

- **ENSEIRB** -
3^{ème} ANNEE OPTION RADIOCOM



**MISE EN ŒUVRE ET ANALYSE DES
PROTOCOLES INTERNET**

Patrice KADIONIK

TABLE DES MATIERES

1. BUT DES TRAVAUX PRATIQUES	4
2. UN PEU D'HISTOIRE	4
3. ARCHITECTURE D'INTERNET	5
3.1. Les protocoles	5
3.2. L'interface de programmation	5
3.3. Les services de base de l'Internet	6
4. LE WWW	6
4.1. Identification des services de base	7
4.2. Identification des nouveaux services	7
4.3. Développement d' une page Web	7
5. PRESENTATION DE L'ANALYSEUR SURVEYOR	8
5.1. Introduction	8
5.2. Principales fonctionnalités	8
5.3. Principe d' utilisation	9
5.4. Getting started	11
5.5. Présentation générale des principales analyses	11
❖ Lancement. Choix de l'interface réseau	11
❖ Mode Monitor	12
❖ Mode Capture	14
Mode Capture & Monitor	19
❖ Mode émission de données	20
6. EX 0 : QUESTIONS DE SYNTHÈSE	25
7. EX 1 : ANALYSE D'UN RESEAU INTERNET	26
8. EX 2 : ANALYSE RESEAU AVEC SURVEYOR	31
9. EX 3 : INTRODUCTION A LA PROGRAMMATION RESEAU	35

PARTIE I

- PRESENTATION GENERALE -

1. BUT DES TRAVAUX PRATIQUES

Le but de ces travaux pratiques est d'étudier la mise en œuvre d'un réseau Ethernet mettant en œuvre les différents protocoles de l'Internet.

L'Internet est un domaine captivant du point de vue matériel mais avant tout logiciel. Le boom du "World Wide Web" en est la preuve actuellement. C'est un domaine qui est porteur d'emplois (votre principale préoccupation à la fin de l'année ?). Internet standard de fait par rapport à d'autres protocoles de l'OSI apparaît pour beaucoup comme le moteur de la 3^{ème} révolution industrielle...

Vous allez être placé(e) dans la peau d'un architecte réseau devant analyser un réseau Internet existant et en trouver les principales caractéristiques. Dans un second temps, vous allez utiliser un analyseur de protocoles Internet pour réaliser un certain nombre d'opérations de maintenance et de surveillance...

Votre base de travail se compose d'un ensemble de documents existants se trouvant sur Internet que vous trouverez en annexe. Dans le monde Internet, les normes s'appellent des "RFC" ("Request For Comments") que l'on peut récupérer à l'adresse <http://www.rfc-editor.org/>.

Les documents donnés en annexe proviennent de l'Unité de Réseaux du CNRS à l'adresse <http://www.urec.fr/cours> (très bons documents).

Vous travaillerez donc à partir de ces documents pour les TP. Les chapitres suivants n'en sont qu'un résumé...

2. UN PEU D'HISTOIRE

Le développement et le succès d'Internet sont liés au développement d'UNIX.

Ces quelques dates en rappellent les principaux moments clés :

1960 - 1970 : projet ARPANET du DOD (Department Of Defense) aux USA. Avoir un réseau capable de supporter une attaque par un mécanisme de routage au mieux. C'est un réseau à commutation de paquets basé sur le protocole IP (Internet Protocol).

1965 : laboratoires Bell, General Electric, MIT : projet Multics. Accès simultanés par plusieurs utilisateurs à un ordinateur de grande puissance de calcul et de grande capacité de stockage pour l'époque.
Système multiutilisateur et multitâche.

1969 : première version de Multics sur un GE645.

1969 : Thomson et Ritchie améliorent Multics pour avoir un meilleur système de fichiers.
Naissance d'UNIX.

1971 : portage d'UNIX sur un PDP11. Naissance du langage C.

1973 : écriture d'UNIX en C pour portage aisé sur d'autres machines. Fourniture des sources d'UNIX aux universités (Berkeley). D'où le schisme entre UNIX ATT et UNIX BSD (Berkeley).

1977 : UNIX System III (ATT).

1981 - 1981 : intégration des protocoles Internet à l'UNIX BSD 4.2 pour l'interconnexion des machines UNIX. Succès.

1983 : UNIX System V., UNIX BSD 4.3.

1989 : projet WWW (World Wide Web) du CERN. Unification de l'ensemble des services de l'Internet : mail, ftp, telnet plus images et son...

1993 : réunification des 2 UNIX : UNIX System V.4 ou BSD 4.4.

3. ARCHITECTURE D'INTERNET

Internet est un standard de fait qui ne respecte pas le modèle d'interconnexion des systèmes ouverts ou modèle OSI ("Open System Interconnexion").

3.1. Les protocoles

Protocole IP (Internet Protocol) : permet l'échange de datagrammes en mode non connecté et non assuré. On ne garantit pas l'arrivée à bon port des datagrammes.

Protocole UDP (User Datagram Protocol) : permet l'envoi par une application de messages en mode non connecté et non assuré.

Protocole TCP (Transport Control Protocol) : permet l'envoi par une application de messages en mode connecté et assuré. Pas de perte ni gain de données. C'est un mode fiable.

3.2. L'interface de programmation

Les sockets (« prise ») sont l'interface de programmation qui permettent de développer des applications utilisant les services de l'Internet.

Concept d'applications client/serveur.

3.3. Les services de base de l'Internet

C'est un ensemble de services de télécommunications de base utilisant Internet en standard sur chaque système UNIX.

Transfert de fichiers ftp (File Transfer Protocol).

Courrier électronique SMTP (Simple Mail Transfer Protocol).

Terminal virtuel telnet (Terminal NETwork protocol).

Commandes « remote » ou r... : rsh, rcp, rlogin, rwho...

Partage de fichiers distants NFS (Network File System).

Appel de procédures distantes RPC (Remote Procedure Call).

Autres : XDR (eXternal Data Representation), talk, finger, Xwindow (multifenêtrage pour écran bitmap).

Le problème crucial pour un non informaticien est l'utilisation complexe de ces outils qui ne possédaient pas à l'origine une interface graphique améliorant le confort d'utilisation.

Projet WWW du CERN : unifier les services de l'Internet pour les rendre plus facile d'utilisation à la communauté de physiciens du CERN avec en plus la possibilité de véhiculer d'autres types d'informations.

4. LE WWW

Le projet World Wide Web (interconnexion de l'ensemble des services client/serveur de l'Internet) se propose d'unifier les services de base de l'Internet et d'en ajouter d'autres.

Ajout de services hypermédia.

Tous les services de l'Internet sont repérés par un lien unique et uniforme : URL (Uniform Resource Locator).

C'est l'équivalent d'un lien hypertexte.

Syntaxe :

type_service://nom_absolu_machine :numéro_de_port/chemin_de_la_ressource

exemple : <http://www.ixl.u-bordeaux.fr> -> accès à un document hypertexte de présentation de l'IXL.

Par défaut, le numéro de port est 80 pour le service HTTP s'il est omis. Le protocole d'échange de documents hypermédia est HTTP (Hyper Text Transport Protocol)

Un document hypermédia est écrit dans un langage particulier HTML ressemblant à Latex (Hyper Text Markup Language).

Dans le cas de l'exemple, le document hypermédia accédé est un document de format HTML (fichier welcome.html).

4.1. Identification des services de base

ftp : ftp://

exemple : ftp://ftp.urec.fr

mail : mailto:adresse_utilisateur

exemple : mailto:kadionik@ixl.u-bordeaux.fr

telnet : telnet://

exemple : telnet://grace@frbdx13.cribx1.u-bordeaux.fr

exemple : telnet://aramis.ixl.u-bordeaux.fr

4.2. Identification des nouveaux services

Document hypermédia : http://

exemple : http://www.ixl.u-bordeaux.fr/gif/pointrouge.gif

Accès base de données WAIS (Wide Area Information Server) : wais://

exemple : wais://zenon.inria.fr :210

Outil de recherche et de récupération de documents GOPHER : gopher://

exemple : gopher://gopher.jussieu.fr/1/infoservers/france

Groupes d'intérêt ou « newsgroup » : news://

exemple : <news://news.enseirb.fr>

4.3. Développement d'une page Web

Cela consiste à développer un document hypermédia HTML en utilisant le langage HTML où il est possible d'intégrer tout type d'informations : texte, image, son...

On peut écrire sa page après maîtrise du langage HTML : mais difficile sauf pour les spécialistes de Latex.

On peut utiliser un programme d'aide plus ou moins compétent, le must étant d'avoir un aide WYSIWYG (« *What You See Is What You Get* »). Cet aide est intégré dans la dernière version de *Netscape Communicator*, icône *composer* en bas à droite. On pourra utiliser le produit *FrontPage* de Microsoft (assez bien fait).

Il est possible en plus d'avoir des séquences interactives écrites avec le langage Java qui un langage orienté objet ressemblant fortement au langage C dans sa syntaxe.

5. PRESENTATION DE L'ANALYSEUR SURVEYOR

5.1. Introduction

L' analyseur de protocoles SURVEYOR de la société SHOMITI est une application qui permet d' analyser et "monitorer" les réseaux Ethernet 10/100 Mb/s. C' est un produit complet qui permet :

- L' analyse expertmulti couches.
- De récupérer les statistiques du réseau Ethernet.
- L' analyse et le décodage des 7 niveaux du modèle OSI.
- De gérer des alarmes sur des paramètres réseau.

SURVEYOR ne permet en standard que d' analyser un segment local (sous-réseau) mais il est possible en option de monitorer un segment distant.

5.2. Principales fonctionnalités

SURVEYOR permet principalement d' utiliser les fonctions suivantes :

- *Capture* : capture des données véhiculées par le réseau dans un buffer en mémoire du PC ou dans le buffer d' une carte spécialisée CMM(*Century Media Module*). Il est possible de créer des filtres de capture pour récupérer seulement les données intéressantes.
- *Capture View* : permet de visualiser les données capturées sous forme graphique (camemberts, colonnes (*Charts*)) ou sous forme de tableaux.
- *Save* : permet de sauvegarder une capture dans un fichier.
- *Log* : compteurs des statistiques Ethernet.
- *Monitor* : vue en temps réel des données circulant sur le segment analysé. Il existe différents formats de présentation.
- *Setting Alarms* : mise en place d' alarmes sur des événements particuliers du segment réseau. Différentes actions sont possibles : messages à l' écran, envoi d' un mail...

Des fonctionnalités en option (ou *Plug-in*) sont possibles :

- *Remote Plug-in* : pour l' analyse des données d' un segment distant.
- *Packet Blaster Plug-in* : permet d' envoyer des données sur le réseau. Ce *Plug-in* est fonctionnel durant ce TP
- *Expert Plug-in* : permet une analyse en mode expert du réseau : analyse des compteurs et diagnostic des problèmes rencontrés. Ce module fournit des solutions envisageables en cas de problèmes détectés.

