Outils d'analyse réseau pour systèmes basés sur Linux

Version : 1.0.2 // 02 mai 2003
Auteur : Thomas Deillon

Sommaire

1. Sources utilisées pour créer ce document

- http://www.robertgraham.com/pubs/sniffing-faq.html

- http://www.hsc.fr/ressources/breves/sniffers.html
- http://lab.erasme.org/ethereal/
- http://www.highsecu.net/dossiers/outils_de_securite.php
- http://lea-linux.org/
- http://www.ethereal.com/ethereal.1.html
- http://www.devveb.org/linux/linux_084.html
- http://www.hsc.fr/ressources/breves/expr-bpf.html
- http://www-eleves-isia.cma.fr/Doc/packages/ngrep/ngrep.html
- http://nezzwerker.net/ports.html

[Retour au sommaire]

2. Introduction

2.1. Comment cela fonctionne-t-il?

Ethernet a été créé autour d'un principe de partage: toutes les machines d'un reseau local partagent le même fil (BNC ou équivalent avec HUB).

Cela implique que toute les machines sont capables de «voir» tout le trafic sur ce fil. Cependant, la carte Ethernet est construite avec un «filtre» qui ignore le trafic qui ne lui appartient pas . Il le fait en ignorant toutes les trames qui n'ont pas son adresse MAC.

Un programme d'écoute enlève ce filtre , mettant ainsi la carte Ethernet en "promiscuous mode" ce qui équivaut à « tous passe ».

Liste des sniffers étudiés ici:

- tcpdump
- wireshark (ex-ethereal)
- ipgrab
- nstreams
- karpski
- tcpflow
- sniffit
- ntop
- netcat
- ngrep
- iptraf
- etherape
- cheops
- netperf
- netpipe

[Retour au sommaire]

3. TCPDUMP

Site Officiel

Il permet de visualiser en temps réel le trafic réseau avec (ou sans) le contenu des paquets. Il n'est pas très convivial mais très puissant. Exemple de résultat obtenu:

```
0x0000   45c0 0030 0000 0000 0211 c7f3 0a0a 05fe        E..0............
0x0010   e000 0002 07c1 07c1 001c 83e0 0000 1003        ................
0x0020   0a6e 0d00 6369 7363 6f00 0000 0a0a 0501        .n..cisco.......
08:34:30.005846 arp who-has pc040.stagiaires.tst tell pc004.stagiaires.tst
0x0000   0001 0800 0604 0001 0050 0435 e3e4 0a64        .........P.5...d
0x0010   1404 0000 0000 0000 0a64 1428 0000 0000        .........d.(....
```

Ecoute un réseau, un port, un protocole ou une composition des trois.

```
Ex: tcpdump -ln -v icmp and not net 10.10.10.0/24
```

Rq: Possibilité d'analyse des trames générées par tcpdump dans wireshark (ex-ethereal) et bien d'autres avec l'option

```
tcpdump -w fichier_de_sortie
```

3.1. Utilisation

On choisit l'interface avec l'option -i.

```
Ex: tcpdump -i eth2
```

On choisit la source ou/et la destination avec src host et dst host.

```
Ex: tcpdump src host 10.100.20.100 and dst host 10.100.25.2
```

On choisit le port avec la commande port:

```
Ex: tcpdump src host 10.100.20.100 and port 80
```

On choisit le protocole:

```
Ex: tcpdump src host 10.100.20.100 and tcp and port 21
```

On peut faire une combinaison de toutes ces options avec « or », « and » et « not »:

```
Ex: tcpdump src host 10.100.20.100 and dst host 10.100.25.2 or dst host 10.100.25.1 and tcp and port 80
```

Exemple vous cherchez le mot de passe de messagerie pop3 de votre voisin (a ne pas faire) :

```
tcpdump -x -X -s 0 -i eth0 dst host Nom_de_la_machine_du_collègue and tcp and port 110
```

et ensuite vous analysez le contenu des trames.

- l'option -x permet de visualiser la trame en hexadecimal
- l'option -X permet de visualiser la trame au format ASCII
- l'option -s 0 permet de tout prendre quelque soit la taille du paquet

Il existe un certain nombre d'options supplémentaires telles que la recherche d'éléments précis dans la trame.

```
Ex : tcpdump 'icmp[0] != 8 and icmp[0] !=0'
```

