier
  - 1999 : Vincent Roca

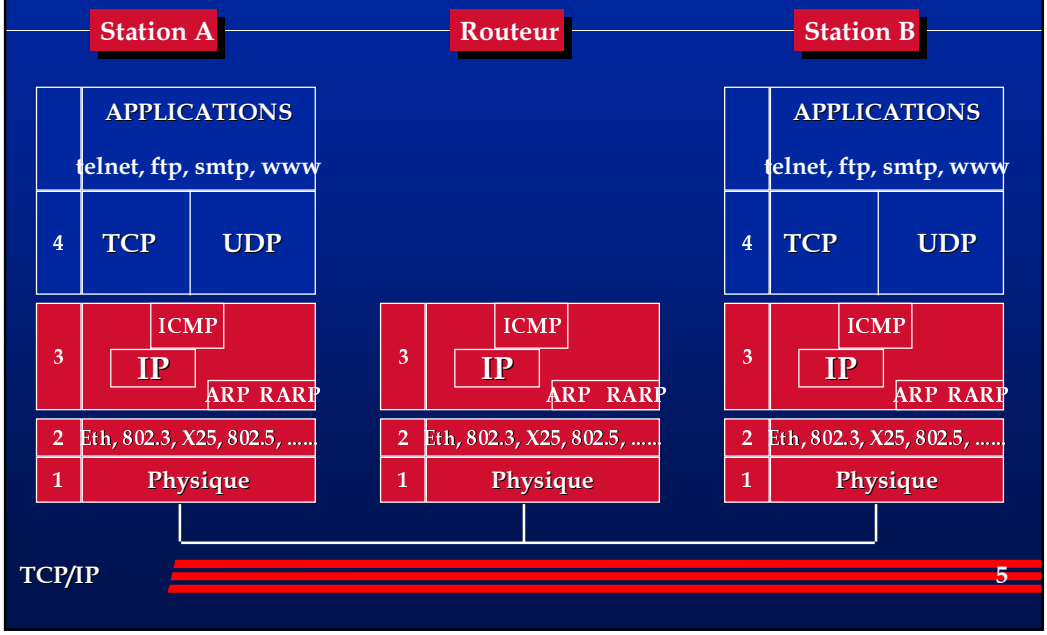
Protocoles de l'Internet : Plan

- Introduction
- Couche réseau : IP (fonctions, adressage, datagramme)
  (Internet Protocol)
- Mapping adresses IP-adresses physiques : ARP, RARP
  (Address Resolution Protocol, Reverse Address Resolution Protocol)
- ICMP
  (Internet control Message System)
- Couche transport : TCP et UDP
  (Transport Control Protocol, User Datagram Protocol)
- Fichiers & commandes sous Unix
- Exemples
- Bilan

Modèle en couches



Modèle en couches



Internet Historique en quelques mots

- Recherches du DARPA
  - Defense Advanced Projects Research Agency
- ARPANET fut le premier réseau à commutation de paquet au milieu des années 1970
  - protocoles sous leur forme actuelle en 1978-79.
  - réseau de liaisons point à point.
    - » exploration sur les liaisons radios et sattelites.
- La mise en oeuvre de TCP/IP en 1980 sur le réseau de recherche de DARPA est le début de l'Internet .
- La transition est complète quand DARPA exige que toutes les machines de ARPANET utilisent TCP/IP
- TCP/IP intégré à l'unix BSD
  - entrée dans le monde universitaire.
  - développement d'applications réseaux avec les *sockets*
- NSFnet en 1986, 12 réseaux régionaux

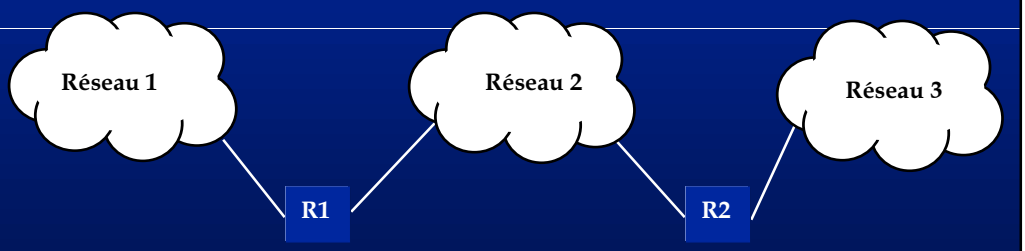
Internet — ISOC,IAB, IETF

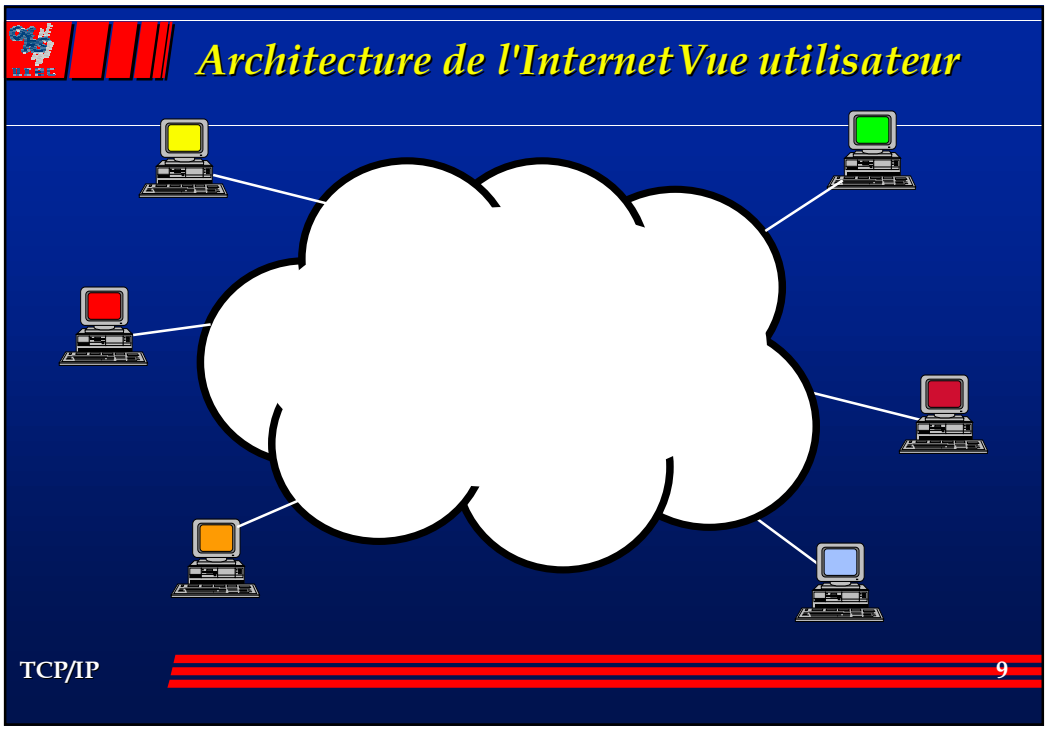
- ISOC
  - Internet Society
  - Pilote l 'IAB et l 'IETF
- Internet Architecture Board
  - organisation autonome, foyer de la recherche et du développement de l'Internet.
  - 10 "task forces".
  - plusieurs rencontres annuelles.
  - le chairman de l'IAB est *l'architecte de l'Internet*
- Internet Engineering Task Force
  - organisée en groupes de travail
  - rapports techniques : drafts Internet, RFCs (Request For Comments)
  - très ouvert (accès libre aux réunions/documents/groupes)
  - ID/RFCs accessibles sur: ftp.lip6.fr

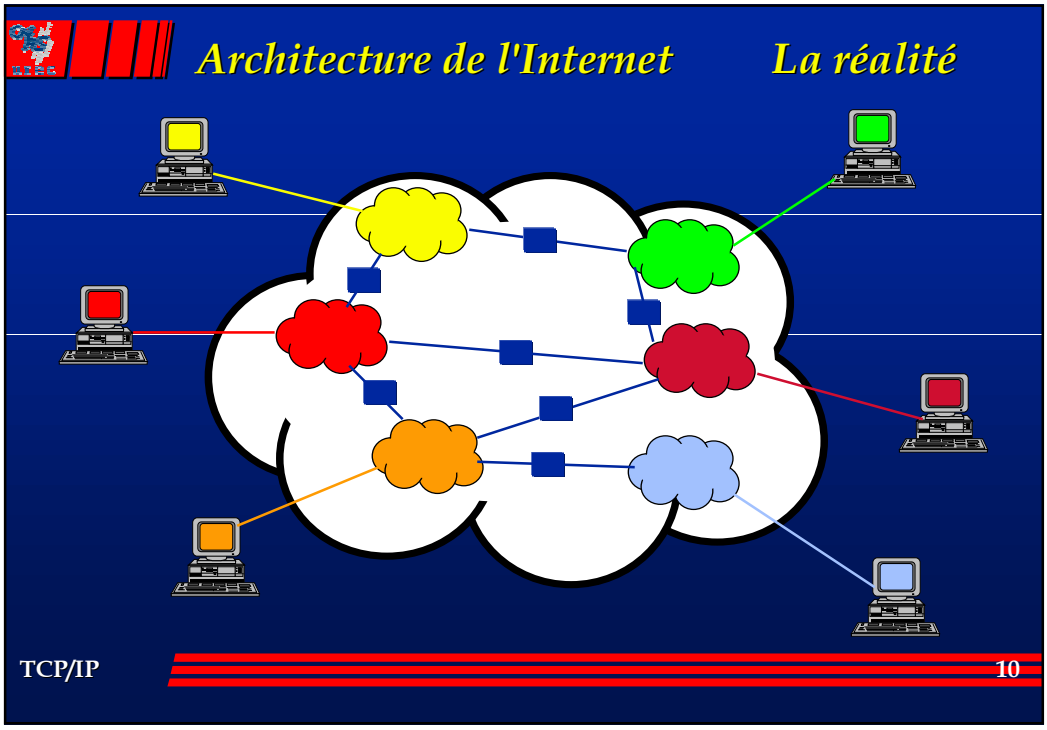
Architecture de l'internet

L'internet est un réseau virtuel construit en nterconnectant des réseaux physiques par des passerelles, les routeurs.