SURVEYOR est considéré comme l'un des meilleurs produits du marché pour l'analyse des protocoles sur Ethernet 10/100 Mb/s.

Il permet en outre l'analyse des protocoles de l'Internet, ce qui nous intéresse ici mais permet aussi l'analyse d'autres protocoles (NetBIOS, Xwindows, DECNET...).

5.3. Principe d'utilisation

SURVEYOR est un logiciel sous Windows qui exploite la carte réseau 10/100 Mb/s d'un PC et ses drivers NDIS. C'est le mode d'utilisation le plus commun. Mais il peut exploiter aussi une carte d'acquisition temps réel CMM2 (*Century Media Module version 2*) plus performante qu'une carte réseau classique. L'ENSEIRB en possède une.

SURVEYOR supporte au plus 4 cartes réseau dans un PC.

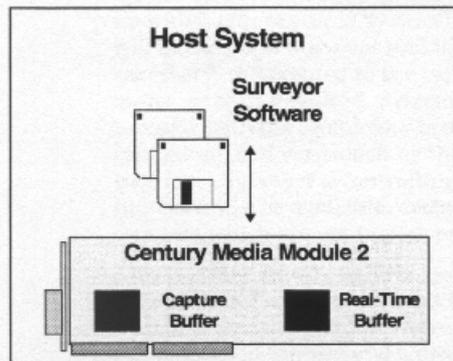
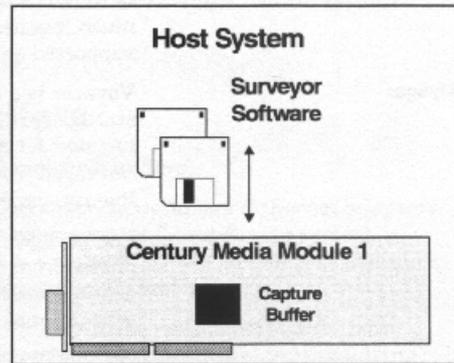
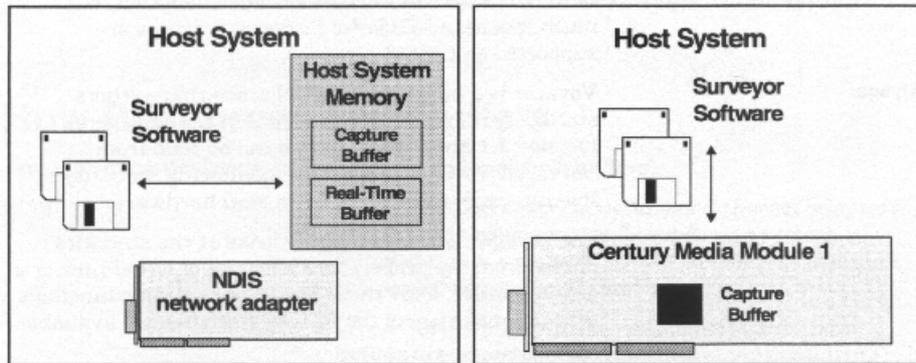
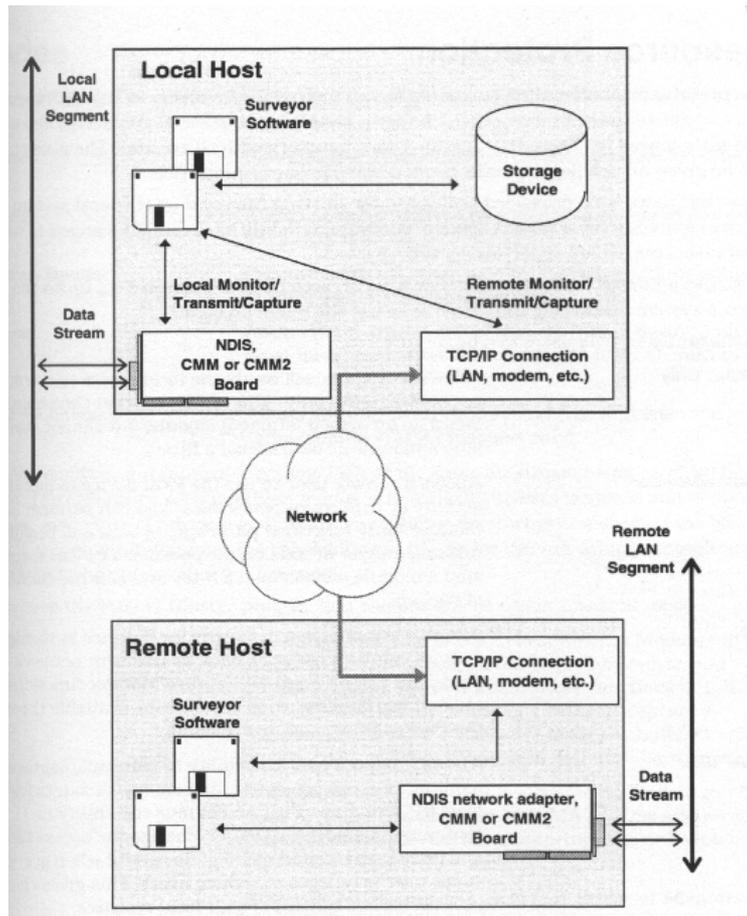
La comparaison des performances globales des 2 types de cartes est donnée ci-après :

	NDIS	CMM2
Taille buffer temps réel	64 Ko	512 Ko
Performance de capture	5- 10 Mb/s si 10 Mb/s	10 Mb/s si 10 Mb/s
Alarmes	Toutes sauf celles non gérées par NDIS	Toutes
Un seul écran full duplex	Non	Oui
<i>Packet slicing</i> (possibilité de capturer qu'une partie d'un paquet depuis son début)	Oui	Oui
Taille buffer de capture	64Ko à 16 Mo	16 Mo
Erreurs trames Ethernet	Non	Oui

On voit ainsi que la carte CMM2 est plus performante mais permet surtout de collecter les statistiques d'erreurs (erreurs de CRC, mauvaises trames...).

Il faut noter que la fonction *packet slicing* est intéressante car elle permet de ne récupérer qu'une partie d'un paquet depuis son début en mode capture ou monitor : 32 premiers octets (couche MAC), 64 premiers octets (couche réseau), 128 premiers octets (couche application) ou tout.

Le principe d'utilisation de SURVEYOR et des cartes réseau est donné sur les figures suivantes.



5.4. Getting started

La prise main du logiciel est aisée. Un *getting started* est donné en annexe. Il faut noter que l'on peut agir de la même manière par les menus ou en cliquant sur les icônes de la barre des boutons. Les menus et boutons sont contextuels. On retrouve souvent les mêmes fonctionnalités dans un contexte différent (impression...).

Se reporter au document *quick start* donné en annexe...

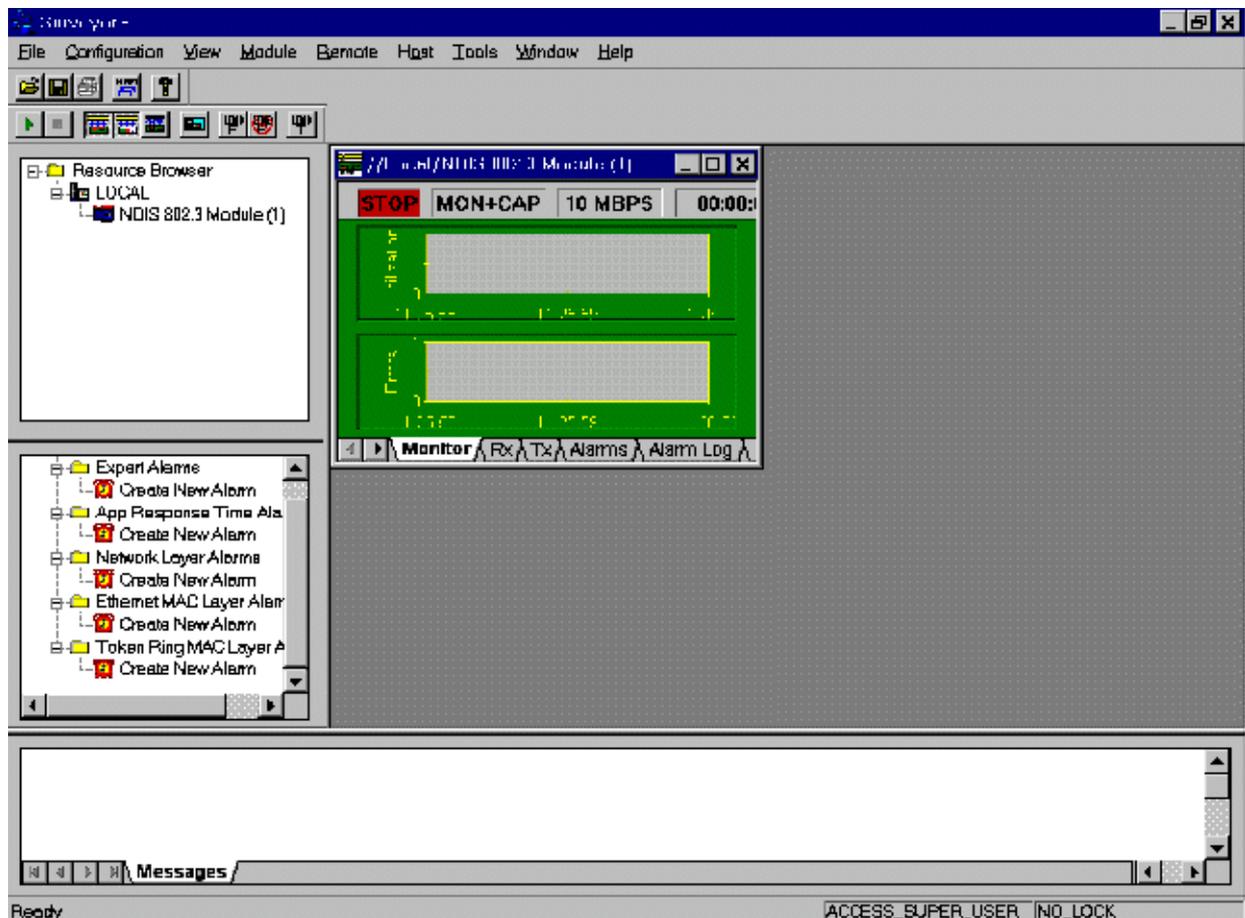
5.5. Présentation générale des principales analyses

Les principales possibilités de SURVEYOR sont balayées. On se reportera au document donné en annexe.

❖ Lancement. Choix de l'interface réseau

Lancer le logiciel SHOMITI SURVEYOR. Si vous avez le PC (pomme1) avec le carte CMM2, choisir la carte CMM2 (module numéro 16 (0x260 d'IO Port)).