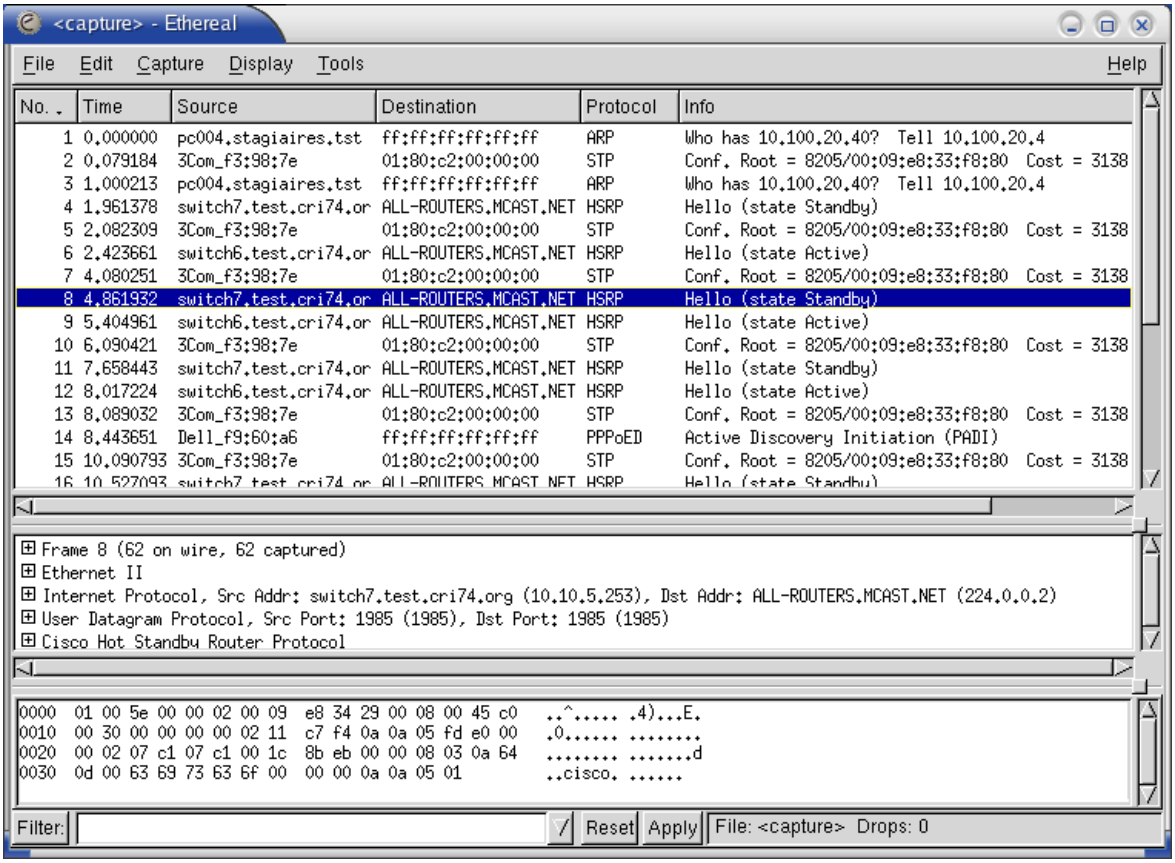
qui affiche tous les paquets icmp qui ne sont ni des "echo request" ni des "echo reply" (c'est à dire qui ne sont pas des ping).

[Retour au sommaire]

4. WIRESHARK (ex-ETHEREAL)

Site Officiel

Très complet avec une interface graphique, il permet de "décortiquer" de nombreux protocoles.



Dans la fenêtre de capture, il y a trois options + tous les filtres que l'on peut ajouter :

Le champ ***File*** permet de spécifier un fichier dans lequel se trouvent les paquets si on veut analyser une trace faite avec tcpdump (option -w) par exemple.

"Update list of packets in realtime" permet (s'il est coché) de voir les paquets s'afficher en temps réel (pendant la capture) dans la fenêtre des paquets disponibles. Cette option ne doit être utilisée que s'il n'y a pas trop de paquets conservés.

Automatic scrolling in live capture permet (s'il est coché) de voir les derniers paquets s'afficher en temps réel (pendant la capture) dans la fenêtre des paquets disponibles en faisant défiler la liste des paquets disponibles. Cette option ne doit être utilisée que s'il y a peu de paquets (surtout sur un portable avec un affichage plus que lent).

Enable name resolution permet (s'il est coché) de demander une traduction des adresses IP en noms. Cette option doit aussi être manipulée avec "précaution" car elle génère des requêtes DNS qui peuvent "encombrer" le réseau et prendre du temps. Surtout s'il y a beaucoup de machines différentes dans les paquets et qu'elles ne sont pas connues dans les DNS.

4.1. Utilisation

Les commandes de filtrage sont les mêmes que celles de tcpdump étant donné qu'elles utilisent la même librairie ***libpcap***. Cependant les options -x -X -s 0 ne sont pas prises en compte car ces options sont des options d'affichage et que wireshark a son propre affichage non modifiable.

Dans cette version « graphique » de tcpdump, il y a quelques options notables: il y a, lors de la capture, un filtre de capture qui permet de capturer que ce que l'on veut, tout le reste étant jeté.

Mais il existe aussi un filtre d'affichage permettant de faire des filtres temporaires pour faire des recherches .

Option de ce filtre d'affichage: on peut utiliser les opérateurs ==, !=, >, <, >= et <= pour comparer les champs avec des valeurs. Les expressions ainsi fabriquées peuvent être combinées avec les opérateurs && (pour un "et logique"), || (pour un "ou logique"), ^^ (pour le "ou exclusif") et ! pour la négation. L'usage des parenthèses est possible.

Tableau des options:

Champ	Type	Signification
ip.addr	Adresse IPV4	Adresse IP source ou destination
ip.dst	Adresse IPV4	Adresse IP destination
ip.flags.df	Booléen	Drapeau IP, ne pas fragmenter
ip.flags.mf	Booléen	Drapeau IP, fragments à venir
ip.ttl	Entier non signé sur 8 bits	Time to Live
nbdgm.src.ip	Adresse IPV4	Adresse IP source d'un paquet NetBios Datagram

nbdgm.src.port	Entier non signé sur 16 bits	Port IP source d'un paquet NetBios Datagram
http.request	Booléen	Requête HTTP
http.response	Booléen	Réponse HTTP
icmp.code	Entier non signé sur 8 bits	Numéro du code d'une commande ICMP
icmp.type	Entier non signé sur 8 bits	Numéro du type d'une commande ICMP
ftp.request	Booléen	Requête FTP
ftp.request.command	Chaîne de caractère	Commande FTP
ftp.reponse.data	Chaîne de caractère	Donnée de transfert FTP
dns.query	Booléen	Requête DNS
dns.reponse	Booléen	Réponse d'une requête DNS

De plus, la comande « Follow TCP Stream » est très utile pour suivre une connexion TCP de A à Z.

4.2. Utilisation recommandée

Pour rechercher quelque chose de pas très bien définie ou pour analyser la sortie du fichier Tcpdump.

