Introduction à l'analyse réseau

Philippe Latu
philippe.latu(at)inetdoc.net

https://www.inetdoc.net

Résumé

L'analyseur de trafic est un outil pédagogique essentiel pour comprendre les mécanismes de fonctionnement des protocoles de communication sur les réseaux contemporains. Ce document comprend deux parties. Dans un premier temps, on trouve une introduction à l'utilisation de l'analyseur Wireshark. Dans un deuxième temps, les travaux pratiques permettent de découvrir l'organisation des informations fournies par cet analyseur.

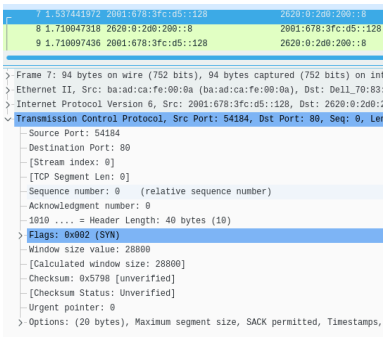


Table des matières

1. Copyright et Licence

Méta-information

Cet article est écrit avec DocBook XML sur un système Debian GNU/Linux. Il est disponible en version imprimable au format PDF : `intro.analyse.pdf`.

2. Capture de trafic avec Wireshark

Avec Wireshark, il est possible de capturer des paquets directement sur les interfaces du système utilisé ou de lire des fichiers de captures sauvegardées. Wireshark supporte les formats de fichiers de capture les plus courants.

Quels sont les protocoles supportés ?

La liste des protocoles supportés par Wireshark évolue de façon continue depuis de nombreuse années. On peut accéder au catalogue soit en consultant la page Protocol Reference qui fournit un classement par famille de tous les protocoles dont les champs sont interprétés, soit via le menu Help → Supported Protocols.

Quels sont les médias supportés ?

Le logiciel Wireshark permet l'analyse des captures réseau de presque toutes les technologies. Les limitations sur les captures sont plutôt dues au système d'exploitation sur lequel on réalise ces captures. Pour obtenir un état des possibilités d'analyse en fonction du système utilisé, il faut consulter la page Network media specific capturing.

Comment accéder aux interfaces ?

Lorsque l'on exécute wireshark en tant qu'utilisateur normal, on ne peut accéder à la liste des interfaces en lançant l'opération Capture. Sur un système d'exploitation correctement administré, un utilisateur normal ne doit pas avoir accès aux interfaces sans conditions. Il existe plusieurs solutions pour donner un accès direct à la liste des interfaces physiques. En voici trois classées par ordre de préférence :

En mode utilisateur avec les paramètres des paquets de la distribution

Les paquets de la distribution Debian GNU/Linux intègrent la délégation des droits de capture de paquets. Pour l'activer, il suffit de reconfigurer le paquet wireshark-common.

```
$ sudo dpkg-reconfigure wireshark-common
```

L'utilisateur doit appartenir au groupe système `wireshark` pour bénéficier de la fonctionnalité. Par exemple, l'ajout de l'utilisateur `etu` au groupe via la commande adduser donne le résultat suivant :

```
$ sudo adduser etu wireshark
Ajout de l'utilisateur « etu » au groupe « wireshark »...
Ajout de l'utilisateur etu au groupe wireshark
Fait.
```

Lors de la connexion suivante avec ce compte utilisateur il sera possible d'utiliser directement les outils wireshark ou tshark.

En mode utilisateur via les Linux Capabilities

On débute par la création d'un groupe système dédié à la capture de trafic réseau.

```
$ sudo addgroup --system pcap
Adding group `pcap' (GID 136) ...
Done.
```

On ajoute un ou plusieurs utilisateur(s) au groupe système.

```
$ sudo adduser phil pcap
Adding user `phil' to group `pcap' ...
Adding user phil to group pcap
Done.
```

Avertissement

Attention ! Cette nouvelle attribution n'est valable qu'après une nouvelle authentification. Nous sommes encore dans le cas classique de création du contexte de travail utilisateur au moment de l'authentification.

On modifie les propriétés du programme dumpcap qui est chargé de la collecte du trafic réseau.

Avant modification du groupe propriétaire, le masque des permissions est le suivant :

```
$ ls -lh `which dumpcap`
-rwxr-xr-- 1 root root 62K  4 mars  18:04 /usr/bin/dumpcap
```

On change le groupe propriétaire et on applique un nouveau masque de permissions. Une fois cette opération faite, les membres du groupe système `pcap` seront les seuls utilisateurs à pouvoir exécuter le programme en mode non privilégié.

```
$ sudo chgrp pcap /usr/bin/dumpcap
$ sudo chmod 750 /usr/bin/dumpcap
$ ls -lh /usr/bin/dumpcap
-rwxr-x--- 1 root pcap 62K  4 mars  18:04 /usr/bin/dumpcap
```

On indique au gestionnaire de paquets Debian que ces nouvelles propriétés doivent être conservées lors des mises à jour à venir.

```
$ sudo dpkg-statoverride --add root pcap 750 /usr/bin/dumpcap
# dpkg-statoverride --list /usr/bin/dumpcap
root pcap 750 /usr/bin/dumpcap
```

On modifie le contexte de travail du programme `dumpcap`.

```
$ sudo setcap cap_net_raw,cap_net_admin=eip /usr/bin/dumpcap
$ sudo getcap /usr/bin/dumpcap
/usr/bin/dumpcap = cap_net_admin,cap_net_raw+eip
```

Les bits `eip` correspondent aux attributs effective, inheritable et permitted.