L'écran principal apparaît :



Dans la fenêtre *Resource Browser*, choisir sa ressource locale en cliquant dessus (soit NDIS, soit CMM2 pour les chanceux).

On peut avoir un bulle d'informations quand on passe la souris sur un icône. Vous avez accès à l'aide en ligne (touche F1) qui possède un glossaire très bien fait des termes et concepts utilisés par le logiciel.

❖ Mode Monitor

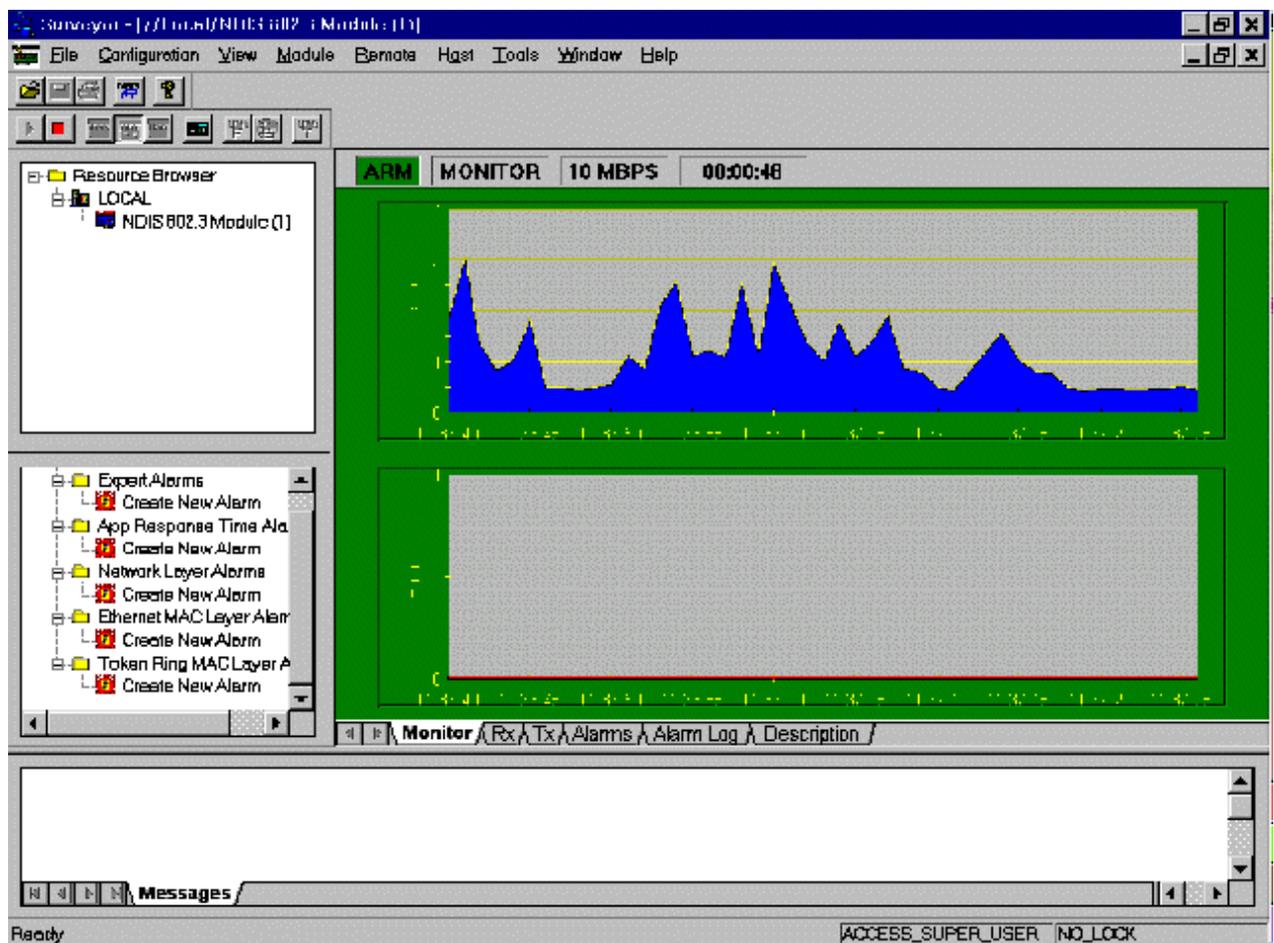
Il suffit de cliquer sur l'icône Monitor



Pour lancer le monitoring, cliquer sur

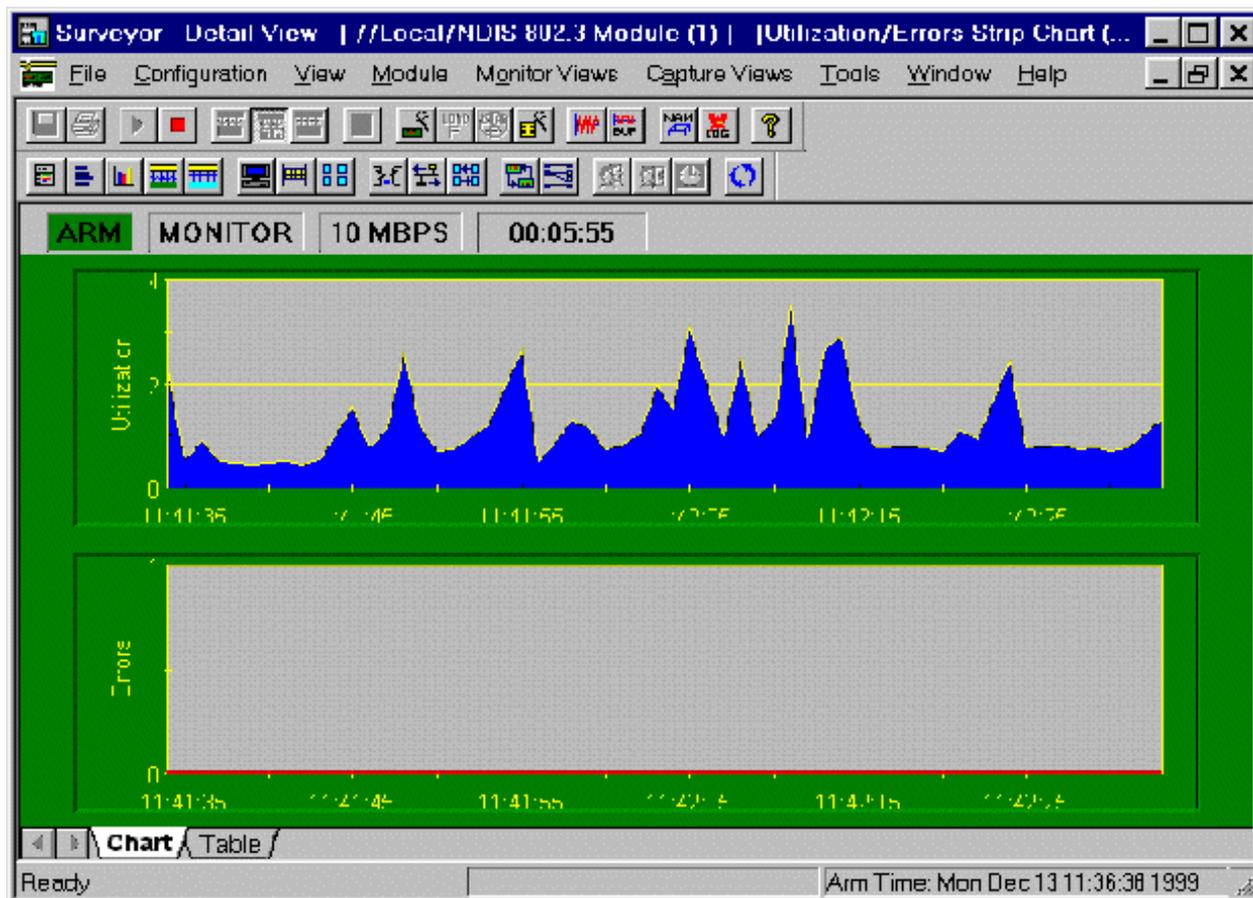


On obtient l'écran suivant :



En cliquant sur les onglets *Rx*, *Tx*, *Alarms*, *Alarm Log*, *Description*, on peut récupérer d'autres informations.

En double cliquant sur cette fenêtre, on ouvre une fenêtre pour le mode détaillé (valable pour les autres modes capture...). On a ainsi l'écran suivant :



Cliquer sur l'onglet *Table*. On remarquera que les compteurs d'erreurs ne s'incrémentent pas pour les PC qui utilisent une carte réseau NDIS (non supporté). Les menus et boutons sont contextuels.