L'internet supporte un service de comunication universel




Architecture de l'Internet Vue utilisateur
TCP/IP
9

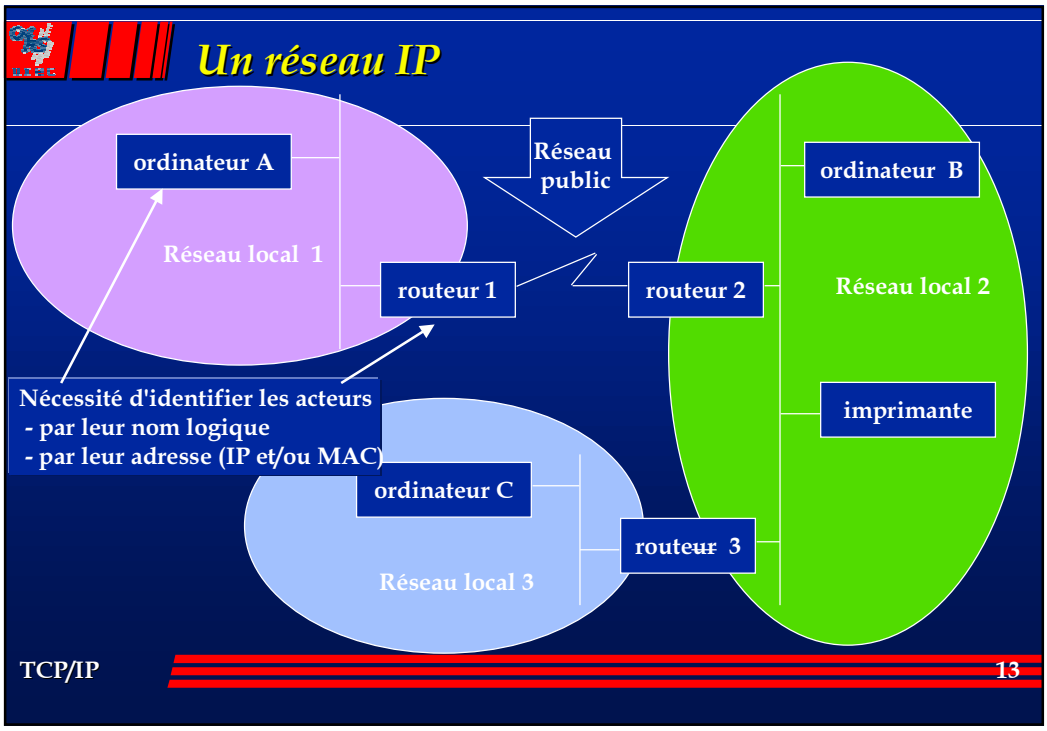

Architecture de l'Internet La réalité
TCP/IP
10

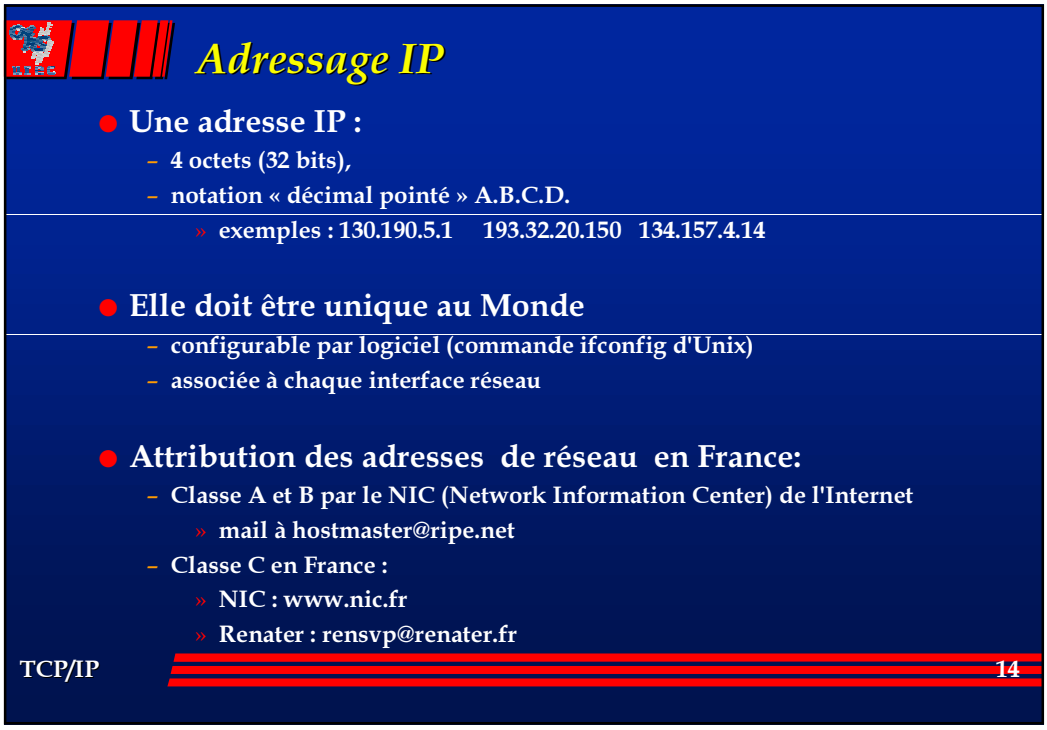
Les services de l'Internet

- ~~Interopérabilité au niveau des applications~~
- Les utilisateurs invoquent les applications sans avoir besoin de connaître la technologie de l'Internet ni son architecture.
- Les plus populaires sont
  - Le courrier électronique (smtp)
  - Le transfert de fichiers (ftp, tftp)
  - L'accès à l'information distante (www)
  - L'accès à des machines distantes (telnet)
  - Les forums (News)

IP

- Internet Protocol (RFC 791)
  - couche 3 du modèle OSI
- « IP au dessus de tout »
  - IP = protocole de convergence
  - Fonctionne sur :
    - ~~Ethernet (RFC894)~~
    - Token-Ring
    - Liaison série de 9.6 Kb/s à 2 Mb/s
      - SLIP (RFC1055)
      - PPP (Point to Point Protocol) (RFC1353), X25 (RFC877)
    - FDDI (RFC1188)
    - Ethernet 100 Mbps
    - ATM (RFC1483, RFC1577, LANE 1.0)
  - Raison : seuls des services d 'émission/réception sans garanties sont nécessaires


Un réseau IP
ordinateur A
Réseau local 1
Réseau public
routeur 1
routeur 2
ordinateur B
Réseau local 2
imprimante
Nécessité d'identifier les acteurs
- par leur nom logique
- par leur adresse (IP et/ou MAC)
ordinateur C
routeur 3
Réseau local 3
TCP/IP
13


Adressage IP
Une adresse IP :
4 octets (32 bits),
notation « décimal pointé » A.B.C.D.
exemples : 130.190.5.1 193.32.20.150 134.157.4.14
Elle doit être unique au Monde
configurable par logiciel (commande ifconfig d'Unix)
associée à chaque interface réseau
Attribution des adresses de réseau en France:
Classe A et B par le NIC (Network Information Center) de l'Internet
mail à hostmaster@ripe.net
Classe C en France :
NIC : www.nic.fr
Renater : rensvp@renater.fr
TCP/IP
14

Adressage IP

- Découpée en deux :
  - adresse de réseau, ou *network id*
    assigné par une autorité, identifie le réseau
  - identificateur local de machine, ou *host id*
    assigné par l 'administrateur du réseau, identifie la machine sur le réseau
  - le découpage précis dépend de la classe d 'adresses...
- Classification :
  - » classe A : N.H.H.H
  - » classe B : N.N.H.H
  - » classe C : N.N.N.H
  - » N = adresse réseau
    H = adresse locale
  - » classe D : cas particulier, pas de distinction network/host
- L'espace d'adressage n'est pas hiérarchisé ou arborescent
  - » à la différence du téléphone, de Transpac, d'ATM, d'IPv6

Adressage IP : classe A

- 7 bits pour le numéro de réseau
  - 1.0.0.0 à 126.0.0.0
- 24 bits pour l'adressage local
  - 254^3 @ locales possibles (16,277,214)
- En France, pas de réseau de classe A
  - Ex : 16.0.0.0 (DEC) 18.0.0.0 (MIT)

Adressage IP : classe B

- 16 bits pour le numéro de réseau
  - 128.1..0.0 à 191.255.0.0
- 16 bits pour l'adressage local
  - 254 x 254 @ locales possibles (65,534)
    - Ex : 129.88 (IMAG) 134.157 (Jussieu)
- quasi épuisée...

