Administration Réseaux

Pr. Abdelhak LAKHOUAJA

Département d'Informatique
Faculté des Sciences
Université Mohammed Premier
Oujda

SMI-S6

Année universitaire : 2018/2019

Support de cours

- Page web du cours :
  http://lakhouaja.oujda-nlp-team.net/teaching/bachlor-level/administration-reseaux/
- Page web personnelle :
  http://lakhouaja.oujda-nlp-team.net/

Chapitre 1
Introduction et rappels

But de l'administration système

- Sécurité et bon fonctionnement du réseau
- Mettre à la disposition des utilisateurs les outils nécessaires
- Veiller à la bonne conduite des utilisateurs

Administration d'un système d'exploitation

- L'utilisation des outils graphiques ne permet de comprendre le comportement des systèmes d'exploitation.
- L'utilisation des **scripts** (bash, perl, powershell (Windows), ...) permet de :
  - créer plusieurs comptes en même temps (voir TP1) ;
  - modifier et sauvegarder plusieurs fichiers ;
  - . . .

Utilisation du compte administrateur

- Pour éviter des erreurs graves, il est déconseillé d'utiliser le compte administrateur pour des tâches courantes.
- Sous Linux :
  - le compte administrateur est **root** et son invite est **#**
  - l'invite d'un utilisateur normal est **$**
  - pour certaines commandes, utilisez des alias :

```
alias mv='mv -i'
alias rm='rm -i'
alias cp='cp -i'
```