Ex: On recupère tout sur le réseau et l'on voit une communication TCP, on se place sur une cette ligne et on utilise l'option "Follow TCP Stream" et wireshark va reconstituer le dialogue TCP .

```
220 kastor Microsoft FTP Service (Version 5.0).
USER tdede
331 Password required for thdei.
PASS xhs7gth
230-*************************************
230-*************************************
230-**                                 **
230-**   Welcome to                    **
230-**                                 **
230-**     Dept. GTR Official          **
230-**     FTP Site                    **
230-**                                 **
230-**                                 **
230-**                                 **
230-*************************************
230-*************************************
230 User tdede logged in.
SYST
215 Windows_NT version 5.0
PORT 10,100,20,100,4,17
200 PORT command successful.
LIST
150 Opening ASCII mode data connection for /bin/ls.
226 Transfer complete.
CWD Linux
250 CWD command successful.
PORT 10,100,20,100,4,18
200 PORT command successful.
LIST
150 Opening ASCII mode data connection for /bin/ls.
226 Transfer complete.
CWD ..
250 CWD command successful.
PORT 10,100,20,100,4,19
200 PORT command successful.
LIST
150 Opening ASCII mode data connection for /bin/ls.
226 Transfer complete.
QUIT
221 Thanks, BYE
```

de même, on peut reconstituer les pages HTML. On recupère le résultat de cette fonction et on le colle dans un fichier HTML. On a alors source de la page web originelle.

[Retour au sommaire]

5. NGREP

Vous connaissez la commande 'grep' pour rechercher des motifs dans des fichiers? Et bien, ngrep fait la même chose, mais sur les trames réseaux. Ngrep utilise les expression régulières comme grep.

Les possibilités de filtrage sont presque (pas les même commandes) les mêmes que celles de tcpdump étant donné qu'elles

utilisent la même librairie libpcap. Mais il y a certaines options en plus sur la recherche de chaine.

Ex:

```
ngrep -d eth0 -i 'USER|PASS' tcp port 21
```

Permet de récupérer facilement les mots de passe en connexion TCP.

- L'option -d sert à choisir l'interface
- L'option -i permet de choisir "case insensitive" (insensible à minuscule/majuscule).

PROBLEME DE NGREP: ne sait pas créer de fichiers lisibles par les autres programmes tel que wireshark d'ou sa limitation d'utilisation.

5.1. Utilisation recommandée

Pour des recherches précises de chaînes de caractères dans des trames. Dans l'exemple on recherche la chaîne PASS ou USER sans se soucier des majuscules (-i).

[Retour au sommaire]

6. IPGRAB

Rq importante : Il n'est pas possible de faire du sniff sur une carte sans adresse. De plus ses options sont inexistentes.

Exemple s'il y a une adresse IP.

```
********************************************************************************
                        IP Header
--------------------------------------------------------------------------------
Version:                4
Header length:          5 (20 bytes)
TOS:                    0x10
Total length:           52
Identification:         50728
Fragmentation offset:   0
Unused bit:             0
Don't fragment bit:     1
More fragments bit:     0
Time to live:           63
Protocol:               6 (TCP)
Header checksum:        13248
Source address:         10.100.20.2

--------------------------------------------------------------------------------
                        TCP Header
--------------------------------------------------------------------------------
Source port:            33547 (unknown)
Destination port:       22 (SSH)
Sequence number:        742086893
Acknowledgement number: 1532083139
Header length:          8 (32 bytes)
Unused:                 0
Flags:                  A
Window size:            60816
Checksum:               42885
Urgent:                 0
Option:                 1 (no op)
Option:                 1 (no op)
Option:                 8 (timestamp)
Length:                10
Timestamp value:       16911240
Timestamp reply:       7500672
```

6.1. Utilisation recommandée

Aucune

[Retour au sommaire]

7. TCPFLOW

Pemet de visualiser en ASCII le contenu des paquets et de rassembler les sessions TCP sur disque.

ex:

```
010.100.020.100.00022-010.100.020.002.40770: ..z.xE..!...i.d....G.f.t.....&......F1B.....Fq..
010.100.020.100.01385-193.048.132.252.00021: USER tdede
...
010.100.020.100.00022-010.100.020.002.40770: ....._.l.V....i.*2E.u.x2.Z.w<........*..UP/q.=..
010.100.020.002.40770-010.100.020.100.00022: c.;...-..]a.|7.+.4../E.. ..\/...Gv..2U%.@.:.xr.S
010.100.020.100.01385-193.048.132.252.00021: PASS xhs7gth
```

[Retour au sommaire]

8. SNIFFIT

Sniffit est assez générique. Il est cependant bien approprié pour surveiller une session telnet par exemple.

Sniffer des paquets et outils de surveillance pour les protocoles TCP/UDP/ICMP. Il permet de donner des détails techniques sur les paquets (SEC, ACK, TTL, Window, ...), mais également sur le contenu dans différents formats (hexa ou texte brut,...).

sniffit -i pour avoir le mode « graphique » en ligne de commande.

[Retour au sommaire]

9. NETCAT

Cette utilitaire est surnommé "le couteau suisse du protocole TCP/IP" (d'après le man). En effet, c'est un simple outil en ligne de commande qui permet de lire et d'écrire au travers de connexions réseaux en utilisant TCP ou UDP. Son implémentation vous permet de l'utiliser seul ou par l'intermédiaire d'autres scripts (perl, shell...); mais, en même temps, il est tellement puissant qu'il peut être utilisé comme un débuggeur réseau ou comme un outil d'exploration. En effet, il supporte n'importe quel type de connexion dont vous pouvez avoir besoin. Cet outil contient encore beaucoup d'autres fonctions qui pourront vous être utiles.

