

-UTC505/USR54D –

Wireshark ou le stéthoscope pour se préparer à ouvrir le réseau

Auteur : E. Gressier-Soudan

Attention : c'est une condition nécessaire, mais pas suffisante !!!



Le "pied de biche" pour ouvrir les trames du trafic Internet

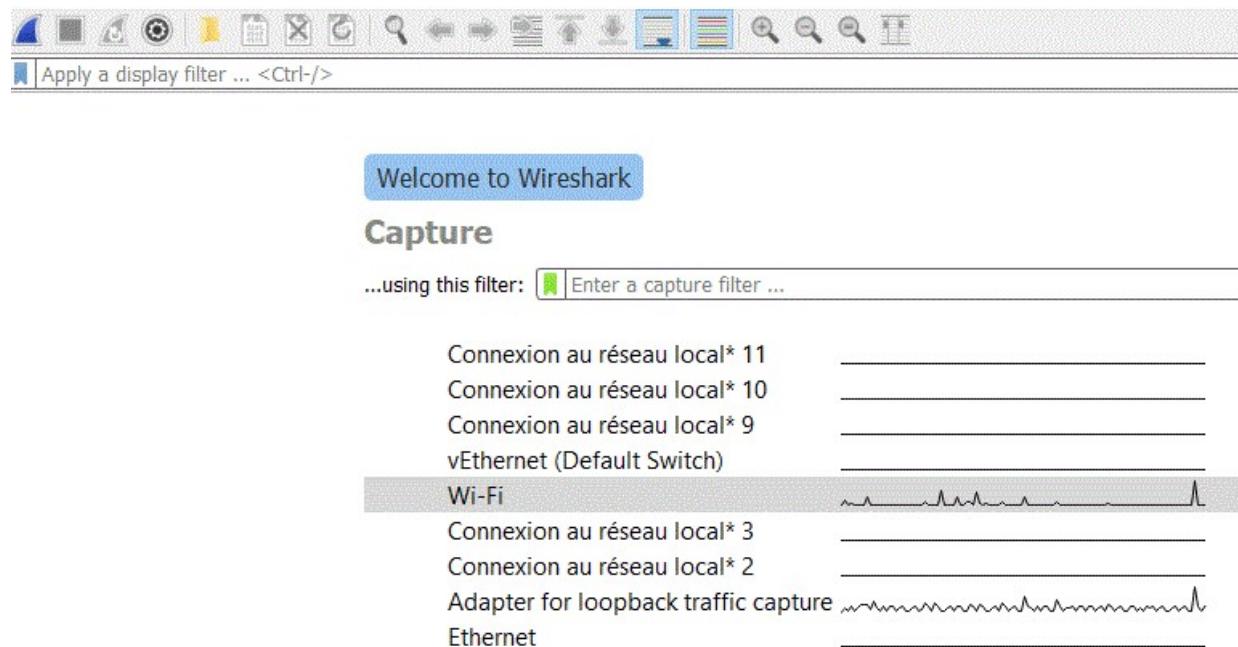
- Comprendre l'encapsulation, c'est comprendre l'enchaînement des couches entre-elles mais pour cela il faut savoir ouvrir le contenu des trames et préalablement les capturer !!!
- L'outil qui va bien : l'analyseur de protocole, on utilise Wireshark (<https://www.wireshark.org/>) qui est gratuit. Pour les différentes versions en fonction de votre système d'exploitation consulter la liste des versions à télécharger : <https://www.wireshark.org/download.html> et le guide utilisateur : https://www.wireshark.org/docs/wsug_html_chunked/, attention à bien installer WinPcap
- Wireshark peut aussi s'utiliser en ligne de commande avec tshark
 - Son ancêtre : Ethereal
- Il y a d'autres outils gratuits :
 - Tcpdump : <https://www.tcpdump.org/>
 - Free Network Analyzer : <https://freenetworkanalyzer.com/>
 - Il n'y a plus d'outil microsoft proposé dans Windows, Microsoft Network Analyser n'est plus... sa dernière version était la 1.4