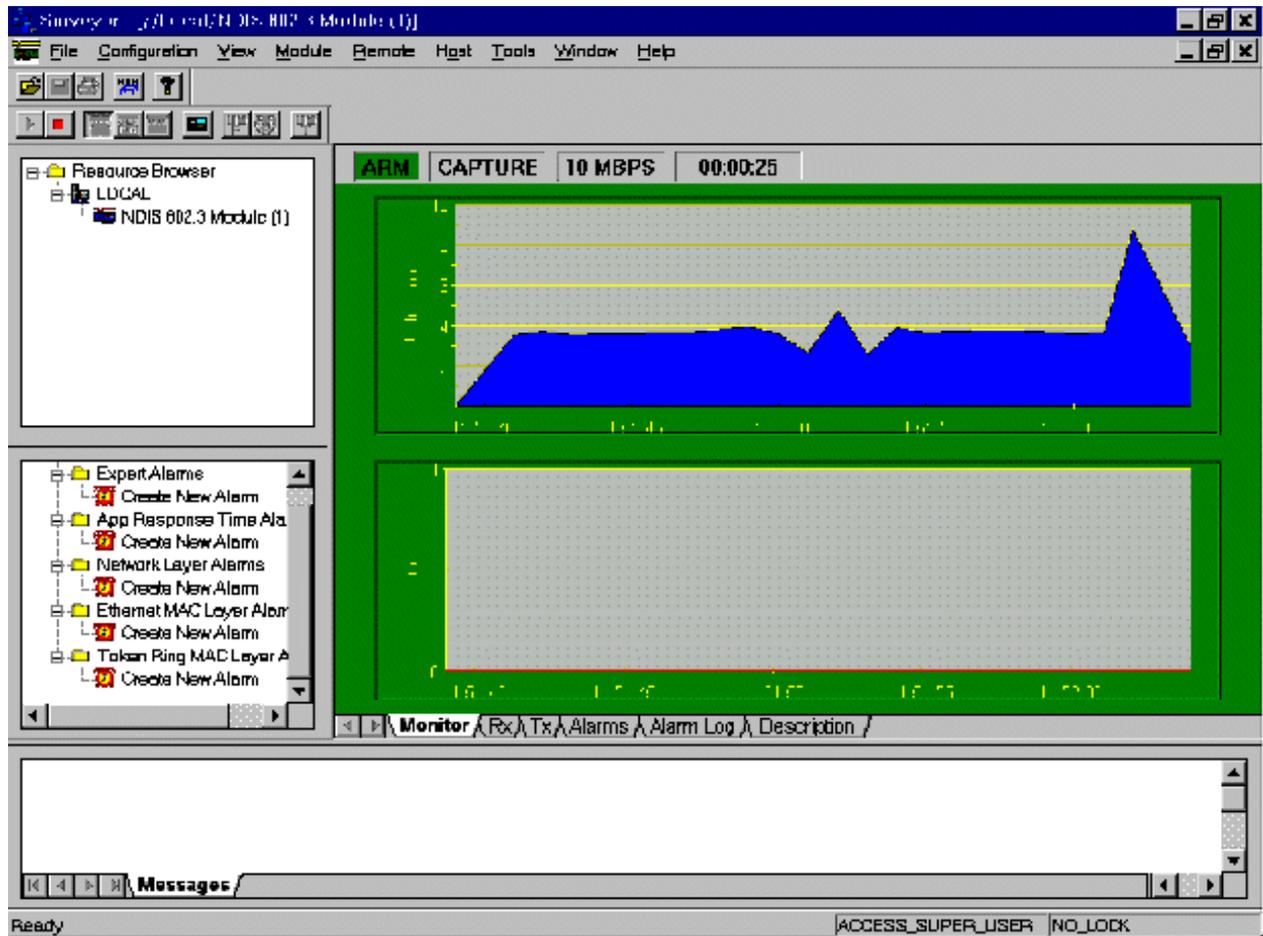
❖ **Mode Capture**

Arrêter l'acquisition en cours en cliquant sur .

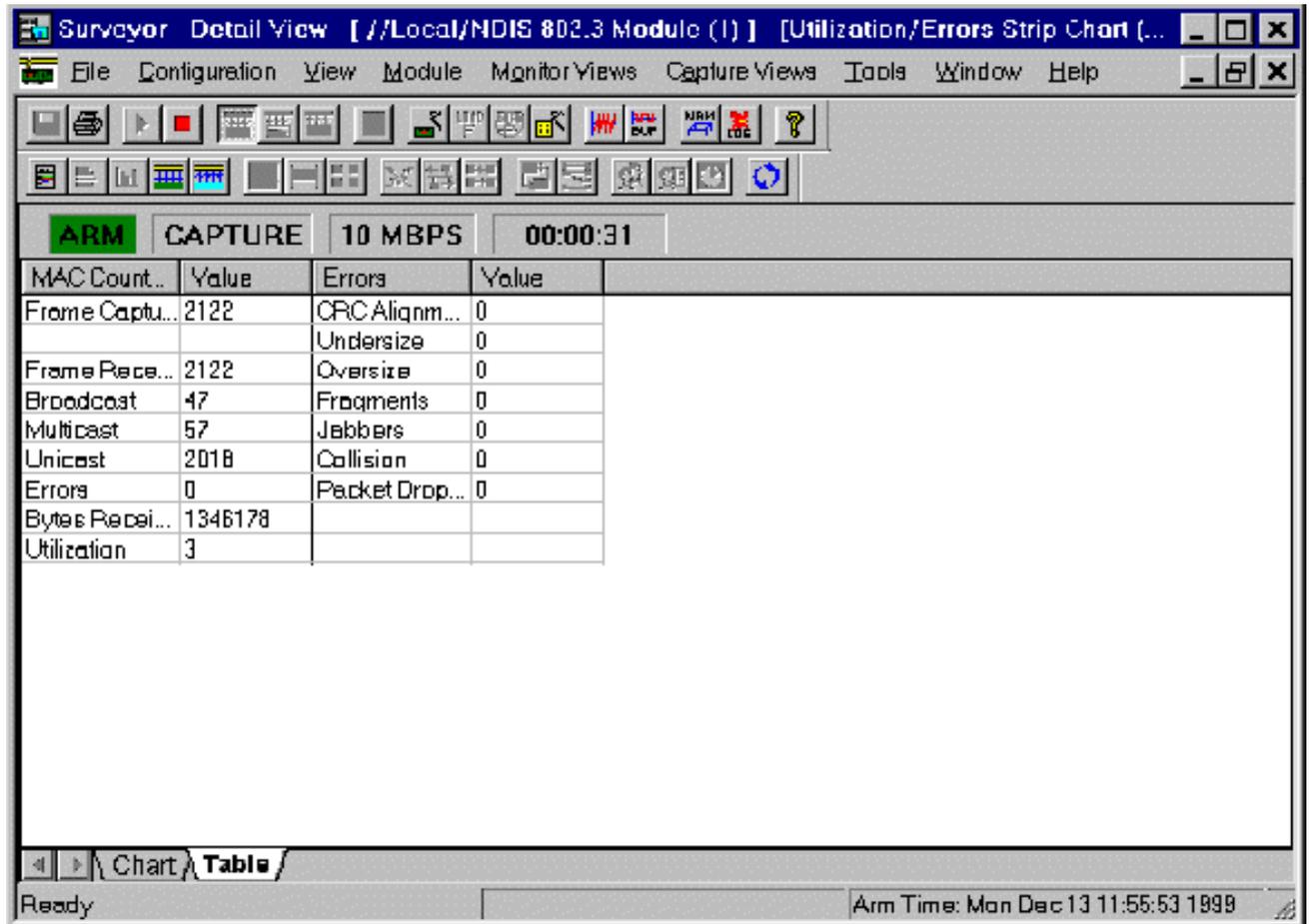
Sélectionner le mode *Capture*. Les données sont ainsi capturées dans le buffer de capture à des fins d'analyse hors ligne.

Cliquer sur .

Lancer l'acquisition. On obtient la figure suivante :

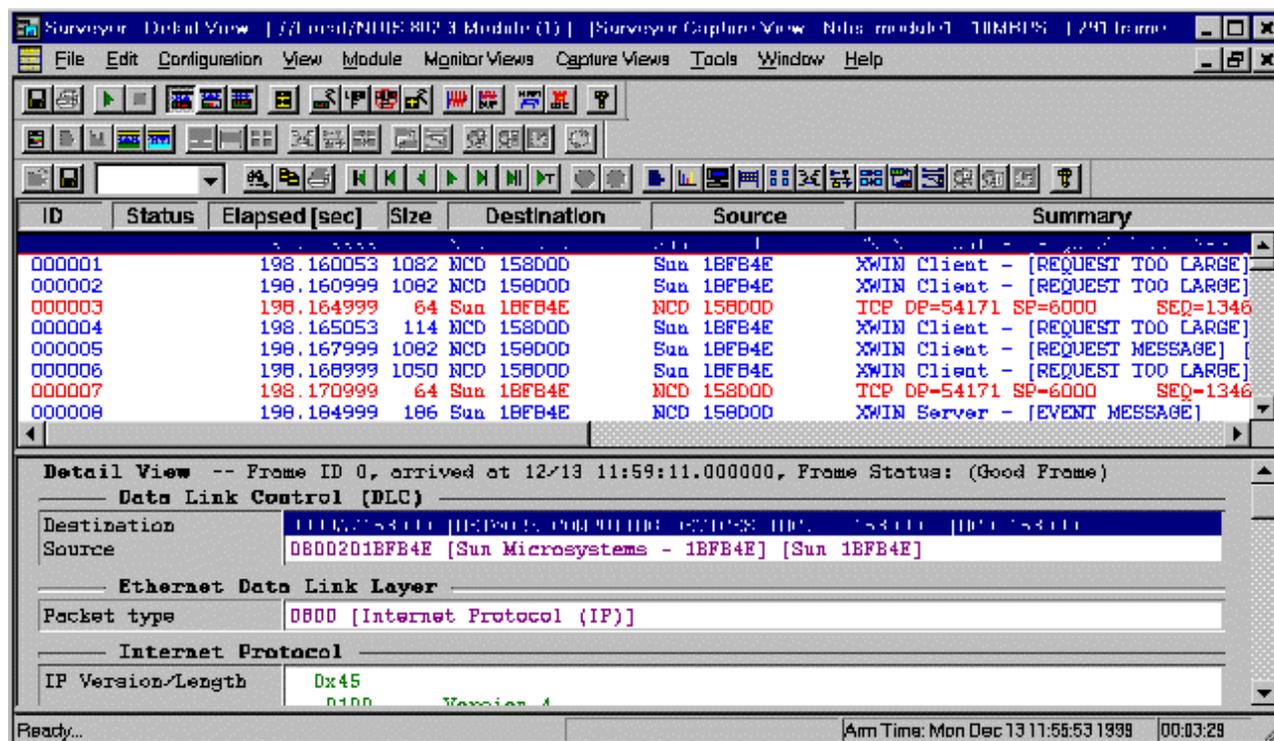


Double cliquer pour accéder au mode détaillé. On obtient quelque chose de similaire vue au point précédent :



On a ici la vue détaillée sous forme de tableau (onglet *Table*).

En arrêtant la capture et en choisissant le menu *Capture Views>Capture View* (touche F9), on peut accéder au buffer de capture et visualiser les trames capturées. On obtient la figure suivante :