1 1 14 16

Adressage IP : classe C

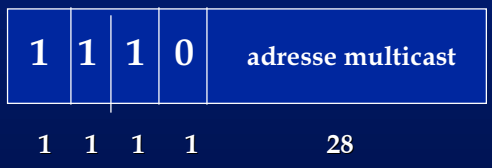
- 24 bits pour le numéro de réseau
  - 192.0.1.0 à 223.255.255.0
- 8 bits pour l'adressage local
  - 254 @ locales possibles
    - Ex : 192.33.181 (IBP) 192.70.89 (CITI2)

1 1 1 21 8

Adressage IP : classe D

- adresses multicast (RFC 1700)
  - transmissions point à multipoint; exemple vidéo-conférence
- réseaux 224 à 231
  - ex : 224.4.4.4
- pas de structuration…

  … car utilisée de façon très spéciale, ponctuelle, sans contrainte d 'unicité, sans organisation gérant leur attribution

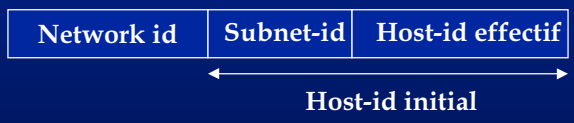


Adressage IP : particularités

- Classe E : 239 < H< 254 Reserved for Futur Use
- Adresses particulières
  - soi-même : 127.0.0.1 (loopback ou localhost)
    - » test logiciels, communication inter-processus sur la station
  - tous les bits de la partie machine à 0 => le réseau
    - » 130.190.0.0 désigne le réseau de classe B : 130.190
  - tous les bits de la partie machine à 1 => tous les hosts du réseau
    - » diffusion, broadcast IP
    - » 130.190.255.255 désigne toutes les machines du réseau 130.190
  - 0.0.0.0 : une machine ne connaît pas son adresse
    - » (station sans disque qui utilise RARP)

Sous-réseaux IP

- Découpage d'un réseau en entités plus petites
  - sous-réseau ou "subnet"
  - permet meilleure structuration du réseau du site
  - décidé par l'administrateur du site
  - adresse de sous-réseaux prélevé sur la partie « host-id »
  - longueur comptée en bits décidée par l 'administrateur

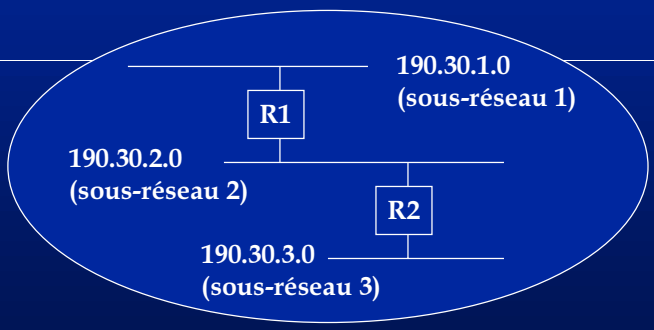


  - tous les équipements réseaux doivent utiliser la notion de sous-réseau (stations, serveurs de terminaux, routeurs, imprimantes…)
  - interconnexion des sous-réseaux impérativement par des routeurs

Sous-réseaux IP

- Exemple :
  - découpage en 3 sous-réseaux, numérotation par le 3ème octet

190.30.0.0 adresse réseau
du site connue de l'extérieur

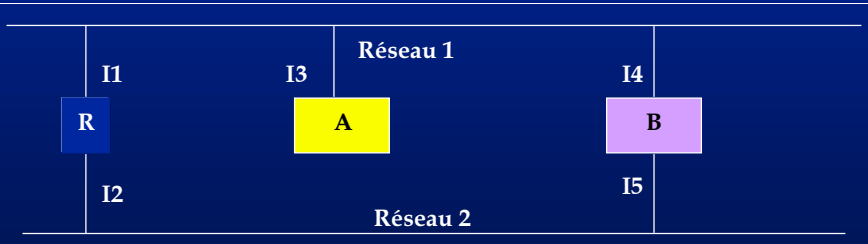


Sous-réseaux IP

- ~~le découpage est inconnu de l'extérieur !~~
- passe par l 'utilisation d 'un subnet-mask
  - même notation que l'adresse IP:
    - bits réseau à 1
    - bits de la partie sous-réseau à 1
    - bits de la partie "host" à 0
  - exemple : 130.190.0.0 , réseau de classe B
    - masque par défaut: 255.255.0.0 si pas de subnet
    - masque 255.255.255.0 si présence de (au plus 254) sous-réseaux
- utilisation :
  - @IP & subnet_mask = adresse network +subnet
    - utilisée pour le routage local au site
  - @IP & ~(subnet_mask) = host id effectif

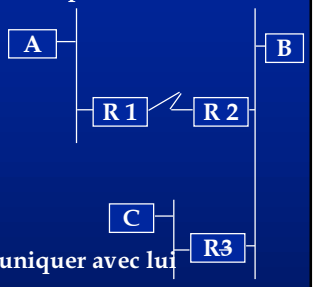
Points faibles de l'adressage IP

- Si une machine change de réseau, son adresse doit changer
- Le routage utilisant la partie réseau de l'adresse, le chemin suivi par les datagrammes vers un host avec de multiples adresses IP dépend de l'adresse utilisée.



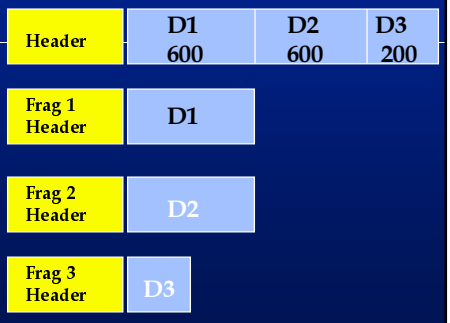
IP : Fonctions

- Transporte des datagrammes de bout en bout
  - Pour aller de l' équipement A à l'équipement C, le datagramme passe par R 1, R 2, R3
  - Chaque datagramme contient
    - » l'adresse IP (Internet) de l'émetteur
    - » l'adresse IP (Internet) du destinataire
    - » chaque interface d'un équipement a une adresse IP
      - la commande ifconfig d'Unix
  - Il faut connaître l'adresse IP d'un équipement pour communiquer avec lui
  - C'est un mode sans connexion
    - » chaque datagramme est traité indépendamment des autres
  - Sans garantie de remise des datagrammes (unreliable)
    - » IP fait au mieux (Best Effort)



IP : Fonctions

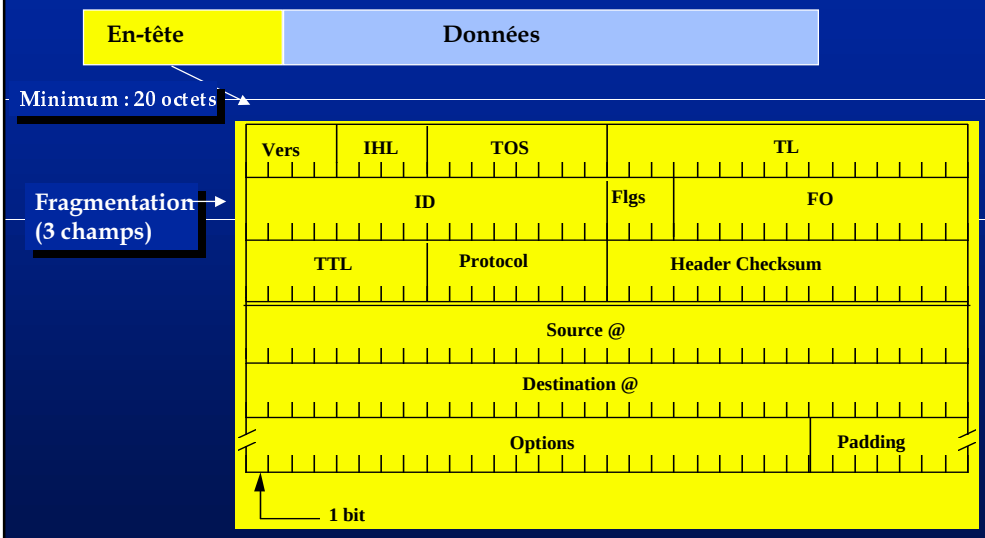
- Assure le routage : savoir où envoyer le datagramme.
  - Les équipements IP ne connaissent que le prochain équipement sur lr chemin (next hop).
- La fragmentation
  - C'est la machine destinataire qui réassemble non le routeur à la frontière d'un type de réseau



IP : Fonctions

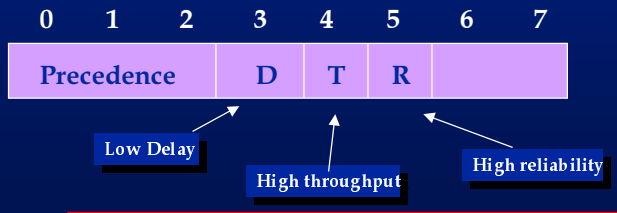
- IP n'assure pas
  - le multiplexage
  - la vérification du séquencement
  - la détection de perte
  - la retransmission en cas d' erreur
  - le contrôle de flux
    - » ICMP assure partiellement cette fonction

Datagramme IP



Datagramme IP

- Vers : Version du protocole
  - Actuellement 4
- IHL : Internet Header Length
  - En nombre de mots de 32 bits
  - Généralement 5, sans options l'en-tête est de 20 octets
- TOS : Type Of Service (qualité de service)
  - Etait prévu pour routage avec contrainte de qualité de service… mais n 'a pas été utilisé !
  - Precedence (0-7 indiquant l'importance du datagramme)



