Analyse des **P**erformances des
Réseaux **I**nformatiques **L**ocaux

Sécurité des **S**ystèmes d'**I**nformation

Analyse de trames réseaux

L'analyseur (sniffer) « WireShark »

Version du document	Auteur
Décembre 2011	Rexy

Table des matières

1. Préambule

La version originale de ce document a été réalisée par Richard REY. Certaines parties exploitent des informations provenant de documents rédigés par les auteurs suivants :

Gisèle BARET - Yan CALIBET – Jérôme FEREYRE

Règles typographiques : les commandes et leurs options, les noms de fichiers et les chemins sont présentés ainsi : */usr/local/nessus/nessus-mkcert.*

Les exemples de mise en oeuvre ainsi que les illustrations présentées dans ce document sont issus des versions V0.99.X de « wireshark ».

2. Présentation

Un analyseur de trames réseaux (network packets sniffer) est un outil (logiciel ou matériel) dont le but et de récupérer les trames transitant sur un réseau pour les présenter de manière à ce qu'elles puissent être analysées par un spécialiste du domaine. L'objectif de ce document est de présenter « WireShark », un des meilleurs analyseurs graphiques de trames, librement disponible aujourd'hui.

Gerald Combs (gerald@ethereal.com) a débuté le développement d'« Ethereal » en 1997 pour aboutir à la première version utilisable en 1998 (v0.2). C'est à partir de cette version que plusieurs contributions sont venues enrichir les fonctionnalités du produit. À ce stade, l'originalité consistait à dissocier les modules de décodage de protocole (dissectors) de la partie principale (menu, affichage, etc.). Cette modularité a rapidement permis à certains développeurs de créer leurs propres décodeurs et ainsi d'augmenter rapidement le nombre de protocoles reconnus par « Ethereal ». En 2006, suite à un problème de « marque déposée », « Ethereal » a été rebaptisé « WireShark (le requin du câble).

Aujourd'hui, « WireShark » évolue au rythme des diverses contributions et améliorations de cette communauté de développeurs. À titre d'exemple, la version 0.99 reconnaît plus de 600 protocoles de communication. Chaque nouvelle version ajoute en général le support d'une dizaine de nouveaux protocoles et corrige ou améliore des décodeurs déjà existants.

« WireShark » ne fonctionne pas seul. Il s'appuie sur un ensemble de fonctions et de procédures de capture de trames de bas niveau. Ces fonctions et ces procédures faisaient initialement partie du projet Unix/Linux « TCPDUMP ». Devenues incontournables dans plusieurs autres projets, elles ont été rassemblées dans une bibliothèque appelée : « libpcap ». La qualité de cette bibliothèque lui a value d'être portée sur d'autres types d'architectures (ex: winpcap = libpcap pour Winxx).

- Le site « WireShark » : www.wireshark.org
- Le site « Libpcap » : www.tcpdump.org
- Le site « Winpcap » : winpcap.polito.it

3. Caractéristiques

Les analyseurs de trames possédant un grand nombre de caractéristiques communes, il semble plus utile de présenter, dans ce chapitre, les avantages de « WireShark » par rapport aux produits concurrents.

- Outre le fait d'être gratuit, « WireShark » est un logiciel libre (licence GPL V2[1]) développé par plusieurs centaines de programmeurs.
- Près de 1000 protocoles sont aujourd'hui traités. La grande réactivité des équipes de développement permet à « WireShark » de traiter un grand nombre de protocoles propriétaires, notamment les plus récents :
  - P2P (eDonkey, Gnutella, etc.)
  - Ludiques (QuakeWorld, Arena, etc.)
  - Messageries instantanées et voix sur IP (AIM, ICQ, jabber, messenger, SIP, H223, etc.)
  - Sécurité (802.1x, 802.1q, Radius, SSL, SSH, Kerberos, etc.)
  - Matériels actifs des réseaux (VRRP, OSPF, SNMP V3, VTP, CDP, etc.)

1 La GPL (General Public Licence) de la FSF (Free Software Foundation) est la licence de référence dans le monde du logiciel libre. Les quatre règles suivantes définissent cette licence :
- liberté d'exécuter le programme pour tous les usages,
- liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins,
- liberté de redistribuer des copies du programme,
- liberté d'améliorer le programme et de publier ces améliorations.

De plus, la FSF a introduit la notion de « copyleft » (par opposition à « copyright ») qui oblige un logiciel GPL modifié à rester sous licence GPL. Cette notion permet de rendre la licence « contaminante » et évite ainsi les dérives propriétaires à but purement lucratif.

- Il est capable de s'appuyer sur une multitude d'interfaces réseau afin de traiter la quasi-totalité des protocoles des couches basses (Ethernet, Appletalk, Token Ring, Frame Relay, ATM, RNIS , 802.11, Fibre channel, GSM, GPRS, PPP, Bluetooth, etc.). L'annexe 1 (cf. §6.1) présente un récapitulatif du niveau de support des interfaces réseau.
- Il fonctionne sur une multitude de plates-formes (Unix, Linux, Windows, BSD, etc.).
- Il est capable d'interpréter automatiquement les fichiers de capture générés par d'autres produits (Snoop, Finisar Surveyor, Novell Lanalyzer, Microsoft Network Monitor, AIX's iptrace, NetXRay, EtherPeek, AiroPeek, RADCOM analyzer, Cisco Secure IDS, Etherwatch, Visual UpTime, CoSine, etc.). De même, il peut exporter ses captures dans divers formats (libpcap, Novell Lanalyseur, Microsoft Network Monitor, Network Associates Sniffer, etc.).

4. Installation

Comme pour l'ensemble des produits sous licence libre, deux méthodes permettent d'installer « WireShark ».

Le première et la plus simple consiste à récupérer le (ou les) fichier(s) d'installation des binaires et de les installer :

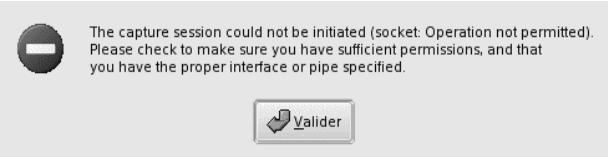
- sous Linux : installer le paquetage « wireshark ». Les dépendances suivantes seront installées « libpcap », « libwireshark0 »;
- Sous WinXX : récupérer et installer « wireshark-setup-x.exe ».

La deuxième méthode consiste à récupérer les fichiers sources et à les compiler sur son système. Sous Linux, la méthode est la même que pour la compilation de n'importe quel autre programme. Soit :

- installer les bibliothèques de fonctions dépendantes « lipbcap0-x.rpm » et « libpcap0-devel-x.rpm » ;
- récupérer et décompresser l'archive *« wireshark-xxx.tgz (ou .tar.bz2) »* dans un répertoire ;
- à partir de ce répertoire, lancer la génération de l'automate de compilation (fichier *« Makefile »*) par la commande *« ./configure »* ;
- enfin, lancer la compilation proprement dite par la commande *« make »* (Attention !!! « WireShark » étant un logiciel relativement conséquent, il faut compter plusieurs minutes de compilation).

5. Utilisation

Comme pour tout analyseur de trames réseau et afin de pouvoir profiter de toutes les fonctionnalités du produit, il est souhaitable de lancer « WireShark » en tant qu'utilisateur avec privilèges (« root » sous Unix/Linux). En effet, certaines fonctions de « WireShark » doivent pouvoir accéder à des zones de la mémoire ou à des fonctions inaccessibles à un simple utilisateur (ex. : la zone d'échange mémoire des pilotes d'interfaces réseau). Lorsque « WireShark » est lancé par un utilisateur sans privilège, les fonctionnalités interdites sont supprimées ou grisées. Ce qui dans certains cas conduit à l'apparition de la boite de dialogue suivante :



5.1. La capture

La fenêtre de lancement de la capture est obtenue en choisissant : menu « Capture » puis « Start ». Cette fenêtre est divisée en cinq sections :

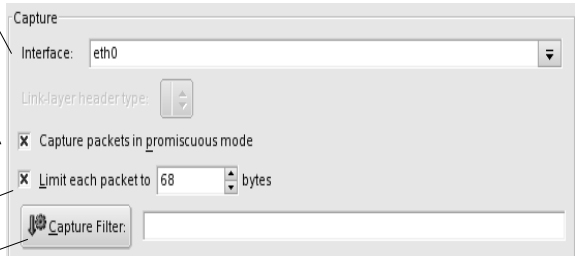
- Section « Capture » : condition de capture

Choix de l'interface réseau de capture (sous Linux : eth0 = 1ère carte ethernet)

Le mode « promiscuité », quand il est accepté par l'interface réseau, lui permet de présenter au système l'ensemble des trames se présentant sur son port d'entrée. Rappel : par défaut, une interface réseau ne présente au système que les trames lui étant destinées (ex : contenant son @MAC pour une carte ethernet) ou les trames de diffusion (broadcast et multicast).