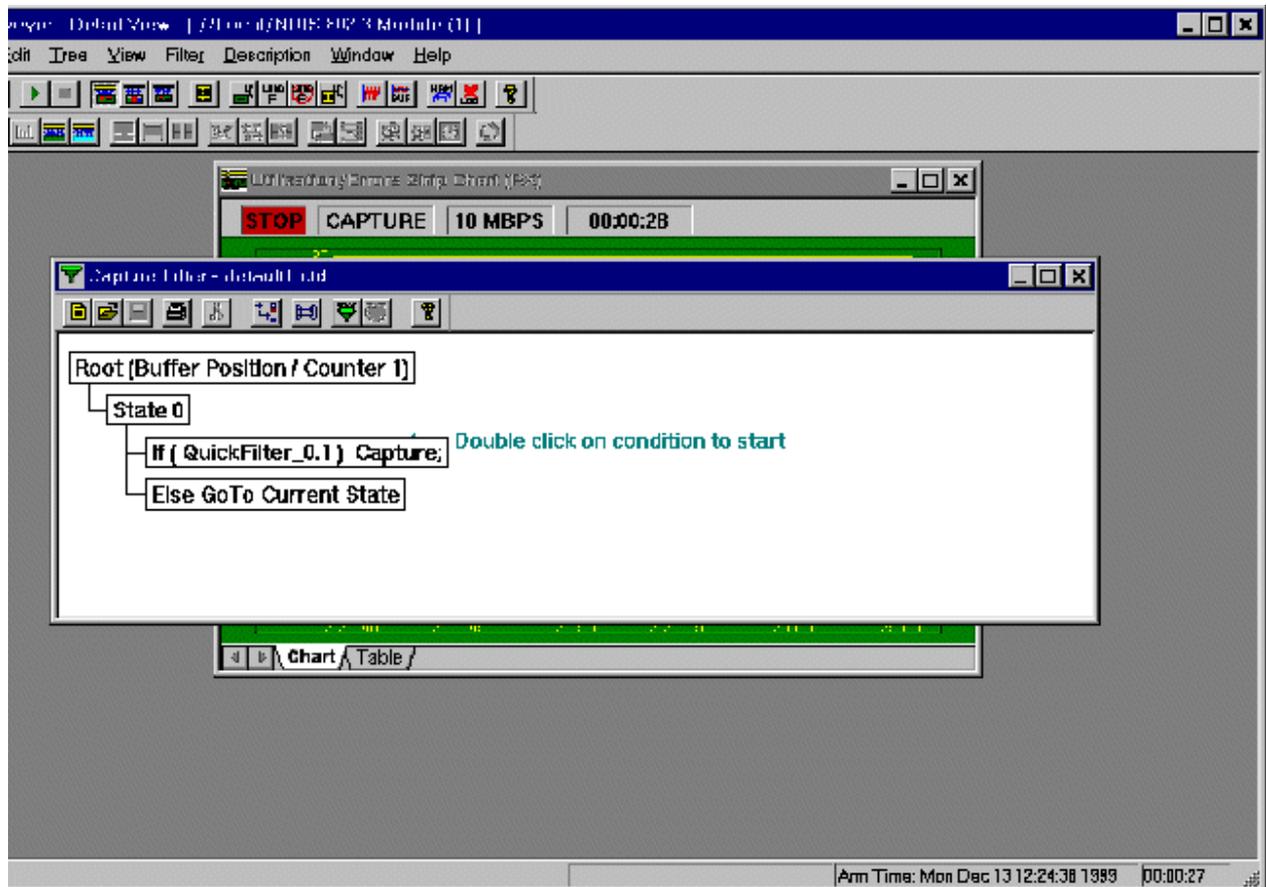


On peut ainsi visualiser le contenu des trames ainsi que leur décodage détaillé.

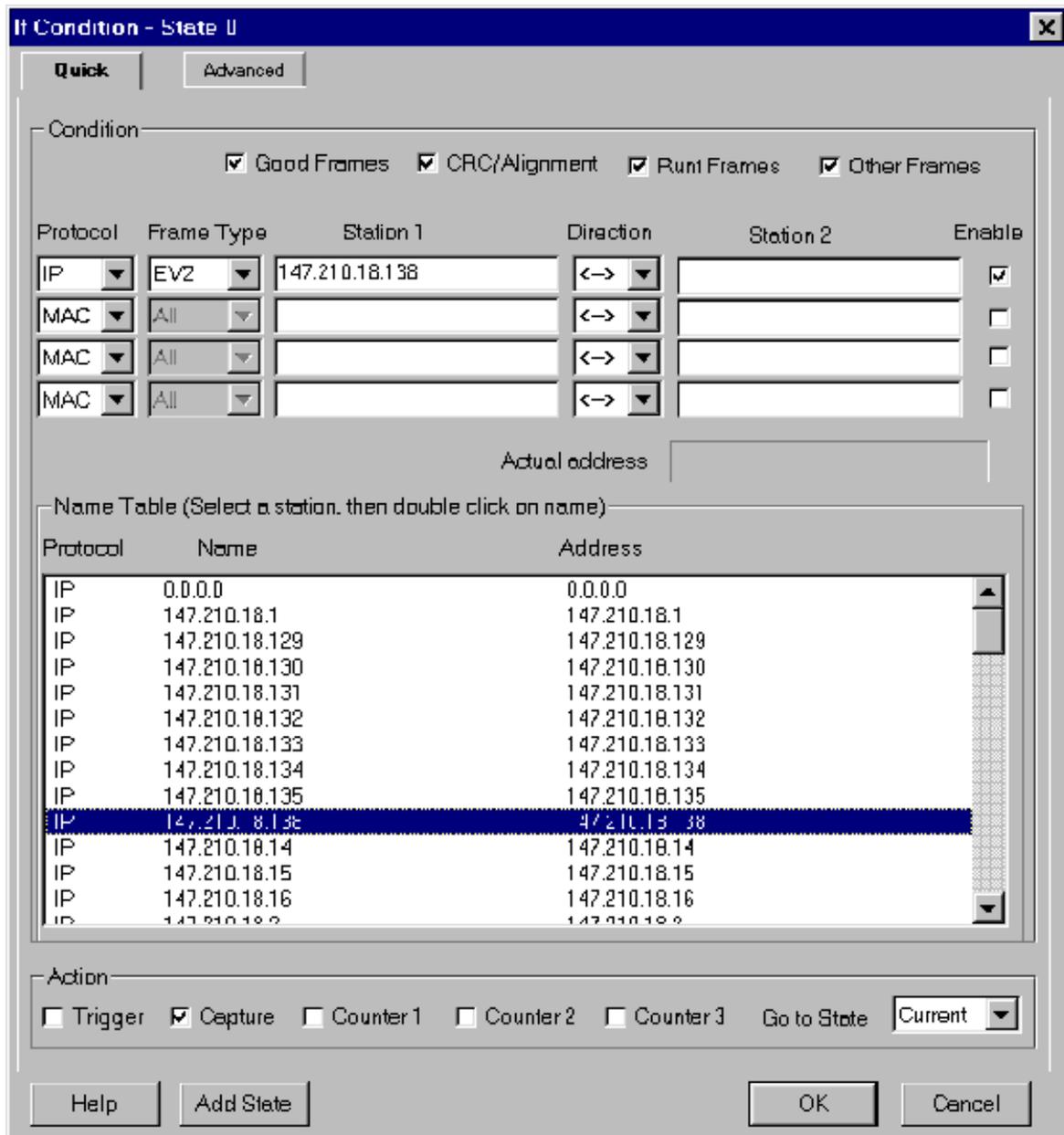
Par le menu *Configuration>Settings* (touche F3), on peut choisir son niveau de *packet slicing*. Pour le mode capture, il est intéressant de capturer toute la trame.

Il est à noter que l'on récupère toutes les trames émises et que l'on peut appliquer des filtres pour ne sélectionner que celles qui sont intéressantes suivant un protocole, un service (numéro de port socket)... On pourra réaliser un filtrage en utilisant le menu *Module>Capture Filter>Create/Modify Capture Filter* pour un filtre de capture (touche F5) ou le menu *Module>Display Filter>Create/Modify Display Filter* pour un filtre de visualisation (touche F4).

La création d'un filtre de capture se fait comme suit. Arrêter l'acquisition en cours. Choisir le menu comme indiqué ci-dessus.



Double cliquer comme indiqué sur la figure. Une fenêtre apparaît pour préciser la condition de filtrage, ici filtrage de l'adresse 147.210.18.138.



L'onglet *Advanced* permet d'accéder à des filtrages avancés et plus fins.

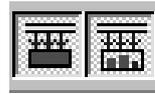


Cliquer sur l'icône pour activer le filtre lors de la prochaine capture. On peut d'ailleurs voir à tout moment si un filtre est actif en utilisant le menu *Module>Active TSP and Capture Filter*.

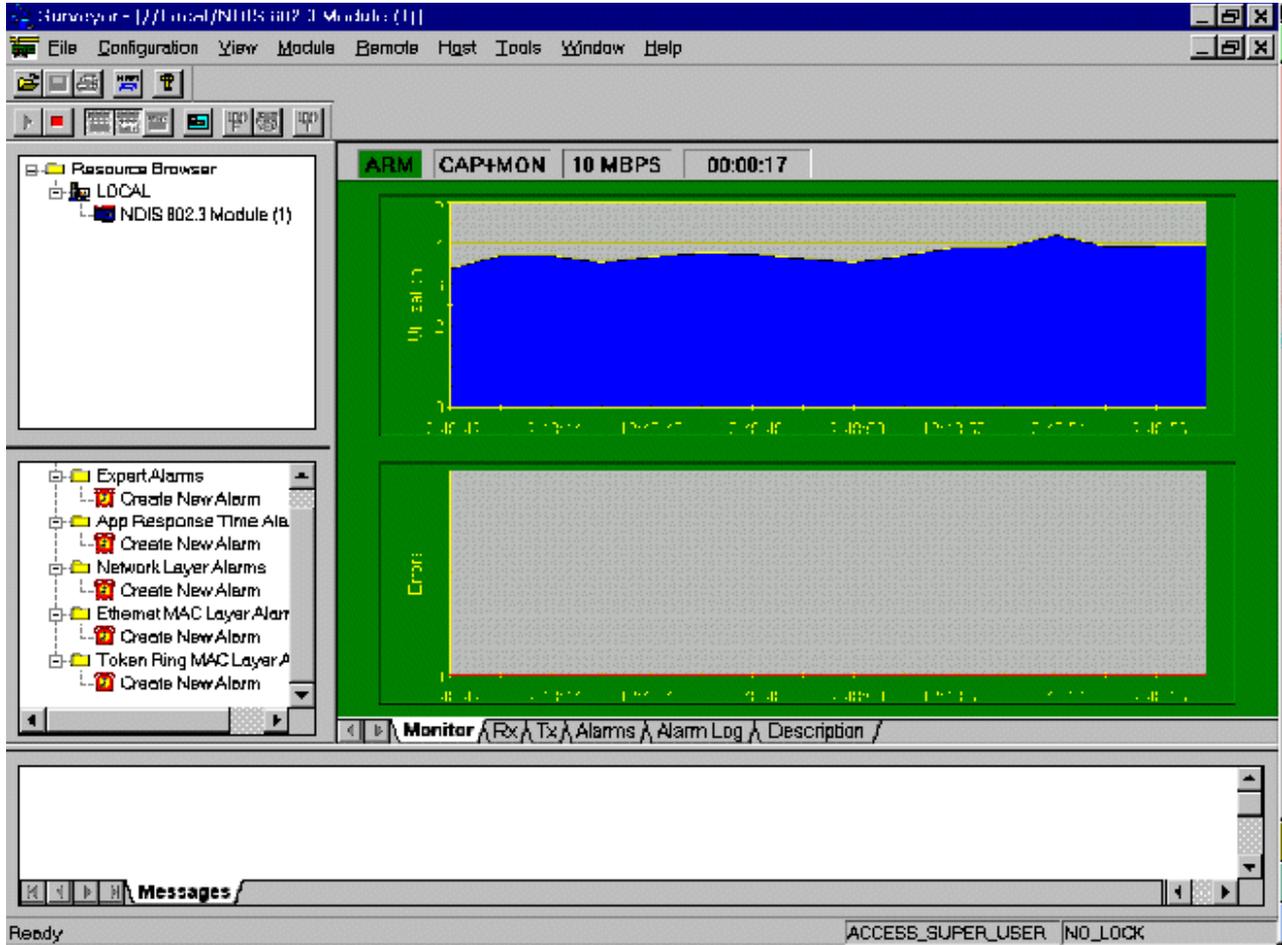
Relancer la capture. Arrêter pour visualiser le buffer d'acquisition.

