Ethereal/Wireshark

un analyseur de protocoles réseau

Licence GFDL

Ce document est soumis à la Gnu Free Documentation Licence. C'est à dire que :

- toute personne a le droit d'utiliser, diffuser et modifier ces documents
- à condition d'indiquer la provenance du document original
- à condition que les documents modifiés ou diffusés soient eux aussi soumis à la Gnu Free Documentation Licence et accessibles en ligne
- j'apprécie d'avoir des retours sur les utilisations de ce document et/ou sur d'éventuelles erreurs/typo/màj/...
- GFDL: http://cesarx.free.fr/gfdlf.html

wireshark: présentation

wireshark est un analyseur de trame.

outil libre en constante évolution

de nombreux greffons lui permettent de décoder de nombreux protocoles

livré avec des outils en lignes de commande permettant la capture de trame, la conversion de formats, ...

wireshark: fonctionnalités

- analyse de protocol réseau
- capture et analyse de trames
- sauvegarde/lecture de capture précédemment sauvegardées
- décompose les différentes couches réseaux présentes dans une trame
- compatible avec les formats de sauvegardes de nombreux logiciels

architecture en couche

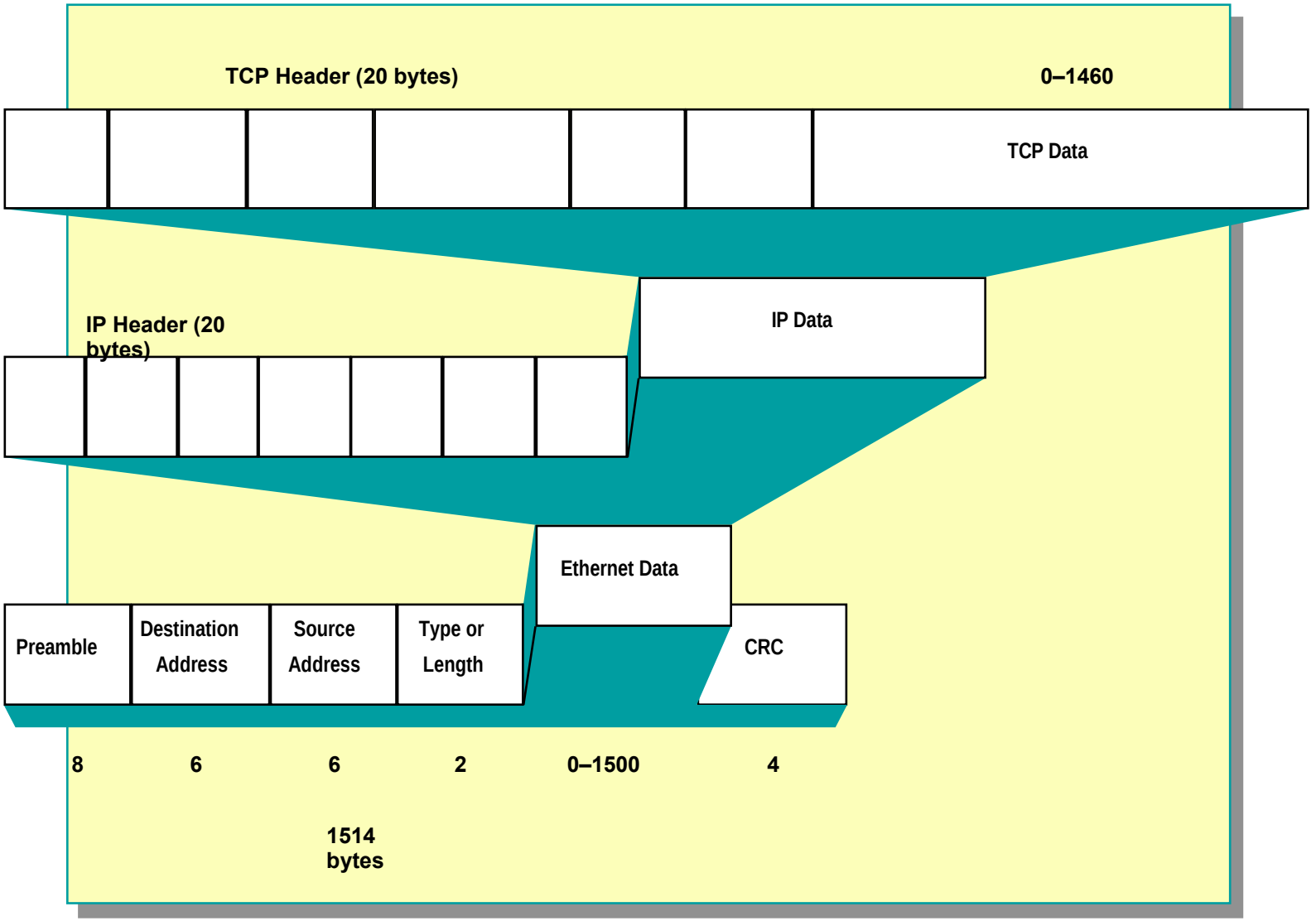