Les étapes d'une session Wireshark (1)

Après avoir installer Wireshark :

- Lancer la capture en choisissant une interface d'écoute :
 - Ethernet si vous êtes connecté sur un cable
 - Wifi si vous êtes en sans fil
 - vEthernet si vous êtes sur une machine virtuelle sous Windows

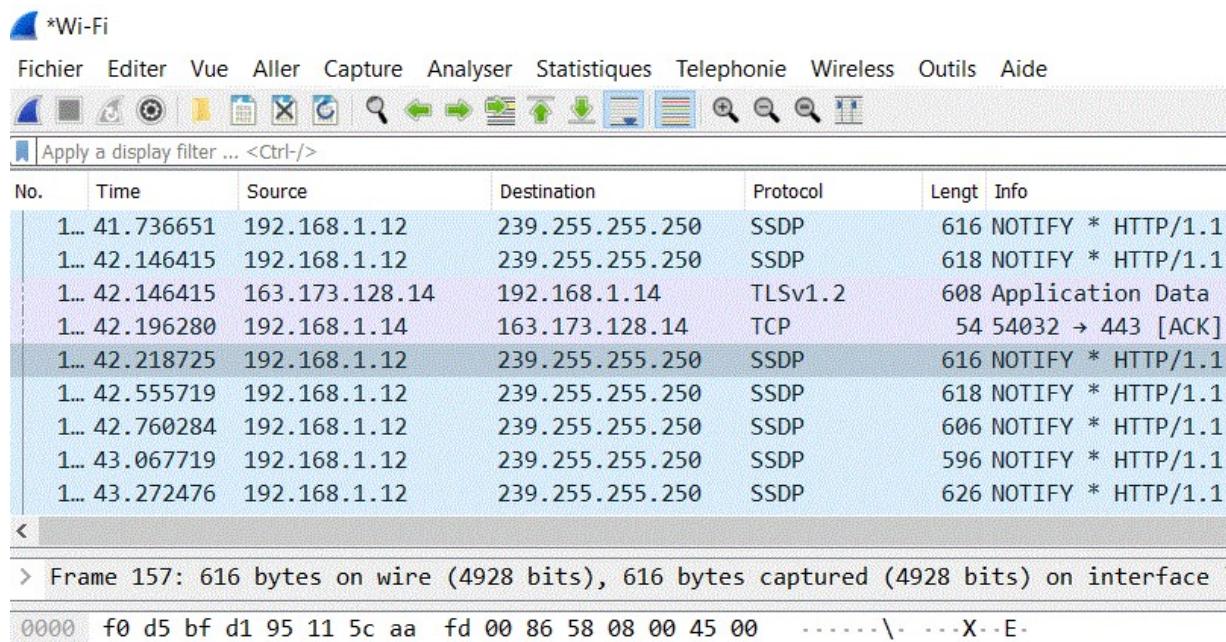


- Parfois capturer du trafic sur un réseau Ethernet Commuté (Switch) n'est pas facile. De nombreuses solutions sont données à la page
https://wiki.wireshark.org/CaptureSetup/Ethernet#Switched_Ethernet
- Vous trouverez aussi une interface qui s'appelle vEthernet, elle est liée à Hyper-V, la machine virtuelle proposée par Microsoft pour Windows, c'est une interface vers un switch virtuel qui permet à toutes les VM de communiquer entre-elles et avec le réseau Ethernet ou Wifi...
tout ça c'est la conséquence de la virtualisation, de premières infos pour commencer à comprendre sur <https://www.altaro.com/hyper-v/the-hyper-v-virtual-switch-explained-part-1/>



Les étapes d'une session Wireshark (2)