9.1. Utilisation

RQ importante: NetCat est un outil à tout faire. Il permet simplement de gérer un socket : d'envoyer... de recevoir... pas plus... Ca veut dire qu'il ne connait aucun protocole comme IRC,FTP,etc... Mais ca veut dire aussi qu'il nous affiche exactemment ce qui se passe dans la connexion, qu'il n'effectue aucun traitement, et qu'il envoie aussi exactemment ce que l'on veut qu'il envoie. (Attention telnet ne fait absolument pas la même chose, telnet est beaucoup moins puissant)

9.1.1. Netcat coté client

Ouverture d'un socket : netcat ftp.univ-savoie.fr 21. tout ce que vous taperez ensuite sera envoyé à l'adresse que vous avez indiquée.

Netcat peut émuler un client telnet:

```
netcat -t <IP_seveur_telnet> 23
```

9.1.2. Netcat côté serveur:

Ouverture du port 23 pour la réception:

```
NETCAT -l -p 23
```

- -l pour dire qu'on est en mode "listen" (on attend une connexion sur un port)
- -p 23 pour dire que le port sur lequel on attend la connexion est le port 23.

-> tout ce qui est écrit sur le client qui ce connecte a ce port sera écrit à l'ecran du serveur et inversement.

9.1.3. Les redirections Entrées/Sorties:

```
netcat -l -p 23 <  toto
```

-> à chaque connection d'un client sur ce serveur, le contenu du fichier toto sera affiché sur son écran.

Nous pouvons aussi bien sûr faire

```
netcat -L -p 23 > toto.log
```

pour renvoyer toutes les informations de connexion dans le fichier toto.log.

```
netcat -l -p 23 -e /bin/bash
```

Toutes les commandes du client seront redirigées vers le programme "bin/bash": ceci est donc un serveur telnet.

9.1.4. Scan de port:

```
netcat -vv -r -z 10.100.25.2  1-100
```

- L'option -vv: sert à avoir des informations sur les connexions
- L'option -z: pour effectuer un scan rapide, qui ne s'arrête pas si le port est ouvert.
- L'option -r: fait un "random" sur les ports de 1 jusqu'à 100 ici. Il fera tous les ports, mais pas dans un ordre précis.
- L'option -i tps_en_sec: Permet d'attendre un certain temps entre essai sur chaque port.

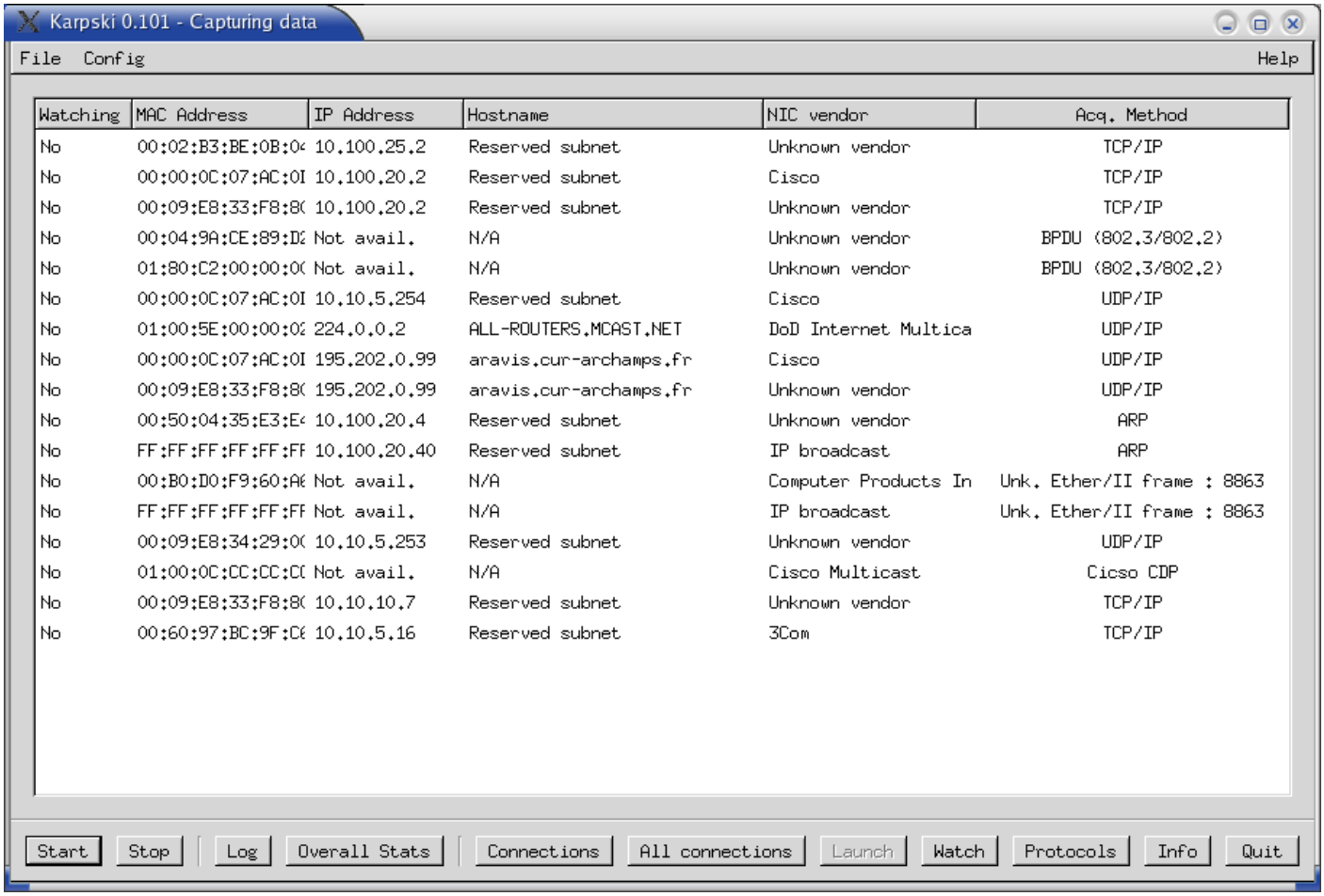
9.2. Utilisation recommandée

Il permet d'ouvrir facilement des connexions quelconques (TCP ou UDP) sans savoir programmer, aussi bien pour créer des petits clients/serveurs, que pour tester un programme. De plus il est utilisable en ligne de commandes, ce qui va permettre de facilement l'incorporer dans des scripts.