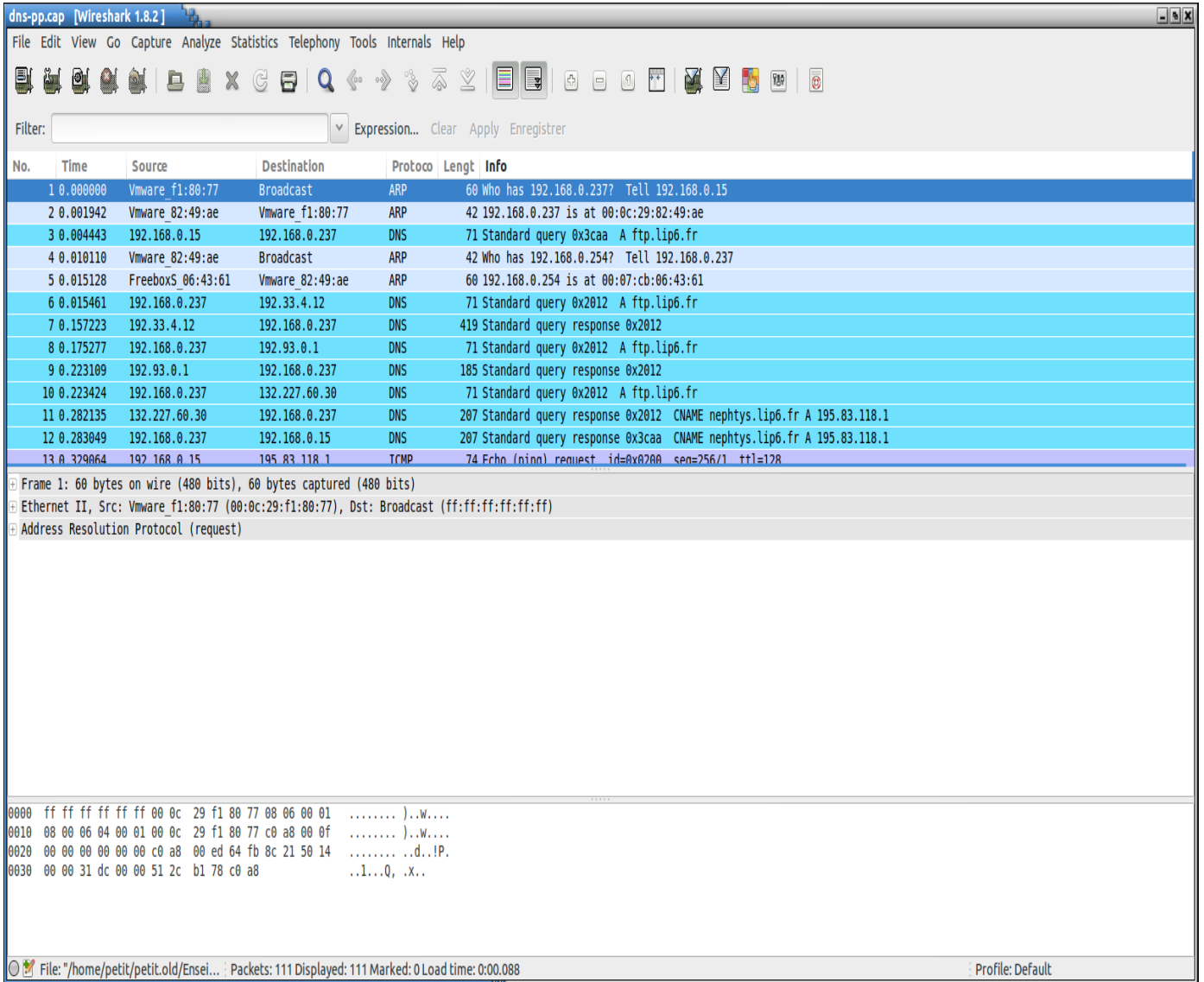
schéma: M. Besson

wireshark: écran

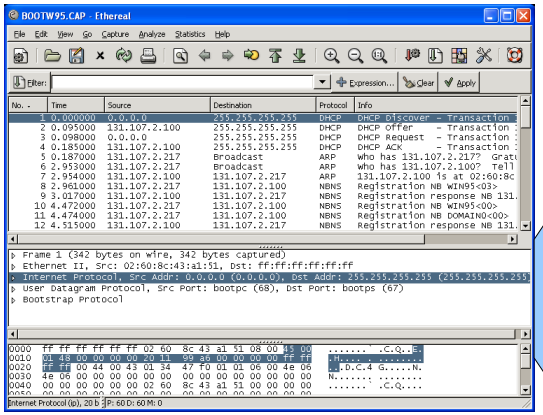
Liste des trames

détail d'une trame

contenu hexa



détail d'une trame




```
▽ Frame 1 (342 bytes on wire, 342 bytes captured)
    Arrival Time: Oct  2, 1996 08:21:08.705000000
    Time delta from previous packet: 0.000000000 seconds
    Time since reference or first frame: 0.000000000 seconds
    Frame Number: 1
    Packet Length: 342 bytes
    Capture Length: 342 bytes
▽ Ethernet II, Src: 02:60:8c:43:a1:51, Dst: ff:ff:ff:ff:ff:ff
    Destination: ff:ff:ff:ff:ff:ff (Broadcast)
    Source: 02:60:8c:43:a1:51 (131.107.2.217)
    Type: IP (0x0800)
▽ Internet Protocol, Src Addr: 0.0.0.0 (0.0.0.0), Dst Addr: 255.255.255.255
    Version: 4
    Header length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 328
    Identification: 0x0000 (0)
  ▷ Flags: 0x00
    Fragment offset: 0
    Time to live: 32
    Protocol: UDP (0x11)
    Header checksum: 0x99a6 (correct)
    Source: 0.0.0.0 (0.0.0.0)
    Destination: 255.255.255.255 (255.255.255.255)
▽ User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)
    Source port: bootpc (68)
    Destination port: bootps (67)
    Length: 308
```