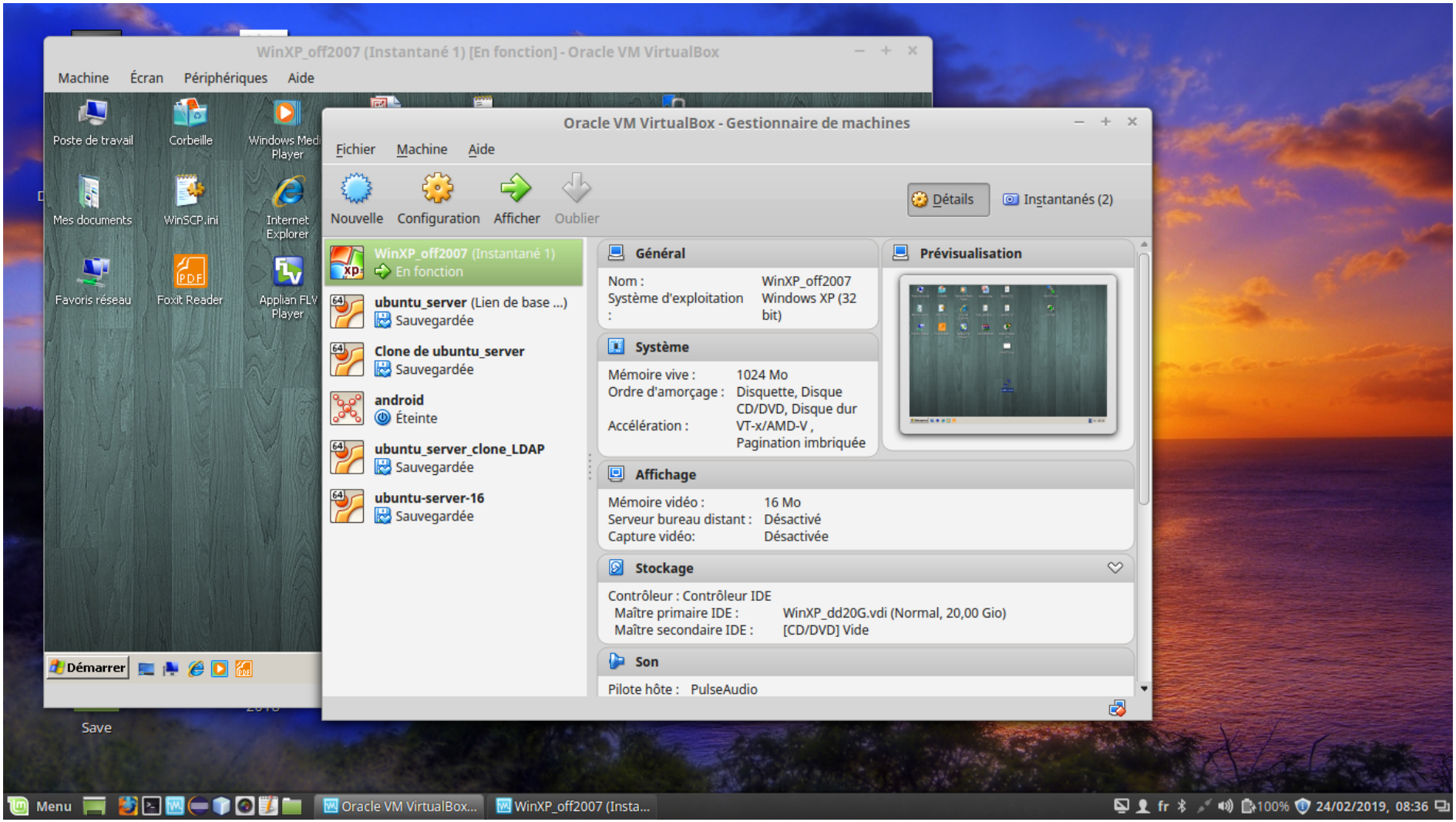
Virtualisation

La virtualisation est utilisé pour exécuter plusieurs systèmes d'exploitation en même temps sans avoir à redémarrer la machine. Elle permet par exemple d'exécuter un système Linux sur Windows ou inversement.

Elle permet de tester différents systèmes et d'exploitation. En cas de problème, la machine n'est pas affecté. Elle permet aussi de tester des tâches d'administration sur un système sur lequel on a des droits limités.

Il existe différentes technologies qui permettent la virtualisation : Xen, Qemu, KVM, Vmware (**www.vmware.com**), VirtualBox (**www.virtualbox.org**) etc.

VirtualBox (Windows XP qui tourne sous Linux)



NetKit

- Netkit est un projet Open-Source qui permet de simuler des réseaux informatiques. Il est basé sur des machines Linux.
- Voir TP.

Gestion des utilisateurs
Création d'un nouveau utilisateur

La commande :

`adduser smi`

permet de créer l'utilisateur **smi**.

Elle a pour effet :

- d'ajouter une ligne à la fin du fichier « /etc/passwd », semblable à : **`smi:x:1001:1001:,,,:/home/smi:/bin/bash`**
- de créer le répertoire : /**home**/**smi**.
- d'ajouter le groupe **smi** dans le fichier « /**etc**/**group** ».
- d'ajouter le mot de passe crypté dans le fichier « /**etc**/**shadow** ». Ce fichier est lisible seulement par root.

$\rightsquigarrow$ Dans un script, il est préférable d'utiliser la commande : **`useradd`** (voir TP).

Gestion des utilisateurs
Suppression utilisateur

- La commande :
  `userdel smi`
  permet de supprimer l'utilisateur **smi**.
- Options :
  - -r : supprime le dossier personnel de l'utilisateur (et son contenu !).
  - -f : forcer la suppression des fichiers de l'utilisateur.

Gestion des groupes

- La commande :
  `groupadd nom_groupe`
  permet d'ajouter un nouveau groupe.
- La commande :
  `groupdel nom_groupe`
  permet de supprimer un groupe.
- La commande :
  `adduser smi nom_groupe`
  permet d'ajouter l'utilisateur **smi** au groupe **nom_groupe**.
- La commande :
  `groups smi`
  permet d'afficher les groupes d'un utilisateur (informations données par la commande **id**)

Administration des utilisateurs

- **su** : pour devenir administrateur (root).
- **su smi** : pour devenir « **smi** ».
- **who** : affiche les utilisateurs connectés.
- **whoami** (qui suis-je ?) : affiche l'utilisateur en cours.
- **last** : affiche les utilisateurs qui se sont connectés au système.

Éteindre/redémarrer le système

- En mode graphique : dépend de la distribution.
- En mode console :
  - **halt** : éteint le système immédiatement (**shutdown -h now**).
  - **reboot** : redémarre le système immédiatement (**shutdown -r now**).
  - **shutdown -h 10** : Éteint le système dans 10 minutes. Les utilisateurs reçoivent un avertissement dans leurs consoles.
  - **shutdown -r 10** : redémarre le système dans 10 minutes.