Permet de limiter la taille maximale des trames capturées.

Permet de définir un filtre de capture (cf. §5.6).

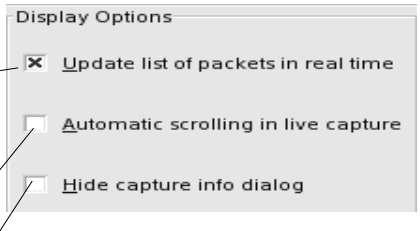


- Section « Display Options » : options d'affichage

Permet d'afficher les trames au moment où elles sont capturées (affichage temps réel). Lorsque cette option est activée, l'option de stockage de la capture dans des fichiers multiples de la section « Capture file » est inhibée (cf. ci-dessous).
Important : Dans le cas d'un réseau rapide ou très chargé, ou dans le cas d'une station peu puissante, la charge de traitement que cette option demande peut engendrer une perte de trames. Dans ce cas, l'inhiber ou utiliser Tethereal pour la capture et Ethereal pour l'analyse.

Lorsque l'affichage « temps réel » est activé, cette option permet de pointer l'affichage sur la dernière trame capturée.

Cache la fenêtre de dialogue pendant la capture (présentée ci-après). Dans ce cas, pour arrêter la capture : menu « Capture » puis « Stop ».



- Section « Capture Files » : options de stockage

Stockage de la capture dans un fichier unique (ignoré si non renseigné).

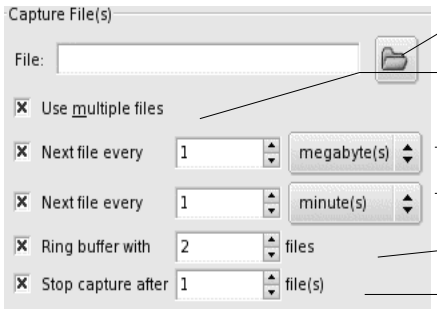
Stockage de la capture dans plusieurs fichiers. Cette option n'est disponible que si l'affichage « temps réel » est désactivé. Les noms de fichiers comportent un numéro incrémenté ainsi que l'heure de début de capture.

Change de fichier quand la condition de taille est atteinte.

Change de fichier quand la condition de temps est atteinte.

Quand le nombre de fichiers est atteint, le plus ancien est supprimé.

Arrêt de la capture après avoir changé n fois de fichier.

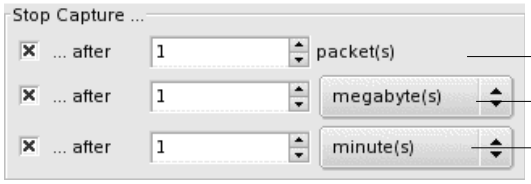


- Section « Stop Capture » : conditions d'arrêt de la capture

Arrêt de la capture après la n^{ième} trame.

Arrêt de la capture après le n^{ième} octet, Koctets, Moctets ou Goctets.

Arrêt de la capture après la n^{ième} seconde, minute, heure ou journée.

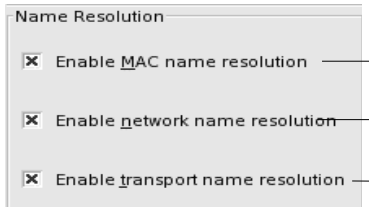


- Section « Name Resolution » : options de résolution de noms

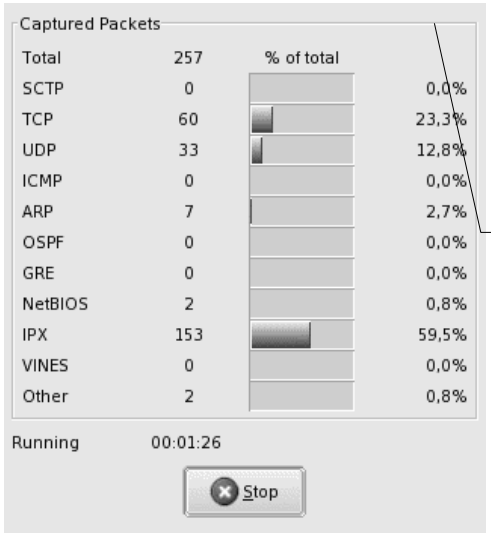
Active la résolution MAC ((@MAC <---> nom de constructeur).

Active la résolution de nom DNS (@IP <---> nom d'hôte DNS).

Active la résolution de nom de services TCP et UDP (N° de port <---> nom de service).

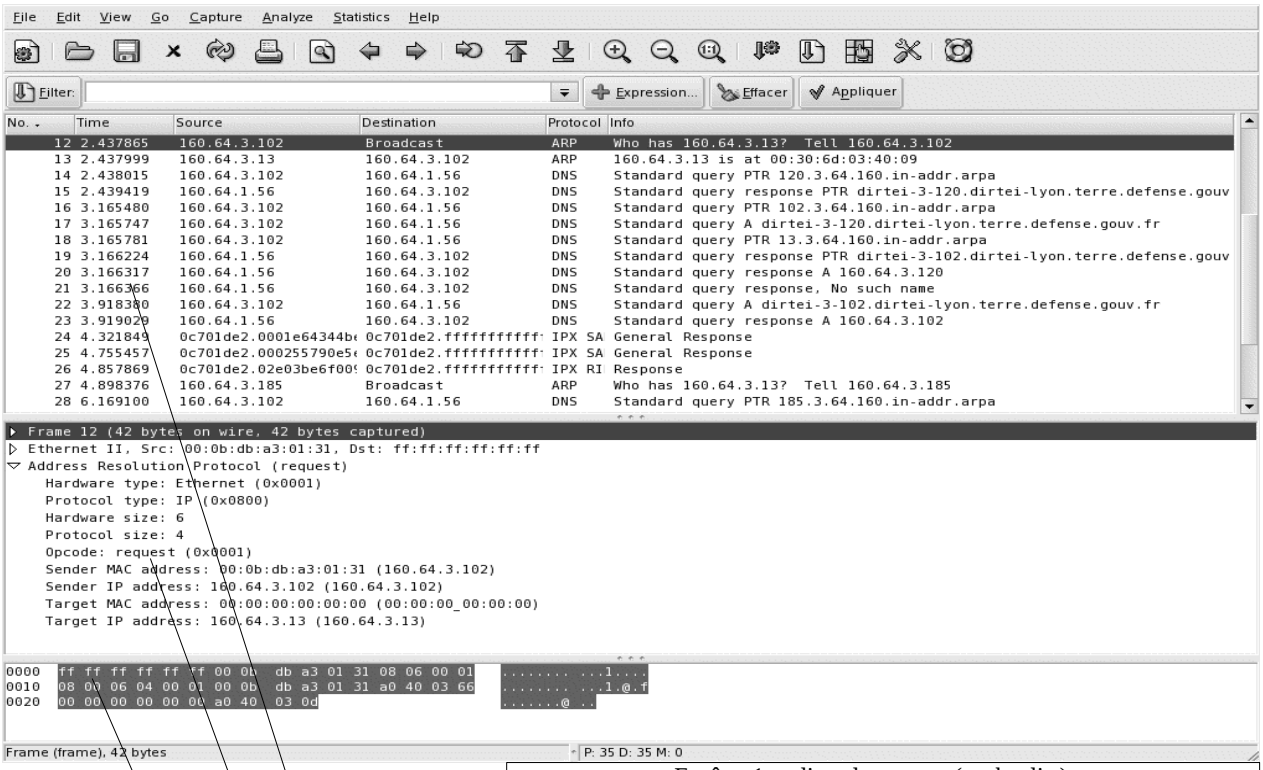


Une fois la capture lancée, la fenêtre de synthèse suivante est présentée :



Cette fenêtre présente une synthèse « temps réel » de la capture. Les informations sont :
- Nombre total de trames reçues
- Distribution des trames par grandes familles :
  - Stream Control Transmission Protocol (OSI 7)
  - Transmission Control Protocol (OSI 4)
  - User Datagram Protocol (OSI 4)
  - Internetworking Contrôle Message Protocol (OSI 3)
  - Adress Resolution Protocol (OSI 2)
  - Open Shortest PathFirst (OSI 3)
  - Generic Routing Encapsulation (OSI 3)
  - Network BIOS = NETBEUI (OSI 3 à 5)
  - Internetworking Packet Exchange (OSI 3)
  - Banyan Virtual Integrated Network Service (OSI 3 à 7)
  - Autres
- Temps de capture

5.2. La fenêtre principale

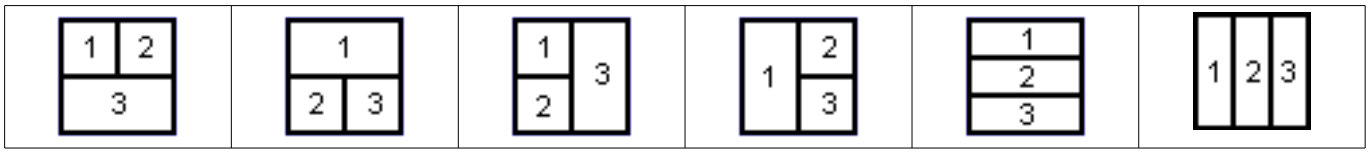


Fenêtre 1 : « liste des trames (packet list) ».
Dans cette fenêtre, une ligne représente une trame reçue. Par défaut les champs disponibles sont :
- le numéro de la trame,
- l'horodatage de la trame (depuis le début de la capture),
- les adresses source et destination de la trame,
- le protocole principal,
- le type de trame dans le protocole principal.