Ethereal: deux types de filtres

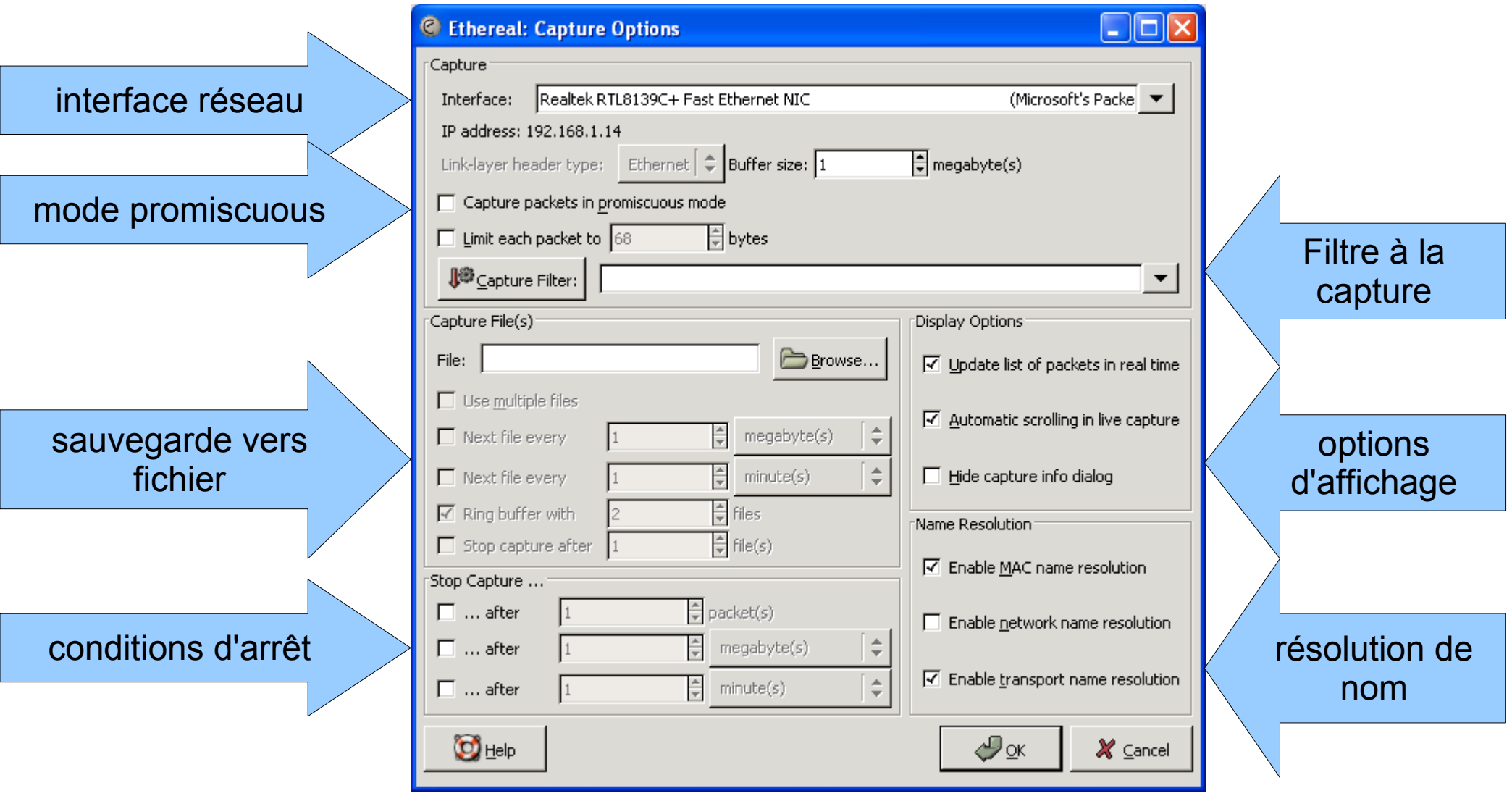
filtres à la capture:

- sélectionner les trames à capturer
- moins pratique et convivial que les filtres d'affichage
- réduit le nombre de trames à capturer

filtres d'affichage:

- langage simple, création avec un assistant
- sélectionner les trames à afficher
- colorier les trames affichées

Lancement d'une capture



Filtres à la capture

langage de filtre de libpcap, utilisable avec tcpdump

forme générale d'un filtre à la capture :

[not] primitive [and|or [not] primitive ...]

