

IP : fragmentation et routage

- Correction -

1. Fragmentation des paquets IP

1.1 Soit un réseau constitué de 5 routeurs IP (R1 ... R5) et de trois stations A, B et C qui doivent communiquer (Figure 1). On suppose que le masque est /8 (255..0.0.0) dans les exercices.

Chaque liaison entre hôtes (station ou routeur) est étiquetée par son MTU (Maximum Transmission Unit). Le MTU définit la taille maximale d'un paquet IP qui peut être véhiculé dans les trames d'un réseau physique particulier. Ce paramètre est rattaché à une interface réseau du hôte (de numéro p_i et d'adresse IP de classe B) pour fragmenter les données avant leur transmission sur la liaison.

On suppose que A doit émettre 1520 octets de données vers B.

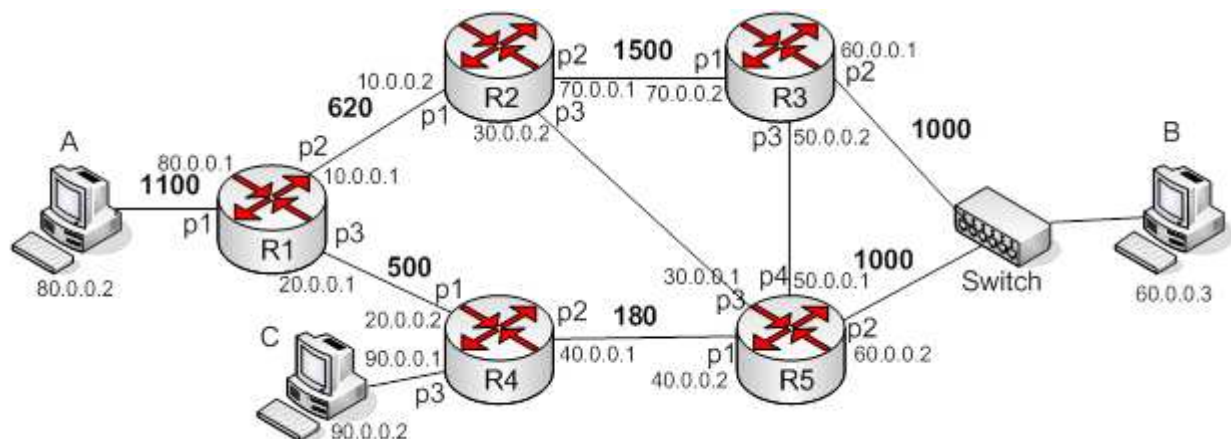
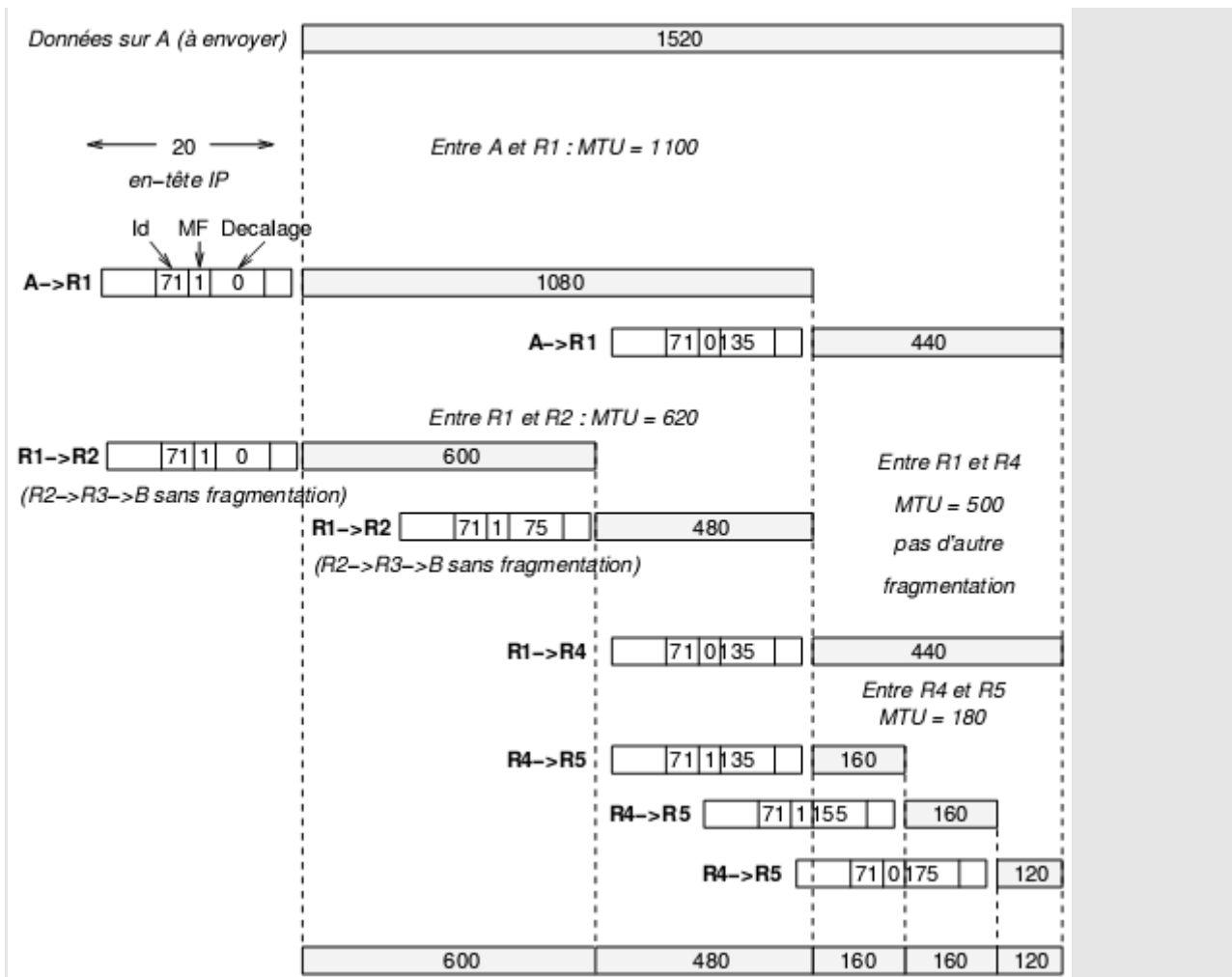