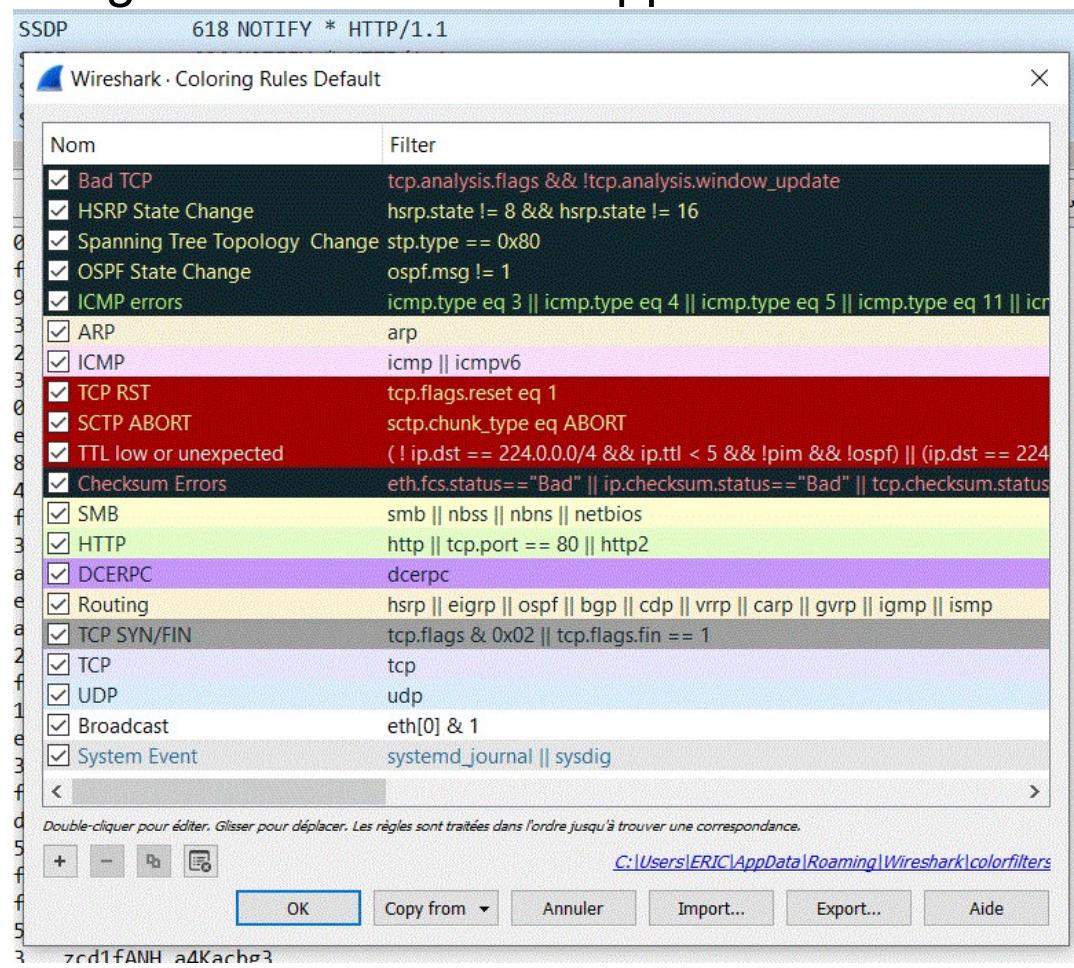
- Souvent on est en Wifi, alors l'interface qu'on choisit est : "Wi-Fi", ça tombe sous le sens
 - On démarre la capture en tapant sur l'aileron bleu en haut à gauche de la barre de menu... et là on attrape tout ce qui passe au niveau de l'interface de notre machine



- On arrête la capture en tapant sur le gros carré devenu rouge après le lancement, à côté de l'aileron bleu en haut à gauche dans la barre de menu

Les codes couleurs...

