TP : Analyse du datagramme IP

1. Introduction

1.1. Objectifs

L'objectif de ce TP est d'analyser et d'interpréter les différents champs du datagramme IP et d'étudier le mécanisme de fragmentation IP.

1.2. Pré-requis

Cours Réseaux Ethernet, modèle OSI et le protocole IP.

1.3. Équipements et logiciels proposés

Un ordinateur de type *PC sous windows 7,*
Un logiciel (*sniffer)* de capture et d'analyse de trames de réseau : *Wireshark.*

1.4. Documents fournis

2. Rappels

2.1. Modèle OSI

C'est le protocole Ethernet / CSMA/CD (norme 802.3) qui est utilisé dans les cartes Ethernet qui sont dans les ordinateurs.

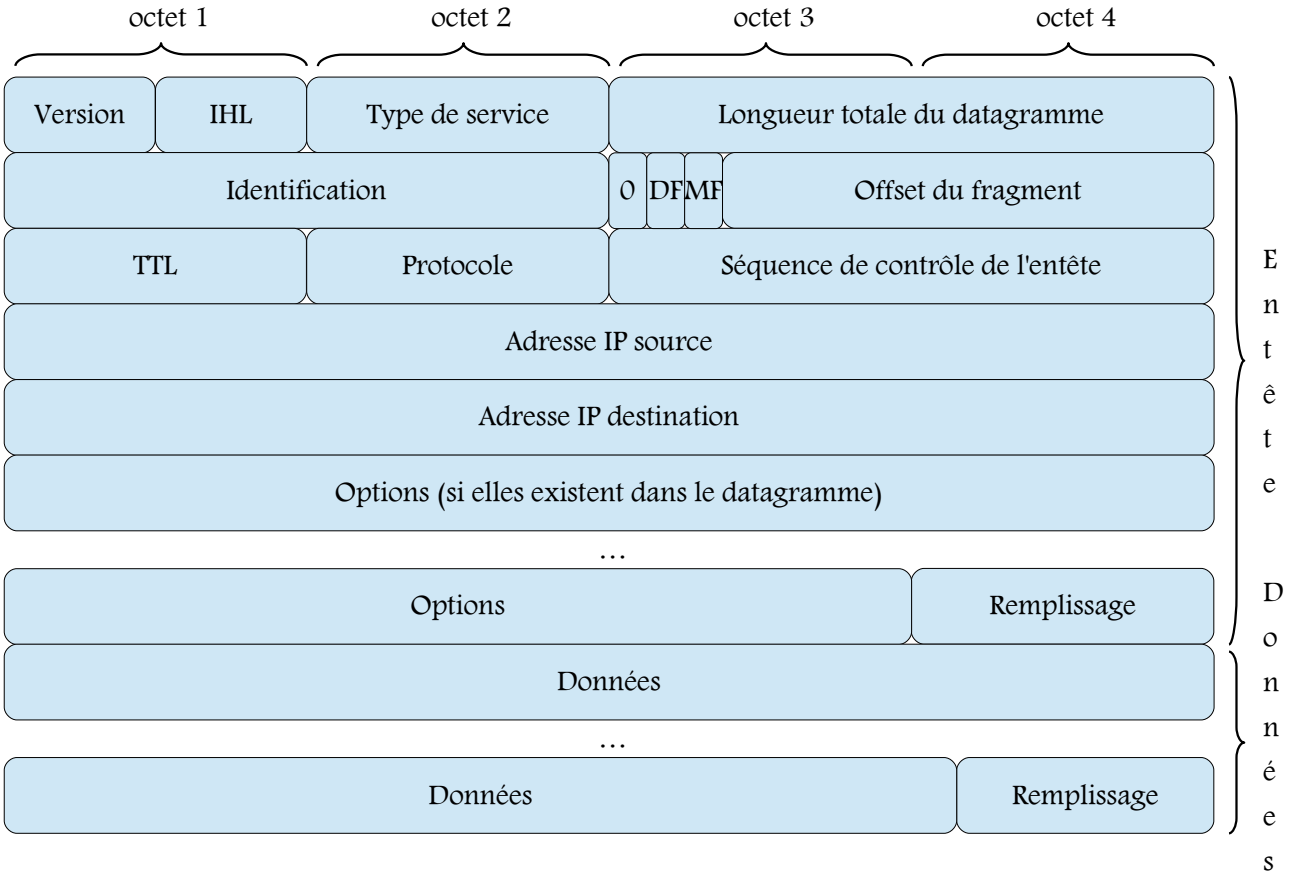
Les protocoles ARP/RARP (Address Resolution Protocol/Reverse Address Resolution Protocole), IP (Internet Protocol) et ICMP (Internet Control Message Protocol) sont associés à la couche 3 du modèle OSI.