❖ **Mode Capture & Monitor**

Il suffit de valider les 2 modes précédents en cliquant sur



. On obtient la fenêtre suivante :

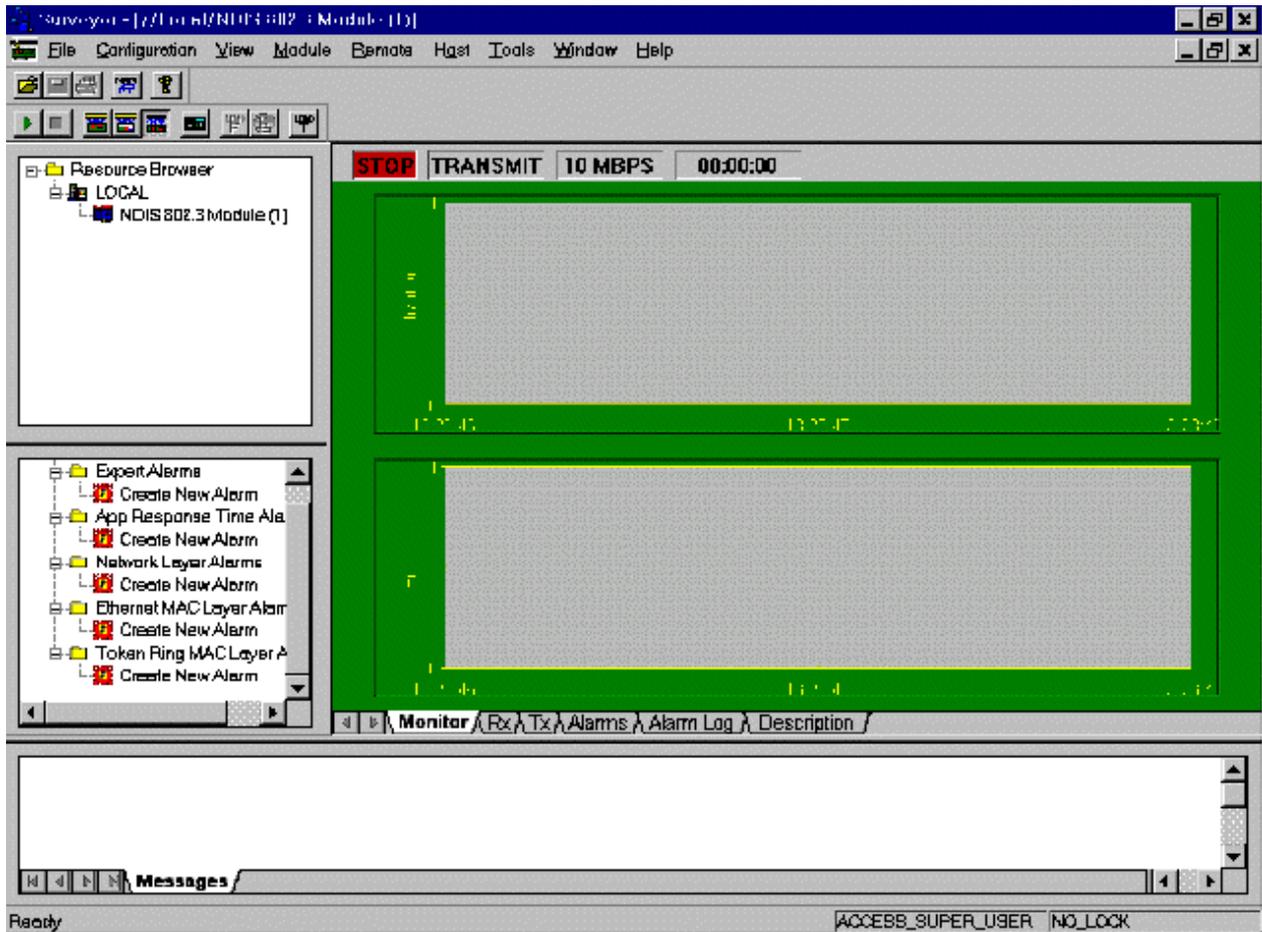


❖ **Mode émission de données**

Il est possible grâce au plug-in *Packet Blaster* d'émettre des données sur le réseau. Elles peuvent être issues d'une capture précédente ou bien être construites à la main.



Cliquer sur l'icône *Transmit*





Cliquer sur l'icône *Create/Modify Transmit Specifications* .

La fenêtre suivante apparaît pour spécifier les trames à émettre :

Transmit Specification: //Local/Ndis_module1 [X]

Defined Streams (Double click to activate/deactivate)

No streams Defined...

MAC Address

DA (Hex) Names...

SA (Hex)

Random Address Mode (values for "X" in DA/SA fields) Random Sequential

Packet Type (Hex) Packet Size

Data (32 bytes)

Seq #

Offset Start Stop Enable Seq#

Stream Mode

Packet Gap Frame Rate Traffic Rate (%util)

Burst Burst Count Burst Gap (msec)

Packet Gap Units micro sec milli sec secnds

Packet Gap

Add Add File... Delete Modify Edit Data...

Repeat Stream Time(s) Auto CRC

Transmission Modes

Transmit Spec (N frames) Time(s).

Transmit Continuously

Transmission Status

Transmission Speed 10Mbps.
 No streams activated.
 Memory used 0 byte(s).

Load Module Open Specs... Save Specs... Template... Cancel Help

Le bouton *Edit Data* permet d'éditer à la main tous les champs de la trame que l'on doit préciser. Le bouton *Template* permet d'avoir un pré remplissage de la trame suivant son type (IP...) et les données du niveau supérieur (TCP...). Il est intéressant de noter que l'on a accès au CRC 32 bits que l'on peut émettre faux par exemple pour générer des erreurs sur les trames. Le bouton *Names* permet de préciser les adresses MAC source et destination.

Voici l'extrait de l'aide en ligne :

To bring up the Transmit Specification dialog box, press the button from the Detail View toolbar. The Transmit Specification dialog box contains:

- List box for viewing defined streams called the Defined Streams list box (top)
- Radio buttons and fields for defining a stream (middle)
- Buttons for adding, modifying, or deleting streams
- Transmission status
- Buttons for loading the resource, opening/saving the specifications, adding stream using templates or Magic Packets™

The Defined Streams list box is a synopsis of the entire specification. Each added stream displays on a separate line. Each defined stream can be activated or deactivated by double-clicking the stream. A check mark beside the stream indicates that it's activated. Only activated streams are loaded to a resource.

Specify the contents and the size of the stream using the DA, SA, Packet Type, Packet Size, and Data fields. DA and SA values can be retrieved from the currently active name table using the Names button. Sequence numbers (Start Seq# and Stop Seq#) are used to number the packets; packet numbering may be useful at the receiving end. When viewing packets at the receiving end, the two-byte sequence number is located at 32H and 33H.

Set the Stream Mode using the radio buttons and the Burst check box. The stream mode defines the rate at which packets are transmitted from a resource and whether bursts of packets with a different rate will be transmitted within the stream.

Set the Repeat Streams field to repeat the stream more than one time. This setting specifies the number of times to repeat one complete stream -- not how many times to repeat transmission of the entire specification, nor the number of bursts within the stream.

The stream mode defines the rate at which packets are transmitted from a resource. The modes are:

Packet Gap The rate is set as an interval of time between packets. The interval can be set in seconds, milliseconds, or microseconds.
Frame Rate The rate is set in number of frames per second.
Traffic Rate The rate is set as a percentage of the maximum speed (10Mbps or 100Mbps) for the resource.

The minimum packet gap is .96 microseconds for 100Mbps networks and 9.6 microseconds for 10 Mbps networks. However, if you want to transmit at faster than line rate, one specific packet-gap value below the minimum. To transmit at faster than line rate, set the packet gap to 3.2 microseconds for 10 Mbps or .88 microseconds for 100 Mbps.

The transmission mode is either Transmit Continuously or Transmit frames 'n' times. Select Transmit Continuously to transmit activated streams in a loop until the resource is stopped.

Select Transmit Spec (N frames) to transmit activated streams a specific number of times. The number of times the entire specification is transmitted is set in the Time(s) field. The number of streams does not necessarily equate to the number of frames transmitted.

WARNING: The transmission mode should always be set prior to loading the resource. The transmission mode is not saved as part of the transmit specification. Unless you set the transmission mode, you may inadvertently flood the network with packets.

There are three ways to repeat frames when transmitting:

Check the bursts box Repeats frames of a stream with a specific timing set between the frames. The special timing is set in the Burst Gap field, the number of repetitions in the Burst Count field.

Repeat Stream n Times Repeats the stream n times. The gap between frames is set as a packet gap, frame rate, or traffic rate. The gap is referred to in the Transmit Specification as the Stream Mode.

Transmission Mode You can set the resource to loop through the entire transmit specification n number of times. Streams are repeated in the specification from first to last until you stop the resource or all streams are transmitted n times.

WARNING: Repeating frames using the transmission mode feature is a software function for CMM1 and NDIS modules. For these types of modules, there is a time gap of about 50ms between each transmission of the entire specification. Use Repeat Stream n Times or Bursts where timing issues are critical when sending frames.

Ways of repeating frames can be used together. For example, assume the following two streams are defined:

```
Stream 1; packet gap=100msec, burst count=4, burst gap=4msec, repeat frame 2 times
Stream 2; packet gap=200msec, no burst
```

The example results in the following:

```
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 104msec
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 100msec
Transmit Stream 1
Wait 104msec
Transmit Stream 2
Wait 200msec
```

If the transmission mode is set to continuous, the entire sequence above is repeated until the resource is stopped.

The Repeat Stream field sets how many times to repeat the current stream.

Cliquer sur le bouton *Load Module* pour charger la trame à transmettre. Il reste alors à lancer l'émission des trames.

Il faut noter que l'on ne peut pas travailler en full duplex avec les cartes réseaux NDIS (soit réception, soit émission) alors que la carte CMM2 le permet.

PARTIE II

- MANIPULATIONS -

6. EX 0 : QUESTIONS DE SYNTHÈSE

A l'issue de la lecture de ce TP, vous devriez pouvoir répondre à ces quelques questions...