- Pour avoir les codes couleurs et leur signification, telle que paramétrée dans Wireshark au moment de l'utilisation, faire dans la barre de menu :
 - Vue > Coloring Rules... et on voit apparaître :



Les étapes d'une session Wireshark (3)

- Filtrage à la capture : il doit spécifié avant de lancer la capture (section capture en même temps que le choix de l'interface) ou en cours d'exécution à l'aide de la barre de menu Capture>Capture Filtres
 - En général, on filtre plutôt à l'affichage qu'à la capture, c'est plus facile
 - Mais si on doit capturer longtemps et de façon ciblée il faut spécifier un filtre de capture
 - http://www.wireshark.org/docs/wsug_html_chunked/ChCapCaptureFilterSection.html ou <https://gitlab.com/wireshark/-/wikis/CaptureFilters>
- Filtrage à l'affichage : il est spécifié dans la case ligne de commande "Apply a display filter... <CTRL-/>"
 - http://www.wireshark.org/docs/wsug_html_chunked/ChWorkDisplayFilterSection.html ou <https://gitlab.com/wireshark/-/wikis/DisplayFilters>

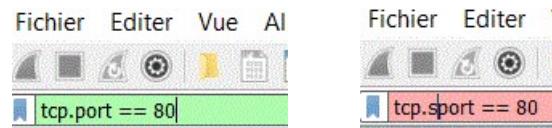
Attention le site d'informations récentes pour Wireshark est : <https://gitlab.com/wireshark/-/wikis/home>

- Pour les deux filtrages, les syntaxes sont voisines, pour un court résumé : https://openmaniak.com/fr/wireshark_filters.php



Syntaxe pour les filtres

- Il y a de nombreuses expressions fonction des attributs protocolaires... on en trouve facilement sur Internet et en particulier sur le site de Wireshark, qq exemples :
 - `tcp` n'afficher que les trames contenant le protocole TCP
 - `p.addr == 192.168.0.1` affiche les paquets IP v4 avec une adresse source ou destination de 192.168.0.1.
 - `tcp.port == 80` Affiche les paquets dont le port TCP source ou destination est égal à 80 (HTTP)
- Si la syntaxe est correcte, le fond de la ligne de commande est vert sinon rose :