Les protocoles TCP (Transmission Control Protocol) et UDP (User Datagram Protocol) sont associés à la couche 4.

<table>
<tr><td>7</td><td>Application</td><td>Ping</td><td>Telnet</td><td>TFTP</td><td></td><td></td></tr>
<tr><td>6</td><td>Présentation</td><td rowspan="3"></td><td rowspan="2"></td><td rowspan="2"></td><td colspan="2" rowspan="3"></td></tr>
<tr><td>5</td><td>Session</td></tr>
<tr><td>4</td><td>Transport</td><td>TCP</td><td>UDP</td></tr>
<tr><td>3</td><td>Réseau</td><td>ICMP</td><td colspan="2">IP</td><td>ARP</td><td>RARP</td></tr>
<tr><td>2</td><td>Liaison de données</td><td colspan="5" rowspan="2">Ethernet
et Interface Matérielle</td></tr>
<tr><td>1</td><td>Physique</td></tr>
</table>

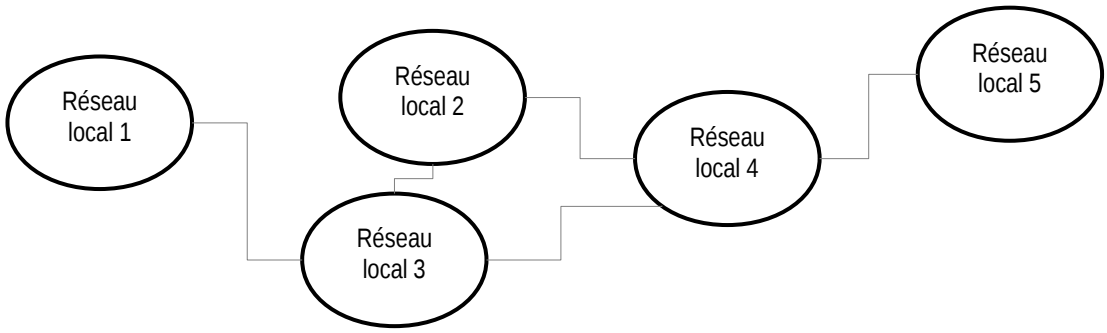
2.2. Protocole IP (couche 3)

Nous rappelons le format du datagramme IP :



Rappelons aussi qu'il existe plusieurs types de réseaux dont :

- Le réseau local (LAN) : il permet de regrouper un ensemble de machines peu éloignées géographiquement.
- Le réseau étendu (WAN) : il permet de relier des réseaux locaux à l'aide d'un lien de longue distance.



Ainsi le protocole IP permet aux couches supérieures de faire l'abstraction de l'ensemble des réseaux (locaux et étendus) qu'il faut parfois traverser pour acheminer le paquet. Les couches supérieures n'ont pas à se soucier de la route parfois compliquée que les paquets doivent emprunter.

Les couches supérieures voient les liaisons comme une liaison directe entre la machine émettrice et la machine réceptrice.

Les **routeurs**, qui sont les matériels actifs de la couche 3, sont utilisés pour la gestion des routes à prendre dans un réseau étendu. Cette action est appelée le **routage**. À chaque fois qu'un datagramme arrive à un routeur, celui-ci décrémente la valeur du champ *TTL* (Time To Live) de un. Si le TTL arrive à 0, alors le datagramme est détruit au niveau de ce routeur et n'ira donc pas plus loin.

