TP Réseau N°3 - MMI 1

Ce TP a pour objectif de mettre en pratique la capture de trame ethernet, et de mettre en évidence la fragmentation de datagrammes.

Introduction

Une première remarque :

Les paquets ont des noms différents selon la couche du modèle DoD :

Message :

créé par la couche application : exemples : HTTP, FTP, SSH...

Segment :

créé par la couche transport : exemples : TCP, ICMP, UDP...

Datagramme :

créé par la couche réseau, exemples : IP, IPX...

Trame :

créé par la couche liaison, exemples : Ethernet, ATM…

Configuration IP

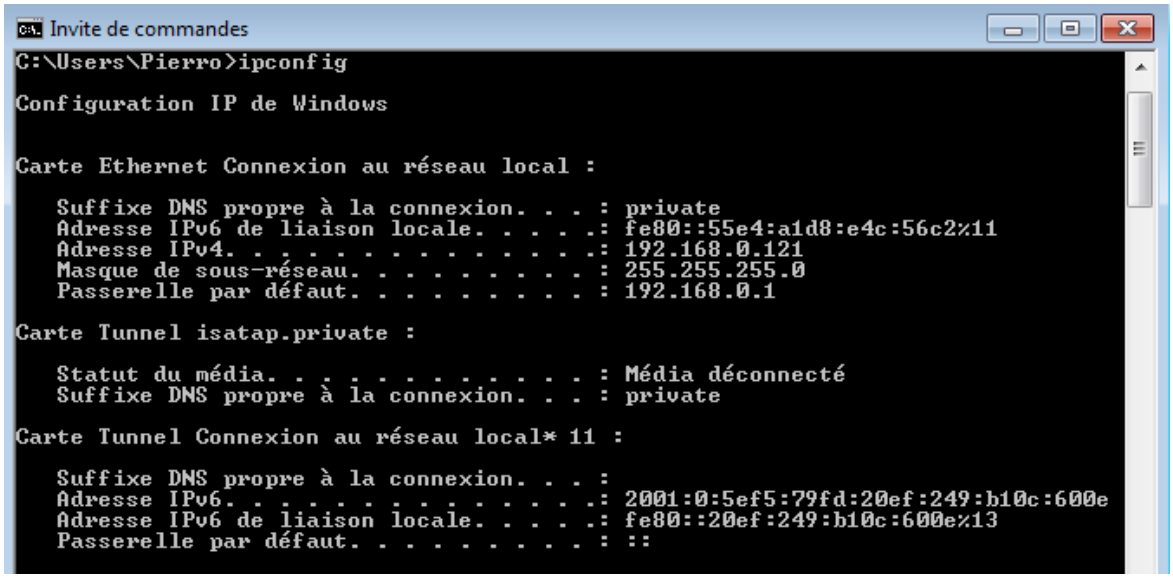
Sous linux en tapant *ifconfig* sans aucun argument, vous allez afficher la configuration IP de votre machine :

```
utilisateur@INF-XXXs:~ > ifconfig
eth0      Link encap:Ethernet  HWaddr XX:XX:XX:XX:XX:XX
          inet addr:XXX.XXX.XXX.XXX  Bcast:XXX.XXX.XXX.XXX  Mask:XXX.XXX.XXX.XXX
          inet6 addr: XXXX::XXXX:XXXX:XXX:XXXX/XX Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:430 errors:0 dropped:0 overruns:0 frame:0
          TX packets:382 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:72789 (72.7 KB)  TX bytes:56495 (56.4 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:329 errors:0 dropped:0 overruns:0 frame:0
          TX packets:329 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:35843 (35.8 KB)  TX bytes:35843 (35.8 KB)
```

Bien entendu, à la place de xxx.xxx.xxx.xxx vous obtiendrez votre adresse IP, votre masque et votre adresse de broadcast.

Sous windows, la commande équivalente est `ipconfig`.



Fragmentation

Reprenez le résultat de la commande `ifconfig`.

1) Relevez la valeur affectée au paramètre MTU (Maximum Transfert Unit) pour l'interface eth0

   -

2) Expliquez le rôle de ce paramètre

   -

3) Grâce à la commande `ifconfig`, modifiez la valeur du paramètre MTU pour l'interface eth0 à 100 octets. Donnez la commande exacte :

   -

On va maintenant envoyer des paquets ICMP de 300 octets de données. Utilisez la commande suivante :

```
ping www.google.fr -s 300
```

Vous obtenez ceci :

```
PING www.google.fr (173.194.34.24) 300(328) bytes of data.

308 bytes from par03s02-in-f24.1e100.net (173.194.34.24): icmp_seq=1 ttl=55
time=36.4 ms

308 bytes from par03s02-in-f24.1e100.net (173.194.34.24): icmp_seq=2 ttl=55
time=36.0 ms
```

```
308 bytes from par03s02-in-f24.1e100.net (173.194.34.24): icmp_seq=3 ttl=55
time=35.4 ms

^C
```

4) Expliquez la valeur 328 affichée, d'où proviennent ces 28 octets supplémentaires ?