Avec l'attribut effective, le noyau ne vérifie pas si l'UID vaut 0 (mode privilégié) si le programme nécessite une opération en mode privilégié.

L'attribut inheritable transmet les aptitudes du processus actuel aux autres processus enfants.

L'attribut permitted indique que le processus peut utiliser les aptitudes étendues du noyau Linux.

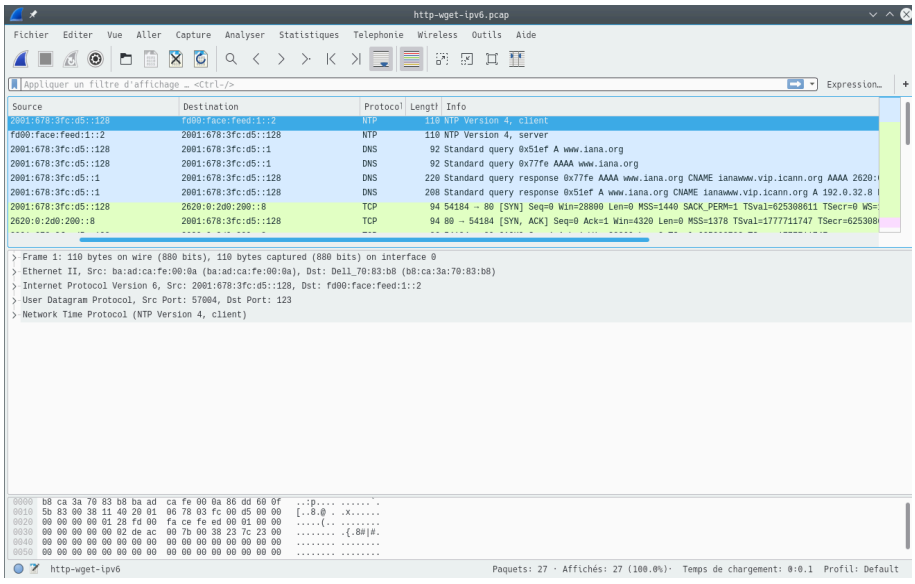
La documentation sur les Linux Capabilities est disponible à partir de la page Not needing root to administer Linux.

En mode super-utilisateur avec une interface graphique

Partant d'une connexion avec un compte utilisateur normal, celui-ci est propriétaire exclusif de son écran (display). Il doit donc autoriser le super utilisateur à accéder à son écran à l'aide de la commande xhost, passer en connexion super-utilisateur avec la commande su puis exécuter l'application Wireshark.

```
$ xhost +local:
$ su -
Password:
# wireshark &
```

3. Interface utilisateur



Écran complet WireShark

L'interface de l'analyseur se décompose en plusieurs barres ou fenêtres :

Barre de menus

On y retrouve la liste classique de menus. Voici une liste des fonctions remarquables accessibles à partir de ces menus.

- Le menu File sert à sauvegarder ou charger un fichier de capture réseau. Une capture peut très bien avoir été réalisée sur une sonde distante ou avec un autre outil et être analysée avec Wireshark à postériori.
- Le menu Capture sert à fixer les paramètres d'une nouvelle capture réseau. Voir Section 4, « Capture d'une série de trame ».
- Le menu Statistics sert à effectuer différents calculs sur les volumes de données et la répartition des protocoles.

Barre des icônes

Cette barre regroupe tous les raccourcis sur les manipulations d'une capture.

Barre de filtrage

Cette barre sert à saisir l'expression de filtrage à postériori d'une capture pour isoler tout ou partie d'un échange réseau.

Fenêtre contenant la liste des trames capturées

Sur chaque ligne on retrouve :

- le numéro du paquet,
- son temps de capture,
- sa source,
- sa destination,
- le protocole de plus haut niveau décodé,
- le résumé des champs caractéristiques de ce protocole.

Fenêtre d'affichage de la pile des protocoles décodés pour la trame sélectionnée

Avant toute opération de développement des champs d'un ou plusieurs protocoles, cette fenêtre donne la liste la pile de protocoles décodés allant du niveau physique (en haut) jusqu'au niveau le plus haut reconnu (en bas). Le protocole de niveau le plus haut reconnu apparaît est celui qui apparaît dans la colonne protocole de la Fenêtre contenant la liste des trames capturées.

- La première ligne ou niveau Frame correspond à une pseudo couche physique. Comme il n'est pas possible de réaliser la capture directement à partir des composants électroniques qui pilotent l'interface réseau sans perturber le fonctionnement du système, l'opération a lieu au niveau liaison à l'aide de la bibliothèque libpcap.

