



## Document technique



# How to Use Wireshark



## Index :

1. Qu'est-ce que Wireshark ?
2. Prérequis d'utilisation
3. Installer Wireshark !
4. Comment fonctionne Wireshark
5. Quelques fonctionnalités supplémentaires

# Qu'est-ce que Wireshark ?

Wireshark est un **analyseur de trame** réseau, il permet d'analyser les échanges que 2 systèmes partagent entre eux : *Un ordinateur et un serveur par exemple.*

Tout ce qui se passe entre une entrée et une sortie, il récupère les « **paquets** », ces derniers contiennent toutes les informations de ce qui est échangé.

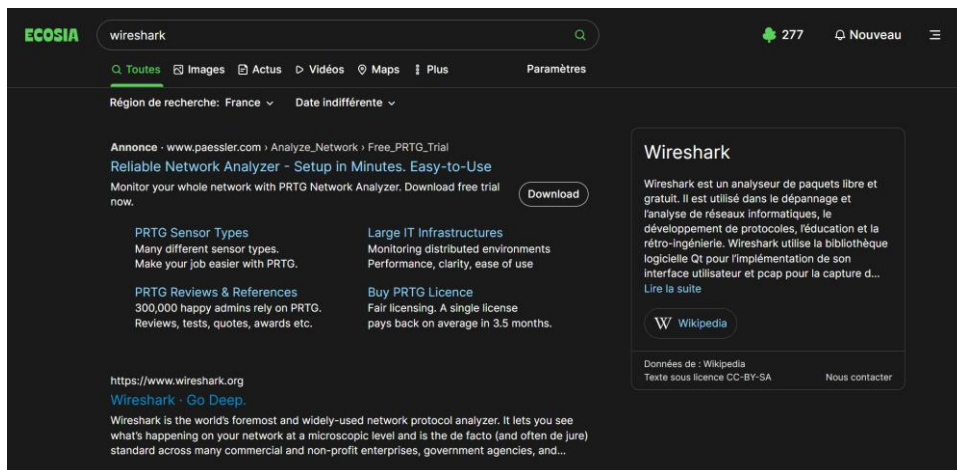
Il est **open source et gratuit** donc accessible à tous. Ces analyses s'effectuent dans une capture en temps réel et on propose la possibilité d'enregistrer la capture. Cependant, cette dernière se doit d'être en temps réel.

Autrefois Ethereal, aujourd'hui Wireshark, cet outils permet de **dépanner un réseau** en trouvant un problème de *liaison et d'efficacité*.

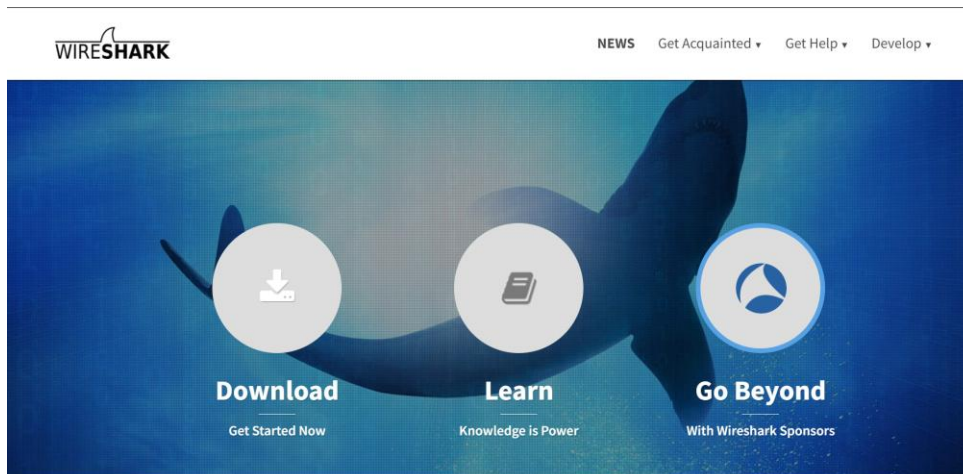
# Prérequis d'utilisation :

- Système d'exploitation **récent**
- 500Mb de **mémoire vive** (RAM)
- Processeur **64/32bits**
- Potentielle **connexion internet** également.