La fonctionnalité du champ TTL est d'éviter de faire circuler des trames en boucle infinie.

D'autre part, ces différents réseaux peuvent être hétérogènes, c'est-à-dire utilisant des protocoles de niveaux 2 différents et des trames de différentes longueurs (différents MTUs). Le protocole IP doit donc aussi s'occuper de découper et de ré-assembler les paquets pour se conformer à ces MTUs. Ce mécanisme s'appelle « fragmentation de paquets ».

Le service de base offert pas IP est l'émission et la réception de paquets appelés « datagrammes ». Ce service est dit non fiable dans la mesure où la perte ou l'altération d'un paquet pendant son transport n'est pas contrôlée par IP.

Aucun mécanisme permettant de récupérer ces erreurs n'existe dans le protocole IP.

3. Travail à effectuer

3.1. Analyse du datagramme IP

3.1.1. Adresses IP

À l'aide de la commande ipconfig, par exemple, relever l'adresse IP de votre poste de travail, ainsi que l'adresse IP du poste de travail voisin et noter le nom du voisin choisi.

Pour l'analyse du datagramme IP, la commande *ping va être utilisée*. Cette commande utilise deux types de message du protocole ICMP

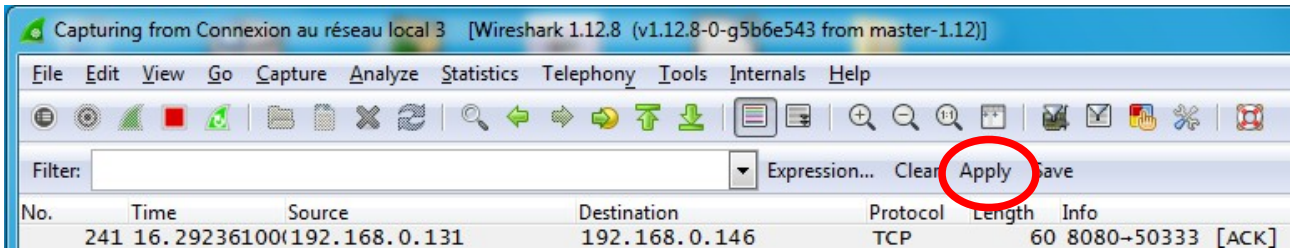
- Le type 8 (*echo request*) est émis vers la machine distante,
- Le type 0 (*echo reply*) est émis par la réponse de la machine distante.

> *ICMP sera vu plus tard. Sachez simplement qu'ICMP rajoute aux données un en-tête de 8 octets dans ce cas*

Afin de configurer le logiciel de capture et d'analyse de réseau *Wireshark* pour afficher uniquement les trames correspondant à la commande ping et au protocole *IP* partant de votre poste de travail, on choisira le filtre d'affichage suivant :

icmp and (ip.src == 192.168.0.x) and (icmp.type == 8) où x est le numéro de votre poste.

Saisir ce filtre et ne pas oublier de l'appliquer en appuyant su "apply"



Lancer une capture de trames (rien ne doit s'afficher tant que votre poste n'envoie pas de trames).

3.1.2. Commande ping

On veut effectuer un ping vers le poste du voisin avec une taille des données = taille par défaut = 32 octets et un TTL=1.

Trouver et écrire sur votre rapport la syntaxe exacte pour cette commande (voir les options nécessaires de la commande "ping" en annexe de ce document).

3.1.3.

Exécuter la commande ping de la question précédente vers le poste du voisin (son adresse IP). Cette requête a-t-elle abouti ?

Exécuter la même commande ping en remplacent le poste du voisin par www.google.fr. Cette requête a-t-elle abouti ?

Expliquer les deux résultats.

3.1.4.

Ping (sous windows) envoie par défaut 4 trames. Relever les valeurs en hexadécimal d'une des trames envoyées à la station voisine (faire attention à ce que ce ne soit pas une réponse à une requête du voisin, modifier le filtre en ajoutant la condition : icmp.type ==8).