[Retour au sommaire]

10. KARPSKI

Version disponible : 0.101

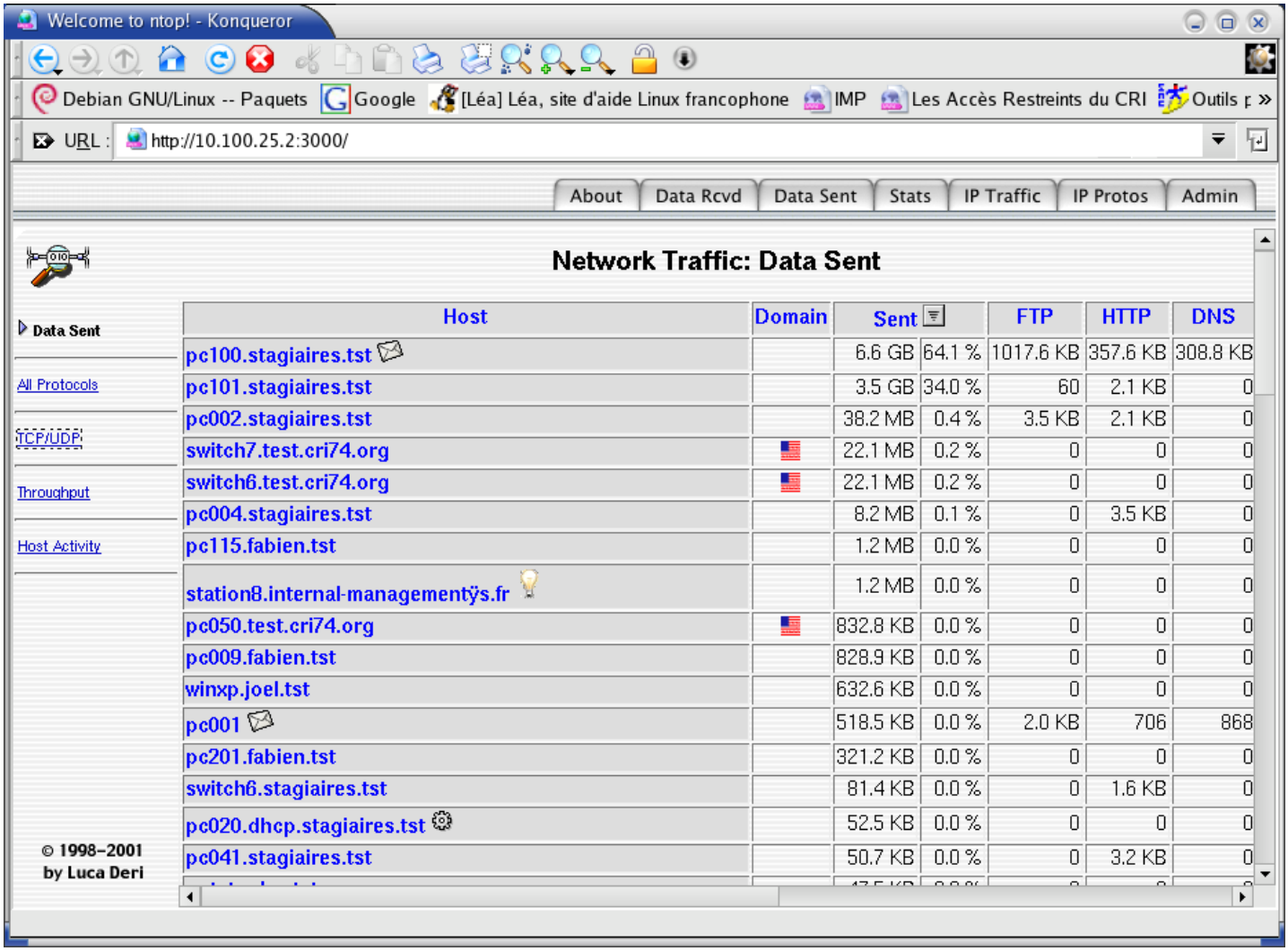


Si un jour il sort une version "sans bug", ce logiciel sera peut-etre et c'est même pas sûr interessant...

[Retour au sommaire]

11. NTOP

Utilisation que sur un réseau faiblement chargé.



Ntop est un programme permettant de collecter des statistiques sur l'utilisation du réseau. Il est capable de différencier les statistiques selon les protocoles, les interfaces utilisées, etc. De plus, ces statistiques sont consultables en temps réel sous forme graphique, avec un navigateur Web. Il permet d'afficher comme le fait 'top' l'utilisation du réseau. Il affiche un résumé de l'utilisation du réseau par machine comme 'top'.

Il existe aussi une version utilisable par l'intermédiaire d'un navigateur Web.

11.1. Utilisation recommandée

Pour les statistiques sur un réseau.