Outils systèmes et réseaux « de base »

- **ifconfig/ipconfig** : pour configurer une interface réseau ou afficher les informations concernant les interfaces réseaux (commandes déjà utilisées en S5).
- **route :** pour afficher/modifier la table de routage.
- **ping :** pour tester la connexion entre deux machines.
- **host :** c'est un utilitaire simple de conversion de noms DNS en adresse IP (et vice versa) :
  Exemple :

```
alkhalil: $ host www.ump.ma
www.ump.ma has address 196.200.156.5
```

Outils systèmes et réseaux « de base »

- **nslookup :** disponible sous Linux et sous Windows, comme la commande **host**, cette commande permet d'afficher l'adresse IP d'un nom DNS.
  Exemple :

```
alkhalil:~$ nslookup www.ump.ma
Server: 127.0.1.1
Address: 127.0.1.1#53

Non-authoritative answer:
Name: www.ump.ma
Address: 196.200.156.5
```

- **netstat**
- **tcpdump/wireshark**

Netstat

netstat -s | -a | -r | -n

Netstat fournit des statistiques sur les :

- paquets émis ou reçus
- erreurs
- collisions
- protocoles utilisés

Netstat

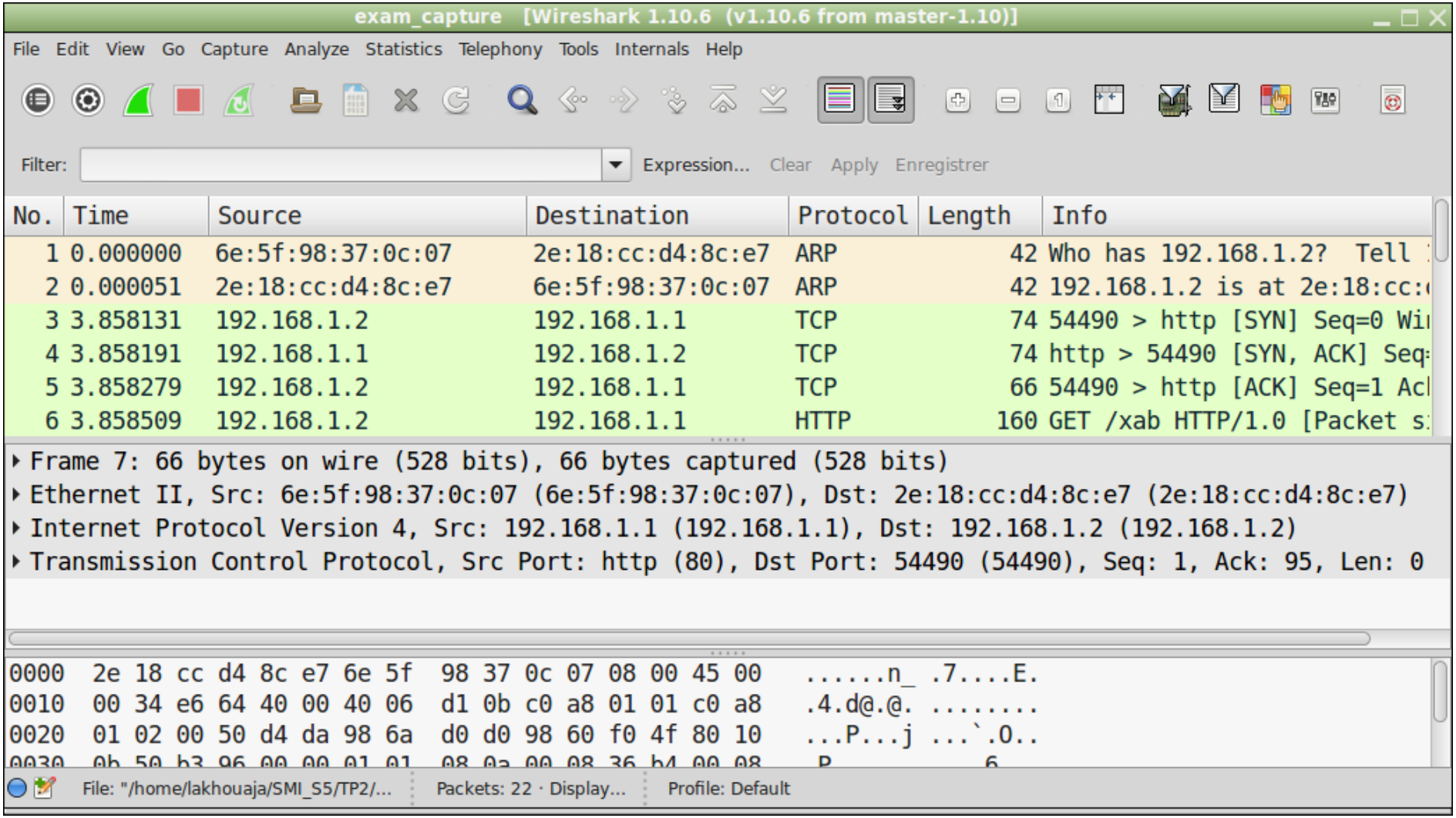
- le nom et l'état des interfaces du système
  `netstat -i`
- le contenu de la table de routage
  `netstat -r | n`
- ainsi que l'état de tous les sockets
  `netstat -a`