Copier cette trame sur votre rapport (pour cela sélectionner la trame, en appuyant sur « Ctl C » au niveau du numéro de ligne et copier en appuyant sur « Ctl V ») et fermer Wireshark.

3.1.5.

Relever l'entête de la trame Ethernet. Vérifier que l'adresse MAC source correspond bien à celle de votre PC.

3.1.6.

Vérifier en relevant la valeur du champ « Type » de la trame Ethernet que celle-ci transporte bien un datagramme IP. Cela nous permettra en effet de pouvoir appliquer le datagramme IP pour décoder le contenu de cette trame.

Relever le contenu en hexadécimal des 20 premiers octets du datagramme IP.

3.1.7.

Donner la valeur du champ « version » et sa signification dans notre cas

3.1.8.

Donner la valeur du champ « IHL » et calculer la taille de l'en-tête IP en octets. À partir de ce résultat indiquer si l'entête du datagramme IP contient le champ « Option ».

3.1.9.

Donner la valeur du champ « Type de service » et sa signification dans notre cas.

3.1.10.

Donner la valeur du champ « Longueur totale du datagramme » en hexadécimal et en décimal.

3.1.11.

Déterminer alors la taille du champ « Données » du datagramme IP. Relevez le champ « Données » du datagramme IP.

À partir de ce résultat, indiquer si les données du datagramme IP contiennent un « Remplissage »

3.1.12.

Donner la valeur du champ « Identification ».

3.1.13.

Donner la valeur des drapeaux DF et MF. Que signifient ces valeurs dans ce cas ?

3.1.14.

Donner la valeur du champ « offset du Fragment ».

3.1.15.

Les données encapsulées dans le datagramme étudié sont-elles fragmentées ? Justifiez la réponse.

3.1.16.

Donner la valeur décimale du champ TTL ou « Durée de vie ». La valeur relevée correspond-elle à la ligne de commande ping des questions 3.1.1 et 3.1.2 ?

3.1.17.

Donner la valeur du champ « Protocole » et indiquer le nom du protocole encapsulé dans le champ de données du datagramme IP.

3.1.18.

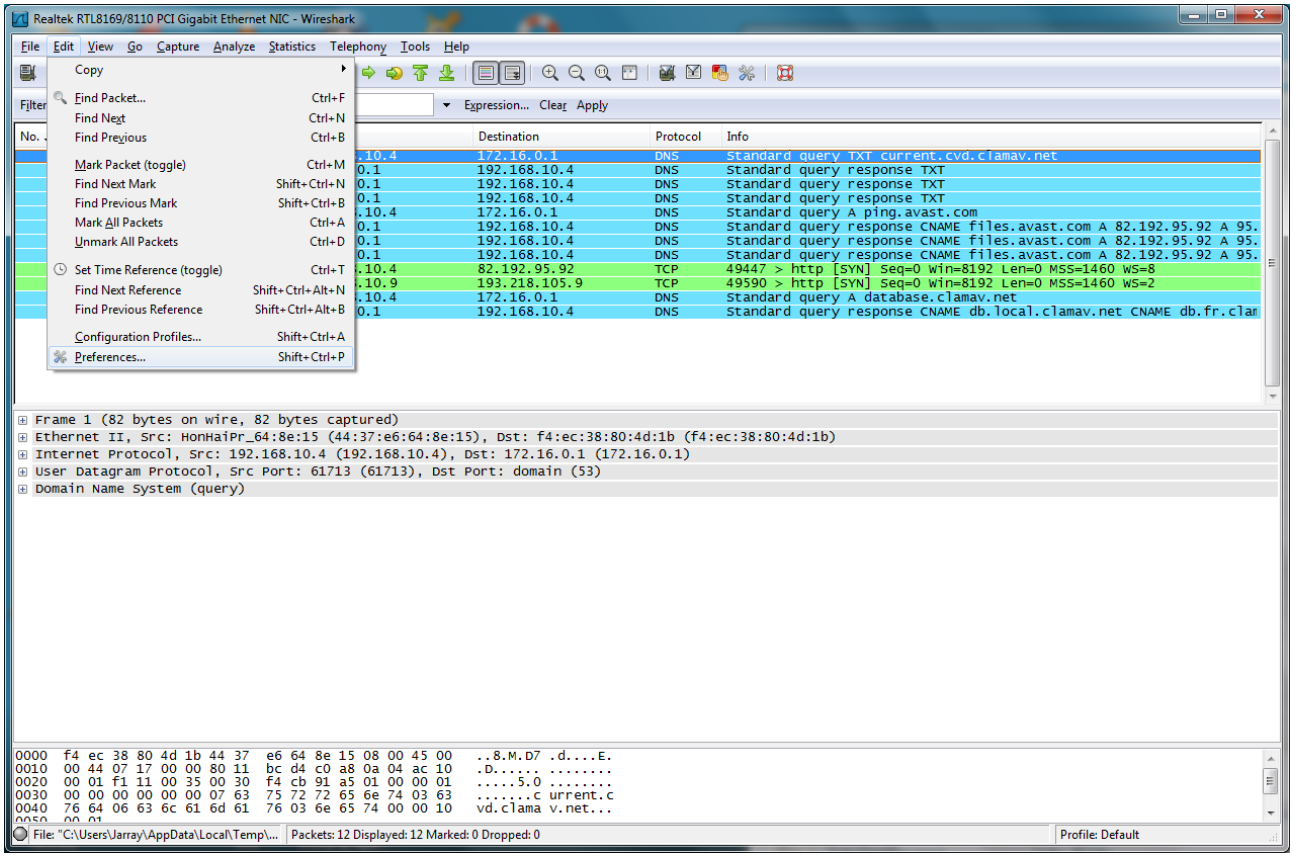
Donner, en notation décimale pointée, l'adresse IP de l'émetteur du datagramme.