Fenêtre 2 : « détail de la trame (packet details) ».
Cette fenêtre présente le détail de la trame sélectionnée dans la fenêtre « liste des trames ».

Fenêtre 3 : « trame binaire (packet bytes) ».
Représentation en héxadécimale et en caractères ASCII de la trame sélectionnée dans la fenêtre « liste des trames »

- L'organisation de ces fenêtres peut être modifiée par le menu « Edit », « Preferences », « User interface » et « Layout ». Les combinaisons suivantes sont possibles :



- Une cinquantaine de champs différents sont disponibles pour la fenêtre « détail de la trame ». Leur sélection s'effectue par le menu « Edit », « Preference », « User interface » et « Columns ». Il est nécessaire de relance WireShark pour que les modifications soient prises en compte. Exemple d'un affichage personnalisé :

N°	G.D.H	Addr src	Port src	Addr dst	Port dst	Proto	Info
8	2003-12-09 17:02:56.934156	192.168.0.137	68	255.255.255.255	67	BOOTP	Boot Request from 00:10:b5:4a:ca:56
19	2003-12-09 17:02:59.684084	160.97.2.100	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_INTRA<20>
29	2003-12-09 17:03:03.415155	160.97.71.120	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
42	2003-12-09 17:03:07.420318	160.97.70.145	137	160.97.255.255	137	NBNS	Name query NB HTTP:<20>
44	2003-12-09 17:03:08.163169	160.97.70.145	137	160.97.255.255	137	NBNS	Name query NB HTTP:<20>
46	2003-12-09 17:03:08.914254	160.97.70.145	137	160.97.255.255	137	NBNS	Name query NB HTTP:<20>
65	2003-12-09 17:03:14.260167	160.97.81.104	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
70	2003-12-09 17:03:15.760795	160.97.74.116	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
74	2003-12-09 17:03:18.490598	160.97.81.105	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
85	2003-12-09 17:03:22.914875	160.97.70.113	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_INTRA<20>
87	2003-12-09 17:03:23.268612	160.97.74.101	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
100	2003-12-09 17:03:27.243016	160.97.1.202	137	160.97.255.255	137	NBNS	Name query NB HTTP:<20>
1	2003-12-09 17:02:55.214971	160.97.70.130	138	160.97.255.255	138	BROWSER	Host Announcement PCT-4RE-SAFGRH, Worksta
40	2003-12-09 17:03:07.133053	160.97.70.122	138	160.97.255.255	138	BROWSER	Host Announcement PCT-4RE-S1CHAN, Worksta
45	2003-12-09 17:03:08.165402	160.97.2.100	138	160.97.255.255	138	BROWSER	Host Announcement SFI_4RE_DEP, Workstatic
69	2003-12-09 17:03:15.695052	160.97.72.101	138	160.97.255.255	138	BROWSER	Host Announcement BZ05237, Workstation, S
76	2003-12-09 17:03:19.085390	160.97.73.107	138	160.97.255.255	138	BROWSER	Host Announcement PCT_4RE_PHARM, Workstat
52	2003-12-09 17:03:11.522308	160.33.33.33	631	160.33.255.255	631	CUPS	ipp://160.33.33.33/printers/hp (idle)

- En cliquant sur le nom du champ, on affecte le tri de la liste des trames à celui-ci. Ainsi, il est possible de trier les trames selon ses propres critères (ex. : par protocoles, par adresse source, etc.). Dans l'illustration précédente, on voit que les trames ont été triées de manière croissante par « port source ».

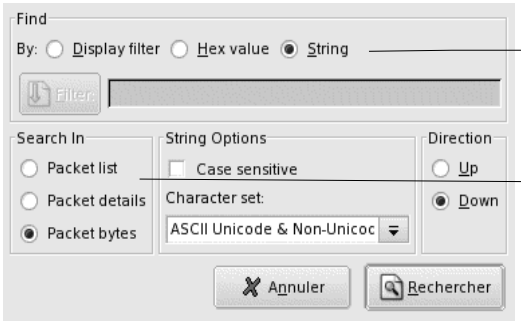
- Les trames considérées comme intéressantes peuvent être marquées (surlignées) afin de pouvoir être plus facilement repérables : menu « Edit » puis « Mark packet » ou au moyen du menu contextuel (clic droit) sur la trame choisie. Pour retirer une marque affectée préalablement à une trame, il suffit de renouveler l'opération. Dans l'illustration suivante, la première trame « ARP » à été marquée. Il est possible d'étendre les possibilités de marquage au moyen de règles de colorisation (cf. §5.4). Le marquage des trames permet aussi à la fonction de sauvegarde de capture (cf. §5.7) d'extraire une partie des éléments d'une capture.

- Il est possible de changer le référentiel de temps afin, par exemple, d'évaluer des temps de réponse : menu « Edit » puis « Time Reference » puis « Set Time Reference » ou au moyen du menu contextuel (clic droit) sur la trame choisie. Pour annuler une référence de temps, il suffit de renouveler l'opération. Dans l'exemple suivant, le référentiel de temps a été positionné sur la première trame « ARP ».

N°	G.D.H	temps	Addr src	Port src	Addr dst	Port dst	Proto	Info
64	2003-12-09 17:03:13.551282	9.209747	160.97.70.145	1026	224.0.1.22	427	SRVLOC	Service Request
65	2003-12-09 17:03:14.260167	9.918632	160.97.81.104	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
66	*REF*	*REF*	00:50:04:09:ac:84		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.81.104? Tell 160.9
67	2003-12-09 17:03:15.554313	1.293972	160.97.70.145	1026	224.0.1.22	427	SRVLOC	Service Request
68	2003-12-09 17:03:15.574301	1.313960	00:09:6b:0c:fc:1c		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.70.65? Tell 160.97
69	2003-12-09 17:03:15.695052	1.434711	160.97.72.101	138	160.97.255.255	138	BROWSER	Host Announcement BZ05237, Worksta
70	2003-12-09 17:03:15.760795	1.500454	160.97.74.116	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>
71	2003-12-09 17:03:15.760963	1.500622	00:50:04:09:ac:84		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.74.116? Tell 160.9
72	2003-12-09 17:03:16.109025	1.848684	00:06:5b:8e:0c:43		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.70.70? Tell 160.97
73	2003-12-09 17:03:17.377910	3.117569	00:10:7b:7f:ee:23		01:00:0c:cc:cc:cc		CDP	Cisco Discovery Protocol
74	2003-12-09 17:03:18.490598	4.230257	160.97.81.105	137	160.97.255.255	137	NBNS	Name query NB SFI_4RE_DEP<20>

5.3. La recherche de trames

La recherche de trames s'effectue par le menu « Edit » puis « Find Packet » ou au moyen des trois icônes suivantes : 



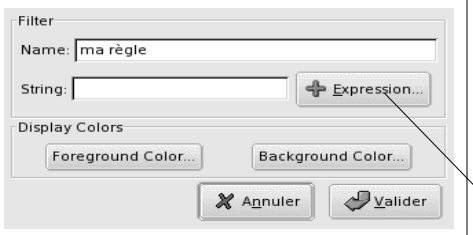
Les trois critères de recherche sont :
- un filtre de type « filtre d'affichage » dont la syntaxe est décrite au §5.4 (ex : tcp.dstport != 80, quakeworld.game, etc.)
- une suite hexadécimale (ex : 12ff45ac2e).
- une suite de caractères ASCII

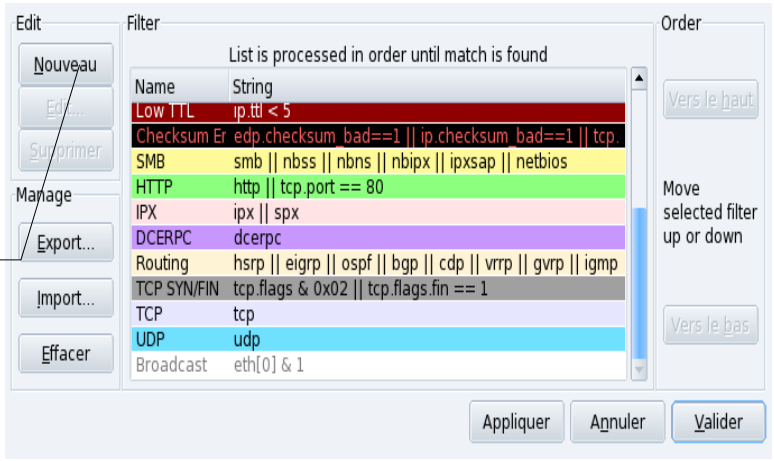
Définit la fenêtre cible de la recherche lorsque le critère de recherche est une chaîne de caractères.