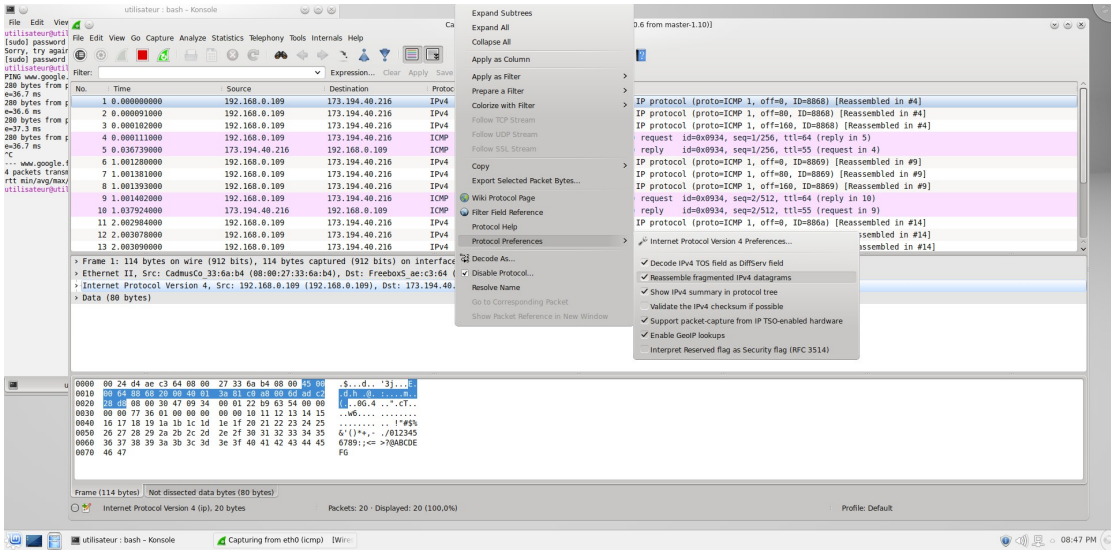
-

5) Est-ce qu'un paquet ICMP de cette taille peut être envoyé dans une seule trame Ethernet de 100 octets ?

-

Gardez la MTU à 100 octets et ouvrez wireshark et capturez le trafic ICMP. Lancez un seul ping de 272 octets.

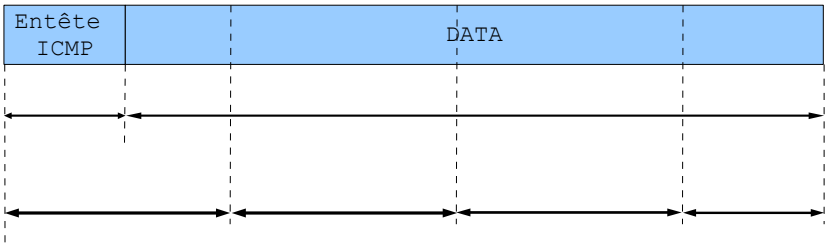
Désactivez l'affichage rassemblé des paquets. Pour se faire, sélectionnez un paquet IP, puis cliquez sur le bouton droit de la souris, Menu 'Protocol Preference', puis décochez 'Reassemble fragmented IPv4 datagramms', comme indiqué ci-dessous :



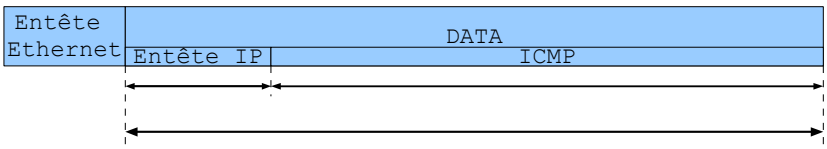
6) Combien de trames Ethernet sont émises ?

-

Indiquez les longueurs de chaque fragment sur le schéma ci-dessous :



7) Chaque fragment est encapsulé par IP, puis émis dans une trame ethernet. Indiquez les longueur ci-dessous :



8) Indiquez le type de message envoyé par ping.