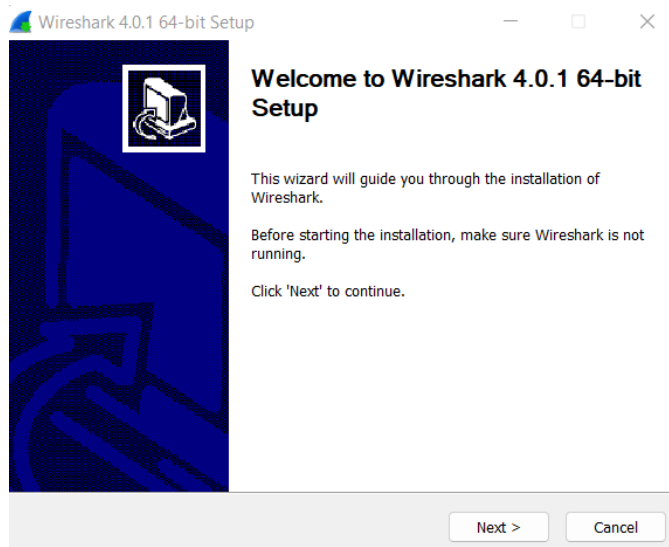
# Installer Wireshark



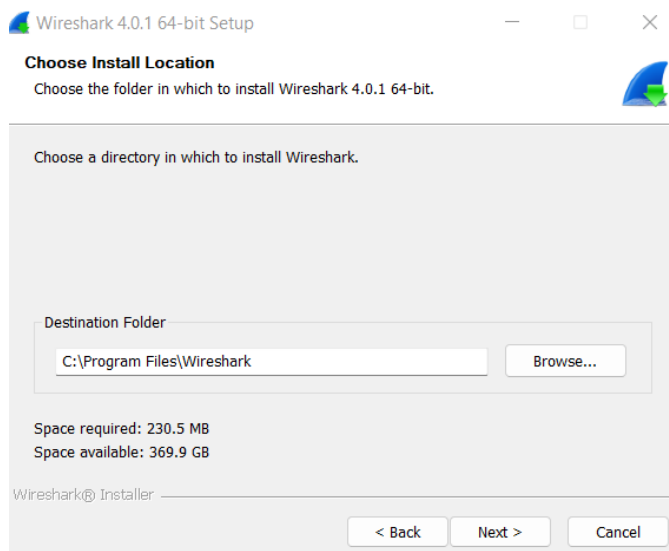
Tout d'abord rendez-vous sur votre navigateur et effectuer la recherche : « Wireshark ». Rendez-vous sur le **site officiel** (*finissant par .org*).



Rendez vous dans les **téléchargements** et choisissez la *version du logiciel* que vous désirez.



Suivez *toutes* les étapes d'installation, rappelez-vous qu'il vous faut **230.5MB d'espace libre** pour supporter l'installation !



Une fois installé, vous êtes *prêt à utiliser* Wireshark !

# Comment fonctionne Wireshark ?

Wireshark est un outils qui récupère les trames du réseaux sur lequel vous vous trouvez, une fois enregistrée vous pouvez **analyser** ces dernières. Pour cela il est important de connaître les **différents types de trames** qui existent. Etant un *environnement très complexe* nous aborderons seulement les *trames Ethernet 802.3 et Ethernet II*.

**Voici un tableau explicatif :**

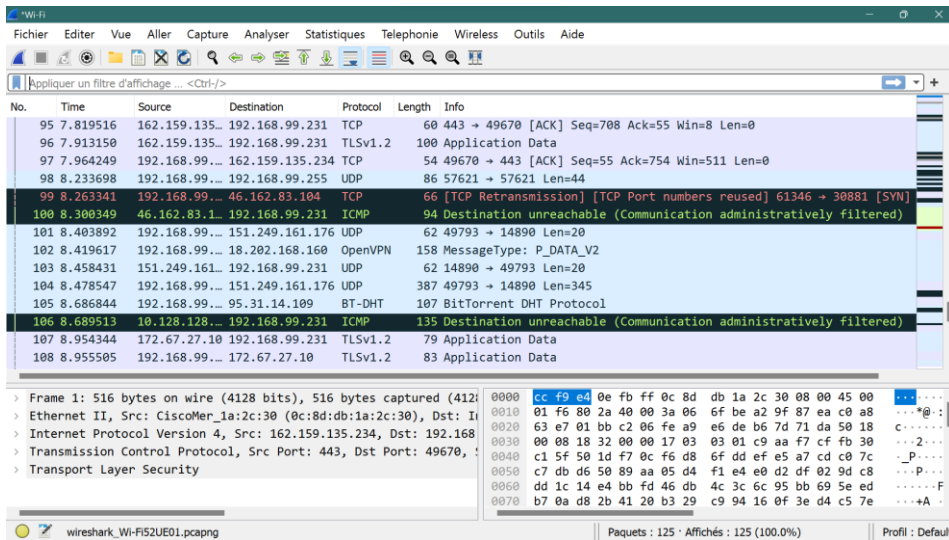
## Trame Ethernet 802.3 :

Adresse Destination	Adresse Source	Longueur des données	Champs de données	Bourrage	FCS
6 Octets	6 Octets	2 Octets	46 à 1500 Octets		4 Octets

## Trame Ethernet II :

Adresse Destination	Adresse Source	Protocole de couche 3	Champs de données (+Bourrage)	FCS
6 Octets	6 Octets	2 Octets	46 à 1500 Octets	4 Octets