  A ce niveau, les informations disponibles sont : la quantité de bits capturés et la date de capture.
- La deuxième ligne correspond au niveau liaison. On y détaille le type et les champs de la trame et les adresses physiques.
- La troisième ligne correspond au niveau réseau. On y détaille les champs du protocole réseau reconnu : adresses logiques et indicateurs d'état.
- La quatrième ligne correspond au niveau transport. On y détaille les champs du protocole de transport reconnu : état de la connexion, numéros de ports utilisés et diverses options.
- La cinquième ligne correspond au niveau application. On y trouve les données utilisateur.

Pour le développement de chacun des champs de la trame, il faut cliquer sur le triangle situé à gauche au niveau de chaque couche.

Fenêtre d'affichage brut de la trame sélectionnée

Cette fenêtre affiche tous les octets de la trame en hexadécimal.

4. Capture d'une série de trame

Après avoir lancé le logiciel Wireshark, la séquence suivante illustre la capture d'une série de 60 trames :

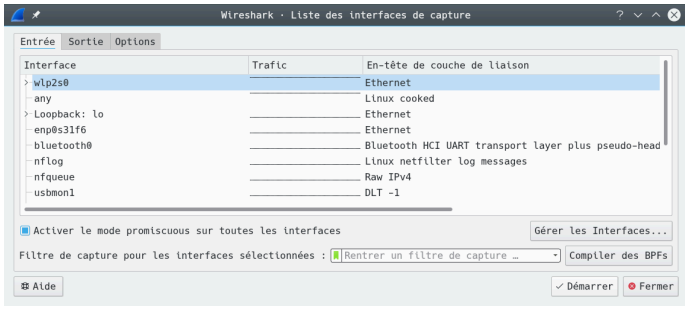
1. Sélectionner Capture puis Options.

2. La ligne Filtre de capture pour les interfaces sélectionnées permet de préciser un filtrage *à priori*. La syntaxe de ce filtrage est identique à celle de la commande tcpdump. La documentation est disponible à partir des pages de manuels de cette commande : `man tcpdump`. Voici 3 exemples :

   - `ip` : en spécifiant le protocole réseau à analyser, on évite la capture des trames des autres protocoles des niveaux réseau et liaison.
   - `host 192.168.0.1` : en spécifiant l'adresse IP d'un hôte, on ne retient que le trafic émis et reçu par cette adresse.
   - `host 192.168.0.1 and host 10.0.0.1` : en spécifiant les adresses IP de 2 hôtes, on ne retient que le trafic entre ces 2 adresses.

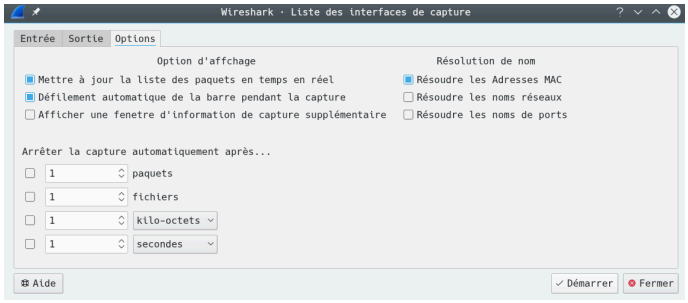
   D'une façon plus générale, on peut combiner plusieurs critères avec les opérateurs logiques `and` et|ou `or`.

   - le type : `host`, `net` et `port`.
   - la direction : `src` et `dst`.
   - le protocole : `ether`, `fddi`, `tr`, `ip`, `ip6`, `arp`, `rarp`, `decnet`, `tcp` et `udp`.



Capture : choix de l'interface et filtrage avant capture

3. La rubrique Options permet de fixer plusieurs critères d'arrêt en fonction du nombre de trames et| ou du volume de données capturées.



Capture : choix des options d'arrêt de capture

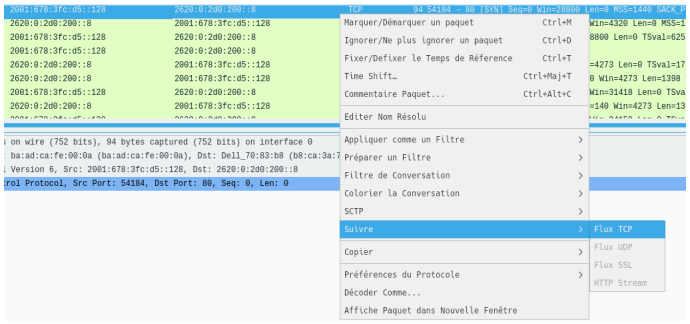
4. Clicker sur le bouton Démarrer pour lancer la capture.

5. Filtrage de l'affichage après capture

Le filtrage *à postériori* permet d'isoler l'information pertinente. La granularité de la syntaxe de filtrage disponible avec Wireshark est très importante. Voici deux exemples assez simples.