5.4. Les règles de coloration

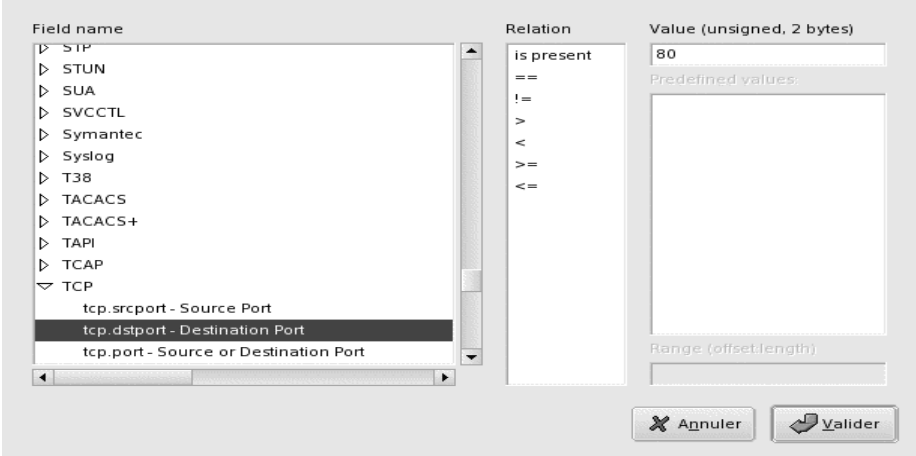
Les règles de coloration permettent, à l'instar du marquage (cf. §5.1), d'améliorer la lisibilité des trames en leur affectant une couleur selon leur type. Cette fonctionnalité est accessible par : menu « View » puis « Coloring Rules ». L'exemple suivant montre quatre règles simples concernant la coloration des trames de type IPX, BOOTP, Netbeui et Appletalk.

La création d'une nouvelle règle de coloration nécessite de définir : le nom de la règle, la couleur d'arrière-plan (Background) ainsi que la couleur des caractères (Foreground). Le critère de coloration est défini dans le champs « String » au moyen d'une chaîne de caractères composée d'expressions logiques. Ce champ peut être renseigné directement (à condition de connaître la syntaxe), ou à l'aide du module « Expression... ».

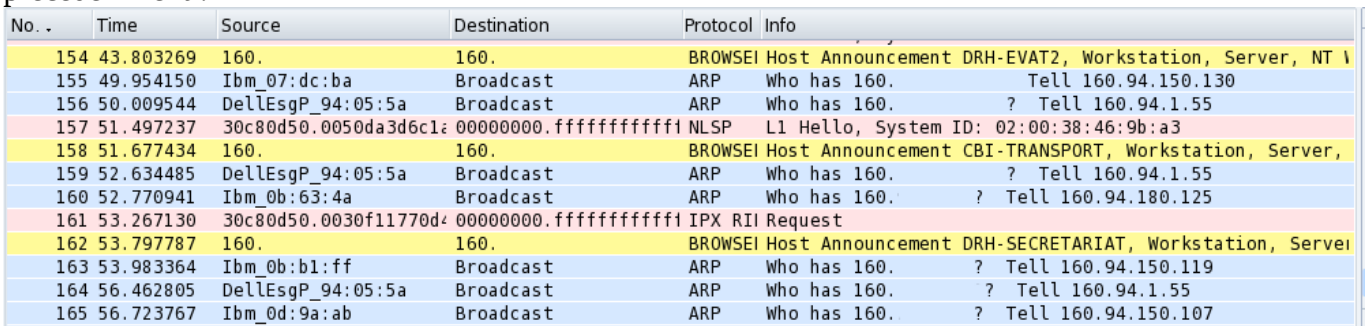




Les expressions logiques peuvent être choisies au moyen de ce module d'aide. Il permet de sélectionner l'attribut d'un protocole et de lui affecter un opérateur et une valeur de test (pour certains attributs, une liste de valeurs prédéfinies est proposée).



L'exemple ci-dessous montre le résultat d'une capture affectée des règles de coloration présentées précédemment :

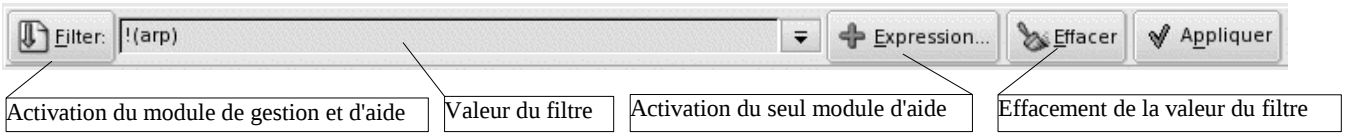


No.	Time	Source	Destination	Protocol	Info
154	43.803269	160.	160.	BROWSEI	Host Announcement DRH-EVAT2, Workstation, Server, NT \
155	49.954150	Ibm_07:dc:ba	Broadcast	ARP	Who has 160. Tell 160.94.150.130
156	50.009544	DellEsgP_94:05:5a	Broadcast	ARP	Who has 160. ? Tell 160.94.1.55
157	51.497237	30c80d50.0050da3d6c1a	00000000.ffffffffffff	NLSP	L1 Hello, System ID: 02:00:38:46:9b:a3
158	51.677434	160.	160.	BROWSEI	Host Announcement CBI-TRANSPORT, Workstation, Server,
159	52.634485	DellEsgP_94:05:5a	Broadcast	ARP	Who has 160. ? Tell 160.94.1.55
160	52.770941	Ibm_0b:63:4a	Broadcast	ARP	Who has 160. ? Tell 160.94.180.125
161	53.267130	30c80d50.0030f11770d4	00000000.ffffffffffff	IPX RII	Request
162	53.797787	160.	160.	BROWSEI	Host Announcement DRH-SECRETARIAT, Workstation, Server
163	53.983364	Ibm_0b:b1:ff	Broadcast	ARP	Who has 160. ? Tell 160.94.150.119
164	56.462805	DellEsgP_94:05:5a	Broadcast	ARP	Who has 160. ? Tell 160.94.1.55
165	56.723767	Ibm_0d:9a:ab	Broadcast	ARP	Who has 160. ? Tell 160.94.150.107

5.5. Les filtres d'affichage

ATTENTION : Ne pas confondre les filtres d'affichage et les filtres de capture (cf. §5.7). Les premiers correspondent à des masques d'affichage que l'on applique sur les trames capturées. Les seconds permettent de filtrer les trames au moment de la capture. Ils n'ont pas le même objectif et leurs syntaxes sont différentes.

L'objectif des filtres d'affichage est de n'afficher que les trames répondant à certains critères. Ces critères sont identiques à ceux utilisés pour créer les règles de coloration (cf. §5.4). Cependant, depuis les dernières versions de WireShark, une méthode encore plus intuitive a été implémentée afin de faciliter le travail de l'analyste. Ainsi, les trois méthodes suivantes permettent de renseigner un filtre d'affichage dans la barre dédiée à cette fonction :



Activation du module de gestion et d'aide

Valeur du filtre

Activation du seul module d'aide

Effacement de la valeur du filtre

NB : tant que le champ de valeur contient des données, le filtre reste actif (que ce soit pour une nouvelle capture ou pendant une analyse).