Datagramme IP

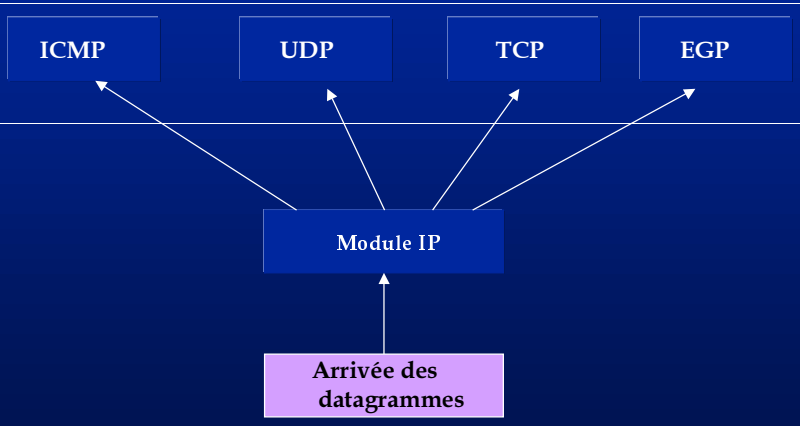
- TL : Total Length
  - Longueur du datagramme, incluant l'entête
  - Unité = octet
  - Maximum : 64 Koctets
  - Recommandé : moins de 576 octets
- TTL : Time To Live
  - Théoriquement exprimé en incréments de 1 seconde
  - L'expéditeur le met à une certaine valeur
  - Décrémenté de 1 (ou plus) à chaque traversée de routeur
  - Le datagramme est détruit quand TTL=0
  - Evite au datagramme de circuler éternellement en cas de boucle

Datagramme IP

- Protocol : Identifie le protocole de la couche supérieure
  - 6 => TCP
  - 17 => UDP
  - 1 => ICMP
- Header Checksum
  - couvre l 'entête IP uniquement
  - but: vérifier son intégrité
  - recalculé par chaque routeur (puisque le champ TTL est modifié)
  - mais ne couvre pas les données (de la responsabilité du transport)

Datagramme IP

- Démultiplexage au niveau de la couche IP
  - champ protocole du l'en-tête



Datagramme IP

- Champs liés à la fragmentation IP
  - ID : Identification du datagramme
    - Utilisé par l'émétteur et le destinataire pour identifier le datagramme
    - Numérotation faite par l'émetteur
    - Uniquement utilisé pour la fragmentation
  - Flags : Flags pour la fragmentation
    - 001 : il y a encore des fragments
    - 000 : dernier fragment (ou pas encore fragmenté)
    - 01X : ne pas fragmenter
  - FO : Fragment offset
    - position du fragment dans le datagramme d 'origine,
    - calculé en unité de 8 octets,
    - premier fragment = 0
    - Le destinataire doit récupérer tout les fragments, si un fragment est perdu tout le datagramme est jeté.

Datagramme IP

- Source @ : Adresse IP de l'émetteur
- Destination @ : Adresse IP du destinataire
- Ce sont les adresses d 'extrémité, pas les nœuds intermédiaires !

Datagramme IP

- Options : variables en taille, permettent des extensions
  - » Certaines options sont standards (décrites dans des RFCs).
    - Exemples : niveau de sécurité, time stamp (chaque routeur ajoute l'heure de passage)
  - » Elles se composent
    - du code de l'option 1 octet
    - de la longueur de l'option 1 octet
    - des données associées

- Padding : Compléte le champs options
  - Pour que la longueur de l'en-tête soit un multiple de 32.

Remarque :
Taille de l'entête est importante (> 20 octets)

- » IP ne peut pas utiliser un lien < 4.8 Kb/s
- » Certains protocoles permettent la compression de l'entête IP

Interface IP-Ethernet (RFC894)

- Datagramme IP dans une trame Ethernet
  - Champ Type = 0800 (Hex)

- Datagramme IP dans une trame IEEE802.3 (RFC1042)
  - LLC(3 octets) = DSAP (=170) + SSAP (=170) + Control (=3)
    - » (SAP : Service Access Protocol)
  - SNAP (5 octets) (Sub-Network Access Protocol)
    - » Protocol ID (=0 sur 3 octets)
    - » Ethernet Type (= 2048 sur 2 octets = 0800 hex)

Interface IP-X25 (RFC877)

- @IP ---> @X25 (@ X121)
  - X25 n'a pas de notion de broadcast.
  - On ne peut donc pas utiliser le protocole ARP
  - On associe une adresse X25 à une adresse IP manuellement
  - C'est statique, donc lourd à gérer
- Les datagrammes IP sont fragmentés à 576 octets
  - Un paquet X25 1984 ne supporte qu'une taille maximale de 128 octets
    - » utilisation du bit M de x25 pour découper le datagramme IP

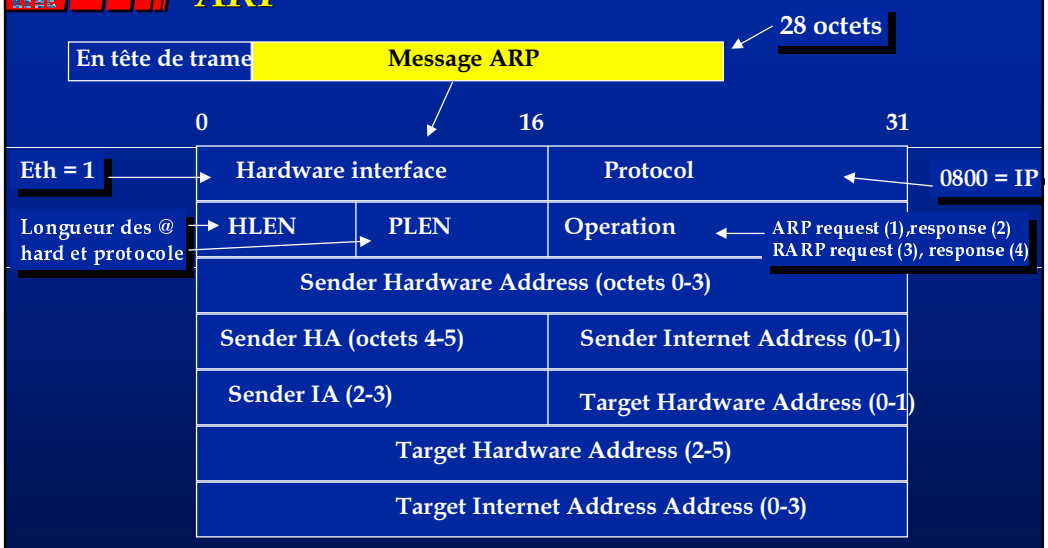
ARP (RFC826)

- Problème : Trouver une adresse MAC à partir de l'adresse IP
- Address Resolution Protocol
  - Permet de trouver l'adresse physique d'une machine sur le même réseau en donnant uniquement son adresse IP
- L'adresse IP est totalement indépendante de l'adresse physique
  - exemple Ethernet : l'adresse MAC est fournie par le constructeur.
- Stockage des @ physiques dans une table ARP (cache).
  - Le cache est remis à jour périodiquement
  - Sous Unix, pour Ethernet, visualisation de la table par la commande : arp -a

ARP

- Soit deux équipements sur le même segment Ethernet.
- La machine A veut envoyer un datagramme à la machine B.
- Elle connait son adresse IP, mais pas son adresse Ethernet :
  - A envoie une trame de broadcast Ethernet qui demande l'adresse Ethernet de B :
    - » adresse destinataire FF.FF.FF.FF.FF.FF avec Type = 0806
    - » en indiquant l'adresse IP de B.
  - Toutes les machines reçoivent la requête.
  - Seul B répond à A en lui donnant son adresse Ethernet.
  - Si c'est une autre machine qui répond à la place de A on parle alors de "Proxy ARP".
    - » utilisation par les routeurs quand le destinataire est dans un autre réseau IP

ARP



RARP (RFC903)

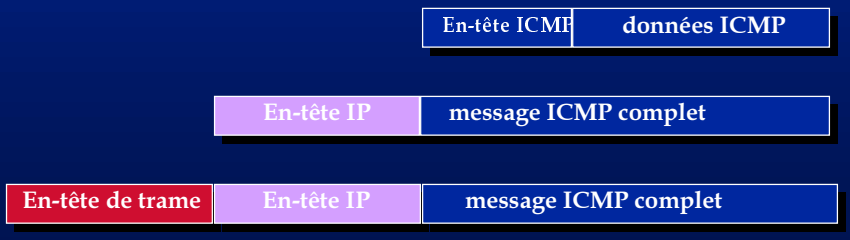
- ~~Problème :~~
  - Trouver une adresse IP à partir de l'adresse Ethernet
- Reverse Address Resolution Protocol
  - Permet de demander une adresse IP en indiquant l'adresse Ethernet.
  - Utilisé au moment du "boot" par certains équipements.
    - » envoie son adresse MAC dans le champ "Target HA"
- Type = 8035 dans la trame Ethernet
- Ethernet type = 32821 dans la trame IEEE802.3
- Utilisé par
  - les Macintosh avec une boite Kinetics
  - les stations sans disque
  - les terminaux X
- Même format de message que ARP

ICMP (RFC792)

- Internet Control Message Protocol
  - Protocole de "gestion" de réseau = *mécanisme de rapport d'erreur.*
- Implémenté sur tous les équipements IP : stations, routeurs.
- Message envoyé par l'équipement destinataire ou un routeur intermédiaire
  - Quand il s'aperçoit d'un problème dans un datagramme
  - Pour avertir l'émetteur afin qu'il modifie son comportement.
  - ex : routeur qui a une mauvaise information de routage.
- Un message ICMP ne doit pas engendrer un autre message ICMP
  - Il ne demande pas de réponse

ICMP

- Un message ICMP est contenu dans un datagramme IP
  - Utilise IP comme un protocole de couche supérieure.
  - Champ protocole du datagramme IP = 1
- Chaque message ICMP a son format
  - 3 champs communs
    - » TYPE (1 octet), 22 types définis
    - » CODE (1 octet), plus d'information sur le champ type.
    - » CHECKSUM (2 octets), sur le message ICMP.