1. Qu'est-ce qu'une adresse Ethernet ?
2. Qu'est-ce qu'une adresse Internet ou IP ? Différences avec la question 1 ?
3. Qu'est-ce qu'IP ? Quel est le mode de commutation et de connexion ?
4. Qu'est-ce qu'UDP ? Quel est le mode de connexion ?
5. Qu'est-ce que TCP ? Quel est le mode de connexion ?
6. Qu'est-ce qu'une "socket" ? Quel est son rôle ?
7. Par quoi est identifié un service de l'Internet ? Donner des exemples ?
8. Quelle est l'interface de programmation standard ou "API" pour Internet ?
9. A quoi correspond une collision sur Ethernet ? Que se passe-t-il alors ?
10. A quoi correspond le mode CSMA/CD ?
11. Est-ce que les protocoles de l'Internet sont liés à un support de transmission particulier ?
12. Dans le monde UNIX, quelles sont les principales commandes utilisées pour analyser le réseau Internet ?

7. EX 1 : ANALYSE D'UN RESEAU INTERNET

1. Booter les machines sous Linux. Se connecter sous le nom guest (mot de passe guest).
2. Vous avez à analyser un réseau Internet basé sur la technologie Ethernet 10Mb/s 10BaseT Twisted Pair mettant en œuvre 2 sous-réseaux et des "HUBS". Proposer le schéma de la configuration matérielle.
3. Retrouver et analyser sous Linux les fichiers de configuration statique du réseau. Remplir le tableau suivant :

Nom machine (" <i>hostname</i> ")	Adresse IP
PC1 :	
PC2 :	
PC3 :	
PC4 :	
PC5 :	

4. En utilisant les commandes UNIX décrites en annexe ?, retrouver les adresses IP des sous - réseaux, les valeurs des *netmasks*, les adresses de diffusion (*broadcast*). Quelle est la classe de réseau IP (A, B, C, D ?) ? Quelle est la taille maximale des paquets MTU(*Maximum Transport Unit*) ? Remplir le tableau suivant :

	Sous-réseau 1	Sous-réseau 2
Nom des machines qui y sont connectées		
Adresse réseau		
Netmask		
Broadcast		
MTU		
Classe réseau		

5. Peut-on accéder du sous-réseau 1 vers le sous-réseau 2 et réciproquement ? Quelle(s) commandes UNIX permet(tent) de le vérifier ? Faire l'essai. Que se passe-t-il si la valeur MTU des 2 sous-réseaux est différente ? Par quoi cela se traduira-t-il ?
6. Quelle commande UNIX va permettre de voir le sous-réseau 2 à partir du sous-réseau 1 et réciproquement ? Réaliser le routage à la main et vérifier alors que vous accédez bien au sous-réseau distant avec la commande UNIX *ping*.

7. Quel est l'intérêt d'introduire une passerelle (*gateway*) dans les tables de routage ? Vérifier vos tables de routage avec la commande UNIX *netstat -nr*.
8. A quoi correspond la route 127.0.0.0 ?
9. Retrouver les adresses Ethernet des interfaces réseau de toutes les machines par utilisation d'un commande UNIX. Que faire si cette adresse n'apparaît pas pour une machine donnée ? Remplir le tableau suivant :

Nom machine (<i>hostname</i>)	Adresse Ethernet
PC1 :	
PC2 :	
PC3 :	
PC4 :	
PC5 :	

10. Par analyse du fichier UNIX de configuration des numéros de port des services, retrouver le numéro de port des services telnet, ftp, mail et www. On peut utiliser la commande telnet autrement que pour se connecter à ce service par défaut en précisant un numéro de port : *% telnet numero_port*
Se connecter au service telnet de la machine par cette méthode et vérifier le bon fonctionnement.
11. Se connecter par telnet au service ftp de la machine. On essayera de faire un transfert ftp en utilisant les commandes *ftp cd, lcd, get, put, bin, asc*. On pourra utiliser le manuel en ligne de ftp (accès par la commande UNIX *% man ftp*).
12. Se connecter par telnet au service www de la machine. Une fois connecté, envoyer le caractère " RETOUR CHARIOT ". Que se passe-t-il ? Même chose en envoyant les caractères ESPACE puis "RETOUR CHARIOT". Que vous renvoie le serveur www ? Quel est le type des données renvoyées par le serveur ? Le protocole HTTP utilisé par un serveur www est structuré sous forme de commandes ASCII dont la structure générale est donnée ci-après (RFC1945) :
GET action HTTP/1.0
Autres infos passées au serveur
Un RETOUR CHARIOT →
Un RETOUR CHARIOT →
Données de l'utilisateur

La commande HTTP peut être GET, PUT, POST et HEAD suivant l'action demandée (généralement GET). Un exemple de données envoyées au serveur www est celui-ci :

```
POST /cgi-bin/HP8510_perform.sh HTTP/1.0
Content-type : application/octet-stream
Content-length: 14
```

→
→
"DATA de 14 OCTETS"

Le serveur en retour renvoie un code d'erreur dont les principaux sont :

200 : OK
204 : No content
400 : Bad request
403 : Forbidden
404 : Not found
408 : Request timeout

Un exemple de données retournées par le serveur www est :

```
HTTP/1.0 200 OK
Date: Mon, 06 Dec 1999 14:50:09 GMT
Server: Apache/1.1.1
Content-type: text/plain
Content-length: 3
Last-modified: Mon, 06 Dec 1999 14:47:55 GMT
```

En vous aidant de l'exemple précédent et en utilisant *telnet*, récupérer le fichier HTML `index.html`. Quel est le code de retour ?

13. Se connecter par telnet au service mail de la machine. Les commandes envoyées au serveur de mail (protocole SMTP *Simple Mail Transfer Protocol*, RFC821) sont structurées sous forme de lignes de commandes ASCII. Un exemple d'échanges de commandes SMTP est proposé ci-après (S: commande à taper, R: réponse du serveur SMTP) :

```
R: 220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready

S: MAIL FROM:<kadionik>
R: 250 OK

S: RCPT TO:<kadionik>
R: 250 OK

S: DATA
R: 354 Start mail input; end with <CRLF>.<CRLF>
S: Blah blah blah...
S: ...etc. etc. etc.
S: .
R: 250 OK

S: QUIT
R: 221 BBN-UNIX.ARPA Service closing
```

En s'aidant de l'exemple, envoyer un mail à l'utilisateur `guest`. Vérifier sa bonne réception en utilisant la commande UNIX *mail*.

14. Utiliser la commande UNIX *traceroute* pour analyser la route prise pour accéder d'une machine du sous-réseau 1 à une machine du sous-réseau 2 et réciproquement. On pourra consulter le manuel en ligne de la commande. A quoi correspond le champ "HOP" ? On en fera le rapprochement avec le champ TTL d'un trame IP.

15. On désire maintenant analyser les données transitant sur le sous-réseau IP en décortiquant les données reçues par l'interface Ethernet. La commande UNIX *tcpdump* permet de faire cette analyse. Regarder la syntaxe de cette commande (*% man tcpdump*). Il est à noter que cette commande ne peut être exécutée que par le super utilisateur en temps normal. On verra pourquoi ensuite...

Que réalisent les commandes suivantes :

```
% tcpdump -a -v net 147.210
% tcpdump -a -v tcp port 21
% tcpdump -a -v -x host ranko and port 80
```

16. On utilise la commande *tcpdump* pour analyser le trafic du port telnet de la machine locale. Quelle est la commande complète UNIX à utiliser ? On obtient une trace comme celle donnée ci-après.

```
User level filter, protocol ALL, datagram packet socket
tcpdump: listening on all devices
11:41:16.234776 eth0 < ranko.enserb.u-bordeaux.fr.telnet >
pythagore.enserb.u-bordeaux.fr.1027: P 20:27(7) ack 2 win 10136
<nop,nop,timestamp 177618092 52722> (DF) (ttl 255, id 2522)
4500 003b 09da 4000 ff06 24cf 93d2 129e
93d2 12d1 0017 0403 d44b 06d4 43c5 db75
8018 2798 8c45 0000 0101 080a 0a96 3cac
0000 cdf2 6c6f 6769 6e3a 20

E^@ ^@ ; ^I.. @^@ ..^F $.. .... ^R..
.... ^R.. ^@^W ^D^C .. K ^F.. C.. .. u
..^X '.. .. E ^@^@ ^A^A ^H^J ^J.. <..
^@^@ .... l o g i n :
11:41:24.804490 eth0 < ranko.enserb.u-bordeaux.fr.telnet >
pythagore.enserb.u-bordeaux.fr.1027: P 37:47(10) ack 12 win 10136
<nop,nop,timestamp 177618949 53579> (DF) (ttl 255, id 2532)
4500 003e 09e4 4000 ff06 24c2 93d2 129e
93d2 12d1 0017 0403 d44b 06e5 43c5 db7f
8018 2798 ffbf 0000 0101 080a 0a96 4005
0000 d14b 5061 7373 776f 7264 3a20

E^@ ^@ > ^I.. @^@ ..^F $.. .... ^R..
.... ^R.. ^@^W ^D^C .. K ^F.. C.. ..^;
..^X '.. .... ^@^@ ^A^A ^H^J ^J.. @^E
^@^@ .. K P a s s w o r d :
11:41:29.050425 eth0 > pythagore.enserb.u-bordeaux.fr.1027 >
ranko.enserb.u-bordeaux.fr.telnet: P 12:13(1) ack 47 win 32120
<nop,nop,timestamp 54003 177618949> (DF) (ttl 64, id 252)
4500 0035 00fc 4000 4006 ecb3 93d2 12d1
93d2 129e 0403 0017 43c5 db7f d44b 06ef
8018 7d78 1bff 0000 0101 080a 0000 d2f3
0a96 4005 74

E^@ ^@ 5 ^@.. @^@ @^F .... .... ^R..
.... ^R.. ^D^C ^@^W C.. ..^; .. K ^F..
..^X } x ^[.. ^@^@ ^A^A ^H^J ^@^@ ....
```

^J.. @^E t

Les données sont présentées sous forme ASCII et hexadécimal. Que peut-on dire sur les traces obtenues concernant la phase d'entrée du mot de passe ? Un réseau local Internet est-il sûr ? En déduire maintenant pourquoi la commande *tcpdump* est réservée au super utilisateur.

17. On utilise la commande *tcpdump* pour analyser le trafic ICMP de la machine locale. Quelle est la commande complète UNIX à utiliser ? Regarder dans la documentation la structure d'une trame ICMP. Pour générer des trames ICMP, on peut utiliser la commande UNIX ping. La mettre en œuvre pour observer le trafic. Quelles sont les données de test émises ?
18. On utilise la commande *tcpdump* pour analyser le trafic IP de la machine locale. Quelle est la commande complète UNIX à utiliser ? Regarder dans la documentation la structure d'une trame IP.
19. On utilise la commande *tcpdump* pour analyser le trafic TCP de la machine locale. Quelle est la commande complète UNIX à utiliser ? Regarder dans la documentation la structure d'une trame IP.
20. On utilise la commande *tcpdump* pour analyser le trafic du service FTP de la machine locale. Quelle est la commande complète UNIX à utiliser ? Pour générer le trafic du service FTP, on peut utiliser la commande UNIX *ftp* par exemple.
21. On utilise la commande *tcpdump* pour analyser le trafic du service SMTP de la machine locale. Quelle est la commande complète UNIX à utiliser ? Pour générer le trafic du service de mail, on peut s'envoyer un message par exemple.
22. Faire un bilan des outils mis à disposition sous UNIX pour gérer un réseau Internet (fichiers de configuration et commandes UNIX). Quel est l'un des problèmes majeurs d'un administrateur Internet ?