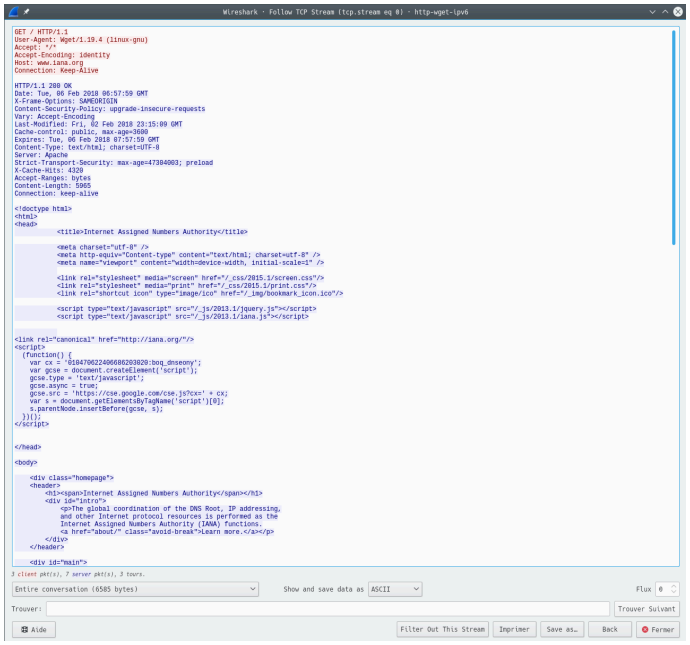
Isoler une connexion TCP

Après avoir réalisé une capture, il est possible d'isoler une connexion TCP en repérant, soit la phase d'établissement de connexion, soit la phase de libération de connexion. En cliquant sur le bouton droit de la souris après avoir sélectionné n'importe quelle trame appartenant à la connexion à isoler, il faut valider l'option Suivre puis l'option Flux TCP.



Isoler une connexion TCP - vue complète

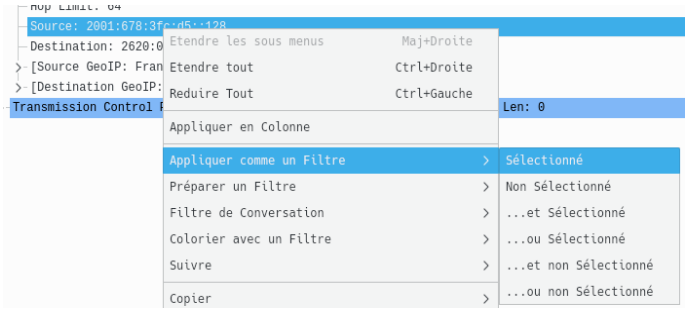
A la suite de cette opération, Wireshark ouvre une nouvelle fenêtre contenant les données vues de la couche transport.



Données vues de la couche transport - vue complète

Syntaxe du filtrage après capture

Il est possible d'utiliser le champ Appliquer un filtre d'affichage pour composer un filtre «à façon». Le champ en question peut être rempli manuellement ou de façon interactive en sélectionnant à la souris un élément d'en-tête de protocole. Voici une copie d'écran qui montre comment sélectionner une adresse et un exemple de syntaxe de filtre.



Saisie interactive d'un filtre d'affichage - vue complète

```
ipv6.addr❶ == 2001:678:3fc:d5::128❷ &&❸ ipv6.addr == 2620:0:2d0:200::8 && tcp.dstport❹ == 80
```

Cette expression est extraite de la Barre de filtrage. Elle tient sur une ligne unique.

❶ `ip.addr` ou `ipv6.addr` : sélection d'une adresse IPv4 ou IPv6.

❷ Exemple d'adresse IP. Les opérateurs utilisables reprennent la syntaxe du langage C pour les tests.

- égalité : `==`
- différence : `!=`
- supérieur ou égal : `>=`
- inférieur ou égal : `<=`

❸ Les opérateurs logiques utilisent eux aussi la la syntaxe du langage C. On peut ajouter des paranthèses pour gérer les priorités.

- et : `&&`
- ou : `||`
- complément : `!`

❹ `tcp.dstport` : sélection d'un numéro de port destination pour le protocole TCP de la couche transport.

La documentation sur l'ensemble des champs des protocoles reconnus utilisables dans les expressions de filtres d'affichage est disponible à l'adresse : Display Filter Reference.

6. Analyse à distance

Lorsque l'on exploite une infrastructure de serveurs avec plusieurs périmètres réseau cloisonnés, il est fréquent de devoir procéder à des captures réseau à distance. De plus, la plupart des serveurs récents sont des lames qui n'ont ni clavier ni écran. Voici donc un exemple de scénario capture réseau réalisée sur un hôte distant exploitée ensuite sur un poste de travail ayant une interface graphique.

Dans la suite de copies d'écran suivante, on considère les éléments suivants :

- Le poste de travail sur lequel l'analyse est effectuée en mode graphique après collecte du fichier de capture est appelé `<my_laptop.myothernet>`.
- Le serveur lame sans écran ni clavier sur lequel la capture réseau est réalisée est appelé `<my_distant_server.mynet>`. On y accède via une console sécurisée SSH.
- On suppose que les deux hôtes ont un compte utilisateur `me`. Le compte utilisateur sur le serveur doit disposer des droits nécessaire à la capture de trames sur les interfaces réseau du serveur. Ces droits sont gérés avec sudo.
- On utilise l'application tshark qui permet d'exécuter l'analyse réseau directement à la console sans recours à une interface graphique. Cette application est fournie par le paquet Debian du même nom. Voir le résultat de la commande `$ apt-cache show tshark` pour obtenir les informations sur ce paquet.