ICMP

- Le champ type
  - 0 Réponse d écho
  - 3 Destination inaccessible
  - 4 « Source Quench » (demande de ralentissement)
  - 5 Redirection
  - 8 Echo
  - 9 Annonce de routeur
  - 10 Sollicitation de routeur
  - 11 TTL expire
  - 12 Problème de paramétrage
  - 13 Horadatage (Time Stamp)
  - 14 Réponse horodatage
  - 15 Demande d 'information
  - 16 Réponse a la demande d 'information
  - 17 Demande de « netmask »
  - 18 Réponse a la demande de « netmask »
  - 30 Traceroute
  - 31 Erreur de conversion des datagrammes
  - 30 Traceroute
  - 31 Erreur de conversion des datagrammes
  - 32 Redirection d 'un équipement mobile
  - 33 Localisation d 'un équipement IPv6
  - 34 Réponse a la demande de localisation d 'un équipement IPv6
  - 35 Demande d 'enregistrement d 'un équipement mobile
  - 37 Réponse a la demande d 'enregistrement d 'un équipement mobile

ICMP

- Type d'indication d'un message ICMP

(type message, code)

- Impossible d'atteindre le réseau (Network unreachable) (3,0)
- Host non atteignable (Host unreachable) (3,1)
- Port TCP or UDP (service) indisponible (Port unreachable) (3,3)
- Demande de ralentir l'émission (Source quench) (4,0)
- Durée de vie dépassée (Time to Live exceeded) (11,0)
- Redirection (Redirect, change a route) (5,0-3)
- Echo et réponse à echo (Echo Echo reply) , commande ping (8)
- Demande de "subnet mask" (Address Mask request) (17)
- Réponse de " subnet mask" (address mask Reply) (18)

- Permet de palier aux manques de services de IP

Couche Transport — Définitions

- Deux protocoles pour la communication entre applications
- TCP : Transmission Control Protocol
  - » protocole en mode orienté connexion
- UDP: User Datagram protocol
  - » protocole en mode sans connexion

Couche Transport — Définitions

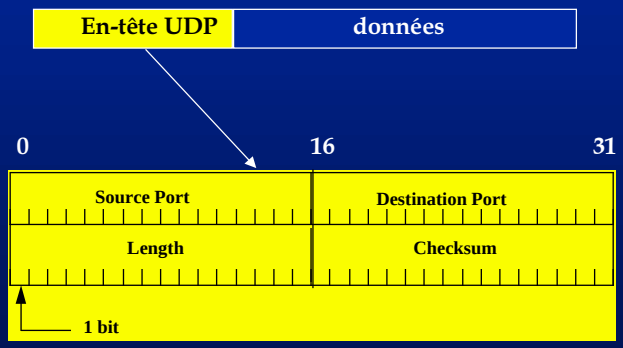
- Identification d'une application : numéro de port
  - le port est une destination abstraite utilisé par le protocole
- socket = Combinaison @ IP - Numéro de port
  - 130.190.5.1 - 23 est le démon telnetd sur la station 130.190.5.1
- La combinaison de 2 sockets définit complètement une connexion TCP ou un échange UDP
  - Exemple 130.190.5.1 - 23 et 147.171.150.2 - 1094
    - » Connexion entre un processus client qui a pris le numéro 1094 sur la machine 147.171.150.2 et le démon telnetd sur la machine 130.190.5.1
    - » Un utilisateur sur 147.171.150.2 a fait un telnet 130.190.5.1
    - » C'est ce que l'on peut voir avec la commande Unix *netstat -a*

Couche Transport — Définitions

- Port pré-définis (RFC 1060 "Assigned numbers") pour les services :
  - 20 FTP-Transfert
  - 21 FTP-Contrôle
  - 23 Telnet
  - 25 SMTP
  - 53 DNS (Domain Name Server)
  - 69 tftp
- Mode client serveur
  - serveur , on parle de démons dans Unix.
  - le client se voit attribué un numéro de port non affecté (>1000) pour éviter toute confusion avec les ports "officiels"
- Tous les équipements TCP/IP respectent cette attribution de ports pré-définis
  - fichier /etc/services d'Unix

UDP (RFC768)

- User Datagram Protocol
  - service sans connection, sans garantie, utilisant IP pour le transport de messages entre machines

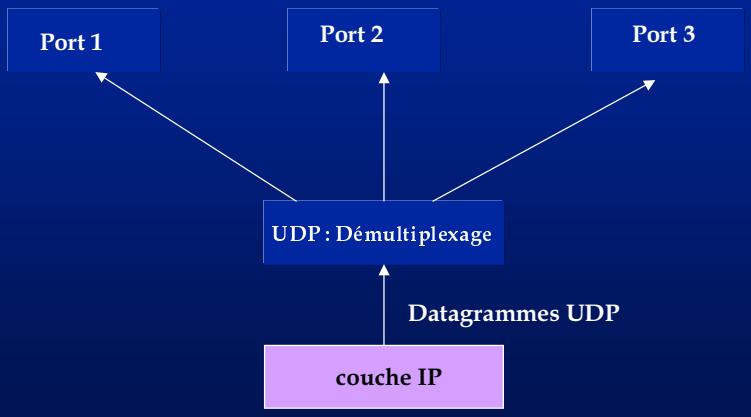


UDP : Champs

- Entête de 8 octets
- Source Port : numéro de port
  - optionnel, identifie un port pour la réponse.
- Destination Port : numéro de port
- Length, taille de l'entête et des données
  - Unité = octet
  - Taille maximale = 64 Koctets
- Checksum : fonction de l'entête et des données
  - optionnel
  - c'est la seule garantie sur la validité des données qui arrivent à destination
- Un datagramme UDP est contenu dans un datagramme IP

UDP

- Par rapport à IP, UDP rajoute l'information indiquant le service



UDP

- Ne fait pas
  - mode connecté,
  - retransmission si erreur ou perte,
  - séquencement,
  - contrôle de flux => très facile de saturer le réseau local et les routeurs !!!
- C'est un protocole de transport non fiable
- Utilisé par NFS
- Utilisé pour la diffusion
  - rwho
  - routed
  - tftp (Trivial FileTransfert Protocol) , port 69
  - ntp (Network Time Protocol), port 123
  - dès que le multicast est nécessaire

TCP (RFC793) — Fonctionnalités

- Transmission Control Protocol
- TCP ne tourne pas dans les routeurs !
  - uniquement aux extrémités (ordinateurs, les imprimantes…)
- Transport
  - De bout en bout entre applications
  - En mode connecté : ouverture ... fermeture (circuit virtuel)
  - Sans erreur : contrôle et retransmission si nécessaire
  - Sans perte : "numérotation" et retransmission
  - Ordonné : bon séquencement
  - Système d'acquittement
  - Avec contrôle de flux (fenêtre d'émission)
  - Full duplex
  - Indication du service par numéro de port

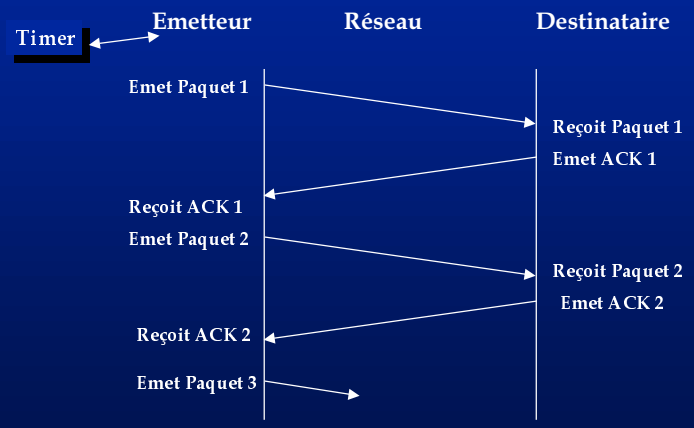
TCP — Fonctionnalités

- Traite les données venant des couches supérieures comme une suite d'octets.
- Découpe cette suite d'octets en segments
  - Taille maximale de 64 Koctets
  - La taille du segment dépend du média devant la station qui émet
- 1 segment TCP est contenu dans un datagramme IP
  - Champ protocole du datagramme IP = 6
- Des segments sont échangés pour :
  - ouvrir les connexions
  - transférer des données
  - envoyés des ACK, gérer le contrôle de flux
  - fermer les connexions

Fiabilité d 'un service réseau

Mécanisme « Send and Wait » : La technique la plus simple…

(on transmet un segment puis on attend l 'acquittement avant de transmettre le suivant)



Fiabilité d'un service réseau

- Mécanisme "Send and Wait"… suite
  - si l'on n'a pas d'acquittement à l'arrivée à échéance du timer, on retransmet puis on attend de nouveau…
  - exploite très mal le réseau
    - » il n 'est utilisé que lors de la transmission !

tt: temps de transmission de la trame d 'information
tp: temps de propagation
efficacité = temps utile / temps total = tt / (tt + 2*tp) = 1 / (1 + 2*tp/tt)

- » si la taille du réseau augmente, tp augmente, l 'efficacité diminue !