Exemple :

tcp port 23 and host 10.0.0.5

cf http://www.tcpdump.org/tcpdump_man.html pour une descriptions complète

Filtres à la capture (2)

[src|dst] host <host>	sélection des paquets selon l'adresse ip source (src) ou destination (dst) ou les deux si on ne précise pas src ou dst. «
ether [src|dst] host <ehost>	idem selon l'adresse ethernet source ou destination
gateway host <host>	paquet utillisant *host* comme routeur: routeur est source ou destination au niveau ethernet mais pas IP.
[src|dst] net <net> [{mask <mask>}|{len <len>}]	sélection des paquets ayant un sous-réseau comme source ou destination. le masque peut être indiqué explicitement ou en notation CIDR
[tcp|udp] [src|dst] port <port>	sélection de paquets selon le port source/destination et le protocole tcp/udp
less|greater <length>	filtrage sur la taille du paquet: « inférieur ou égaux » ou « supérieur ou égaux »
ip|ether proto <protocol>	sélection du protocole soit de la couche IP soit de la couche ethernet

Filtres à l'affichage

langage de filtre différent de celui des filtres à la capture: &&, ||, (,) et des expressions

sert à la sélection des trames affichés et à la colorisation des trames

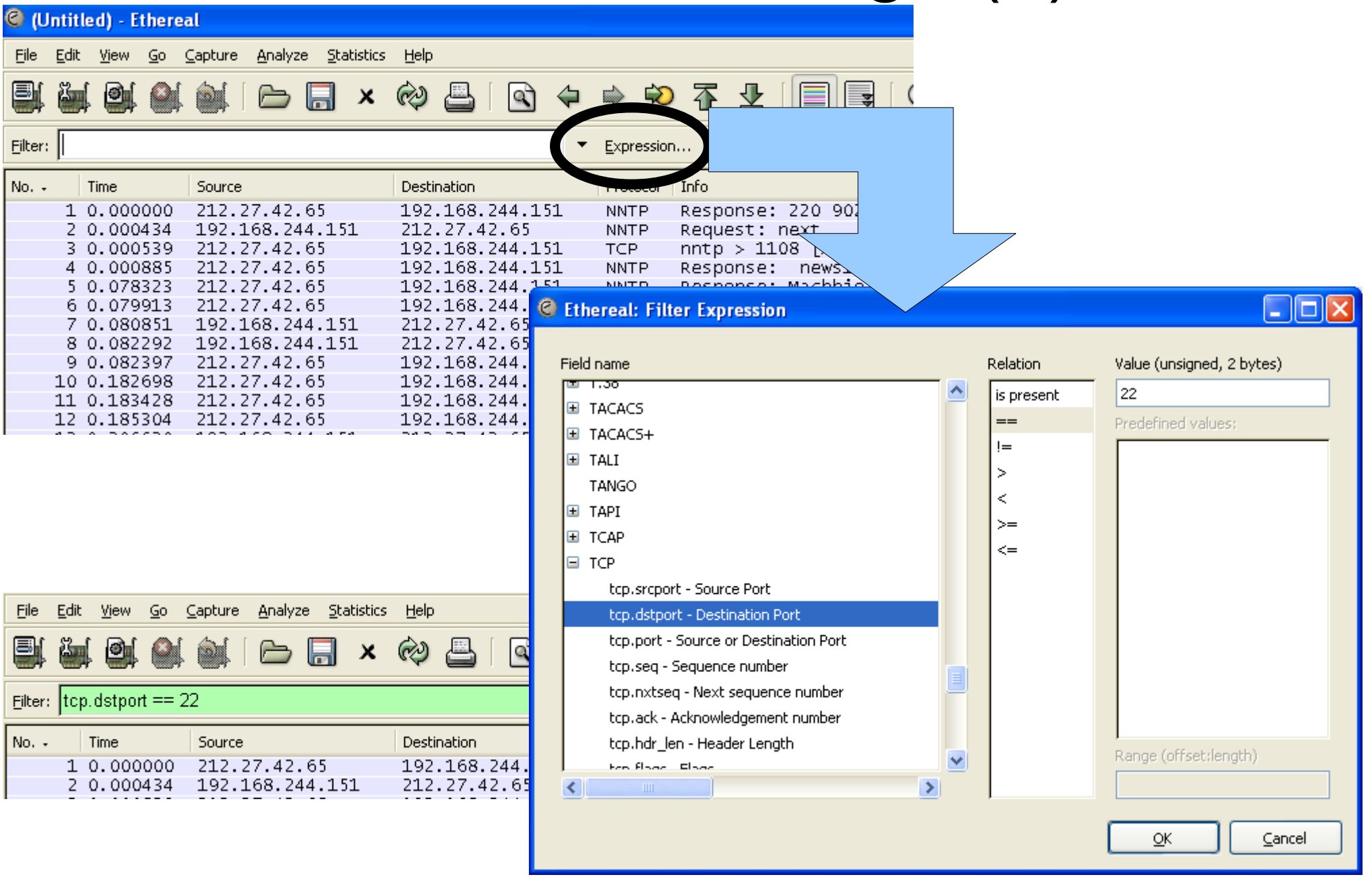
dépend des routines de décodage de chaque protocole