- Les indications données ci-dessous ne peuvent se substituer aux pages de manuels de l'application. Il est vivement conseillé de les consulter pour adapter l'analyse réseau à ses besoins : `man tshark`.

Connexion au serveur depuis le poste de travail

Comme indiqué ci-avant, on accède au serveur via une console sécurisée SSH. À partir le Windoze, l'outil `putty` permet d'effectuer la même opération.

```
me@<my_laptop>:~$ ssh me@<my_distant_server.mynet>

Linux <my_distant_server> 2.6.15 #1 SMP Mon Mar 13 14:54:19 CET 2006 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
No mail.
Last login: Tue Mar 21 10:45:38 2006 from <my_laptop.myothernet>

me@<my_distant_server>:~$
```

Lancement de la capture réseau dans un nouveau shell

Un utilisateur «normal» n'ayant pas les droits suffisants pour accéder directement aux interfaces réseau, on doit lancer l'analyseur de réseau via sudo : `$ sudo tshark`.

On lance cette commande dans un nouveau shell en ajoutant le symbole `&` à la fin de la ligne. De cette façon, on conserve la possibilité de lancer d'autres commandes sur la console obtenue lors de la connexion au serveur.

```
me@<my_distant_server>:~$ sudo tshark -q -i _eth0 -w distant.pcap \
  -a filesize:4096 tcp and ! host <my_laptop.myothernet> &
```

- L'option `-q` rend la capture «silencieuse». Il s'agit surtout de supprimer l'affichage du compte des paquets enregistrés pendant la capture. Cet affichage est gênant si l'on souhaite conserver la console pour effectuer d'autres manipulations en cours de capture.
- L'option `-i _eth0` désigne l'interface réseau sur laquelle la capture est réalisée.
- L'option `-w distant.pcap` désigne le fichier dans lequel les paquets capturés sont enregistrés. Sans spécification du format de fichier avec l'option `-F`, les paquets capturés sont enregistrés directement (mode raw).
- L'option `-a filesize:4096` donne le critère d'arrêt de l'enregistrement. Ici, le critère retenu est la taille du fichier de capture. Cette taille est comptabilisée en multiple du kilooctet (1024 octets) ; soit 4096ko dans cet exemple.
- Les options suivantes correspondent au filtrage à priori des paquets à enregistrer. On spécifie le protocole de transport `tcp` ***et*** on n'enregistre pas les paquets de l'hôte qui à ouvert la console sécurisée : `! host <my_laptop.myothernet>`. Sans cette dernière précaution, l'enregistrement ne contiendra pratiquement que les échanges SSH. Ces échanges sont sans intérêt puisqu'ils correspondent aux communications entre les deux hôtes utilisés pour l'analyse distante.

«Initiation» du trafic réseau à capturer.

Cette commande n'est qu'un prétexte pour remplir le fichier de capture. Avec le téléchargement d'une image des sources du noyau Linux, on est sûr de faire transiter un volume suffisant ;-).

```
me@<my_distant_server>:~$ wget \
  http://kernel.org/pub/linux/kernel/v2.6/linux-2.6.16.tar.bz2
--11:14:29--  http://kernel.org/pub/linux/kernel/v2.6/linux-2.6.16.tar.bz2
           => `linux-2.6.16.tar.bz2'
Résolution de kernel.org... 204.152.191.5, 204.152.191.37
Connexion vers kernel.org|204.152.191.5|:80...connecté.
requête HTTP transmise, en attente de la réponse...200 OK
Longueur: 40 845 005 (39M) [application/x-bzip2]

100%[===============//====================>] 40 845 005   296.19K/s    ETA 00:00

11:16:58 (292.09 KB/s) - « linux-2.6.16.tar.bz2 » sauvegardé [40845005/40845005]
```

Fin de la capture et visualisation du fichier

Comme indiqué ci-avant, l'enregistrement s'arrête lorsque le fichier atteint la taille de 4096ko.

```
[1]+  Done  sudo tshark -q -i _eth0 -w distant.pcap \
                -a filesize:4096 tcp and ! host <my_laptop.myothernet>

me@<my_distant_server>:~$ ls -lAh
-rw-------  1 root latu   4,1M 2006-03-21 11:14 distant.pcap
-rw-r--r--  1 latu latu    39M 2006-03-20 07:22 linux-2.6.16.tar.bz2

me@<my_distant_server>:~$ sudo chmod 640 distant.pcap

me@<my_distant_server>:~$ exit
logout
Connection to <my_distant_server.mynet> closed.
```

L'enregistrement sur fichier ayant été réalisé avec l'identité du super-utilisateur via la commande sudo, il faut changer le masque des permissions de ce fichier ou son propriétaire. Dans cet exemple, c'est le masque des permissions d'accès qui a été étendu pour que l'utilisateur normal puisse lire le fichier de capture et le transférer sur son poste de travail.