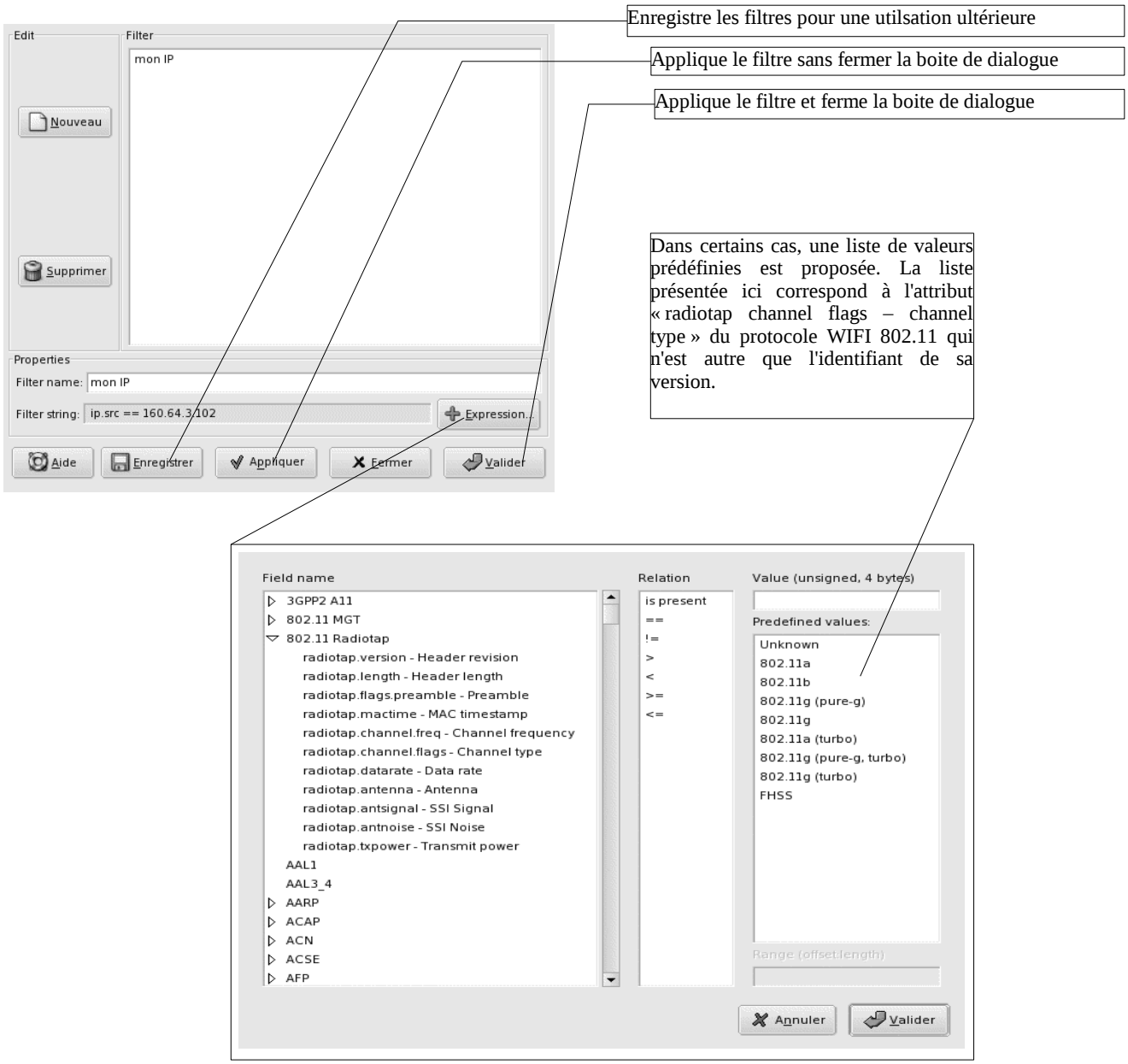
a. la connaissance syntaxique

Avec un peu d'expérience, il est possible de renseigner directement le champ « valeur » de la barre de filtres en suivant la syntaxe adéquate. Dans les deux copies d'écran précédentes, WireShark filtre toutes les trames de type « ARP » (l'opérateur « ! » correspond au « non » logique). Ainsi, dans la copie d'écran du §5.4, les trames de type « ARP » ont été masquées par ce filtre. Pour les faire réapparaître, il suffit de blanchir le champ « valeur » à l'aide du bouton « Effacer ». Pour information, les exemples suivants donnent un aperçu de quelques éléments de syntaxe élémentaires :

- eth.addr==08.00.08.15.ca.fe (ne garde que l'@ MAC : 08.00.08.15.ca.fe) ;
- ip.addr==192.168.0.10 (ne conserve que l'@ IP : 192.168.0.10) ;
- tcp.port==80 (ne garde que les trames dont le port source ou le port destination est 80) ;
- ip.addr==192.168.0.10 && tcp.srcport==80 (combinaison de deux conditions précédentes).

b. le module d'aide

Le module d'aide est identique à celui utilisé pour créer des règles de coloration (cf.§5.4). Il permet de créer rapidement des filtres complexes et de pouvoir les sauvegarder afin de les réutiliser lors de captures ultérieures.



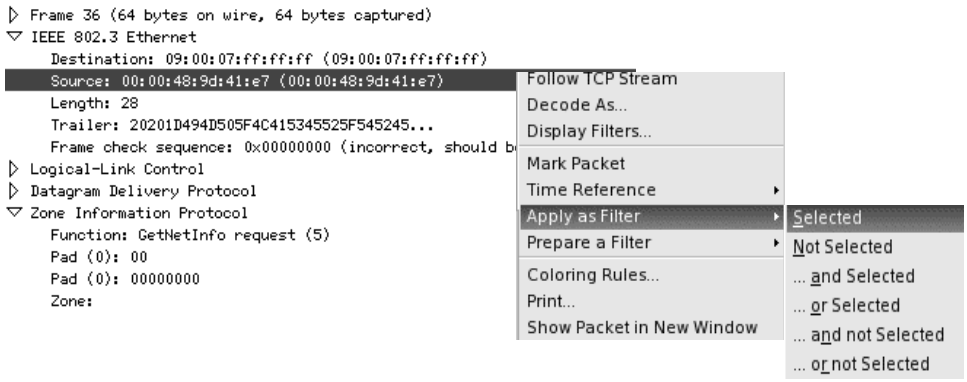
c. le menu contextuel

L'utilisation du menu contextuel (clic droit) de la fenêtre de détail des trames permet de façonner à peu près n'importe quel filtre de manière très simple. Il suffit de lancer ce menu sur le champ de la trame qui doit servir de condition au filtre. L'utilisation d'un exemple permet de mieux comprendre l'intérêt de cette fonctionnalité.

La capture suivante montre la présence des trames « ZIP » (pour « Zone Information Protocol » membre de la famille « Appletalk »). Ce protocole n'étant pas a priori utile sur le réseau local considéré, il est nécessaire de repérer les systèmes ayant émis ce protocole afin de corriger éventuellement leur configuration.

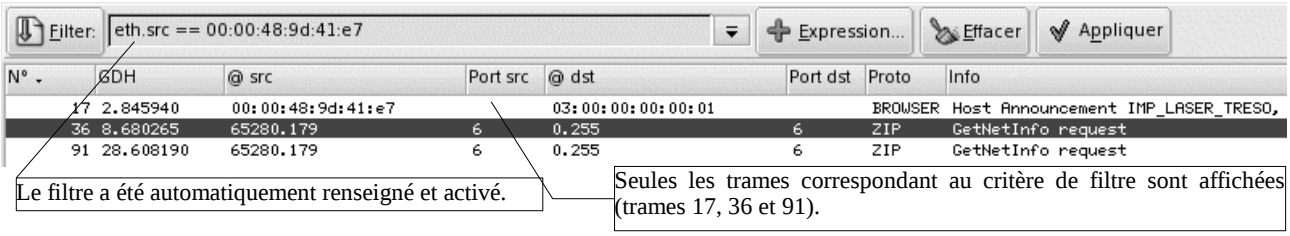
GDH	@ src	Port src	@ dst	Port dst	Proto	Info
32 8.371873	00:60:b0:a4:82:97		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.91.60? Tell 160.97.1.43
33 8.400904	00:60:b0:a4:82:97		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.73.60? Tell 160.97.1.43
34 8.429680	00:60:b0:a4:82:97		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.1.60? Tell 160.97.1.43
35 8.675946	65280.31	6	0.255	6	ZIP	GetNetInfo request
36 8.680265	65280.179	6	0.255	6	ZIP	GetNetInfo request
37 8.697826	65280.151	6	0.255	6	ZIP	GetNetInfo request
38 8.753824	00000000.0008c71152c0	0x0452	00000000.ffffffffffff	0x0452	IPX SAP	General Response
39 9.126564	00:30:84:1b:97:19		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.1.5? Tell 160.97.70.6
40 11.918082	160.97.70.122	138	160.97.255.255	138	BROWSER	Host Announcement PCT-4RE-S1CHAN, Workstatio
41 12.171082	00:10:b5:98:63:88		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.1.60? Tell 160.97.82.106
42 12.205347	160.97.70.145	137	160.97.255.255	137	NBNS	Name query NB HTTP:<20>
43 12.543526	00:10:b5:93:48:c1		ff:ff:ff:ff:ff:ff		ARP	Who has 160.97.1.41? Tell 160.97.1.202

En analysant le contenu des trames « ZIP » dans la fenêtre de détail (capture ci-dessous) on s'aperçoit que hormis l'adresse « MAC », aucune information ne permet d'identifier d'emblée le système ayant émis ces trames (aucune adresse IP, IPX, etc.).



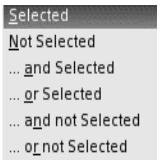
L'idée est donc de chercher parmi toutes les trames capturées celles émises par le même système (même @MAC) afin de rechercher dans celles-ci des informations complémentaires. Il faut donc mettre en place un filtre d'affichage afin de ne voir que les trames ayant la même « @MAC source » que la trame incriminée.