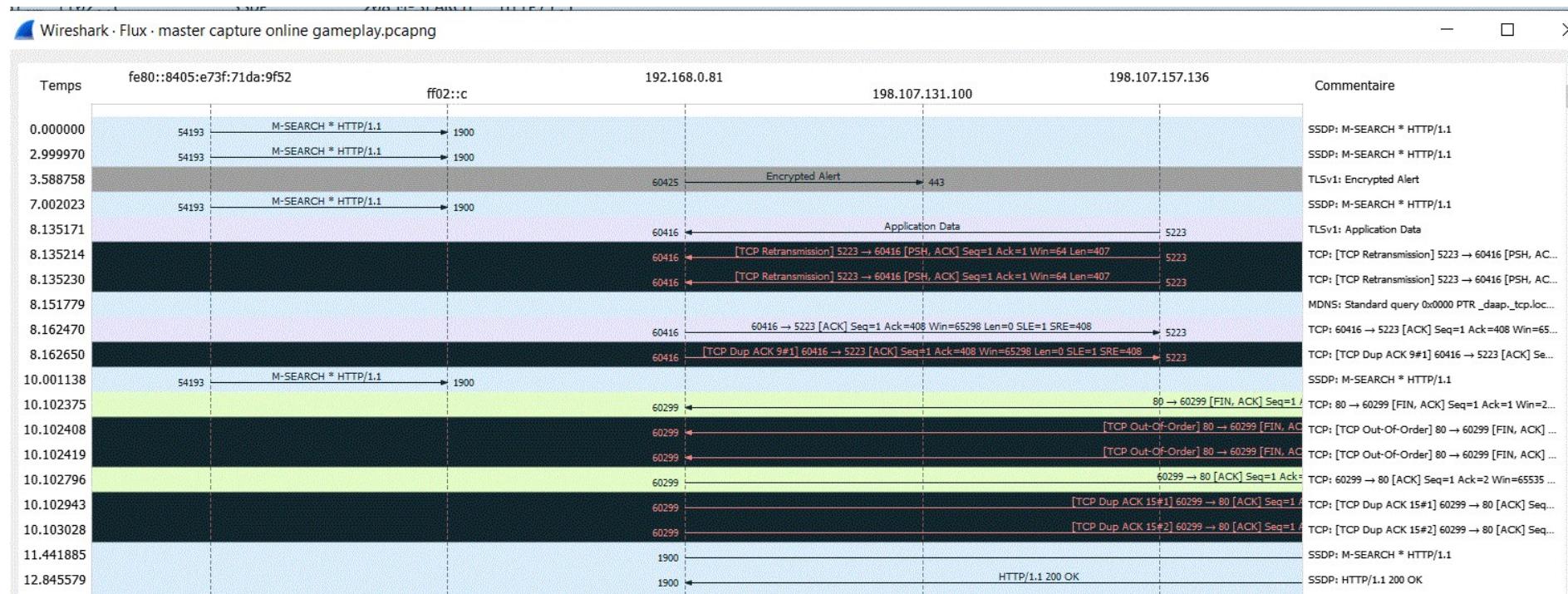
- Liste des protocoles Internet :
<https://gitlab.com/wireshark/wireshark/-/wikis/InternetProtocolFamily>
- Liste de tous les protocoles :
<https://gitlab.com/wireshark/wireshark/-/wikis/ProtocolReference>
- Liste des ports pour la couche transport Internet :
<https://gitlab.com/wireshark/wireshark/-/wikis/PortReference>
- Et plus globalement :
<https://gitlab.com/wireshark/wireshark/-/wikis/home>



Les étapes d'une session Wireshark (4)

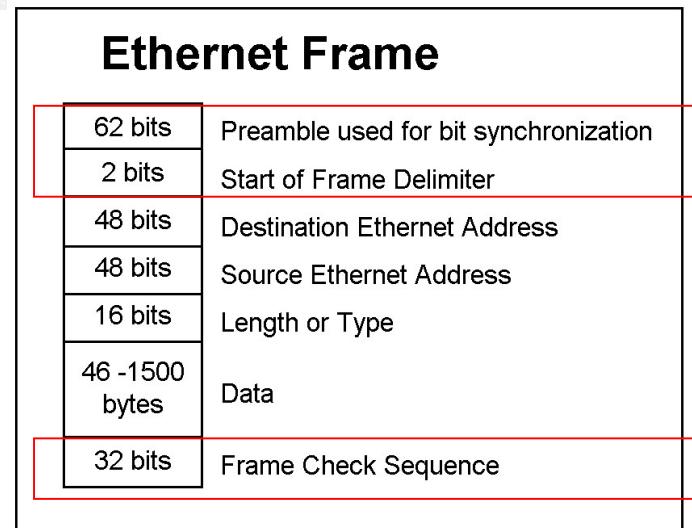
- C'est suivre certains flux capturés
 - Une commande intéressante dans la barre de menu :
 - Statistiques> Graphique des Flux
 - Elle permet d'avoir une cartographie de tous les échanges capturés
- Après filtrage, le graal c'est d'inspecter les trames qui nous intéressent et les différents protocoles qu'elles contiennent





Une trame Ethernet capturée par Wireshark

```
Frame Number: 58
Frame Length: 66 bytes (528 bits)
Capture Length: 66 bytes (528 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:tcp]
[Coloring Rule Name: TCP SYN/FIN]
[Coloring Rule string: tcp.flags & 0x02 || tcp.flags.fin == 1]
0000  00 04 76 9f fa 3a 00 26  6c 9d 84 fd 08 00 45 00  . .
0010  00 34 61 a0 40 00 80 06  86 42 a3 ad e7 6b a3 ad  . .
0020  04 17 08 70 00 00 00 00  00 00 00 00 00 00 00 00  . .
. . .
. . .
. . .
```