Récupération du fichier de capture sur le poste de travail

```
me@<my_laptop>:~$ scp me@<my_distant_server.mynet>:~/distant.pcap .
distant.pcap                                  100% 4097KB 682.8KB/s   00:06

me@<my_laptop>:~$ wireshark -r distant.pcap
```

La commande scp illustre le transfert du fichier de capture réseau via SSH. On peut effectuer la même opération à partir de Windoze avec l'outil `WinSCP`.

Enfin, il est possible de lire le fichier de capture directement au lancement de l'analyseur réseau avec l'option `-r`.

7. Travaux pratiques : messages de contrôle internet (ICMP)

Protocoles et outils étudiés

- Internet Control Message Protocol ou ICMP ; messages de type : `Echo`, `Echo Reply` et `Time Exceeded`.
- Internet Protocol ou IP ; champ de l'en-tête IP : `Time to Live`.
- Commande ping.
- Commande traceroute.

Marche à suivre

Commande ping

1. Lancer Wireshark.
2. Lancer la capture des trames sans restrictions d'adresses, de protocoles ou de volume.
3. Lancer une console et taper une commande du type `ping -c10 www.phrack.org`. L'option `-c10` limite le nombre de requêtes ICMP à 10. Bien sûr, le choix de l'adresse à contacter est totalement libre.
4. Arrêter la capture lorsque l'invite de commande réapparaît à la console.
5. Sauvegarder le fichier de capture.

Commande traceroute

1. Lancer Wireshark.
2. Lancer la capture des trames sans restrictions d'adresses, de protocoles ou de volume.
3. Lancer une console et taper une commande du type `traceroute www.phrack.org`. Bien sûr, le choix de l'adresse à contacter est totalement libre.
4. Arrêter la capture lorsque l'invite de commande réapparaît à la console.
5. Sauvegarder le fichier de capture.

La plage de ports UDP utilisée par défaut par la commande traceroute est de plus en plus fréquemment bloquée par les équipements d'interconnexion.

Analyse avec ping

Pour répondre aux questions suivantes, utiliser le résultat de la capture issue de l'étape précédente ou charger un fichier de capture.

Protocoles capturés

Q1. Quels sont les protocoles indiqués dans la colonne Protocol de la fenêtre de liste des trames capturées ?

Il est probable que les paquets ICMP soient précédés d'un jeu de question/réponse DNS.

Q2. Relever l'adresse IP renvoyée avec la réponse DNS.

Message ICMP «Echo Request»

Étude du paquet IP qui correspond au premier message ICMP `Echo Request`.

Q3. Quelle est l'adresse IP destination du paquet ? Quelle est la valeur du champ `Protocol Type` ? Quelle est la valeur du champ `Time to Live` ?

Étude du message ICMP.

Q4. Quel est le type de message ICMP ? Quel est l'identificateur de message ? Quel est le numéro de séquence ?

Q5. Sélectionner à la souris les octets de données du message de requête. Comparer ces données avec celles affichées dans la fenêtre d'affichage brut.

Message ICMP «Echo Reply»

Étude du paquet IP qui correspond au premier message ICMP `Echo Reply`.

Q6. Quelle sont les adresses IP source et destination du paquet ? Quelle est la valeur du champ `Protocol Type` ? Quelle est la valeur du champ `Time to Live` ?

Étude du message ICMP.

Q7. Quel est le type de message ICMP ? Comparer l'identificateur de message et le numéro de séquence du message de réponse avec les valeurs du message de requête.

Q8. Sélectionner à la souris les octets de données du message de requête. Comparer ces données avec celles affichées dans le message de requête.

Messages ICMP restants

Reprendre les 2 points précédents pour les messages ICMP `Echo Request` et `Echo Reply` restants.

Q9. Comment les champs d'identification et de numéro de séquence évoluent dans le temps ?

Q10. Est-ce que les séquences de données des requêtes et des réponses changent ?

Q11. Calculer l'écart de temps entre l'émission de chaque message `Echo Request` et la réception de chaque message `Echo Reply`. Comparer les résultats avec les valeurs maximum, moyenne et minium fournies par la commande ping.

Analyse avec traceroute

Pour répondre aux questions suivantes, utiliser le résultat de la capture issue de l'étape précédente ou charger un fichier de capture.

Protocoles capturés

Q12. Quels sont les protocoles indiqués dans la colonne Protocol de la fenêtre de liste des trames capturées ?

Il est probable que les paquets ICMP soient précédés d'un jeu de question/réponse DNS.

Q13. Relever l'adresse IP renvoyée avec la réponse DNS.

Message UDP

Q14. Quelle est l'adresse IP destination du premier paquet contenant le message UDP ? Quelles sont les valeurs des champs `Protocol Type` et `Time to Live` ?

Comparer l'adresse IP destination relevée avec celle de la réponse DNS. Noter les valeurs caractéristiques de l'en-tête IP en vue d'une utilisation ultérieure.

Q15. Combien d'octets de données sont présents dans ce message de requête ?

Noter la séquence de caractères présente dans la troisième fenêtre.

Message ICMP «Time Exceeded»

Q16. Quelles sont les adresses IP source et destination du paquet de la première réponse ICMP `Time Exceeded` ?