-

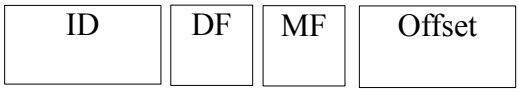
9) Y-aurait-il eu fragmentation avec une MTU à 1500 octets ?

-

10) En vous aidant de la capture réalisée avec wireshark et en analysant que l'envoi du message ICMP, compléter le schéma ci-dessous en précisant :

a) le numéro de trame,
b) le décodage des champs ID, DF, MF, Offset de l'en-tête IP

Numéro de trame :



Numéro de trame :

Numéro de trame :

Numéro de trame :

11) Indiquez dans quel ordre les datagrammes seront rassemblés. Répondez en donnant le numéro des trames correspondantes

-

12) Est-ce que l'ordre de réception des datagrammes a une importance pour la reconstitution du datagramme initial ?

-

13) Quels sont les champs de l'entête IP qui permettent d'effectuer le réassemblage ?

-

14) La probabilité de perdre un datagramme augmente-t-elle avec la fragmentation ?

-

15) Quelle est l'influence de la MTU sur les couches supérieure ?

-

Remettez la valeur de votre MTU à 1500.

TTL

Lisez la page de manuel de ping. Quelle est la signification de TTL ?

-

Lancez un ping vers www.google.fr. Déduisez-en le nombre de routeurs traversés :

-

Une utilisation astucieuse de ce mécanisme est utilisée par l'utilitaire `tracepath`. En mettant le champ TTL à 1, le premier routeur rencontré rejette le paquet et averti l'émetteur en retournant un message ICMP d'erreur.

On renvoie alors un paquet avec un champ TTL à 2, afin d'atteindre le routeur suivant, et ainsi de suite. On récupère ainsi les adresse IP de tous les routeurs traversés.

Mis en œuvre :

Dans wireshark, filtrez les trames UDP et ICMP. Ensuite, exécutez la commande suivante et observez le résultat.

```
tracepath www.google.fr
```

ARP

16) A quoi sert le cache ARP

-

Visualisez le cache ARP, grâce à la commande *arp*. Vérifiez avec vos voisins que les adresses hardware sont correctes.

17) Trouvez la commande pour vider du cache ARP l'adresse de votre voisin de droite.

-

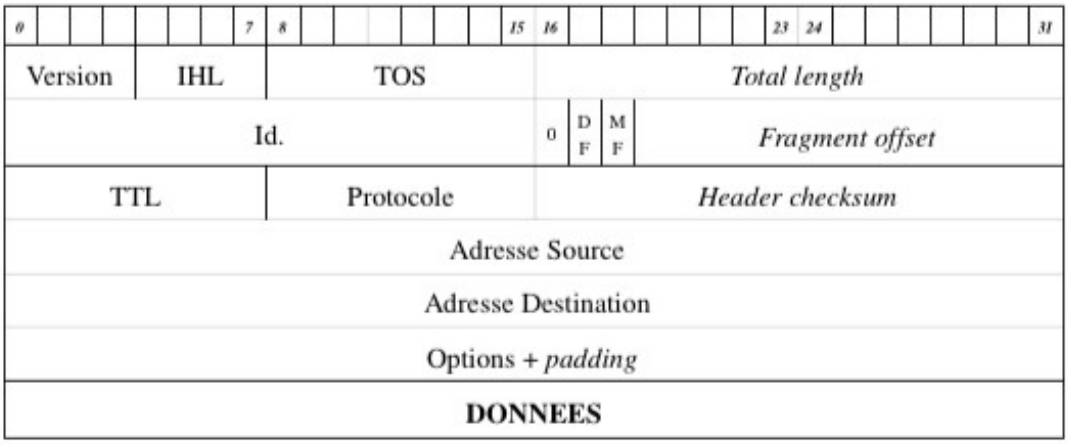
18) Lancez un ping vers le PC de votre voisin de droite. Visualisez le cache ARP. Que remarquez vous ?