=> évolue beaucoup d'une version à l'autre

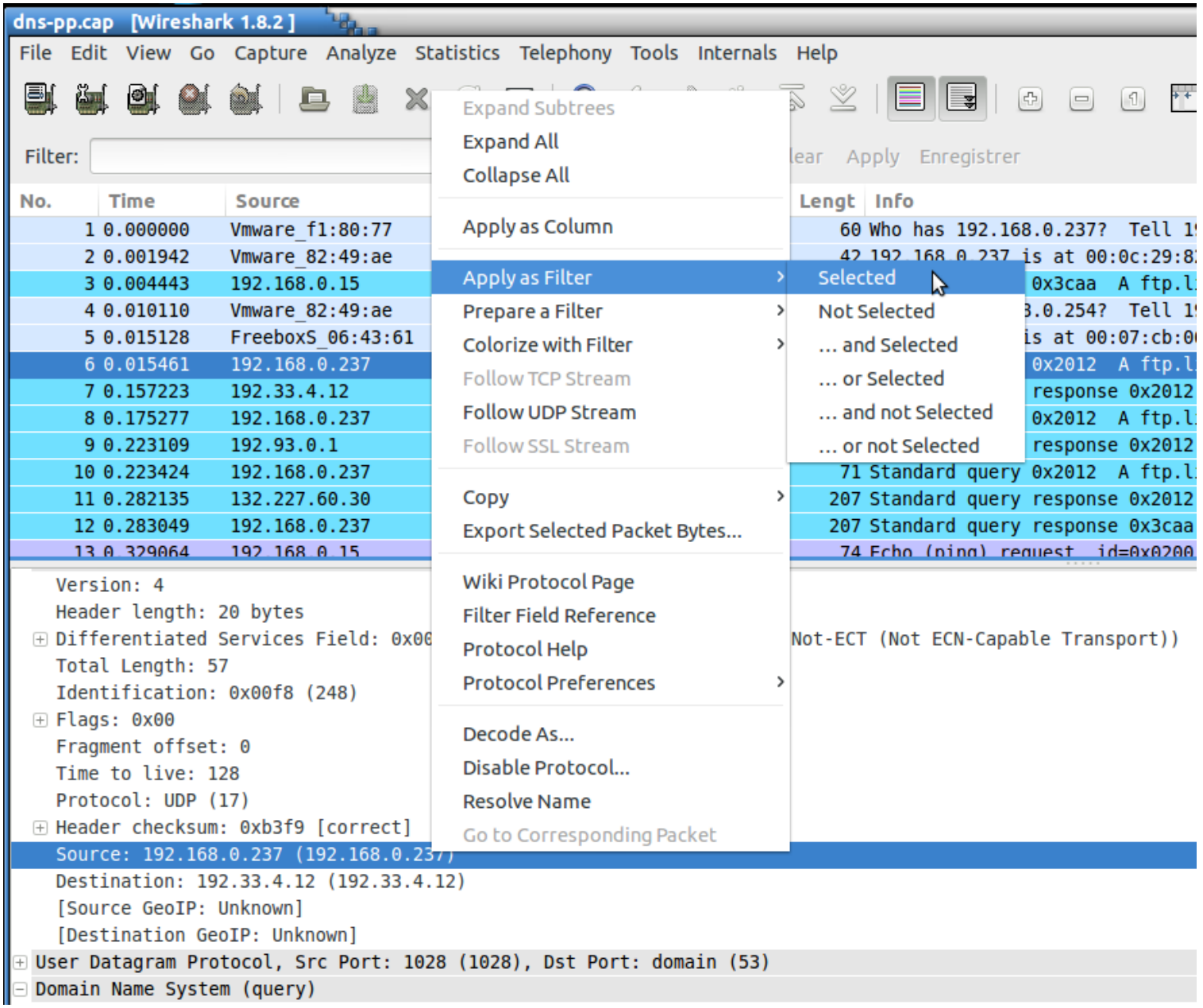
guide de référence du filtre d'affichage: http://www.ethereal.com/docs/dfref/

ne pas oublier de cliquer sur « Apply » pour appliquer le filtre courant

Filtres à l'affichage (2)



Filtres à l'affichage (3)



Filtres à l'affichage (4)

- il est possible de sélectionner les paquets ayant le même critère qu'un paquet existant grâce au menu contextuel
- cf diapo précédente

Exercices

charger « bootw95.cap » situé dans captures_base

sélectionner les trames tcp

sélectionner les trames dhcp (voir Bootp/Dhcp)

les trames dont l'adresse ip destination est 255.255.255.255

coloriage et divers

- coloriage: colorier les trames vérifiant certains filtres
  - couleur de la trame = celle du premier filtre auquel correspond elle correspond
  - via « View/coloring rules »
- « set time reference » (menu edit): l'horodatage des trames suivants se fait en référence à cette trame
- « Edit/mark Packet »: marquer la trame pour la repérer