Étude du message ICMP.

Q17. Quel est le type de message ICMP ?

Les champs `Type`, `Code` et `Checksum` sont suivis par plusieurs octets à zéro puis par l'en-tête IP du message ICMP `Echo Request`. Comparer les valeurs caractéristiques de cet en-tête avec celles notées ci-avant.

Q18. Est-ce que le message ICMP contient de nouveaux octets de données ?

Evolution du champ TTL

Q19. Combien de messages UDP sont émis avec la même valeur de champ `TTL` dans l'en-tête de paquet IP ?

Q20. Quelles sont les adresses IP source des paquets ICMP `Time Exceeded` ?

Comparer ces adresses avec celles données lors de l'exécution de la commande traceroute.

Q21. Quel est le type du message ICMP reçu lorsque l'hôte destinataire est atteint ?

Q22. Comment calculer les temps affichés par la commande traceroute à partir des valeurs données dans la colonne `Time` de la fenêtre des trames capturées ?

Utiliser les pages de manuels de la commande traceroute pour obtenir la signification des différentes valeurs de temps pour atteindre une destination.

8. Travaux pratiques : consulter une page Web (HTTP)

Protocoles & mécanismes d'adressage étudiés

- Les adressages matériel (MAC ou Ethernet) et logique (IPv4 ou IPv6) servent à identifier les hôtes client et serveur pour chaque service utilisé.
- La requête et les réponses du service de noms de domaines (DNS) permettent de faire la correspondance entre le nom et l'adresse IP du serveur consulté.
- Le chargement de la page Web illustre le fonctionnement du protocole TCP avec les phases d'établissement, de maintien et de libération d'une connexion. On étudie aussi la notion de numéro de séquence utilisée pour l'acquittement des quantités d'octets transmises entre client et serveur.
- Les requêtes et réponses HTTP illustrent dialogue entre le navigateur et le serveur Web au niveau application.

Marche à suivre pour réaliser une capture

1. Lancer Wireshark.
2. Lancer la capture des trames sans restrictions d'adresses, de protocoles ou de volume.
3. Lancer un navigateur Web et saisir une adresse de site (URL) de votre choix.
4. Une fois la page complètement chargée, arrêter la capture. Sauvegarder un fichier de capture.
5. Passer aux questions de la section suivante.

Suivant le contexte d'accès à l'Internet, le volume d'information capturé varie énormément en fonction du raccordement à un réseau local filaire ou radio et à la variété des protocoles présents. Il est cependant préférable d'effectuer la première capture sans aucune restriction *à priori* de façon à avoir une image exacte du trafic. Si l'information utile est vraiment noyée dans du «bruit», il est toujours possible de reprendre la capture avec un filtre ; voir Section 4, « Capture d'une série de trame ».

À titre d'exemple, voici deux fichiers de capture qui contiennent tous les éléments nécessaires au traitement des questions suivantes.

- Capture IPv4 : `http-wget-iana-ipv4.pcap`
- Capture IPv6 : `http-wget-iana-ipv6.pcap`

Questions sur le chargement d'une page Web

Pour répondre aux questions suivantes, utiliser le résultat de la capture issue de l'étape précédente ou charger un fichier de capture.

Identifier les protocoles capturés

Q23. Quelle est la liste des protocoles indiqués dans la colonne Protocol de la fenêtre de liste des trames capturées ?

Confirmer que la capture contient bien la séquences des protocoles DNS, TCP et HTTP.

Identifier les rôles à partir des adresses

Analyser la trame correspondant au premier message de requête DNS émis par le client Web.

Q24. Quelles sont les adresses (MAC|Ethernet) et IP du client ?

Q25. Quel est le contenu du champ type de la trame Ethernet ?

Q26. Quelles sont les adresses destination (MAC|Ethernet) et IP ?

Q27. À quelles machines correspondent ces adresses ?

Repérer les informations de l'en-tête IP

Analyser l'en-tête IP du premier message de requête DNS émis par le client Web.

Q28. Quelle est la taille de l'en-tête ? Quelle est la longueur totale du paquet ?

Q29. Repérer le champ «type de protocole» dans l'en-tête. Quel est le type de protocole de la couche transport présent dans les données du paquet ?

Repérer les informations de l'en-tête UDP

Analyser l'en-tête UDP du premier message DNS émis par le client Web.

Q30. Quels sont les numéros de ports du client et du serveur ? Quelles sont les particularités de ces valeurs ? Quel est le protocole de couche application présent dans les données du message ?

Q31. Quelle est la valeur indiquée dans le champ longueur de l'en-tête UDP ? Est-ce qu'elle correspond à l'information donnée dans l'en-tête du paquet IP ?

Reconnaître la requête posée par le client DNS

Analyser le message de requête DNS émis par le client Web.

Q32. Quel est le champ qui indique si le message est une requête ou une réponse ?

Q33. Quelle est l'information présente dans le corps de la requête ? Identifier le type et la classe de la requête.

Q34. Quel est l'identificateur de transaction de la requête ?

Caractériser la réponse du serveur DNS

On considère maintenant la réponse à la requête précédente.