Donner, en notation décimale pointée, l'adresse IP de la station de destinataire du datagramme.

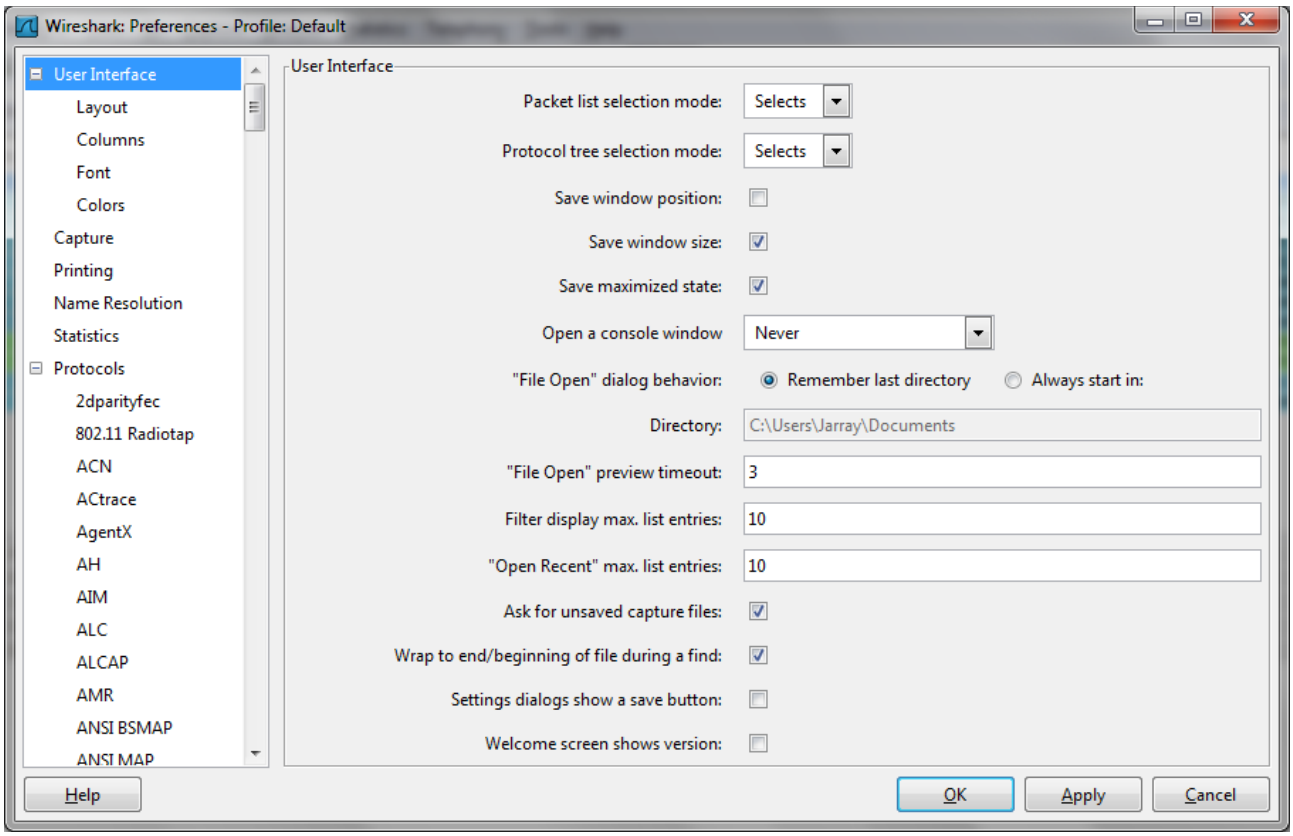
3.2. Fragmentation des datagrammes IP

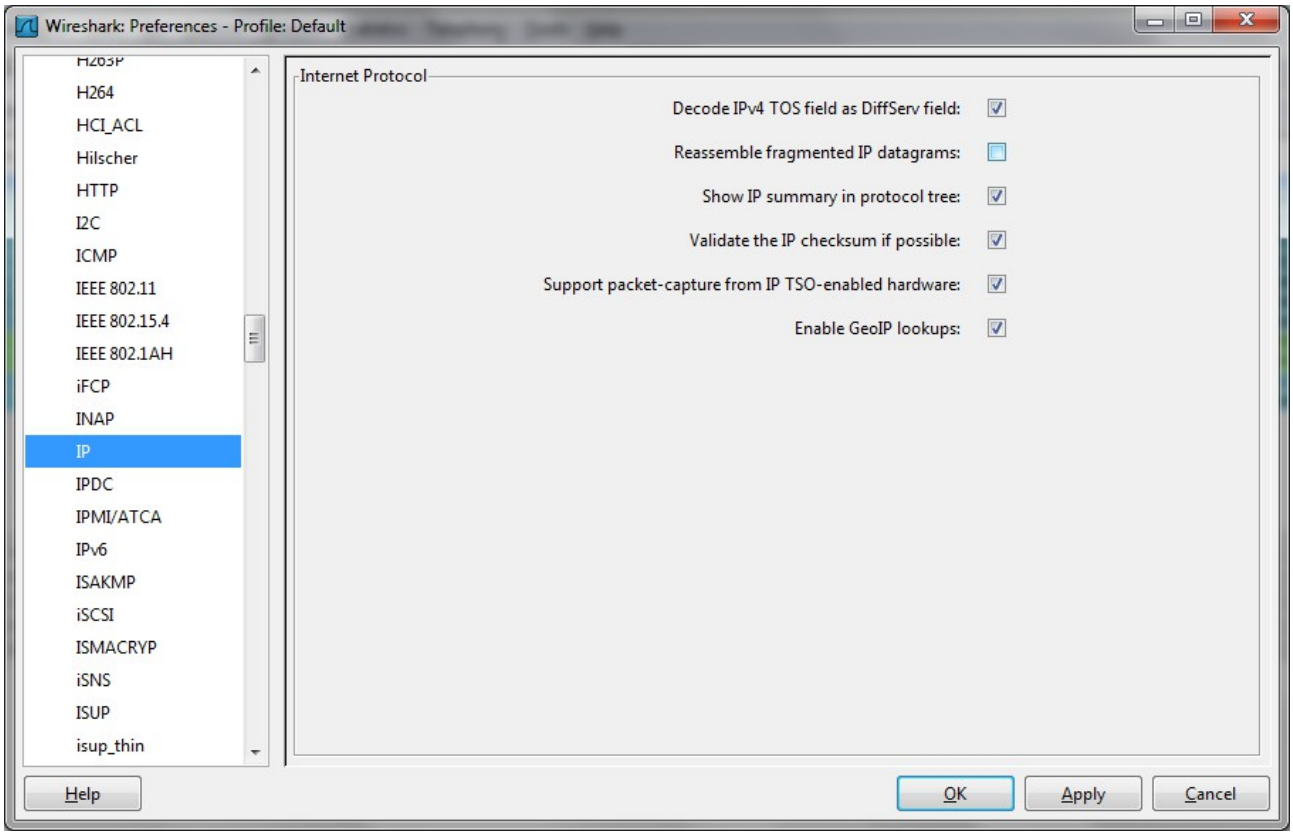
Désactivation du ré-assemblage : Par défaut, Wireshark ré-assemble les fragments d'un même message et ne montre sur son écran d'affichage que la trame correspondant au message ré-assemblé. Pour mettre en évidence la fragmentation éventuelle et voir tous les fragments, il faut désactiver le ré-assemblage Wireshark de la façon suivante :