Statistiques: protocol hierarchy

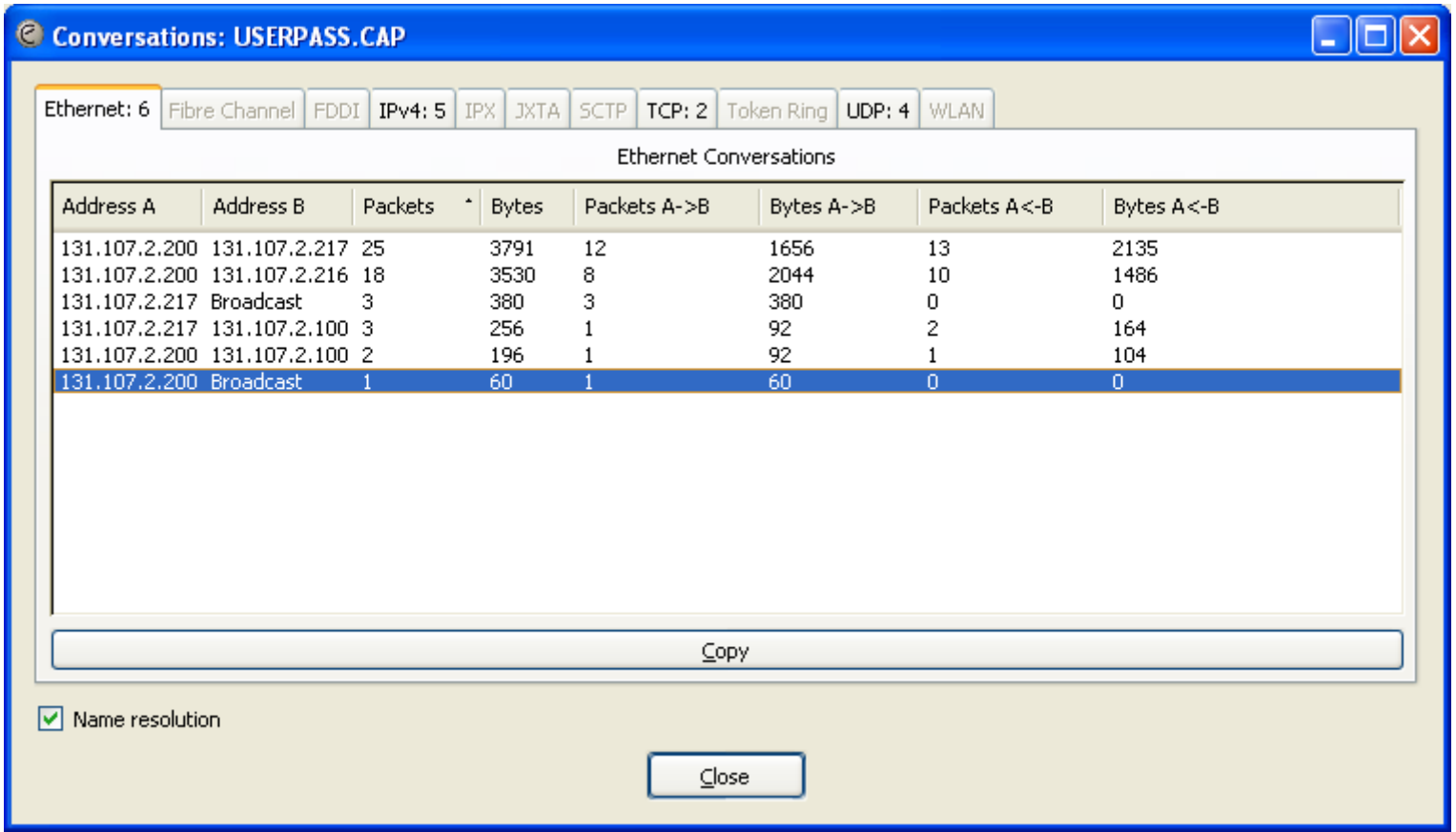
« protocol hierarchy »: nombre de trames, débit, ... présenté hiérachiquement selon le modèle en couche

Protocol	% Packets	Packets	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s
Frame	100,00%	52	8213	0,002	0	0	0,000
Ethernet	100,00%	52	8213	0,002	0	0	0,000
Address Resolution Protocol	11,54%	6	360	0,000	6	360	0,000
Internet Protocol	88,46%	46	7853	0,001	0	0	0,000
User Datagram Protocol	11,54%	6	1044	0,000	0	0	0,000
NetBIOS Name Service	7,69%	4	392	0,000	4	392	0,000
NetBIOS Datagram Service	3,85%	2	652	0,000	0	0	0,000
SMB (Server Message Block Protocol)	3,85%	2	652	0,000	0	0	0,000
SMB MailSlot Protocol	3,85%	2	652	0,000	0	0	0,000
Microsoft Windows Browser Protocol	1,92%	1	260	0,000	1	260	0,000
Microsoft Windows Logon Protocol (Old)	1,92%	1	392	0,000	1	392	0,000
Transmission Control Protocol	76,92%	40	6809	0,001	12	720	0,000
NetBIOS Session Service	53,85%	28	6089	0,001	4	372	0,000
SMB (Server Message Block Protocol)	46,15%	24	5717	0,001	14	2357	0,000
SMB Pipe Protocol	19,23%	10	3360	0,001	0	0	0,000
Microsoft Windows Lanman Remote API Protocol	11,54%	6	1852	0,000	6	1852	0,000
DCE RPC	7,69%	4	1508	0,000	2	396	0,000
Microsoft Network Logon	3,85%	2	1112	0,000	2	1112	0,000