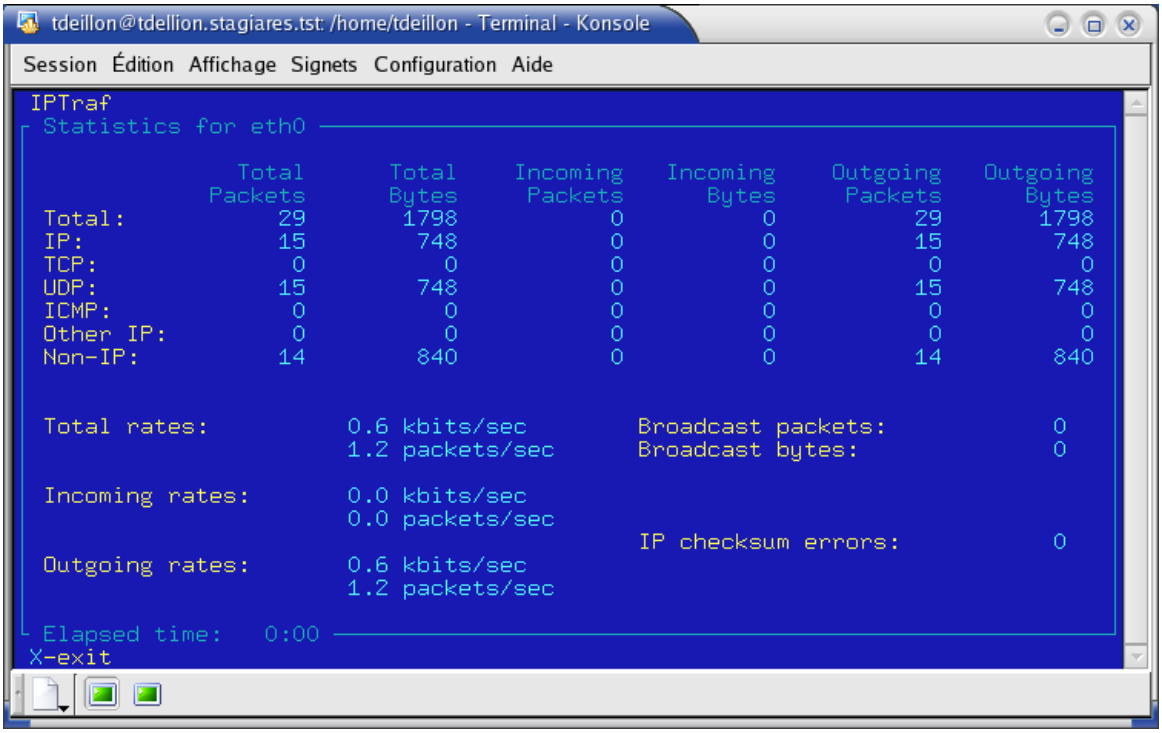
Logiciel très complet : statistiques par rapport aux protocoles, aux machines qui envoient, aux machines qui reçoivent, ...

rq: trou de sécurité quand on utilise l'interface Web sur une machine distante. Nécessite donc un firewall pour définir qu'elles sont les personnes qui ont le droit de voir les informations.

[Retour au sommaire]

12. IPTRAF

Outil en ligne de commande utilisant la librairie "ncurse" pour afficher en utilisant des couleurs des informations sur l'usage du réseau (TCP info, UDP counts, ICMP et OSPF information, Ethernet load info, node stats, IP checksum errors, ...).



12.1. Utilisation recommandée

Si vous avez l'impossibilité d'ouvrir une console graphique et que vous ne voulez ou pouvez pas installer ntop, ce logiciel vous permettra de faire quelques statistiques sur l'utilisation du réseau (ex bande passante).

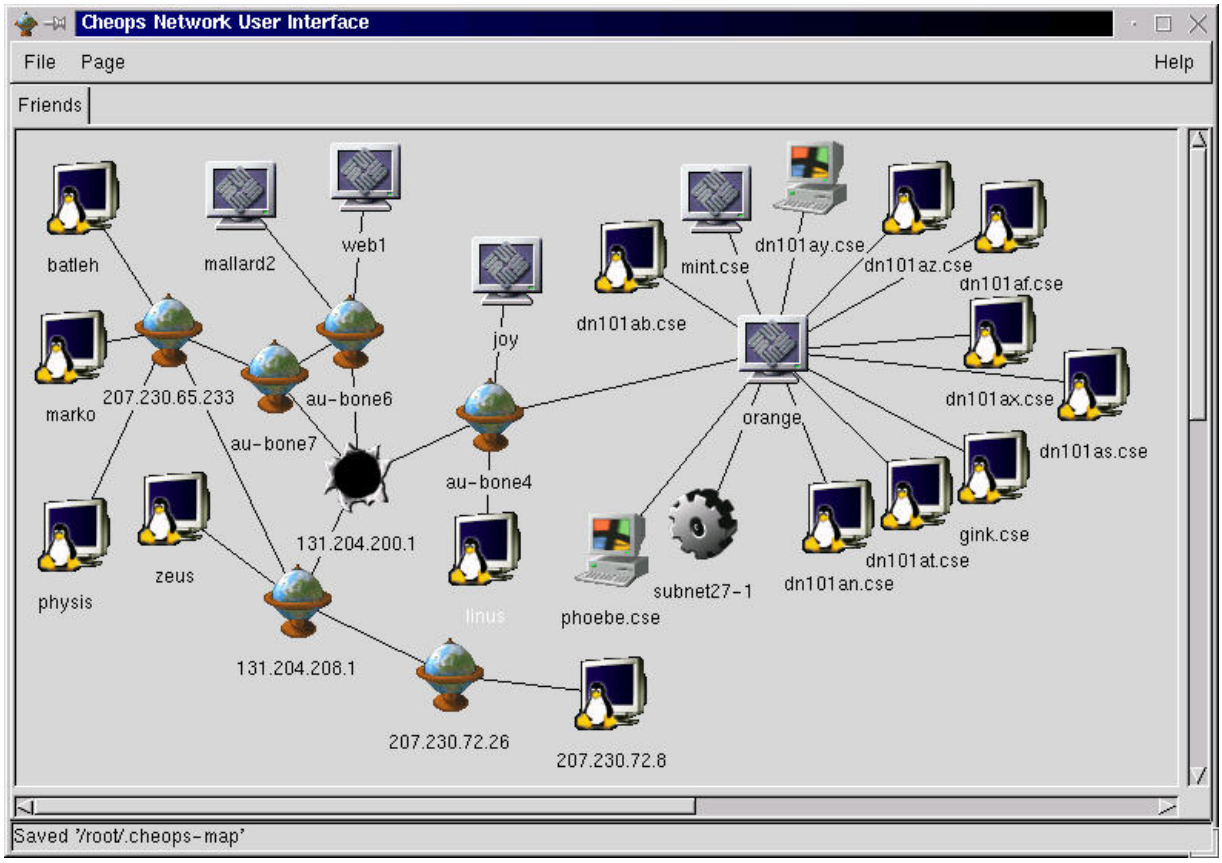
Il est assez convivial mais ses fonctionalités sont réduites.