Aller dans le menu « Edit » puis dans « Preferences »



Ensuite étaler le menu « Protocols » en appuyant sur le signe + à gauche de ce menu. Aller sur le choix IPv4 et décocher l'option « Reassemble fragmented IPv4 datagrams ».


Wireshark: Preferences - Profile: Default
User Interface
Layout
Columns
Font
Colors
Capture
Printing
Name Resolution
Statistics
Protocols
2dparityfec
802.11 Radiotap
ACN
ACtrace
AgentX
AH
AIM
ALC
ALCAP
AMR
ANSI BSMAP
ANSI MAP
User Interface
Packet list selection mode: Selects
Protocol tree selection mode: Selects
Save window position:
Save window size:
Save maximized state:
Open a console window Never
"File Open" dialog behavior: Remember last directory Always start in:
Directory: C:\Users\Jarray\Documents
"File Open" preview timeout: 3
Filter display max. list entries: 10
"Open Recent" max. list entries: 10
Ask for unsaved capture files:
Wrap to end/beginning of file during a find:
Settings dialogs show a save button:
Welcome screen shows version:
Help
OK
Apply
Cancel


Wireshark: Preferences - Profile: Default
H263P
H264
HCI_ACL
Hilscher
HTTP
I2C
ICMP
IEEE 802.11
IEEE 802.15.4
IEEE 802.1AH
iFCP
INAP
IP
IPDC
IPMI/ATCA
IPv6
ISAKMP
iSCSI
ISMACRYP
iSNS
ISUP
isup_thin
Internet Protocol
Decode IPv4 TOS field as DiffServ field:
Reassemble fragmented IP datagrams:
Show IP summary in protocol tree:
Validate the IP checksum if possible:
Support packet-capture from IP TSO-enabled hardware:
Enable GeoIP lookups:
Help
OK
Apply
Cancel

Changer votre filtre d'affichage en **((ip or icmp) and ip.src == x.x.x.x)** (x.x.x.x = votre IP)

3.2.1.

Soit à effectuer un ping vers le poste du voisin avec une taille de données = 3000 octets et TTL =1
Donner la syntaxe exacte pour cette commande (voir les options nécessaires de la commande « ping » en annexe de ce document).

3.2.2.

Lancer une capture Wireshark et taper la commande ping de la question précédente (avec ses options nécessaires). À partir des trames capturées par WireShark :

Donner le nombre de trames envoyées par la station émettrice à la station voisine.

Pourquoi la station émettrice ne peut-elle pas envoyer les données en une seule fois ?

Justifier le nombre de trames envoyées par la station émettrice.

3.2.3.

Pourquoi avoir mis (ip or icmp) et non pas seulement (icmp) dans le filtre d'affichage ?

3.2.4.

Quel est le nombre maximum d'octets à envoyer avec la commande « ping » en deux fragments ?