Figure 1 – Architecture de réseau

1.1 Décrivez les fragmentations réalisées pour la transmission d'un paquet IP émis par A à destination de B, en supposant que le routeur R1 transmet alternativement les trames qu'il reçoit vers R2-R3 puis vers R4-R5. Précisez pour chaque fragment de paquet, les valeurs des champs (Identification, More Fragment Flag, Offset). On supposera que la valeur initiale de l'Identifiant du paquet est 71.



1.2 Justifiez le fait que la régénération des paquets fragmentés n'a lieu que sur la station destinataire et non sur les routeurs.

Certains routeurs ne voient pas passer la totalité des fragments d'un paquet IP, ils ne peuvent donc pas reconstruire le datagramme original. De plus, l'opération de défragmentation nécessite de la mémoire et du calcul. La philosophie de IP est de simplifier les fonctions dans le réseau (c'est-à-dire exécutées par les routeurs) et de repousser les traitements complexes (contrôle de flux, contrôles d'erreurs, chiffrement, fragmentation, etc ...) aux extrémités du réseau (c'est-à-dire exécutés par les stations).

2. Routage des paquets IP

Soit le réseau de la Figure 1 (question 1 ci-dessus). A chaque liaison, supposée symétrique, est associée une distance égale à 1. On supposera que les routeurs mettent en œuvre un protocole de routage de type vecteur de distance avec l'algorithme Bellman-Ford.