Statistiques: conversations

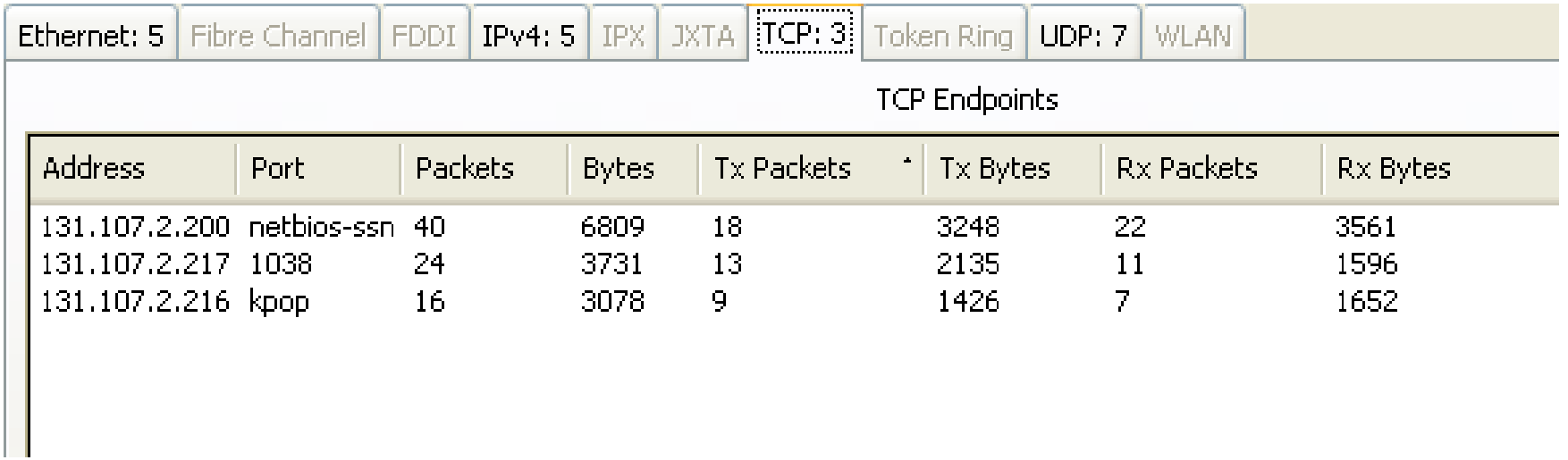
qui cause à qui: résumés par couche

chaque onglet peut s'obtenir séparemment via « conversation lists »