Pour réaliser cette action facilement, il suffit d'activer le menu contextuel (clic droit) au niveau du champ servant de critère de filtre (l'@MAC source dans notre cas : cf. capture d'écran ci-dessus), puis « Apply as filter » et « Selected ». Le résultat de ce filtrage et le suivant :



Le résultat du filtrage nous permet d'identifier une trame NetBEUI émise par la même @MAC. Les informations contenues dans cette trame nous permettent d'identifier une imprimante laser particulière (IMP_LASER_TRESO).

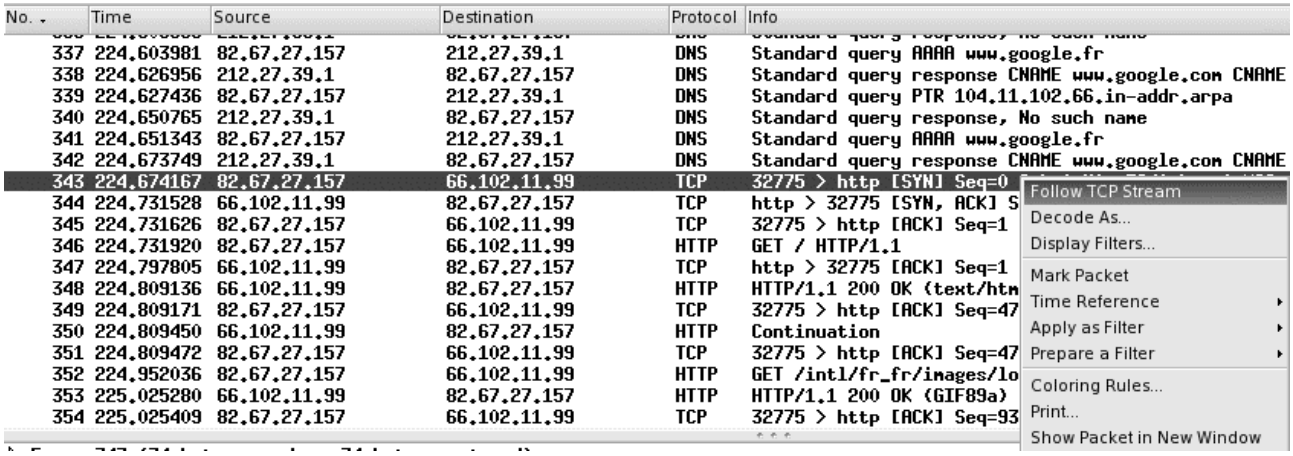
Cette méthode d'élaboration des filtres est très puissante. Elle permet, entre autre, d'enrichir un filtre existant en lui affectant un opérateur logique et une autre condition (ex : je veux filtrer les trames dont l'adresse MAC source est xxxxxxx, **et** dont le port destination est 43). Pour cela, il suffit de ré-ouvrir le menu contextuel (click droit) et d'ajouter le nouveau critère de filtre affecté d'un opérateur logique (non, et, ou, non et, non ou).



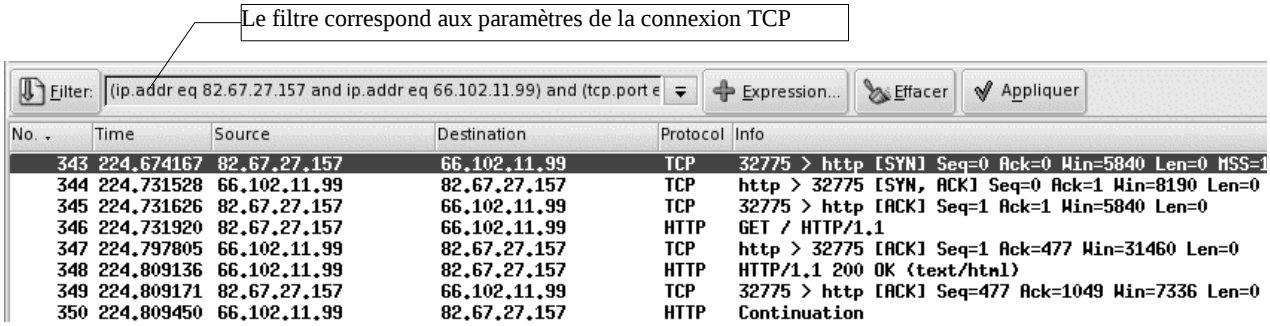
5.6. Le suivi de session

Le suivi de session est une fonctionnalité intéressante qui permet de filtrer au sein d'une capture les échanges de trames entre deux stations identifiées et d'afficher les données transférées sous différents formats (texte ASCII, hexadécimal, etc.). Il est évident que seuls les échanges de type connectés (s'appuyant sur TCP) peuvent être suivis. Cette fonctionnalité n'est en fait qu'une utilisation poussée du filtre d'affichage.

Exemple : l'illustration suivante montre une demande de connexion de type WEB (HTTP) entre un navigateur (client) et un serveur WEB (google). L'activation du menu contextuel sur la trame TCP de demande de connexion vers le serveur WEB permet de choisir l'option « Follow TCP Stream).



« WireShark » va alors affecter les paramètres de cette connexion (@ip source + @IP destination + port TCP source + port TCP destination) à la valeur du filtre d'affichage. Le flux ainsi filtré va apparaître :



Une deuxième fenêtre s'ouvre afin de montrer le contenu des données échangées :

Dans cette deuxième fenêtre, on peut voir l'échange HTTP :

- la requête HTTP du navigateur (GET / HTTP/1.1) ;
- la réponse HTTP du serveur WEB (HTTP/1.1 200 OK) ;
- la page WEB (le serveur « google » envoie ses pages WEB de manière compressée comme il le spécifie dans sa réponse HTTP (Content-Encoding: gzip).

5.7. Les filtres de capture (BPF)

ATTENTION : Ne pas confondre les filtres d'affichage (cf. §5.5) et les filtres de capture. Les premiers correspondent à des masques d'affichage que l'on applique sur les trames capturées. Les seconds permettent de filtrer les trames au moment de la capture. Ils n'ont pas le même objectif et leurs syntaxes sont différentes.

Les filtres d'entrée BPF (Berkeley Packet Filter) (ou filtres de capture) sont utilisés par « WireShark » et de nombreuses applications (comme « Ntop », « Snort », « tcpdump », etc.). Ils servent à sélectionner les paquets selon des critères précis (hôte, protocole, port, etc.). En fait, toutes les applications réseau développées à partir de la bibliothèque « libpcap » (unix/linux) et « winpcap » (win32) peuvent potentiellement utiliser ces filtres.

En comparaison avec les filtres d'affichage vus précédemment, les filtres de captures sont beaucoup plus pauvres (ils ne permettent pas d'atteindre aussi finement tous les attributs des trames réseau).

Le standard BPF est le suivant :

- une expression consiste en une ou plusieurs primitives ;
- une primitive est un identifiant « id » (nom ou nombre) précédé par un ou plusieurs critères ;
- il existe trois sortes de critères différents :
  - « type » : ce critère indique à quoi correspond le nom ou nombre. Les types possibles sont host (hôte), net (réseau) et port. Si le critère « type » n'est pas spécifié, « host » sera utilisé par défaut.
    Exemples : « *host mon_nom_d'hôte* », « *net 192.168* », « *port 20* ».
  - « direction » : ce critère permet de spécifier une direction (depuis ou vers un id). Les directions sont « src », « dst », « src or dst » et « src and dst ». Si le critère « direction » n'est pas spécifié, « src or dst » sera utilisé par défaut.
    Exemples : « *src test.labo.org* », « *dst net 192.168* », « *src or dst port ftp-data* », « *dst port 80* ».
  - « protocole » : ce critère permet de spécifier un protocole particulier. Les protocoles possibles sont ether, fddi , ip, arp, rarp, decnet, lat, moprc, mopdl, tcp et udp. Si le critère « protocole » n'est pas spécifié, tous les protocoles cohérents avec le « type » seront pris en compte.
    Exemples : « *ether src test.labo.org* », « *arp net 192.168* », « *tcp port 21* ».
    Exemple d'équivalence : « *src test.labo.net* » <=> « *(ip or arp or rarp) src test.labo.org* »
    « *net 192.168* » <=> « *(ip or arp or rarp) net* »
    « *port 53* » <=> « *(tcp or udp) port 53* »

Il existe quelques primitives logiques ou mots clés spéciaux : « gateway », « broadcast », « less », « greater », « and », « or » et « not »
Exemple : « *host test.labo.org and not port ftp and not port ftp-data* »

Suivant les cas, certains critères n'ont pas besoin d'être répétés.
Exemple : « *tcp dst port ftp or ftp-data or domain* » <=> « *tcp dst port ftp or dst port ftp-data or tcp dst port domain* »