- On introduit la notion de fenêtre d 'anticipation…

(ou fenêtre de transmission)

Fiabilité d 'un service réseau

- Mécanisme du *glissement de fenêtre (sliding window)*

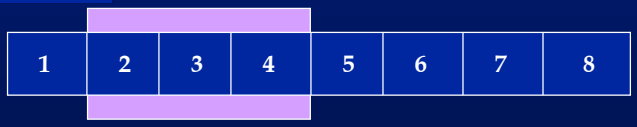
fenêtre initiale

émetteur et destinataire sont concernés

L'émetteur peut envoyer 3 paquets avant de reçcevoir un acquittement

L'acquittement du paquet 1 arrive, la fenêtre glisse

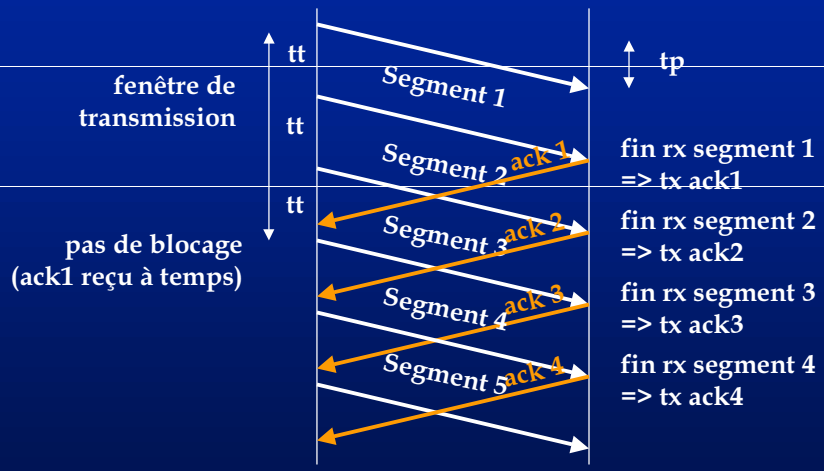
glissement →



Les performances sont fonctions de la taille de la fenêtre et de la vitesse à laquelle le réseau accepte les paquets.

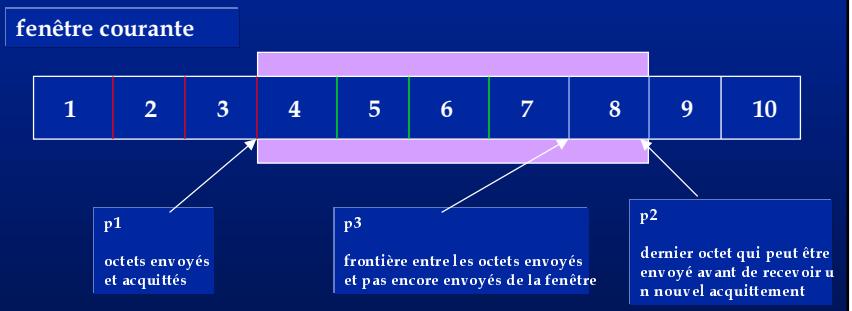
Fiabilité d'un service réseau

- Si la fenêtre a une taille suffisante, il n'y a pas de blocage !



TCP Fiabilité du service

- Mécanisme de "sliding window"
  - au niveau de l'octet
  - les octets du flot de données sont numérotés séquentiellement.
  - l'émetteur gère 3 pointeurs



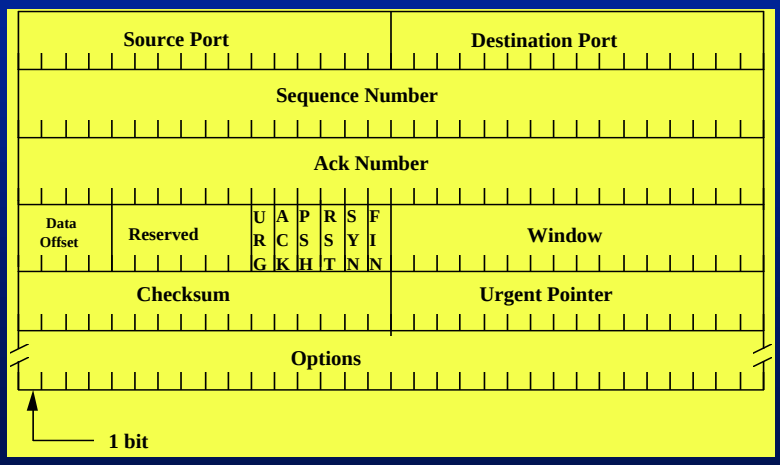
TCP est full duplex => 2 fenêtres pour chaque connexion

TCP Contrôle de flux/congestion

- Le destinataire joue avec la place disponible dans les buffers pour réduire la transmission (où l'augmenter)
  - champ « window advertisement » dans les ACK
  - donne le nombre d'octets que le receveur est prêt à accepter à ce moment
  - on modifie la fenêtre d'émission en conséquence
- Contrôle de flux/congestion sont indispensables à l'Internet
  - hétérogénéité des machines pour les communications de bout en bout...
    - » TCP résoud ce problème avec le contrôle de flux
  - réseaux et routeurs de différentes capacités.
    - » TCP résoud ce problème avec l 'algorithme de contrôle de congestion du « slow start »
      - perte de segment interprétée comme signe de congestion
      - => on réduit drastiquement la fenêtre de congestion

Segment TCP — en-tête

- Taille minimale de 20 octets
  - plus 20 octets en-tête IP => en-tête TCP/IP : 40 octets



Segment TCP

- Unité de transfert TCP pour :
  - Etablir des connexions
  - Transférer des données
  - Envoyer les acquitements
  - Avertir de la taille des fenêtres
  - Fermer les conexions

Segment TCP Champs

- Source Port (16 bits)
  - Numéro du port TCP qui identifie l'application du coté émetteur du segment TCP
- Destination Port (16 bits)
  - Numéro du port TCP qui identifie l'application du coté destinataire du segment TCP
- Sequence Number (32bits)
  - Nombre donné en octets.
  - Par rapport au début de la connexion; référe nce au flux dans la même direction que le segment
  - Assure le bon séquencement.
- Acknowledgment Number (32 bits)
  - Nombre donné en octets, c'est une information en retour.
  - Identifie la position du dernier octet reçu en donnant le numéro du prochain octet que lui,destinataire, espère reçevoir.
  - Les octets précédents ont été reçus sans erreur ni perte

Segment TCP Champs

- Data Offset ou HLEN, longueur de l'en-tête (4bits)
  - Donné en multiple de 32 bits , souvent = 5
- URG : Urgent (1bit)
  - exemple interruption, Ctrl C dans telnet
- ACK : tenir compte de l'Acknowledgment Number (1bit)
- PSH : délivrer immédiatement les données (1bit)
  - l'émetteur ne prévoit pas d'envoyer d'autres données dans l'immédiat
  - exemple : après une fin de ligne sous telnet en mode ligne)
- RST (reset) : reprise d' une connexion au départ (1bit)
  - après plusieurs SYN incompréhensifs ou un crash, ...
- SYN : Désire établir une connexion (1bit)
- FIN : Termine la connexion (1bit)

Segment TCP Champs

- **Window :**
  - Nombre d'octets que l'émetteur peut envoyer par rapport à l'Acknowlegment Number sans recevoir d'acquittement (espace libre dans le buffer de réception). C'est une information de retour.
  - Permet le contrôle de flux
- **Checksum :**
  - Fonction sur l'entête et les données
  - Permet de vérifier que le transport s'est effectué sans erreur
  - Si le récepteur s'aperçoit d'une erreur, il fait comme si le segment avait été perdu, il ne l'acquitte pas

Segment TCP Champs

- **Urgent Pointer , indique le dernier octet de données urgentes quand URG = 1**
- **Options :**
  - Permet d'ajouter des options.
  - Certaines sont standards (décrites dans des RFCs).
  - Une option utilisée lors de l'établissement de la connexion est de négocier la taille maximum de segment que l'on souhaite recevoir

Les accusés de réception

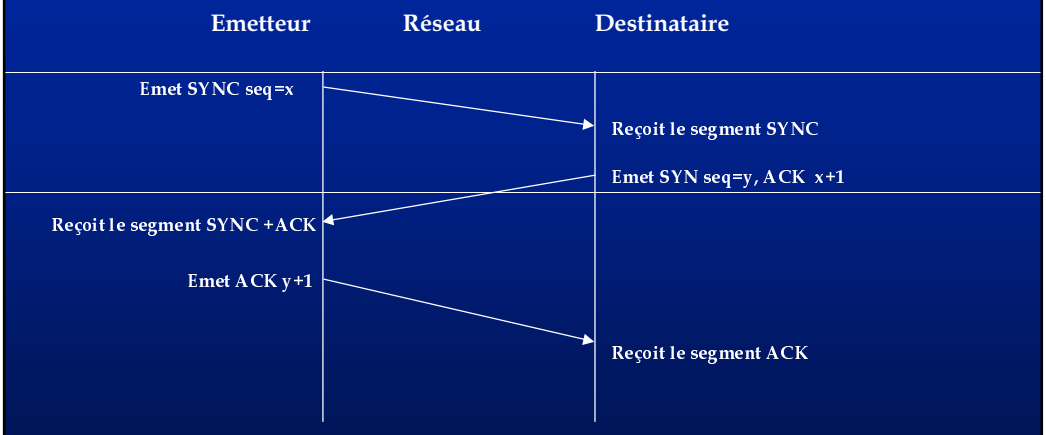
- Peuvent être transportés avec des données
- Acquittent un nombre d'octets de données (et non obligatoirement un segment entier)
- Ne sont pas obligatoires à chaque segment reçu

Segment TCP Champs

- ACK et retransmission
  - le shéma d'ACK de TCP est cumulatif
    - » référence au nombre d'octets déjà reçu
  - avantages
    - » facile à générer et sans ambibuité.
- Les délais de retransmission
  - pour chaque segment envoyé il y a un timer de déclenché, mais la structure de l'Internet impose des timers variable.
  - algorithme adaptatif
    - » ajustement automatique et dynamique, tout au long de la connexion,
    - » en fonction des délais d'acquittement des segments précédents ("segment round-trip time") .