[Retour au sommaire]

13. CHEOPS

Véritable "couteau suisse" pour analyser des réseaux, le tout avec un rendu agréable à l'aide d'une interface GTK. Cet outil vous permet d'avoir accès de manière aisée à la plupart des outils réseaux et il permet même d'afficher l'OS des machines.

Ce programme permet de faire un plan du réseau. Il envoie des trames UDP/ICMP sur le réseau en demandant qui ils sont .



13.1. Utilisation recommandée

En regardant le site constructeur, cet outil a l'air merveilleux , mais en le testant ici au CRI, il est arrivé a ne trouver que 5 machines en se trompant sur leur OS.

Mais s'il fonctionne (?!?) il doit être très utile.

[Retour au sommaire]

14. NESSUS

Site Officiel

Nessus est un scanner de vulnérabilités qui effectue un balayage réseau sur une cible pour chercher des vulnérabilités dans le réseau, comme des erreurs de programmation, des backdoors, etc...

14.1. Utilisation recommandée

Logiel de recherche de trous de sécurité réseau. Rq: Très facile d'utilisation.

[Retour au sommaire]

15. NSTREAMS

ntreams permet de synthétiser (en nature) les flux. C'est à dire d'afficher les protocoles utilisés sur un réseau indépendemment du nombre de fois où ils ont été utilisés.

Cet outil permet de faire la cartographie des flux écoutés sur le réseaux.

15.1. Utilisation

Liste des services utilisés comme ci-dessous:

```
nstreams -f fichier_sortie tcpdump
```

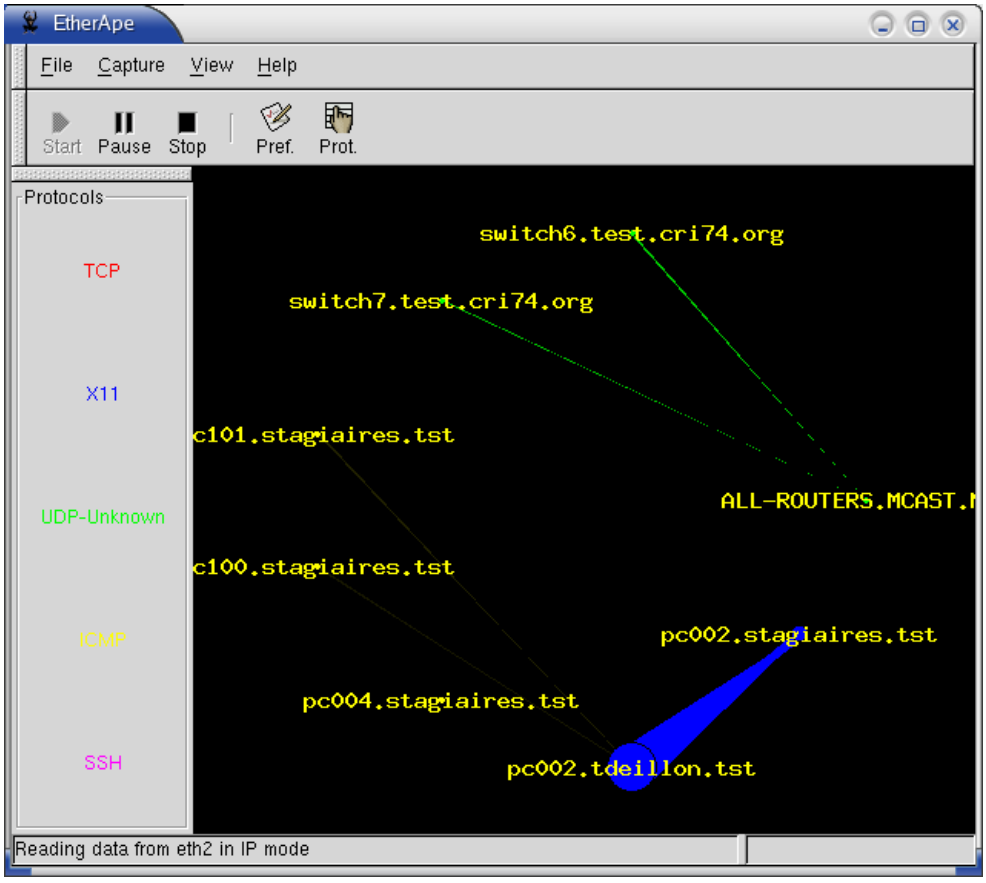
```
ssh (version 2 or windows or masqueraded) traffic between 10.100.20.2 and 10.100.20.100
dns traffic between 10.100.20.100 and 195.202.0.99
netbios-ns (udp) traffic between 10.100.105.255 and 10.100.105.9
Unknown tcp traffic between 10.100.20.100:1375 and 10.100.20.2:6000
Unknown tcp traffic between 10.100.20.2:6000 and 10.100.20.100:1375
Unknown tcp traffic between 10.100.20.2:6000 and 10.100.20.100:1376
Unknown tcp traffic between 10.100.20.2:6000 and 10.100.20.100:1377
Unknown tcp traffic between 10.100.20.2:6000 and 10.100.20.100:1378
netbios-ns (udp) traffic between 10.10.5.255 and 10.10.5.51
http traffic between 10.100.20.100 and 216.239.53.99
ftp traffic between 10.100.20.100 and 193.48.132.252
netbios-ns (udp) traffic between 10.100.105.255 and 10.100.105.115
```