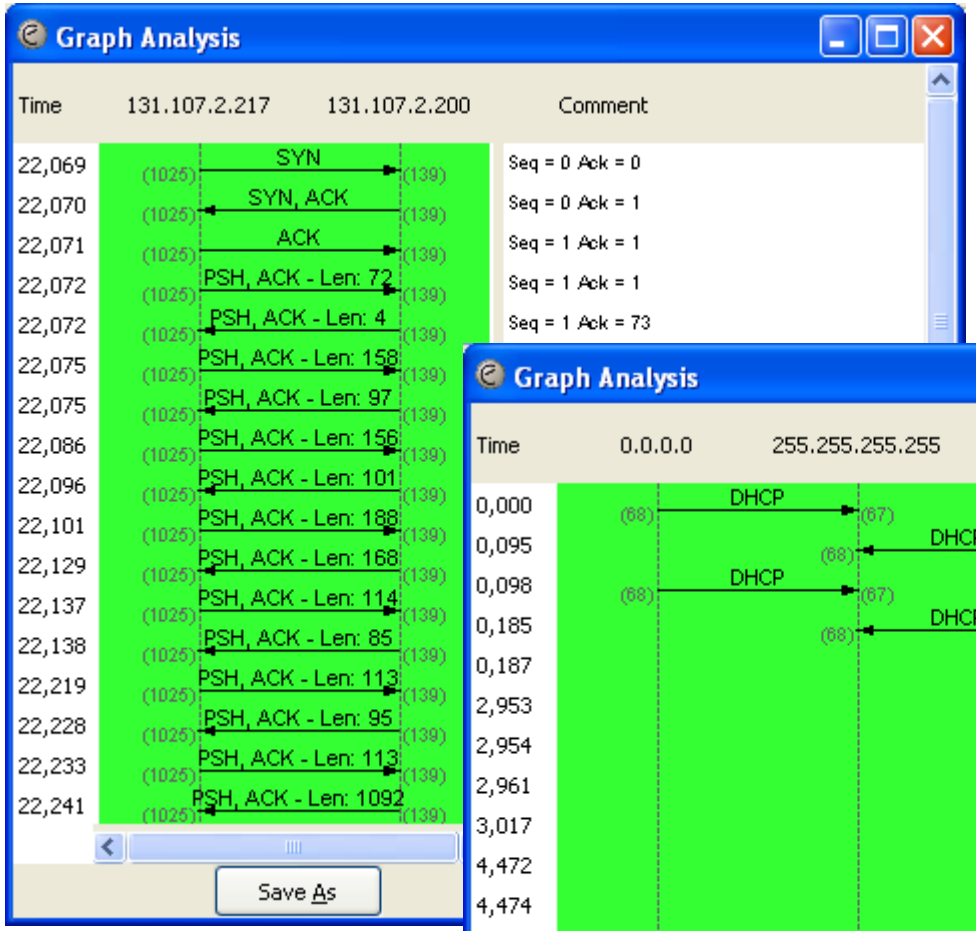
Statistiques: EndPoints

- indique les destinations des divers traffic. La notion dépend de la couche considérée: adresse MAC pour ethernet, adresse IP pour IP, adresse IP+port pour tcp ou udp, ...
- chaque onglet peut s'obtenir séparemment via « EndPoints lists »

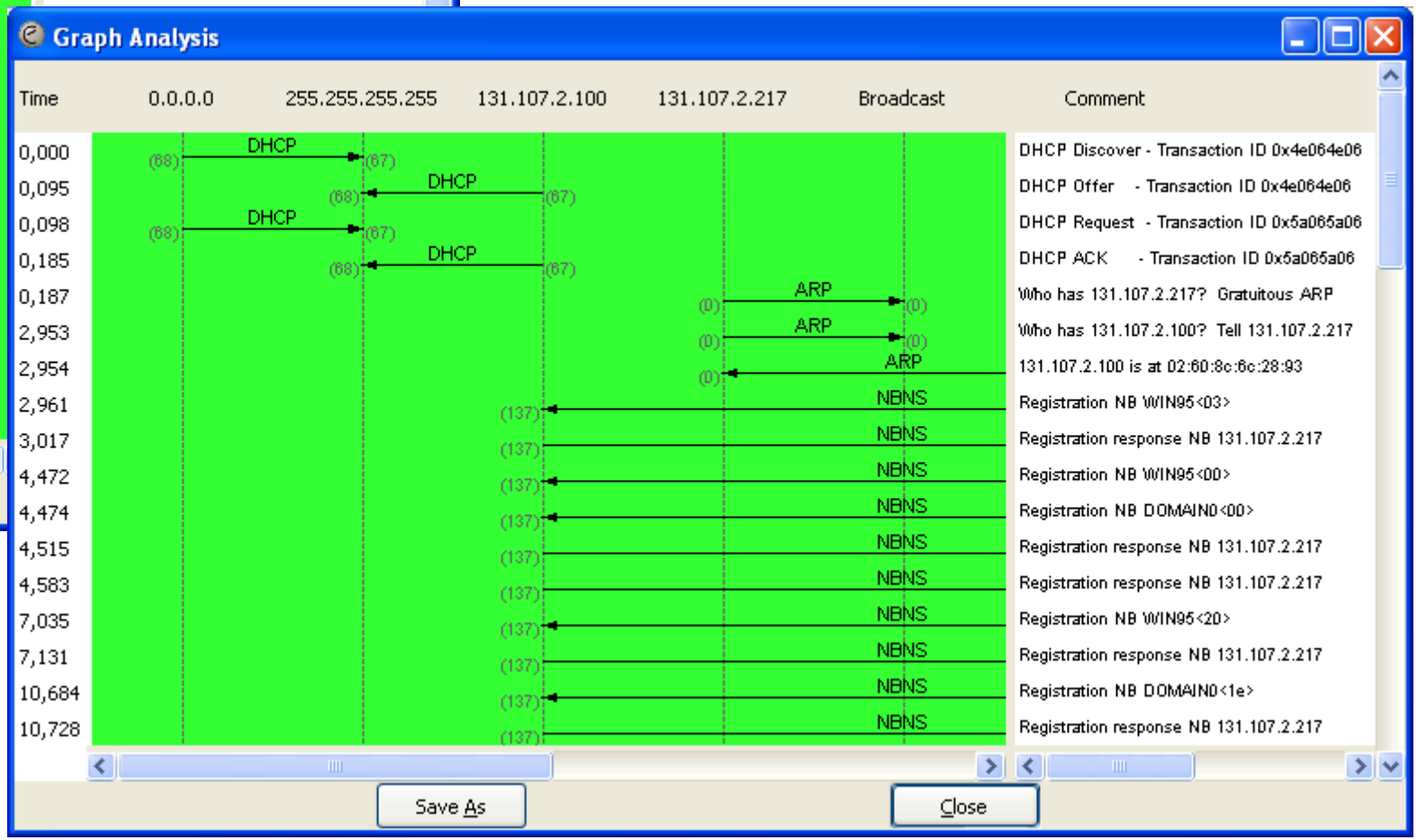


Statistiques: diagramme de flot

TCP



Tout le traffic



Ethereal: performances

perte de trames: ethereal n'arrive plus à suivre

Solutions possibles

- désactiver l'affichage en temps réel des trames
- désactiver les filtres à la capture si la quantité capturée est grande
- activer les filtres à la capture si seul une faible part des trames est utile
- arrêter les autres programmes (antivirus, daemon chargés, ...)
- utiliser un outil dédié à la capture (tethereal, tcpdump, ...) puis analyser le fichier sauvé avec ethereal