Ceci permet à TCP de s'adapter sans paramétrage, à tous les débits et à tous les temps de réponse, donc à tous les réseaux

TCP Etablissement d'une connexion



x et y sont les numéros de séquence initiaux

Fichiers etCommandes sous Unix

- Principaux fichiers et commandes utilisables sur un système Unix, pour gérer TCP/IP
- Sur les systèmes UNIX System V et BSD
  - les commandes sont identiques,
  - les fichiers de configuration peuvent être différents ou situés ailleurs dans l'arborescence des fichiers (en fonction du système et des constructeurs).

Fichiers sous /etc

- *rc**
  - on trouve les commandes ifconfig, route et le démarrage des daemons tels que inetd, Sendmail, routed, named
    - même philosophie avec System V
- *hosts*
  - permet aux applications d'avoir le numéro IP à partir d'un nom de machine. (Presque) inutile si on utilise resolv.conf
    - Ex : 127.0.0.1 localhost loghost # The machine
    - 129.88.32.1 imag mailhost # Gould NP1 BatB
    - 129.88.32.24 hal hal-gw # Sps7-300 Spix32
- *networks*
  - fait la correspondance nom de réseau et numéro de réseau IP (idem host mais pour les réseaux)

Fichiers sous /etc

- *inetd.conf*
  - sous-démons lancés par inetd (telnetd, ftpd,..)
- *sendmail.cf*
  - règles de reécritures utilisé par le démon sendmail pour
    - écrire les adresses de messagerie
    - aiguiller le message vers la bonne destination (définir le mailer)
- *named.boot*
  - premier fichier de configuration pour le serveur de noms
  - les autres sont traditionnelement dans le répertoire /var/named

Commandes

- ifconfig :
  - indique le numéro IP du coupleur, le masque de subnet, le broadcast IP
- route
  - pour mettre à jour la table de routage
- netstat
  - affiche de très nombreuses informations d'état et de valeurs de compteurs :
    - -r : visualise la table de routage.
- ping :
  - ping serv1.grenet.fr
    - Envoit un datagramme ICMP echo à la machine
    - Cela permet de savoir si la machine est accessible
    - Indique aussi le temps de réponse.

Commandes (suite) :

- arp -a
  - visualise la table ARP
    - cisco-viallet (147.171.149.128) at 0:0:c:0:5b:37
    - alpe (147.171.149.3) at 8:0:2b:8:47:f8
    - alize (147.171.149.11) at 8:0:20:0:bc:eb
- arp -s
  - fixe une adresse Ethernet pour une @ IP
    - Ex : arp -s 147.171.200.200 08:00:2B:00:BC:09
- traceroute
  - Permet de suivre le chemin emprunté par les datagrammes
  - Utilise le champs Time To Live de IP.
- whois -h mcsun.eu.net nom_ou_numéro_de_réseau
  - Permet d'avoir des informations sur un réseau (coordonnées des responsables administratifs et techniques ...).

Exemple d'application : FTP (RFC959) 1

- File Tranfer Protocol
  - Application pour transférer des fichiers
- Utilise 2 connexions TCP (---> transfert fiable) :
  - 1 de contrôle (commandes et réponses) : *port 21*
  - 1 de transfert de données : *port 20*
    - cette connexion est ouverte puis fermée à chaque transfert
- Modes
  - *client* : processus d'un utilisateur, par exemple
  - *serveur* : démon ftpd qui est lancé par *inetd* sous Unix
- Le client ouvre la connexion
  - le serveur attend

FTP 2

- Vérification de l'identité du client
  - Nom et mot de passe à l'ouverture de la connexion.
  - Le client doit être déclaré sur le serveur.
  - Ses droits d'accès découlent de cette déclaration.
- Dialogue entre le client et le serveur
  - Vous pouvez tester ce dialogue avec la commande Unix :
    - » telnet nom_de_machine 21
- Commandes de FTP
  - Une commande utilisateur est envoyée presque sans modification, sous forme d'une suite de caractères terminée par CRLF (identique à telnet).
  - Dirigés vers le port 21
  - Exemples : help reste HELP, ls devient LIST, quit reste QUIT

FTP 3

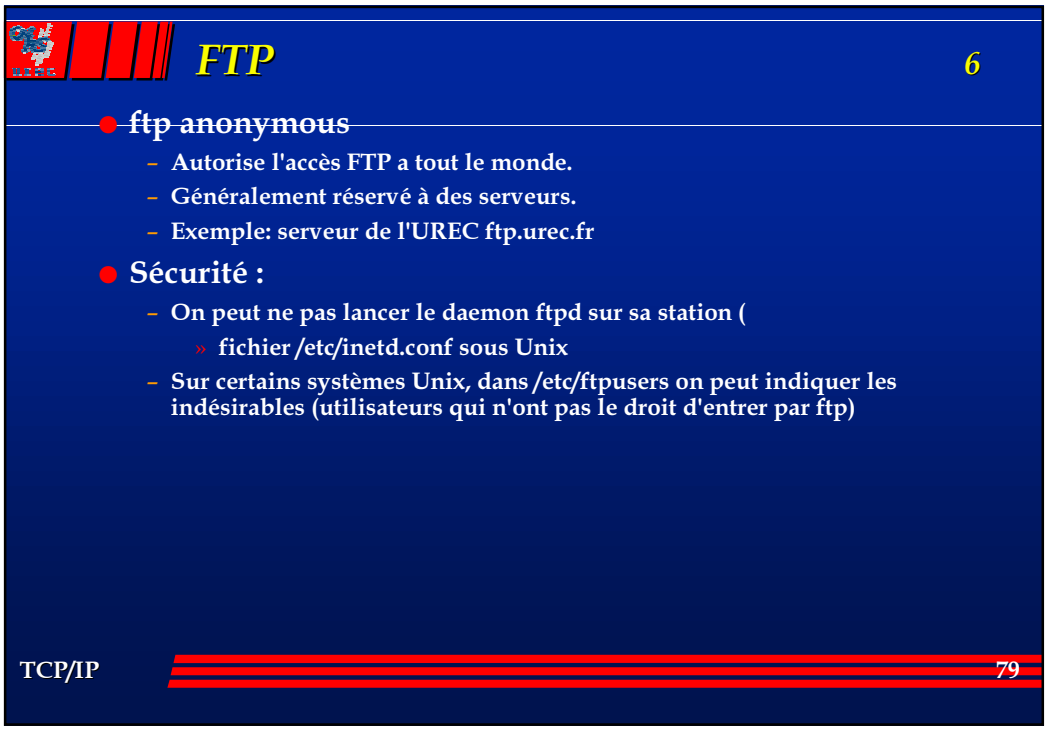
- La réponse du serveur :
  - Précédée d'un nombre de 3 caractères :
    - » 1er chiffre permet de savoir à quoi se rapporte la réponse
- 1XX :
  - la commande commence a être exécutée, il y aura une autre réponse.
  - Ex :150 Opening data connection for ... (message en début de transfert)
- 2XX :
  - La commande a été exécutée avec succès.
  - Vous pouvez envoyer une autre commande.
  - Ex: 226 Transfer complete (message en fin de transfert)
- 5XX :
  - Commande non acceptée.
  - Ex: 550 fichier: No such file or directory en réponse à "get fichier"