2.1 Quelle est la longueur maximale d'une route dans un réseau IP utilisant le protocole de routage RIP ? Comment s'assurer qu'un paquet IP ne bouclera pas sans fin en raison d'une table erronée ?

La route la plus longue traverse au maximum 16 routeurs. Les paquets IP transmis par une source sont initialisés avec le champ TTL (Time To Live) égal à 16. A chaque passage dans un routeur, le champ TTL du paquet est décrémenté de 1. Lorsqu'un paquet arrive dans un routeur avec une valeur TTL égale à 1, le routeur détruit ce paquet, limitant les routes à 16 nœuds traversés au maximum.

2.2 On supposera que le réseau vient d'être mis en service par l'administrateur et que chaque routeur n'a qu'une connaissance locale de la topologie du réseau (il ne connaît que ses routeurs voisins et ses sous-réseaux voisins). Donner les tables de routage initiales des différents routeurs telles que configurées par l'administrateur, en suivant le format de table ci-dessous.

Adresse IP du réseau destination	Masque du reseau	adresse IP du prochain routeur	Métrique (Hop Count) nbre de sauts	numéro de l'interface

A T=0

L'administrateur réseaux se connecte à distance sur chaque routeur Ri et programme la table de routage de chacun d'eux pour ajouter des routes statiques en utilisant la commande « route »

Exemple pour le routeur R1. Il me faut ajouter 3 routes statiques au moyen de la commande système « route ». Je me connecte a distance ou en locale a R1 et j'exécute les commandes suivantes :

Sous linux:

Je commence par afficher la table de routage initiale : **\$>route**

Puis je tape les 3 commandes ci-dessous pour ajouter les 3 routes :

route add -net 10.0.0.0 netmask 255.0.0.0 dev eth2 metric 0
route add -net 20.0.0.0 netmask 255.0.0.0 dev eth3 metric 0
route add -net 80.0.0.0 netmask 255.0.0.0 dev eth1 metric 0

Je peux ajouter une 4eme route statique vers le sous-reseaux 70.0.0.0/8.

route add -net 70.0.0.0 netmask 255.0.0.0 gw 10.0.0.2 dev p2 metric 1
ou **route add -net 70.0.0.0 netmask 255.0.0.0 gw 10.0.0.2 dev 10.0.0.1 metric 1**
ou **route add -net 70.0.0.0 netmask 255.0.0.0 gw 10.0.0.2 dev eth2 metric 1**

Enfin je peux supprimer la 4eme route au moyen de la commande suivante:

route del -net 70.0.0.0 netmask 255.0.0.0 gw 10.0.0.2 dev 10.0.0.1 metric 1

sous Windows:

route ADD 10.0.0.0 MASK 255.0.0.0 10.0.0.2 IF p2 METRIC 1
route ADD 10.0.0.0 MASK 255.0.0.0 10.0.0.2 IF 13 METRIC 1

route DELETE 10.0.0.0

Table R1

Dest.Network/ Mask	Mask	Next Hop / Passerelle	Interface / IF	Metric (Hop Count)
10.0.0.0	255.0.0.0	On-link (*)	P2 (eth0)	0
20.0.0.0	255.0.0.0	On-link (*)	P3 (eth1)	0
80.0.0.0	255.0.0.0	On-link (*)	P1 (eth2)	0

Table R2

Dest.Network/ Mask	Next hop	Interf	metric
10.0.0.0/8	direct	P1	0
30.0.0.0/8	direct	P3	0
70.0.0.0/8	direct	P2	0

R3

Dest.Network/ Mask	Next hop	Interf	metric
50.0.0.0/8	direct	P3	0
60.0.0.0/8	direct	P2	0
70.0.0.0/8	direct	P1	0

R4

Dest. network	Next hop	Interf	metric
20.0.0.0/8	direct	P1	0
40.0.0.0/8	direct	P2	0
90.0.0.0/8	direct	P3	0

R5

Dest. network	Next hop	Interf	metric
30.0.0.0/8	direct	P3	0
40.0.0.0/8	direct	P1	0
50.0.0.0/8	direct	P4	0
60.0.0.0/8	direct	P2	0

2.3 Donner le vecteur de distance du routeur R1, que l'on notera VR1.

L'administrateur démarre le daemon RIPd sur chaque routeur, ce qui permettra d'ajouter dynamiquement les autres routes dans les tables.