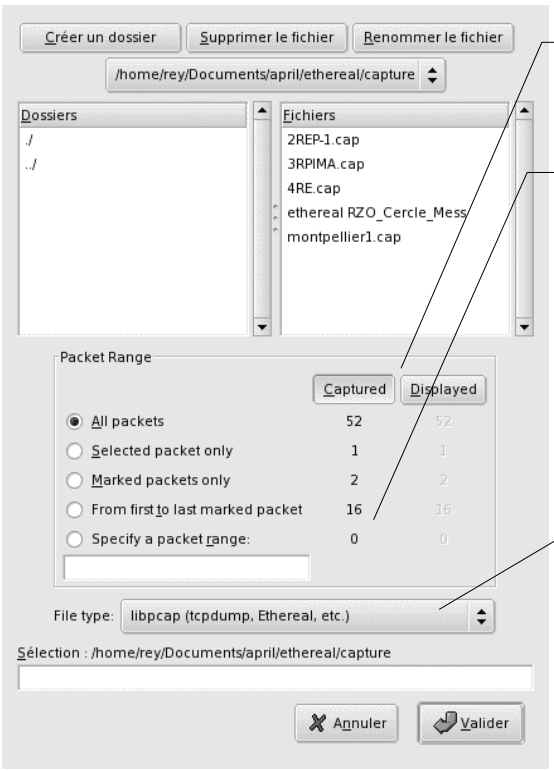
Exemple de primitives

- *dst host <hôte>* : vrai si le champ destinataire du paquet IP contient <hôte>, lequel doit être un nom ou une adresse.
- *src host <hôte>* : vrai si le champ émetteur du paquet IP contient <hôte>.
- *host <hôte>* : vrai si le champ émetteur ou destinataire du paquet IP contient <hôte>.

- *ether dst <eth_hôte>* : vrai si l'adresse ethernet destination est <eth_hôte>. <eth_hôte> doit être une entrée dans le fichier « /etc/ethers » ou une adresse ethernet numérique.
- *ether src <eth_hôte>* : vrai si l'adresse ethernet source est <eth-hôte>.
- *ether host <eth_hôte>* : vrai si l'adresse ethernet source ou destination est <eth_hôte>.
- *gateway <hôte>* : vrai si le paquet utilise <hôte> comme passerelle. Traduction : l'adresse ethernet source ou destination est <hôte> alors que ni l'adresse IP source, ni l'adresse IP destination n'est celle d'<hôte>. <hôte> doit être un nom et doit être dans les fichiers « /etc/hosts » et « /etc/ethers ». Cette primitive est équivalente à la suivante : « *ether host <eth_hôte> and not host <hôte>* ».
- *dst net <réseau>* : vrai si l'adresse IP destination du paquet appartient au réseau <réseau> (nom ou numérique).
- *src net <réseau>* : vrai si l'adresse IP source du paquet appartient au réseau <réseau>.
- *net <réseau>* : vrai si l'adresse IP source ou destination du paquet appartient au réseau <réseau>.
- *dst port <p>* : vrai si le paquet est un paquet tcp ou udp et qu'il contient comme port destination <p>. <p> peut être un nombre ou un nom utilisé dans « /etc/services ». Si <p> est un nom, le port et le protocole seront contrôlés. Si <p> est un nombre ou un nom ambigu, seul le port sera contrôlé.
- *src port <p>* : vrai si le paquet possède comme port source <p>.
- *port <p>* : vrai si le paquet possède un port source ou destination égal à <p>.
- *tcp src port <p>* : ne retient que les paquets tcp ayant comme port source <p>.
- *less <longueur>* : vrai si le paquet a une longueur inférieure ou égale à <longueur>.
- *greater <longueur>* : vrai si le paquet a une longueur supérieure ou égale à <longueur>.
- *ip proto <protocole>* : vrai si le paquet est un paquet ip de protocole <protocole>. <protocole> peut être un nombre ou un des noms suivants : « icmp », « udp » ou « tcp ». Ces identifiants doivent être précédés de « \ ».
- *ether broadcast* : vrai si le paquet est un paquet ethernet broadcast. Le mot clé « ether » est facultatif.
- *ip broadcast* : vrai si le paquet est un paquet ip broadcast. (utilise le masque de sous réseau local).
- *ether multicast* : vrai si le paquet est un paquet ethernet multicast. Le mot clé « ether » est facultatif.
- *ip multicast* : vrai si le paquet est un paquet ip multicast.
- *ether proto <protocole>* : vrai si le paquet est un paquet ethernet utilisant le protocole <protocole>. <protocole> peut être un nombre ou un nom comme « ip », « arp », « rarp », « decnet », etc. Ces identifiants sont des mots clés et doivent être précédés de « \ ».

5.8. La sauvegarde et le chargement de fichiers de capture

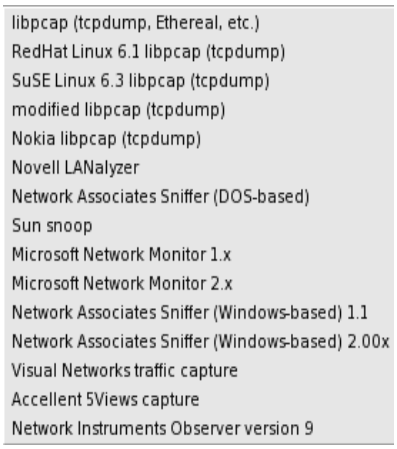
- La sauvegarde d'une capture dans un fichier est effectuée par le menu « File » puis « Save as ... ». Par habitude, les extensions des fichiers de capture sont souvent : « .cap ».



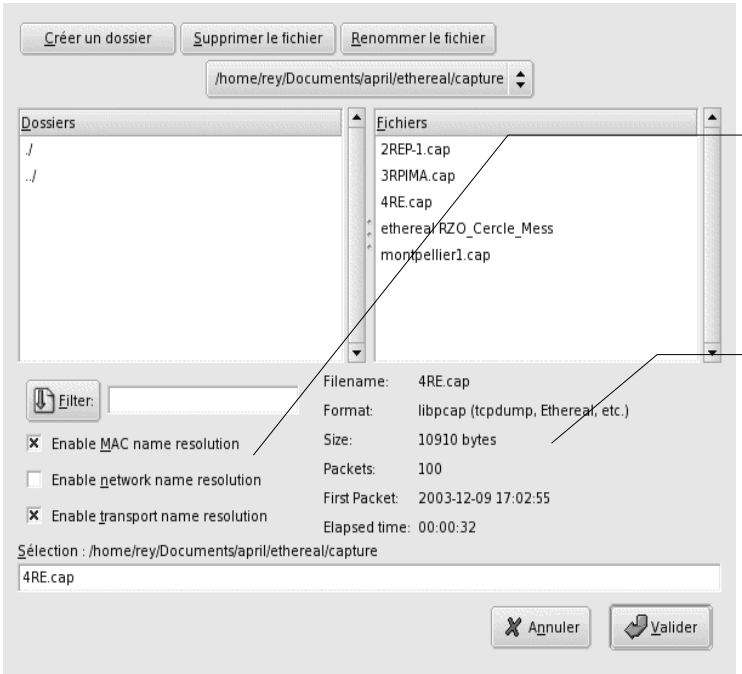
Choix des trames à sauvegarder parmi les trames capturées ou parmi les trames affichées (filtre d'affichage cf. §5.5).

La sauvegarde concerne :
- toutes les trames,
- seulement les trames sélectionnées
- seulement les trames marquées (cf §5.1)
- de la première à la dernière trames marquées (cf §5.1)
- les N° de trame spécifiés (ex : 2-32,112,113,1000-1500).

Choix du format du fichier de sauvegarde parmi la liste (permet à d' autres analyseurs de trames de lire ces fichiers). Le format natif d'Ethereal est : « libpcap ».



- Le chargement d'un fichier de capture est effectué par le menu « File » puis « Open... ».



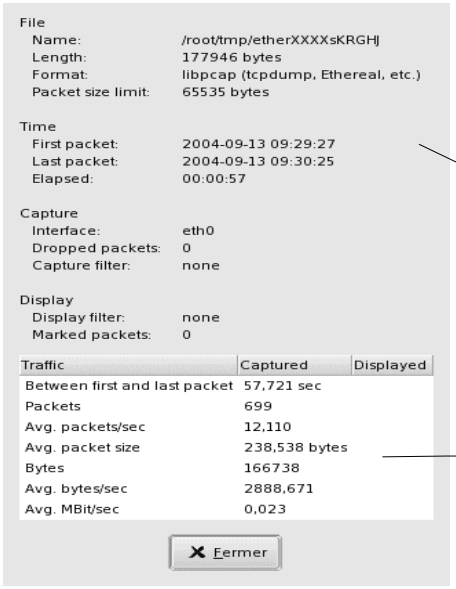
Comme pour une capture « temps réel » (cf. §5.1), les options de chargement permettent de définir un filtre de capture (cf. 5.6) et l'activation des procédures de résolution de nom.