Q35. Quelles sont les adresses (MAC|Ethernet) et IP de la réponse DNS ?

Vérifier que les adresses attendues sont présentes.

Q36. Quel est le nombre d'octets contenus dans les données du paquet IP ? Pourquoi la quantité de données est-elle plus importante que celle du paquet de requête ?

Q37. Quel est l'identificateur de transaction de la réponse ? Est-ce qu'il correspond à la requête ?

Q38. Combien de réponses sont disponibles dans le message de réponse ? Quelle est la signification des valeurs TTL (Time-to-live) ?

Q39. Pour synthétiser cette partie, faire un croquis des piles de protocoles des couches physique à application pour le client et le serveur DNS. Identifier les adresses, les numéros des protocoles présents dans les en-têtes et les unités de données (PDUs) bout en bout.

Caractériser l'établissement de la connexion TCP

Sur la fenêtre du haut de l'écran Wireshark, repérer la ligne qui correspond au premier segment TCP de l'établissement de connexion en trois étapes (three ways handshake) entre le client et le serveur HTTP.

Q40. Quels sont les hôtes identifiés par les adresses MAC et IP de cette ligne de capture ? Quelles sont les valeurs des champs `type` et `protocol` respectivement attendues pour cette trame et ce paquet ?

Vérifier que ces champs et adresses correspondent au rôle de chacun des deux hôtes en communication.

Q41. Quels sont les numéros de ports utilisés par le client et le serveur ? Quelle est la signification de ces deux valeurs ?

Q42. Quel est le numéro de séquence choisi par le client ? Quelle est la taille maximale de segment (Maximum Segment Size ou MSS) proposée par le client ?

Q43. Quelle est la signification de l'indicateur d'état SYN ?

Identifier la ligne de capture qui correspond au second segment TCP dans l'établissement de la connexion en trois étapes (three ways handshake).

Q44. Quel est le numéro de séquence choisi par le serveur ? Quelle est la taille maximale de segment (Maximum Segment Size ou MSS) renvoyée par le serveur ?

Q45. Quelle est la signification des indicateurs d'état SYN et ACK ?

Q46. Pourquoi le numéro de séquence du client a-t-il été incrémenté à 1 ?

Identifier la ligne de capture qui correspond au dernier segment TCP dans dans l'établissement de la connexion.

Q47. Quelle est la signification de l'indicateur d'état ACK ?

Q48. Que peut-on conlure sur l'état de la connexion entre le client et le serveur HTTP à partir des deux numéros de séquence ?

Caractériser les éléments de la requête HTTP GET

Identifier la ligne de capture qui correspond au message HTTP GET.

Q49. Quelles sont les valeurs des numéros de séquence et d'acquittement de l'en-tête TCP ?.

Vérifier que tout correspond aux valeurs attendues.

Q50. Quels sont les indicateurs d'état actifs de l'en-tête TCP ? Expliquer pourquoi.

Q51. Quelles sont les longueurs de l'en-tête et de la «charge» du message TCP ?

On considère maintenant le contenu du message HTTP GET.

Q52. Comparer le texte décodé dans la fenêtre d'affichage de la pile de protocoles avec le contenu de la fenêtre d'affichage brut.

Q53. Compter le nombre d'octets du message et vérifier que ce nombre correspond au champ longueur de l'en-tête TCP.

Q54. Quel est le prochain numéro de séquence attendu dans le message suivant émis par le serveur HTTP ?

Caractériser les éléments de la réponse HTTP

Q55. Déterminer si le serveur répond avec un message HTTP ou un segment TCP ACK ?

Q56. Quel est le numéro de séquence émis par le serveur HTTP ? Est-ce qu'il correspond à la valeur attendue ?

On considère maintenant l'en-tête du message réponse HTTP.

Q57. Quelle est la longueur de la «charge» indiquée dans l'en-tête TCP ?

Q58. Quels sont les indicateurs d'état actifs de l'en-tête TCP ? Expliquer pourquoi.

Q59. Quel est le prochain numéro de séquence attendu dans le message suivant émis par le client ?

On considère maintenant le corps du message réponse HTTP.

Q60. Quel est le code dans le message de réponse ?

Q61. Sélectionner ce code avec la souris dans la fenêtre d'affichage de la pile de protocoles et comparer avec ce qui est affiché sur la page du navigateur Web.

Cette opération revient à suivre la démarche présentée dans la la section intitulée « Isoler une connexion TCP ».

9. Documents de référence

Guide de l'utilisateur

Le Wireshark User's Guide est la référence la plus complète sur l'utilisation de notre analyseur de trafic favori !

Protocoles

- Le fichier PDF TCP/IP and tcpdump Pocket Reference Guide est une «antisèche» sur les champs des en-têtes des protocoles essentiels ; un document ***indispensable*** pour la pratique de l'analyse réseau.

Travaux pratiques

- Le support Configuration d'une interface de réseau local présente les outils de lecture de la configuration d'une interface réseau Ethernet.
- Le site Wireshark Labs présente les travaux pratiques proposés dans le livre Computer Networking: A Top-Down Approach, 7th ed..