Wireshark

Wireshark [1] est un outil d'analyse des réseaux qui permet de capturer et d'analyser les paquets qui circulent sur le réseau. Il peut être utilisé pour capturer les paquets qui circulent sur une interface ou pour visualiser le contenu d'un fichier qui des paquets capturés par un autre utilitaire tel que **tcpdump**. Il est multi-plateforme, il fonctionne sous Linux, Windows, MacOS, ...

1. Site officiel : https://www.wireshark.org/

interface de wireshark



Colonnes

Les colonnes se présentent comme suit :

- **No. :** représente le numéro du paquet ;
- **Time :** représente le temps de capture du paquet ;
- **Source :** représente l'adresse IP ou MAC de la source ;
- **Destination :** représente l'adresse IP ou MAC destination ;
- **Protocol :** représente le type du protocole capturé ;
- **Length :** représente la taille du paquet (en octets) ;
- **Info :** représente une brève information concernant le paquet.

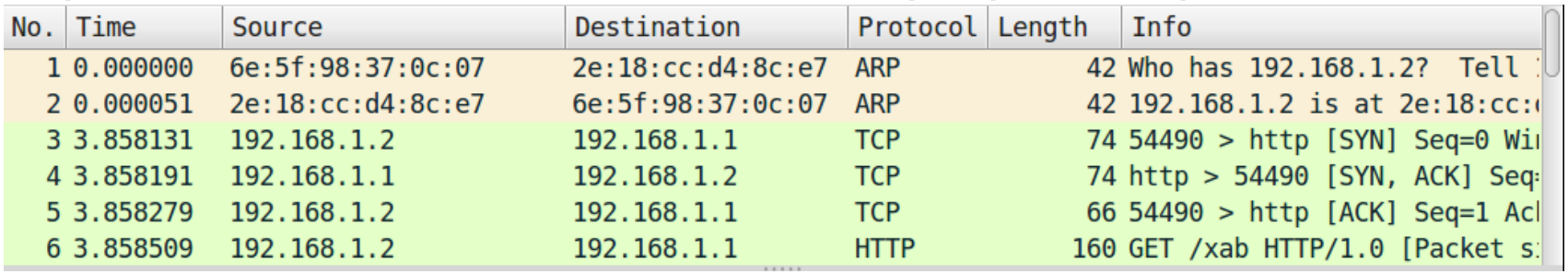
Remarque

sous Linux, comme pour la commande **tcpdumd**, **wireshark** ne peut pas être utilisé pour capturé des données en mode simple utilisateur. Pour capturer des données il faut passer en mode administrateur.