Vérifier cela par une capture Wireshark et copier les trames sur votre rapport.

3.2.5.

Vérifier par une capture Wireshark qu'il suffit d'ajouter 1 à ce nombre maximum pour obtenir 3 fragments et copier les fragments sur votre rapport.

3.2.6.

Relever dans un tableau, la valeur des champs « `Identification` », des bits `DF` et `MF` » et « `Offset du fragment` » de tous les datagrammes émis par la station émettrice.

	Identification	DF	MF	Offset
Fragment 1				

Fragment 2				
.....				

3.2.7.

Les valeurs du champ « `Identification` » des trames émises par votre station sont-elles les mêmes ? Pourquoi ?

3.2.8.

Que signifie la valeur du bit `DF` des trois datagrammes ?

3.2.9.

Que signifie la valeur du bit MF du dernier datagramme ?

3.2.10.

Expliquer la valeur du champ « Offset » ou « Place du fragment » des deux premiers datagrammes ?

4. Analyse de l'acheminement des paquets sur le réseau Internet

Faire un ping sur le serveur google.fr : `ping google.fr`

4.1.

Relever la valeur `TTL` obtenue par la commande `ping`.

4.2.

À partir de cette valeur, donner le nombre de routeurs que le message de réponse du serveur a traversé pour arriver à votre station de travail.

> *On rappelle que les grands serveurs, comme Google, utilisent souvent le système d'exploitation UNIX qui à un TTL par défaut de 64.*

Utilisation de la commande « tracert » :

La commande `tracert` permet de connaître la route (l'ensemble des routeurs) empruntée par le paquet avant d'arriver à la destination.

Lancer une capture de trames et taper la commande : `tracert google.fr`

4.3.

À partir de la réponse de la commande tracert, relever le nombre de routeurs que le datagramme de requête a traversé.

4.4.

Faire la correspondance avec la valeur `TTL` donnée par la commande `ping`. Expliquer les deux résultats.

> *On rappelle que les datagrammes ne prennent pas forcément les mêmes chemins à chaque commande ping et que le nombre de routeurs trouvés n'est qu'approximatif.*

4.5.

Établir la correspondance entre les trames relevées et les sauts successifs affichés lors de la commande `tracert`.

5. Annexes

5.1. Commande ping:

La commande *ping* utilise le protocole `ICMP` (le protocole ICMP comporte une entête de 8 octets) pour vérifier la présence sur un réseau, d'un hôte dont on connaît l'adresse IP. Par défaut (sous Windows), elle envoie 4 trames contenant chacune une donnée de 32 octets.

```
Utilisation :

ping [-t] [-a] [-n échos] [-l taille] [-f] [-i vie] [-v TypServ]
   [-r NbSauts] [-s NbSauts] [[-j ListeHôtes] | [-k ListeHôtes]]
[-w Délai] NomCible

Options :
  -a         Recherche les noms d'hôte à partir des adresses.
```

```
-n échos   Nombre de requêtes d'écho à envoyer.
-l taille  Envoie la taille du tampon.
-f         Active l'indicateur Ne pas fragmenter dans le paquet.
-i vie     Durée de vie.
-r NbSauts Enregistre l'itinéraire pour le nombre de sauts.
-s NbSauts Dateur pour le nombre de sauts.
```

5.2. Trame Ethernet

Préambule	Adresse Mac de Destination	Adresse Mac Source	EtherType	Données	FCS
	6 octets	6 octets	2 octets		4 octets

Préambule avant la trame = suite de 1 et de 0 (10101010...), servant à la synchronisation.
Taille minimum de la trame = 64 octets, taille maximum = 1518 octets
Taille minimum des données = 46 octets, taille maximum = 1500 octets

EtherType

0x0000 à 05DC	IEEE 802.3 : Longueur des données
0x0800	Internet IP (IPv4)
0x0806	ARP
0x6003	DECNET
0x86DD	IPv6
0x880B	PPP
0x8847	MPLS Unicast
0x8848	MPLS Multicast
0x8100	802.1Q TPID for VLAN Frame
0x884C	MPOA