Le protocole RIP s'exécutant sur chaque routeur construit un vecteur de distances VRi par extraction des paramètres de la table de routage statique.

VR1 = ([10.0.0.0, 255.0.0.0, 0], [20.0.0.0, 255.0.0.0, 0], [80.0.0.0, 255.0.0.0, 0])

ou VR1= ([10.0.0.0, 255.0.0.0, 0], [20.0.0.0, 255.0.0.0, 0], [80.0.0.0, 255.0.0.0, 0])

Cela correspond au 3 paramètres (adresse réseau destination, masque, metrique) pour chaque route (ligne de la table statique)

2.4 On considérera la séquence d'échange de vecteurs de distances suivante:

Toute les 30 secondes, RIPd du routeur R1 envoie ce vecteur VR1 vers tous les routeurs voisins pour partager ses routes locales et distances qui transitent par lui.

<u>Instant</u>	<u>Événement</u>
T ₁	R2, R4 reçoivent VR1 (vecteur de distances de R1)
T ₂	R1, R3, R5 reçoivent VR2

Donnez la table de routage des routeurs suite à l'échange des vecteurs de distances VR1 et VR2.

À T1, R2 et R4 reçoivent VR1 = ([10.0.0.0/8, 0], [20.0.0.0/8, 0], [80.0.0.0/8, 0])

R2 mettra à jour sa table en ajoutant 2 nouvelles routes vers les réseaux 80.0.0.0/8 et 20.0.0.0/8

Network/Mask	Next hop	Interface.	metric
10.0.0.0/8	direct	P1	0
30.0.0.0/8	direct	P3	0
70.0.0.0/8	direct	P2	0
80.0.0.0/8	10.0.0.1	P1	1
20.0.0.0/8	10.0.0.1	P1	1

Les routes statiques sont en gris, et les routes dynamiques sont en jaune dans la table.

R4 mettra à jour sa table en ajoutant 2 nouvelles routes vers les réseaux 80.0.0.0/8 et 10.0.0.0/8

Network/Mask	Next hop	Interface	metric
20.0.0.0/8	direct	P1	0
40.0.0.0/8	direct	P2	0
90.0.0.0/8	direct	P3	0
10.0.0.0/8	20.0.0.1	P1	1
80.0.0.0/8	20.0.0.1	P1	1

À T2, R1, R3, R5 reçoivent VR2 = ([10.0.0.0/8, 0], [20.0.0.0/8, 1], [30.0.0.0/8, 0], ([70.0.0.0/8, 0], [80.0.0.0/8, 1])

R1

Network	Next hop	Int.	metric
10.0.0.0/8	direct	P2	0
20.0.0.0/8	direct	P3	0
80.0.0.0/8	direct	P1	0
30.0.0.0/8	10.0.0.2	P2	1
70.0.0.0/8	10.0.0.2	P2	1

R3

Network	Next hop	Int.	metric
50.0.0.0/8	direct	P3	0
60.0.0.0/8	direct	P2	0
70.0.0.0/8	direct	P1	0
10.0.0.0/8	70.0.0.1	P1	1
20.0.0.0/8	70.0.0.1	P1	2
30.0.0.0/8	70.0.0.1	P1	1
80.0.0.0/8	70.0.0.1	P1	2

R5

Network	Next hop	Int.	metric
30.0.0.0/8	direct	P3	0
40.0.0.0/8	direct	P1	0
50.0.0.0/8	direct	P4	0
60.0.0.0/8	direct	P2	0
10.0.0.0/8	30.0.0.2	P3	1
70.0.0.0	30.0.0.2	P3	1
20.0.0.0	30.0.0.2	P3	2
80.0.0.0	30.0.0.2	P3	2

2.5 Pourquoi et au bout de combien de temps une route est elle considérée comme non valide puis supprimée de la table d'un routeur ?