-

Annexes

IP

Le format du datagramme IP, est organisé en champs de 32 bits :

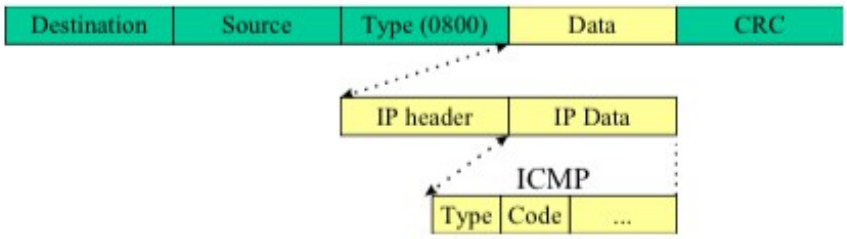


- Le champ "Version" (4 bits) identifie la version du protocole IP. Elle est fixée actuellement à 4.
- Un champ "Internet Header Length" (4 bits) spécifie la longueur de l'entête en mots de 32 bits. Cette longueur IHL varie de 5 à 15, 5 étant la longueur normale lorsqu'aucune option n'est utilisée.
- Le champ "Type de service" TOS (8 bits) définit la priorité du paquet et le type de routage souhaité. Cela permet à un logiciel de réclamer différentes qualité de services pour un datagramme : temp réel, best effort...
- Le champ "Total Length" (16 bits) définit le nombre d'octets contenus dans le paquet y compris l'entête IP. Puisque ce champ est codé sur 16 bits, la taille max. d'un paquet IP est de 65535 octets (64 Ko).
- Le champ "Identification" (16 bits) contient une valeur entière utilisée pour identifier les fragments d'un datagramme. Ce champ doit être unique pour chaque nouveau datagramme.
- "Flags" (3 bits) est utilisé pour contrôler la fragmentation des paquets. Le bit de poids faible à zéro indique le dernier fragment d'un datagramme et est baptisé "More Flag" ou MF bit. Le bit du milieu est appelé "Do not Fragment flag" ou DF bit. Le bit de poids fort n'est pas utilisé.
- "Fragment Offset" (13 bits) sert à indiquer la position qu'occupait les données de ce fragment dans le message original.
- Le TTL ou "Time To Live" (8 bits) est l'expression en secondes de la durée maximale de séjour du paquet dans un réseau. Il existe 2 manières de faire baisser cette valeur : lors du réassemblage du paquet dans un routeur, sa valeur est décrémentée chaque seconde ou alors chaque routeur qui traite ce paquet décrémente le TTL d'une unité (compteur de routeurs). Si le TTL devient nul, son paquet IP n'est plus relayé : c'est souvent l'indication d'une erreur de paquet qui boucle. Une utilisation détournée de ce champ permet de tracer la route empruntée par un paquet. En mettant le champ TTL à 1, le premier routeur rencontré rejette le paquet et signale sa présence en retournant un paquet ICMP d'erreur vers l'émetteur. On renvoie alors le paquet avec le champ TTL à 2 afin d'atteindre le routeur suivant et ainsi de suite. A chaque fois, on récupère l'adresse IP du routeur.

- Le champ "Protocole" (8 bits) identifie la couche de transport propre à ce datagramme : 17 pour UDP, 6 pour TCP, 1 pour ICMP, 8 pour EGP, 89 pour OSPF , ... (voir /etc/protocols)
- Le "Header checksum" ou champ de contrôle de l'entête (16 bits) contient le "complément à un" du total "en complément à un" de tous les mots de 16 bits de l'en-tête.
- Les adresses IP source et destination sont codées sur 32 bits
- A la rubrique "Options", sont stockées des demandes spéciales pour requérir un routage particulier pour certains paquets. + le champ "padding" (bourrage) est habituellement rempli de 0 de manière à aligner le début des données sur un multiple de 32 bits.

ICMP

Le protocole ICMP (Internet Control Message Protocol RFC 792) utilise les datagrammes IP pour transporter ses messages.



Le protocole ICMP permet par exemple :

- le contrôle de flux : le récepteur débordé par un émetteur trop rapide, envoie un message ICMP Source Quench pour arrêter temporairement l'émission
- la détection de destinations inaccessibles dénoncée par un message Destination Unreachable
- la redirection de routes pour avertir une machine hôte d'utiliser un autre gateway.

ICMP fournit d'intéressantes données pour le diagnostic d'opérations du réseau. ICMP utilise des datagrammes IP pour véhiculer des messages allerretour entre noeuds concernés. Un message d'erreur ICMP est généré par une machine hôte réalisant qu'il y a un problème de transmission et renvoyé à l'adresse de départ du datagramme ayant provoqué le problème.

Le protocole ICMP est utilisé notamment par la commande ping qui :

- emet un paquet ICMP "demande d'écho" (type=8 et code=0) et
- reçoit, si la machine distante est active (alive), un paquet ICMP "réponse d'écho" (type=0 et code=0).