Description de l'interface

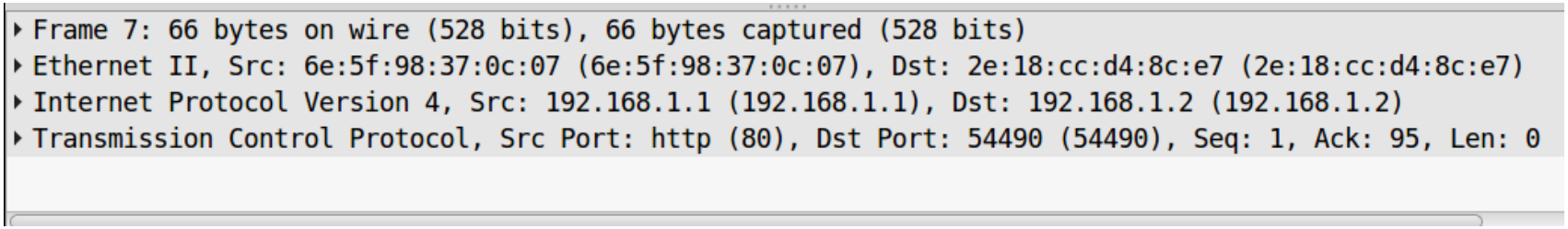
L'interface est découpé en trois zones :

1. supérieure : contient l'ensemble des paquets capturés



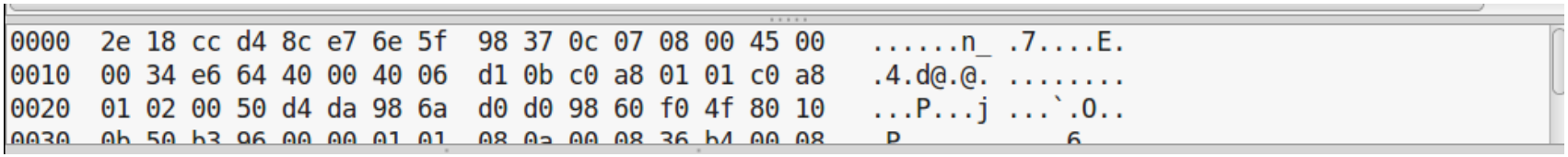
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	6e:5f:98:37:0c:07	2e:18:cc:d4:8c:e7	ARP	42	Who has 192.168.1.2? Tell
2	0.000051	2e:18:cc:d4:8c:e7	6e:5f:98:37:0c:07	ARP	42	192.168.1.2 is at 2e:18:cc:
3	3.858131	192.168.1.2	192.168.1.1	TCP	74	54490 > http [SYN] Seq=0 Wi
4	3.858191	192.168.1.1	192.168.1.2	TCP	74	http > 54490 [SYN, ACK] Seq
5	3.858279	192.168.1.2	192.168.1.1	TCP	66	54490 > http [ACK] Seq=1 Ac
6	3.858509	192.168.1.2	192.168.1.1	HTTP	160	GET /xab HTTP/1.0 [Packet s

2. centrale : affiche les détails d'un paquet sélectionné sous forme de couches



```
▸ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
▸ Ethernet II, Src: 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07), Dst: 2e:18:cc:d4:8c:e7 (2e:18:cc:d4:8c:e7)
▸ Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
▸ Transmission Control Protocol, Src Port: http (80), Dst Port: 54490 (54490), Seq: 1, Ack: 95, Len: 0
```

3. Zone inférieure : présente le paquet sous format octale et ASCII



```
0000  2e 18 cc d4 8c e7 6e 5f  98 37 0c 07 08 00 45 00   ......n_ .7....E.
0010  00 34 e6 64 40 00 40 06  d1 0b c0 a8 01 01 c0 a8   .4.d@.@. ........
0020  01 02 00 50 d4 da 98 6a  d0 d0 98 60 f0 4f 80 10   ...P...j ...`.O..
0030  0b 50 b3 96 00 00 01 01  08 0a 00 08 36 b4 00 08    P          6
```

Filtres

Il est possible de ne pas afficher tous les paquets en les filtrant. Par exemple, on peut afficher juste les paquets **http**, en tapant **http** dans la zone **Filter :**. Il est possible aussi d'utiliser des expressions.

Exemples de filtres

Filtre/expression	**Signification**
tcp	afficher seulement les paquets TCP
ip.src==192.168.1.2	afficher seulement les paquets qui sortent de 192.168.1.2
ip.dst==192.168.1.1 && http	afficher les paquets HTTP qui partent vers 192.168.1.1
ip && !udp	afficher les paquets IP mais n'afficher pas les paquets UDP