Afin de pouvoir retrouver une capture plus facilement, un résumé de celle-ci est présentée. Il comprend :
- le nom du fichier de capture,
- la taille (en octets)
- le nombre de trames
- la date et l'heure de la première trame
- le temps de capture

5.9. Les outils de statistique

Les outils de statistique sont disponibles à travers le menu « Statistics ». Ces outils fonctionnent sur la base des trames capturées (toutes les trames) ou sur la base des trames affichées (filtre d'affichage activé). Les informations suivantes sont présentées :

- Synthèse des trames affichées (« Summary ») :



Rubrique « fichier » (file) : nom du fichier de capture (dans le cas du chargement d'une archive, sinon nom d'un fichier temporaire), taille du fichier (en octets), format du fichier, limite de la taille des trames capturées.
Rubrique « temps » (time) : GDH de la première et de la dernière trame, temps de capture.
Rubrique « capture » : interface réseau de capture, nombre de trames perdues, valeur du filtre de capture.
Rubrique « affichage » (display) : valeur du filtre d'affichage, nombre de trames marquées.

Analyse du trafic capturé
Temps de capture, nombre de trames, débit moyen (en trames par seconde), taille moyenne des trames, taille cumulée, débit moyen en octets/sec et en Moctets/s.

- Distribution par protocole des trames affichées (« Protocol Hierarchy ») :

Protocol	% Packets	Packets	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s
Frame	100,00%	65	7219	0,007	0	0	0,000
Ethernet	100,00%	65	7219	0,007	0	0	0,000
Internet Protocol	100,00%	65	7219	0,007	0	0	0,000
Transmission Control Protocol	81,54%	53	6013	0,006	33	2168	0,002
Data	27,69%	18	1874	0,002	18	1874	0,002
Hypertext Transfer Protocol	3,08%	2	1971	0,002	1	530	0,001
Line-based text data	1,54%	1	1441	0,001	1	1441	0,001
User Datagram Protocol	18,46%	12	1206	0,001	0	0	0,000
Domain Name Service	18,46%	12	1206	0,001	12	1206	0,001

- Liste des dialogues point à point des trames capturées (« Conversations ») :
  Différents types de dialogues sont disponibles dans les onglets de cet outil (liste par adresses MAC, IP, IPX, FDDI, FC ou par ports UDP/TCP). La création d'un filtre d'affichage via le menu contextuel (click droit) est disponible avec cet outil (cf. §5.5.c).

Ethernet: 1 | Fibre Channel | FDDI | IPv4: 12 | IPX | TCP: 11 | Token Ring | UDP: 1

TCP Conversations

Address A	Port A	Address B	Port B	Packets ˄	Bytes	Packets A->B	Bytes A->B	Packets A<-B	Bytes A<-B
82.67.27.157	34411	217.94.123.221	8888	11	836	6	457	5	379
82.67.27.157	33455	81.152.253.201	8888	6	548	3	342	3	206
82.67.27.157	33998	68.73.231.239	9999	6	408	3	198	3	210
82.67.27.157	34413	66.102.9.104	http	6	2217	4	712	2	1505
82.67.27.157	33584	80.162.0.70	8888	6	614	3	270	3	344
82.67.27.157	33539	81.171.8.64	8886	4	340	2	204	2	136
82.67.27.157	34374	24.2.17.180	8888	4	340	2	204	2	136
82.67.27.157	33570	62.59.166.219	8888	4	308	2	180	2	128
82.67.27.157	34410	80.135.221.233	6134	2	132	1	66	1	66
82.67.27.157	34412	82.50.126.214	8888	2	138	1	74	1	64
82.67.27.157	34408	217.19.50.74	http	2	132	1	66	1	66

L'exemple ci-dessus montre les dialogues par ports TCP d'une station connectée à Internet. Le port destination (Port B) montre des connexions WEB (http) et des connexions P2P du réseau EMULE/NAPSTER (8888, 8886, 9999).

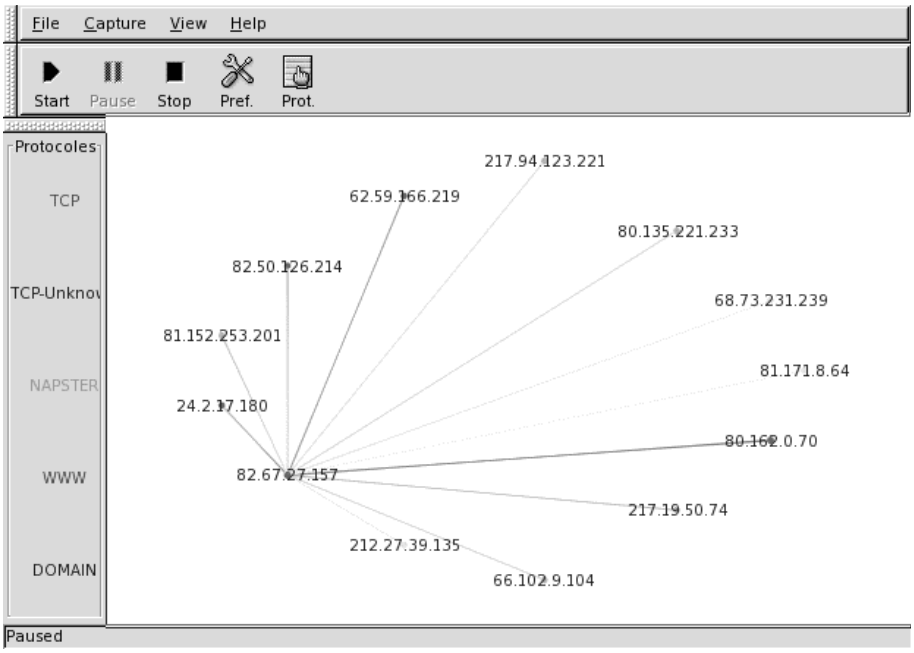
- Liste par points de destination des trames capturées (« Endpoints ») :
  Cette liste est une autre représentation de la liste précédente.
- Graphe d'entrée/Sortie (« IO Graph ») :
  Ce graphe est une représentation des trames capturées par protocole et en fonction du temps. Il fonctionne sur le principe des filtres d'affichage (cf.§5.5).

L'exemple ci-dessus, montre le « poids » que représentent les trames « IPX » (graph 2) par rapport à l'ensemble des trames capturées (graph 1).

6. Utilisation d'outils tiers

6.1. « Etherape »

« Etherape » est un analyseur graphique de dialogues réseau point à point. Il peut être considéré comme la représentation graphique de l'outil statistique « Conversations » de « WireShark » présenté au §5.8. Il est capable de lire un fichier au format WireShark (libpcap) et de représenter en « temps réel » les différents dialogues capturés :



6.2. « tcpdump »

« tcpdump » est un outil à interface texte simple permettant de capturer le trafic réseau. Il permet, soit

d'afficher ce trafic sous forme de ligne de texte, soit de l'envoyer dans un fichier afin de pouvoir être analysé en mode « hors ligne » par « wireshark » par exemple. Son interface texte lui permet d'être utilisé à distance sur un système ne comportant pas d'interface graphique.

- Utilisé en mode interactif : « tcpdump {-i interface_de_capture} »
- Utilisé pour stocker une capture dans un fichier : « tcpdump {-i interface_de_capture} -s 1500 -w fichier_de_capture » (l'option « -s 1500 » permet de définir la taille des trames capturées. Elle est fixée par défaut à 68 octets).

7. Annexes

7.1. Annexe 1 : Support des interfaces réseau

Le support des interfaces réseau est intimement lié au système d'exploitation qui les gère au moyen des pilotes de périphériques (device drivers). WireShark » s'appuie sur des bibliothèques d'accès aux périphériques dont les possibilités sont naturellement liées à la capacité du système d'exploitation à gérer ces interfaces. Ainsi, en fonction de chaque système, le gestion des cartes réseau est différemment supportée. Le tableau suivant résume la situation à la date de rédaction de ce document (2004) pour quelques systèmes d'exploitation et pour les familles de cartes les plus utilisées.

	802.11 (WiFi)	**ATM**	**Ethernet**	**FDDI**	**Frame Relay**	**Loop-back**	**Série**	**Token Ring**
IBM AIX	Non testé	Non testé	oui	Non testé	Non testé	Non testé	Non testé	oui
Free BSD Open BSD Net BSD	Non testé	Non testé	oui	Non testé	Non testé	oui	Non testé	oui
Mac OS X	oui[2]	non	oui	non	non	oui	oui[3]	Non
SUN Solaris	Non testé	oui	oui	oui	non	non	non	oui
Win 32	oui[2]	Non testé	oui	Non testé	non	non[4]	Partiellement[5]	oui
Linux	oui	oui	oui	oui	oui[6]	oui	oui[3]	oui

2 Seulement pour les trames de données (les trames de contrôle sont ignorées). Et seulement les trames en provenance de ou vers la station qui analyse.
3 Seulement pour les trames de données (les trames de contrôle sont ignorées).
4 Ce type d'interface n'existe pas nativement sous WIN32.
5 Seulement pour la famille WinNT et avec des versions de la bibliothèque « WinPcap » différentes de 3.x.
6 Seulement avec les dernières versions de la bibliothèque « LibPcap ».