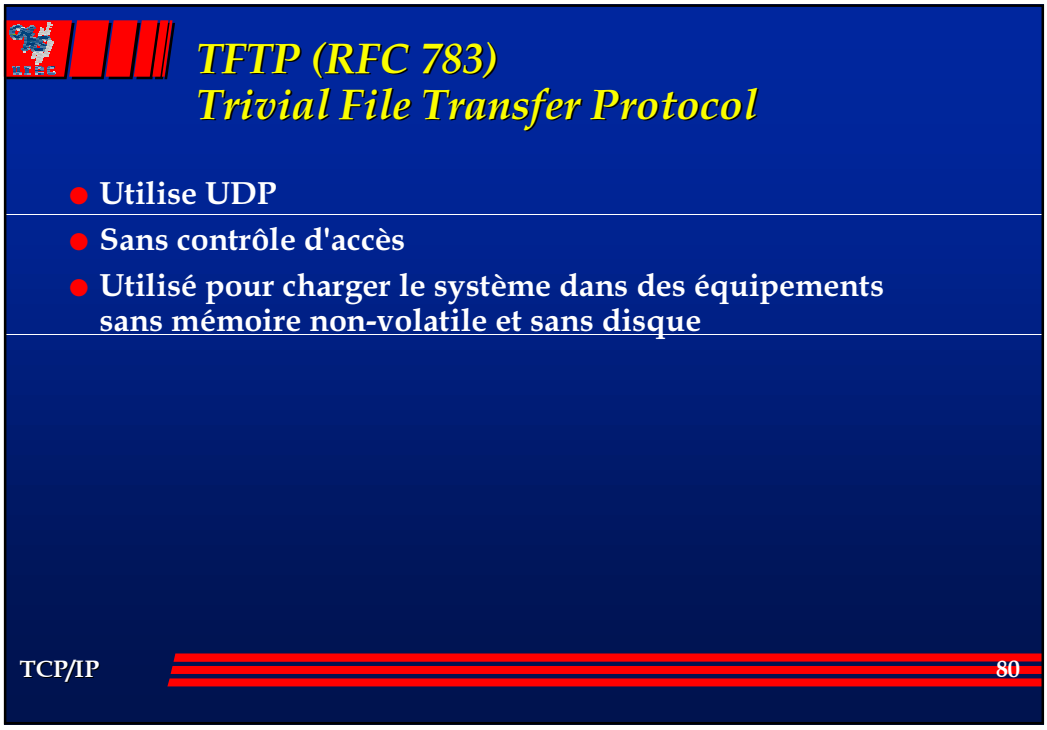
FTP 4

- Permet de transférer des fichiers
  - En mode "stream" ou "block" (Unix toujours en mode stream)
  - Texte (ASCII sous Unix)
    - » Suite d'octets avec 7 bits significatifs.
    - » Les fins de ligne, de page ... sont détectées et transformées si besoin pour être adaptées à la machine cible.
    - » Il peut y avoir un transcodage: ASCII-EBCDIC
  - Binaire (Image)
    - » Suite d'octets avec 8 bits significatifs.
    - » Aucune transformation est apportée

FTP 5

- Principales commandes (SunOs 4.1.3)
  - help ou ? liste des commandes
  - status état des connexions
  - open ouvrir une connexion (transparent)
  - user entrer un nom d'utilisateur (transparent)
  - passwd envoyer le mot de passe (transparent)
  - ls fait un ls sur la machine distante
  - cd changer de répertoire sur la machine distante
  - get rapatrier un fichier (ouvre une connexion TCP)
  - put envoyer un fichier (ouvre une connexion TCP)
  - type indiquer le type de fichier (ASCII ou binaire)
  - delete effacer un fichier sur la machine distante
  - quit fermer une connexion





Exemple d'application TELNET (RFC854) 1

- TErminaL NETwork protocol
  - terminal virtuel, remote terminal, terminal à distance
- Utilise une connexion TCP
  - Fiable mais gourmand en bande passante
  - Utilise le *port 23* pour le serveur
- Mode
  - *client* : processus d'un utilisateur
  - *serveur* : daemon telnetd qui est lancé par inetd sur Unix
- Le client ouvre la connexion

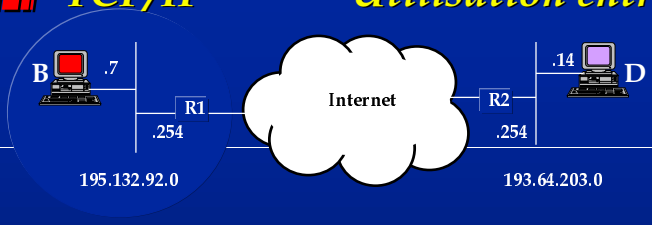
TELNET 2

- Vérification de l'identité du client
  - Nom et mot de passe à l'ouverture de la connexion.
  - Le client doit être déclaré sur le serveur.
  - Ses droits d'accès découlent de cette déclaration.
- Lorsque la connexion est ouverte
  - C'est une suite d'octets qui s'échangent.
  - Elle permet de transporter des octets avec 8 bits significatifs.
  - On peut donc émuler n'importe quel type de terminal
    - » VT100, IBM 3270 (logiciel TN3270), ...
- Pour insérer des commandes, le protocole utilise le caractère 255 comme échappement
  - ce caractère est appelé IAC (Interpret As Command)

TELNET 3

- Options négociées entre le client et le serveur :
  - Type de terminal
  - Echo local ou distant
  - Transmission binaire
  - Longueur de ligne
  - Taille de la page
  - Tabulations horizontales et verticales
  - ....
- rlogin est le concurrent de telnet sur Unix

TCP/IP Utilisation entre 2 stations



Sur la machine B, "telnet Machine D" Que se passe-t-il ?

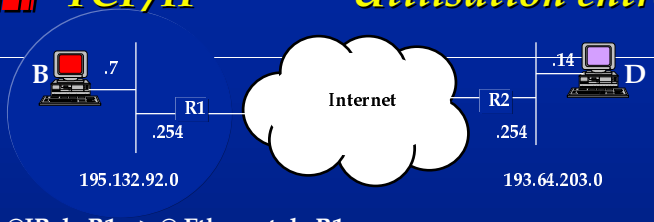
- Nom --> @ IP
  - » La machine B traduit le nom "machine D" en 193.64.203.14
    - table hosts ou cache ou interrogation d'un DNS.

Si elle ne trouve pas, message "Host unknown".

- Comment atteindre 193.64.203.14 (1er saut) ?
  - » Ce n'est pas un numéro 195.132.92.X
  - » Il faut donc passer par un routeur.
  - » B consulte sa table de routage : il faut passer par 195.132.92.254

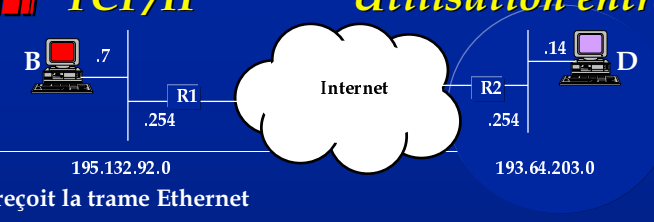
Si elle n'a pas ce renseignement, message "Network unreachable"

TCP/IP — Utilisation entre 2 stations



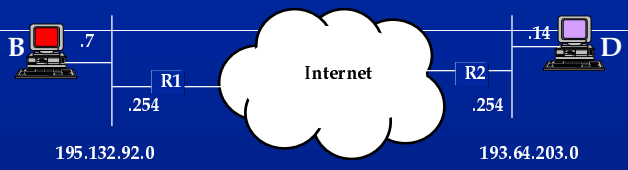
- @IP de R1 --> @ Ethernet de R1
  - » B émet une trame de "broadcast" Ethernet après avoir vérifié qu'elle n'a pas déjà l'information dans sa table ARP
  - » Contenant une trame ARP avec la question :
  - » quelle est l'adresse Ethernet de 195.132.92.254 ?
  - » R1 répond à B : l'@ Ethernet de 195.132.92.254 est 0:0:c:0:5b:37
- B envoie une trame Ethernet avec l'@ destination 0:0:c:0:5b:37
  - » Incluant un datagramme IP (@ orig 195.132.92.7 et @ dest 193.64.203.14
    - Contenant un segment TCP
      - Avec un numéro de port destinataire 23 (telnetd)
      - SYN=1 (ouverture d'une connexion TCP)

TCP/IP — Utilisation entre 2 stations



- R1 reçoit la trame Ethernet
  - » Extrait le datagramme IP, l'@ IP du destinataire (193.64.203.14) et cherche où l'envoyer.
  - » Il a une interface sur l'Internet (par Renater ou un autre opérateur)
  - » Les protocoles de routage font leur travail et le datagramme IP arrive sur R2
- R2 recherche alors l'@ Ethernet de 193.64.203.14
  - » S'il ne la trouve pas dans sa table ARP, il envoie un broadcast ARP
  - » Il peut ensuite envoyer le datagramme IP à la machine D
- D reçoit le datagramme IP
  - » Extrait le segment TCP.
  - » Ouvre une session TCP.
  - » Avec l'indication de port numéro 23, il appelle la partie telnetd du démon inetd.

TCP/IP — Utilisation entre 2 stations

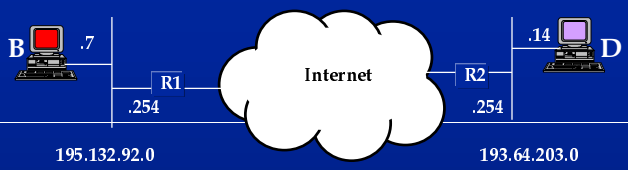


- telnetd demande le nom de l'utilisateur.
  - La question est transportée par un segment TCP, dans un datagramme IP
    - @origine 193.64.203.14, @ destination 195.132.92.7.

Pour envoyer ce datagramme

- *la machine D cherche l'itinéraire avec la même méthode que la machine B au départ*
  - table de routage, ARP, ...

TCP/IP — Utilisation entre 2 stations



- **Remarques**
  - D ne tient pas compte de la précédente arrivée d'un datagramme IP pour trouver l'itinéraire de la réponse. Il refait le raisonnement, comme s'il n'avait rien reçu
  - Pour tester la connectivité IP, il n'est pas utile de tester un appel de B vers A, si on a déjà testé un appel de A vers B
  - Le broadcast ARP n'est utilisé que lors de la première recherche d'adresse Ethernet
  - Quand il y a un problème, la recherche d'erreur est difficile quand on n'a pas ce mécanisme en tête

Bilan — TCP/IP aujourd'hui

- C'est le protocole le plus utilisé actuellement, sur tous les types de réseaux (locaux et longues distances)
- Quel sera son remplaçant ?

BILAN — Avantages de TCP/IP

- Gratuit
- Indépendant des constructeurs
- Disponible sur tous les types de matériel
  - micro, station, super calculateur et équipements de réseaux
- Facile à installer
- Produits éprouvés depuis longtemps dans un monde hétérogène
- Inclut de très nombreuses applications
- Bien standardisé et documenté
- Les protocoles sont simples mais efficaces