Toutes les 30 secondes, le daemon RIP (routed) qui s'exécute sur le routeur, transmet sur le port 520 (UDP) sa table de routage vers tous ces routeurs voisins, sous la forme de paquets IP broadcast (255.255.255.255). Si un routeur voisin ne répond pas au bout de 180 sec (3 mns), toutes les routes passant par ce routeur sont invalidées avec un coût mis à 16 (infini). Ces routes sont conservées durant encore 120 sec (2 mns), puis supprimées de la table si aucun paquet RIP n'est reçu de ce routeur.

2.6 Travaux pratiques : RIP

2.6.1 - Visualiser la table de routage sur le serveur et sur le client, avec la commande : **route**

La commande **route**, tout comme **ifconfig** sert à la fois à connaître l'état de la table de routage de l'hôte et à configurer de nouvelles routes statiques au besoin.

Exemple de table de routage sous windows avec la commande système (**C:\route print**)

Description	Destination du réseau	Masque de réseau	Passerelle	Interface	Métrique
Itinéraire par défaut	0.0.0.0	0.0.0.0	10.0.0.1	10.0.0.169	1
Réseau de bouclage	127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
Réseau local	10.0.0.0	255.0.0.0	10.0.0.169	10.0.0.169	1
Adresse IP locale	10.0.0.169	255.255.255.255	127.0.0.1	127.0.0.1	1
Adresses multicast	224.0.0.0	240.0.0.0	10.0.0.169	10.0.0.169	1
Adresse de diffusion limitée	255.255.255.255	255.255.255.255	10.0.0.169	10.0.0.169	1

Exemple de table de routage sous Linux/UNIX avec la commande système (**\$route -n**)

```
# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.1.0    0.0.0.0        255.255.255.0  U        0      0        0 eth0
0.0.0.0        192.168.1.1    0.0.0.0        UG        0      0        0 eth0
```

La route vers la destination 0.0.0.0 correspond à la route par défaut. Ainsi si un paquet vers une destination n'a pas trouvé de route explicite (statique ou dynamique) vers le réseau destination, le routeur local transmettra ce paquet sur la route par défaut qui transitera vers la passerelle par défaut.

2.6.2 – Sur le serveur ajouter une route vers le réseau 192.168.119.0/24 au travers de la passerelle 192.168.119.1 et l'interface locale « eth0 » d'un coût de 1 HC (Hop count) au moyen de la commande ci-dessous, et visualiser la nouvelle table:

route add -net 192.168.119.0 netmask 255.255.255.0 gw 192.168.119.1 IF eth0 metric 0

- Lancer l'analyseur de protocole Wireshark sur le serveur. Puis lancez le daemon RIP « **routed** » sur le serveur grâce à la commande ci-dessous. Analyser les paquets RIP transmis par le serveur:

[prof@machine] **\$/etc/init.d/routed start.**

Exercice 3. Table de routage IP : Jeu des 7 erreurs

Rappels :

- On peut adresser directement tout poste connecté physiquement à son réseau et partageant la même adresse réseau au niveau IP.
- Un poste qui émet un paquet à destination d'un autre réseau IP, utilise une passerelle (routeur) qui se trouve sur son réseau.
- Les adresses de diffusion générale (255.255.255.255) ne passent pas les routeurs.
- L'adresse 0.0.0.0 est l'adresse par défaut (default route sous unix), elle signifie "ailleurs", dans le sens où on utilise cette ligne de table pour router les paquets dont l'adresse de destination ne correspond à aucune adresse de la table de routage.
- L'adresse 127.0.0.1 est l'adresse de loopback, elle permet à un poste de "s'auto adresser".
- Chaque ligne de la table de routage se lit de la façon suivante : Pour atteindre l'adresse réseau (colonne 2) de masque réseau (colonne 3), je passerai par la passerelle (colonne 4) en utilisant la carte réseau locale d'adresse IP (colonne 5).