## Voici comment se présente un enregistrement classique :

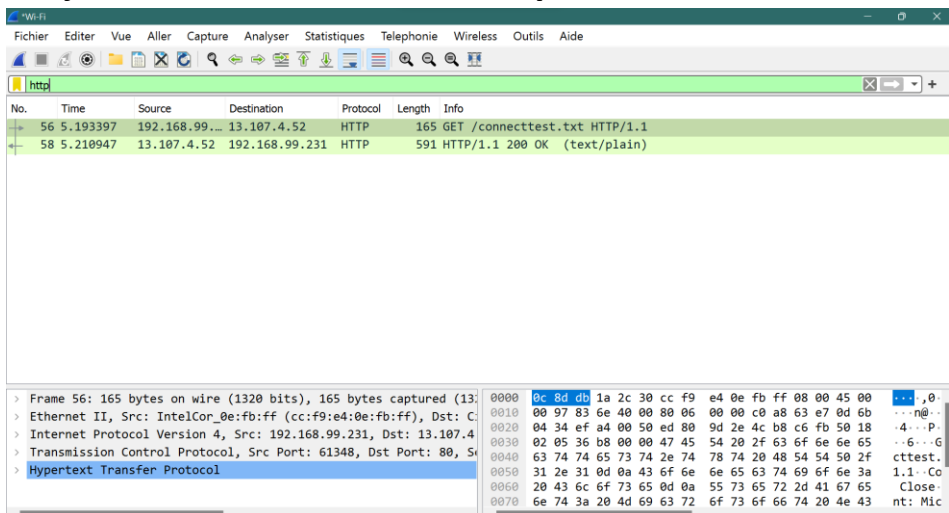


The image shows a Wireshark packet capture window. The top menu bar includes 'Fichier', 'Editer', 'Vue', 'Aller', 'Capture', 'Analyser', 'Statistiques', 'Telephonie', 'Wireless', 'Outils', and 'Aide'. Below the menu is a toolbar with icons for file operations, capture, and analysis. A filter bar at the top of the packet list shows 'Appliquer un filtre d'affichage ... <Ctrl-/>'. The packet list table has columns: No., Time, Source, Destination, Protocol, Length, and Info. It displays several packets, including TCP ACKs, TLS data, UDP messages, ICMP unreachable messages, OpenVPN data, and BitTorrent DHT messages. The packet details pane on the right shows the structure of the selected packet (No. 106), including Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol fields. The packet bytes pane at the bottom shows the raw hex and ASCII data of the selected packet.

No.	Time	Source	Destination	Protocol	Length	Info
95	7.819516	162.159.135...	192.168.99.231	TCP	60	443 → 49670 [ACK] Seq=708 Ack=55 Win=8 Len=0
96	7.913150	162.159.135...	192.168.99.231	TLSv1.2	100	Application Data
97	7.964249	192.168.99...	162.159.135.234	TCP	54	49670 → 443 [ACK] Seq=55 Ack=754 Win=511 Len=0
98	8.233698	192.168.99...	192.168.99.255	UDP	86	57621 → 57621 Len=44
99	8.263341	192.168.99...	46.162.83.104	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 61346 → 30881 [SYN]
100	8.300349	46.162.83.1...	192.168.99.231	ICMP	94	Destination unreachable (Communication administratively filtered)
101	8.403892	192.168.99...	151.249.161.176	UDP	62	49793 → 14890 Len=20
102	8.419617	192.168.99...	18.202.168.160	OpenVPN	158	MessageType: P_DATA_V2
103	8.458431	151.249.161...	192.168.99.231	UDP	62	14890 → 49793 Len=20
104	8.478547	192.168.99...	151.249.161.176	UDP	387	49793 → 14890 Len=345
105	8.686844	192.168.99...	95.31.14.109	BT-DHT	107	BitTorrent DHT Protocol
106	8.689513	10.128.128...	192.168.99.231	ICMP	135	Destination unreachable (Communication administratively filtered)
107	8.954344	172.67.27.10	192.168.99.231	TLSv1.2	79	Application Data
108	8.955505	192.168.99...	172.67.27.10	TLSv1.2	83	Application Data

Vous remarquez que l'on peut faire appel à des **filtres**, ces derniers permettent de sélectionner des types d'informations.

Ici, j'ai filtré avec l'élément **http** :



The image shows the same Wireshark capture window, but with a filter 'http' applied in the filter bar. The packet list now only shows two packets: a GET request (No. 56) and a 200 OK response (No. 58). The packet details pane on the right shows the structure of the selected packet (No. 56), including Ethernet II, Internet Protocol Version 4, and Hypertext Transfer Protocol fields. The packet bytes pane at the bottom shows the raw hex and ASCII data of the selected packet.

No.	Time	Source	Destination	Protocol	Length	Info
56	5.193397	192.168.99...	13.107.4.52	HTTP	165	GET /connecttest.txt HTTP/1.1
58	5.210947	13.107.4.52	192.168.99.231	HTTP	591	HTTP/1.1 200 OK (text/plain)

Une fois ces informations en tête, vous serez apte à chercher certaines informations au sein des trames, on peut par exemple, **reconstituer des images** envoyer sur des *messengeries privées*.

**A vous de décider de l'usage que vous en ferez !**