15.2. Utilisation recommandée

Aide à la cartographie des flux, permet de synthétiser les flux d'un réseau. Sert à savoir les ports utilisés pour la création d'un FireWall.

[Retour au sommaire]

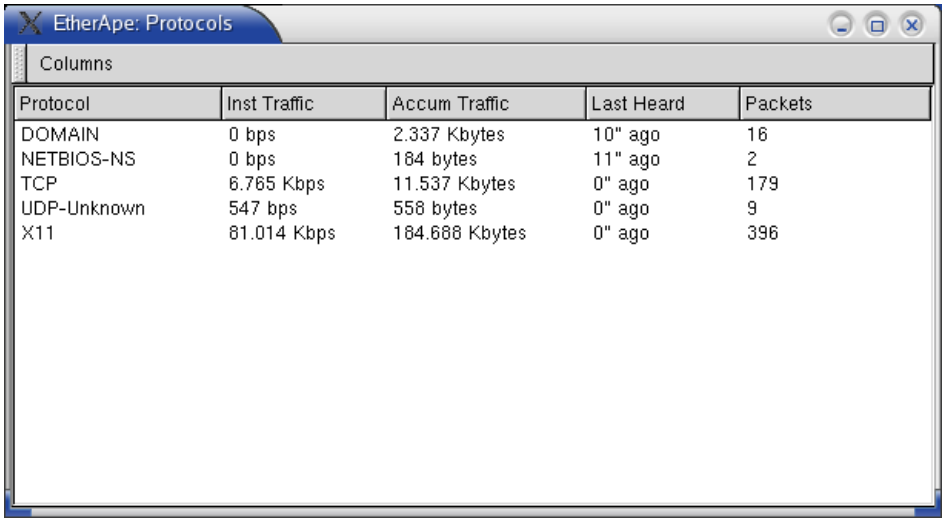
16. ETHERAPE



Logiel qui permet de voir l'utilisation du réseau en temps réel. ***Attention***, c'est ce que voit le PC sur lequel le programme est lancé. (attention donc au switch).

16.1. Utilisation

L'option view-> protocols permet de voir les protocoles utilisés et le débit en temps réel.



16.2. Utilisation recommandée

Pour se donner une idée de l'utilisation du réseau en temps reél.

[Retour au sommaire]

17. NETPERF

Logiciel permettant de faire des statistiques sur la vitesse du réseau.

17.1. Utilisation

Sur le serveur, vous lancez la commande :

```
netserver -p 12866
```

Sur le client vous lancez la commande:

- netperf -l 60 -H 10.100.20.101 -t UDP_STREAM
- netperf -l 60 -H 10.100.20.101 -t TCP_STREAM

- l'option -l est pour la durée en seconde: ici 60 secondes.
- l'option -H est pour l'adresse IP du serveur: ici 10.100.20.101
- l'otption -t est pour spécifier le type de trame à tester.
- l'option -p est pour définir le port sur lequel on fait le test (port du serveur)

17.2. Utilisation recommandée

voir le ralentissement d'un élément sur le réseau ou tout simplement pour voir la qualité du réseau.

[Retour au sommaire]

18. NETPIPE

Sert à faire des statistiques de vitesse par rapport à la taille des paquets.

18.1. Utilisation

Attention pour l'utilisation: il faut utiliser la commande Nptcp.

- Chez client: NPtcp -t -h IP_serveur -P -o ouptup_file
- Chez serveur: NPtcp -r

Pour exploiter les résultats, utiliser Gnuplot qui sert à faire des graphiques.

Une fois dans Gnuplot, tapez :

```
plot "try1" using 4:2 with linespoints
```

ce qui permettra de prendre la 4ème ligne pour l'axe des X et la 2ème ligne pour l'axe des Y avec les traits entre les points.

[Retour au sommaire]

19. Tableau Recapitulatif des meilleurs logiciels

Utilisation	Programmes
Enregitrement dans un fichier d'information circulant sur le réseau ou analyse en live	TcpDump
Analyse de trame (pas en live)	Wireshark (ex-Ethereal)
Recherche d'informations précises (surtout chaine de caracteres) dans des trames	Ngrep
Pour tout ce qui touche à clients/serveurs et ports	Netcat
Statistiques complets sur le traffic réseaux (en temps réel)	NTop
Statisques rapides sur le traffic (en temps réel)	IPTraf
Construire le plan du réseau	Cheops
Pour rechercher les failles d'une ou plusieurs machines	Nessus
Visualise en temps réel l'utilisation du réseau + statistiques rapides	Etherape
Pour le calcule de vitesse sur le réseau (ex: pour comparer avec ou sans un fireWall)	NetPerf
Pour calcul de vitesse suivant la taille des paquets envoyés (ex d'utilisation: comparer avec ou sans un firewall)	NetPipe

[Retour au sommaire]

Document généré avec les cri-doctools