Question : huit erreurs se sont glissées dans la table de routage suivante, saurez vous les retrouver ?

Numéro de ligne	Adresse réseau destination	Masque réseau	Adresse de la prochaine passerelle	Interface locale
1 (route boucle locale)	0.0.0.0 127.0.0.1	0.0.0.0	127.0.0.1	127.0.0.1
1 (route par défaut)	0.0.0.0	0.0.0.0	200.100.40.2	200.100.40.1
2	200.100.40.0	255.255.255.0	200.100.40.2 directe	200.100.40.1
3	200.100.40.1	255.255.255.255	127.0.0.1	127.0.0.1
4	200.100.40.255	255.255.255.255	200.100.40.2 directe	200.100.40.1
5	200.100.50.0	255.255.255.0	200.100.50.1	200.100.50.1
6	200.100.50.1	255.255.255.255	127.0.0.1	127.0.0.1
7	200.100.50.255	255.255.255.255	200.100.50.1	200.100.50.1
8	200.100.60.0	255.255.255.0	200.100.60.1 200.100.40.2	200.100.40.1
9	200.100.70.0	255.255.255.0	200.100.40.2	200.100.40.1
10	200.100.80.0	255.255.255.0	200.100.40.2 200.100.50.2	200.100.50.1
11	200.100.90.32	255.255.255.224	200.100.40.2	200.100.40.1
12	200.100.90.64	255.255.255.224	200.100.40.2	200.100.40.1
13	200.100.90.128	255.255.255.0 255.255.255.128 (/25)	200.100.40.2	200.100.40.1
14	201.0.91.0	255.255.255.0	200.100.50.2	200.100.50.1

15	201.0.91.255	255.255.255.255	200.100.50.2 200.100.50.1 ou direct	200.100.50.1
16	255.255.255.255	255.255.255.255	200.100.50.2 directe	200.100.50.1

Réponses :

- 1) Ligne 1 : la route par défaut renvoie à l'ordinateur lui-même, il s'agit trivialement d'une boucle
- 2) Ligne 2 : On passe par un routeur pour atteindre le réseau dont fait partie l'ordinateur. La colonne « Passerelle » doit être renseigné avec « DIRECT »
- 3) Ligne 4 : on passe par un autre routeur pour transmettre les paquets de diffusion dirigés (directed broadcast) alors qu'il s'agit de paquets destinés au réseau IP dont fait partie l'ordinateur. Même erreur que « Ligne 2 ». La colonne « Passerelle » doit être renseignée avec « DIRECT »
- 4) Ligne 8 : Le réseau IP du routeur est différent du réseau IP de l'interface, on ne peut donc pas atteindre le routeur. La colonne « Passerelle » doit être renseignée avec une adresse d'interface dans le réseau « 200.100.40.0/8 »)
- 5) Ligne 10 : Même motif que la ligne 8,
- 6) Ligne 13 : Le masque de sous-réseau n'est pas bon. Dans tous les cas l'application du masque (255.255.255.0) sur l'adresse de destination IP d'un paquet ne donnera pas l'adresse 200.100.90.128. Il faudrait corriger la colonne masque avec « 255.255.255.128 »
- 7) Ligne 16 : 255.255.255.255 est une adresse de diffusion limitée au réseau local, elle ne peut franchir un routeur. La colonne « Passerelle » doit être renseignée avec « DIRECT »