8. EX 2 : ANALYSE RESEAU AVEC SURVEYOR

1. Rebooter les machines sous Windows NT. Se connecter sous le nom guest (mot de passe guest).
2. Ouvrir une fenêtre MSDOS. Il existe sous Windows NT l'équivalent des commandes UNIX *route*, *netstat*, *ping*, *telnet* et *arp*. Par utilisation de ces commandes, retrouver les tables de routage.
3. Peut-on réaliser le routage du sous-réseau 1 vers le sous-réseau 2 ou inversement ? Retrouver la syntaxe de la commande *route* sous Windows NT.
4. Par utilisation des commandes *ping* et *arp*, retrouver les adresses Ethernet des différentes machines connectées.
5. Par utilisation du menu général Windows, lancer l'outil SURVEYOR (sous-menu TP E3 télécom). Choisir la carte Ethernet pour l'analyse ou la carte CMM2 pour ceux qui ont le PC *pomme1*. Lancer le monitoring. Analyser le flux des services IP en présence en cliquant sur le bouton *Protocol distribution*.
6. Retrouver les adresses Ethernet des différentes machines connectées en cliquant sur le bouton *Host Table*.
7. Retrouver les adresses IP des différentes machines connectées en cliquant sur le bouton *Network Layer Host Table*.
8. Retrouver la matrice de connexion au niveau Ethernet des différentes machines connectées en cliquant sur le bouton *Host Matrix*.
9. Retrouver la matrice de connexion au niveau IP des différentes machines connectées en cliquant sur le bouton *Network Layer Matrix*.
10. Retrouver la matrice de connexion au niveau application entre les différentes machines connectées en cliquant sur le bouton *Application Layer Matrix*. On utilisera par la suite ces différentes fonctionnalités (points 6 à 10) comme aide à l'analyse

11. Analyse du flux ARP. Lancer le monitoring. A quoi correspondent les ARP sur les adresses IP 192.9.200.254 ou bien 192.9.201.254 ? Expliquer.

12. Lancer une capture puis arrêter. Analyser le buffer de capture et décortiquer une trame ARP en la comparant à sa structure théorique donnée en annexe. Quelle est l'adresse Ethernet de broadcast qui est utilisée ? Quel est le type paquet pour une trame ARP ?

13. Quelles est la version du protocole IP employé ?

14. Analyse du flux ICMP. Lancer la capture. A partir d'une fenêtre MSDOS, faire un *ping* vers une machine du réseau (par exemple *citron* d'adresse IP 192.9.202.1). Arrêter la capture. Retrouver dans le buffer de capture les trames correspondant aux échanges *ping*. Quel est le type de la trame émise ? Quel est le type paquet ? Comparer la structure de la trame émise avec la structure théorique donnée en annexe. Combien de trames sont échangées pour un *ping* ? Quelle est la valeur TTL et quel est son rôle ?

15. Quelle est la taille du CRC utilisé pour les trames IP ? Quel est le pouvoir de détection théorique d'un tel code ? Est-on conforme à la norme IP ?

16. Analyse du flux *telnet*. Lancer la capture. A partir d'une fenêtre MSDOS, faire un *telnet* vers une machine du réseau (par exemple *citron*). Arrêter la capture. Retrouver dans le buffer de capture les trames correspondant aux échanges *telnet*. Quel est le type de la trame émise ? Quel est le type paquet ? Quel protocole de transport est utilisé ? Comparer la structure de la trame émise avec la structure théorique donnée en annexe. Préciser les numéros de port source et destination. La valeur du port destination est-elle normale ? Combien de trames sont échangées pour établir la liaison ? A quoi correspondent les numéros d'acquittement ? Quelle est la grande différence avec un numéro de séquence dans LAP-D par exemple ? Retrouver dans le flux de données la séquence d'échange *username/passwd*. Le mot de passe est-il crypté ? Que peut-on dire sur la sécurité d'un réseau Internet ?

17. Analyse du flux *ftp*. Lancer la capture. A partir d'une fenêtre MSDOS, faire un *ftp* vers une machine du réseau (par exemple *citron*). Arrêter la capture. Retrouver dans le buffer de capture les trames correspondant aux échanges *ftp*. Quel est le type de la trame émise ? Quel est le type paquet ? Quel protocole de transport est utilisé ? Comparer la structure de la trame émise avec la structure théorique donnée en annexe. Préciser les numéros de port source et destination. La valeur du port destination est-elle normale ? Combien de trames sont échangées pour établir la liaison ? Retrouver dans le flux de données la séquence d'échange *user/passwd*. Le mot de passe est-il crypté ? Que peut-on dire encore sur la sécurité d'un réseau Internet ?

18. Analyse du flux HTTP. Lancer la capture. Lancer *netscape* et se connecter au serveur web d'une machine du réseau (par exemple *citron*). Arrêter la capture. Retrouver dans le buffer de capture les trames correspondant aux échanges HTTP. Quel protocole de transport est utilisé ? Comparer la structure de la trame émise avec la structure théorique donnée en annexe. Préciser les numéros de port source et destination. La valeur du port destination est-elle normale ? Retrouver dans le flux de données les commandes HTTP ? Retrouver dans le flux de données les pages HTML échangées ainsi que les éventuelles images de format GIF.

19. Construction et émission de trames IP. On travaillera pour les manipulations suivantes par groupe de 2 binômes : un s'occupera de la partie émission de trames IP tandis que l'autre s'occupera de la partie réception/analyse. Le binôme qui a le PC *pomme1* peut travailler tout seul. Il choisira de travailler avec la carte CMM2 en réception/analyse et avec la carte NDIS en émission. Ces limitations d'utilisation viennent du fait que les cartes réseaux NDIS ne sont utilisables qu'en mode half duplex. Dans tout ce qui suit, le binôme réception/analyse se placera en mode monitoring/capture. Les explications données s'adressent maintenant au binôme émission. Quelle est la taille minimale d'une trame IP ? Pourquoi cette limitation ? Créer une trame IP d'émission de taille minimale. On prendra comme adresse Ethernet source (SA) \$000002 et comme adresse Ethernet destination (DA) \$000001. Le CRC sera laissé égal à 0. La valeur du CRC est-elle correcte ? Après avoir chargé le buffer d'émission, lancer l'émission de trames en mode continue. Lancer la capture puis arrêter. Retrouver dans le buffer de capture les trames émises. En analyser une en la comparant à sa structure théorique. Le CRC émis est-il correct ? Trouver une explication à cette différence.

20. Emission de trames ARP fantaisistes. Construire une trame ARP en utilisant les *templates* disponibles. On prendra SA égale à \$000002 et DA égale à \$000001. Générer le bon CRC. Après avoir chargé le buffer d'émission, lancer l'émission de trames en mode continue. Lancer la capture puis arrêter. Retrouver dans le buffer de capture les trames émises. En analyser une.

21. Emission de trames ARP . Construire une trame ARP en utilisant les *templates* disponibles. On prendra SA égale à celle du PC et DA égale à celle du PC *citron*. On renseignera tous les champs en s'aidant de la capture préalable d'une trame ARP émise sur le réseau. On notera la valeur des champs suivants : *Hardware Type=\$0, Protocole Type=\$80, Hardware Size=\$6, Protocole Size=\$4, Operation=\$1*. Après avoir chargé le buffer d'émission, lancer l'émission de trames en mode continue. Lancer la capture puis arrêter. Retrouver dans le buffer de capture les trames émises. En analyser une. Regarder sur la console système de *citron* les traces du programme *tcpdump*. A-t-on une réponse dans les traces du buffer de capture ou sur la console système de la part de *citron* ?

22. Emission de trames ICMP . Construire une trame ICMP en utilisant les *templates* disponibles. On prendra SA égale à celle du PC et DA égale à celle du PC *citron*. On renseignera tous les champs en s'aidant de la capture préalable d'une trame ARP émise sur le réseau. Après avoir chargé le buffer d'émission, lancer l'émission de trames en mode continue. Lancer la capture puis arrêter. Retrouver dans le buffer de capture les trames émises. En analyser une. Regarder sur la console système de *citron* les traces du programme *tcpdump*. A-

t-on une réponse dans les traces du buffer de capture ou sur la console système de la part de *citron* ?

23. Emission de trames HTTP . Lancer *netscape*. Lancer l'analyseur en mode analyse/capture. Se connecter au serveur web de citron. Arrêter la capture. Retrouver les trames correspondant aux échanges de type HTTP. Analyser le buffer de capture. Sauvegarder la première trame de requête HTTP dans un fichier de nom `http` (suffixe `.CAP` par défaut) (icône *Save* de la fenêtre *Capture View*). Construire une requête HTTP en utilisant le fichier de capture précédemment créé lors de la création du buffer de capture (bouton *Add File*). Après avoir chargé le buffer d'émission, lancer l'émission de trames en mode continue. Lancer la capture puis arrêter. Retrouver dans le buffer de capture les requêtes HTTP émises. Analyser les échanges. Regarder sur la console système de *citron* les traces du programme *tcpdump*. A-t-on un échange complet et correct ? On pourra s'en persuader en regardant les traces du serveur web de *citron* en analysant le fichier `/etc/httpd/logs/access_log` sur *citron*.

24. Classer les protocoles ARP, ICMP, TCP, HTTP, *ftp* et *telnet* dans la catégorie protocoles sans états ou catégorie protocoles avec états.

9. EX 3 : INTRODUCTION A LA PROGRAMMATION RESEAU

A venir...

- ANNEXE -
DOCUMENTS DE TRAVAIL

- DOCUMENTS 1 -

INTRODUCTION A L'INTERNET COMPRENDRE L'INTERNET

- DOCUMENTS 2 -

LES FICHIERS ET LES COMMANDES DE CONFIGURATION UNIX DES SERVICES DE L'INTERNET

- DOCUMENT 3 -

PRESENTATION D'ETHERNET 802.3

- DOCUMENT 4 -

INTRODUCTION TO THE INTERNET PROTOCOLS TCP-IP : PROTOCOLES DE L'INTERNET

- DOCUMENT 5 -

NOTIONS DE ROUTAGE IP LE RUTGERS : INTRODUCTION TO ADMINISTRATION OF AN INTERNET-BASED LAN

- DOCUMENT 6 -

LE WORLD WIDE WEB

- DOCUMENT 7 -

L'API SOCKET : PROGRAMMATION RESEAU

- DOCUMENT 8 -

GUIDE D'UTILISATION DE L'ANALYSEUR IP SURVEYOR