N'apparaît que sur le medium mais pas dans Wireshark

Source :
<http://www.networksecurity.org/members-area/glossary/e/ethernet-frame.html>

N'apparaît pas dans Wireshark, seules les trames sans erreur sont affichées



Pour vous exercer : plusieurs traces disponibles

Pour lancer Wireshark il suffit d'ouvrir les fichiers. Il y en a plusieurs issus de <https://github.com/hashploit/packet-captures> une trace de battlefield 2 modern Combat sur Playstation 2

Pour s'exercer le fichier "1-1-2-Wireshark-bf2-2015.pcapng" est mis à votre disposition dans l'environnement de formation. Vous pouvez explorer les autres traces à cet endroit puis aller sur le site pointé par l'url ci-dessus.



No.	Time	Source	Destination	Protocol	Lengt	Info
15	1.770420	192.168.137.227	65.112.87.186	TCP	75	65494 → 28910 [PSH, ACK] Seq=355 Ack=645 Win=17520 Len=9
16	1.771405	192.168.137.227	65.112.87.186	TCP	84	65494 → 28910 [FIN, PSH, ACK] Seq=364 Ack=645 Win=17520 Len=0
17	1.772338	192.168.137.227	65.112.87.186	TCP	74	65493 → 28910 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 WS=1 TS=
18	1.941282	65.112.87.186	192.168.137.227	TCP	66	28910 → 65494 [ACK] Seq=645 Ack=383 Win=65536 Len=0 TSval=
19	1.968796	65.112.87.186	192.168.137.227	TCP	888	28910 → 65494 [PSH, ACK] Seq=645 Ack=383 Win=65536 Len=82
20	1.969662	192.168.137.227	65.112.87.186	TCP	60	65494 → 28910 [RST] Seq=383 Win=0 Len=0

> Frame 17: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{049C1D03-B59D-45B0-A0F1-744EB1536

▼ Ethernet II, Src: SonyInte_fc:79:1d (00:15:c1:fc:79:1d), Dst: Giga-Byt_7b:a8:93 (74:d4:35:7b:a8:93)

- > Destination: Giga-Byt_7b:a8:93 (74:d4:35:7b:a8:93)
- > Source: SonyInte_fc:79:1d (00:15:c1:fc:79:1d)
- Type: IPv4 (0x0800)

▼ Internet Protocol Version 4, Src: 192.168.137.227, Dst: 65.112.87.186

- 0100 = Version: 4
- 0101 = Header Length: 20 bytes (5)
- > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
- Total Length: 60
- Identification: 0x028f (655)
- > Flags: 0x40, Don't fragment
- Fragment Offset: 0
- Time to Live: 64
- Protocol: TCP (6)
- Header Checksum: 0x5477 [validation disabled]
- [Header checksum status: Unverified]
- Source Address: 192.168.137.227
- Destination Address: 65.112.87.186

> Transmission Control Protocol, Src Port: 65493, Dst Port: 28910, Seq: 0, Len: 0

0000	74 d4 35 7b a8 93 00 15 c1 fc 79 1d 08 00 45 00	t-5{.....y...E-
0010	00 3c 02 8f 40 00 40 06 54 77 c0 a8 89 e3 41 70	-<...@... Tw...Ap
0020	57 ba ff d5 70 ee fb 68 4b 2d 00 00 00 00 a0 02	W...p..h K-----
0030	40 00 69 47 00 00 02 04 05 b4 01 03 03 00 01 01	@.iG-----
0040	08 0a 00 00 06 b0 00 00 00 00

Attention, les dates dans la
colonne "Time" sont en secondes



A vous d'essayer c Wireshark !!!



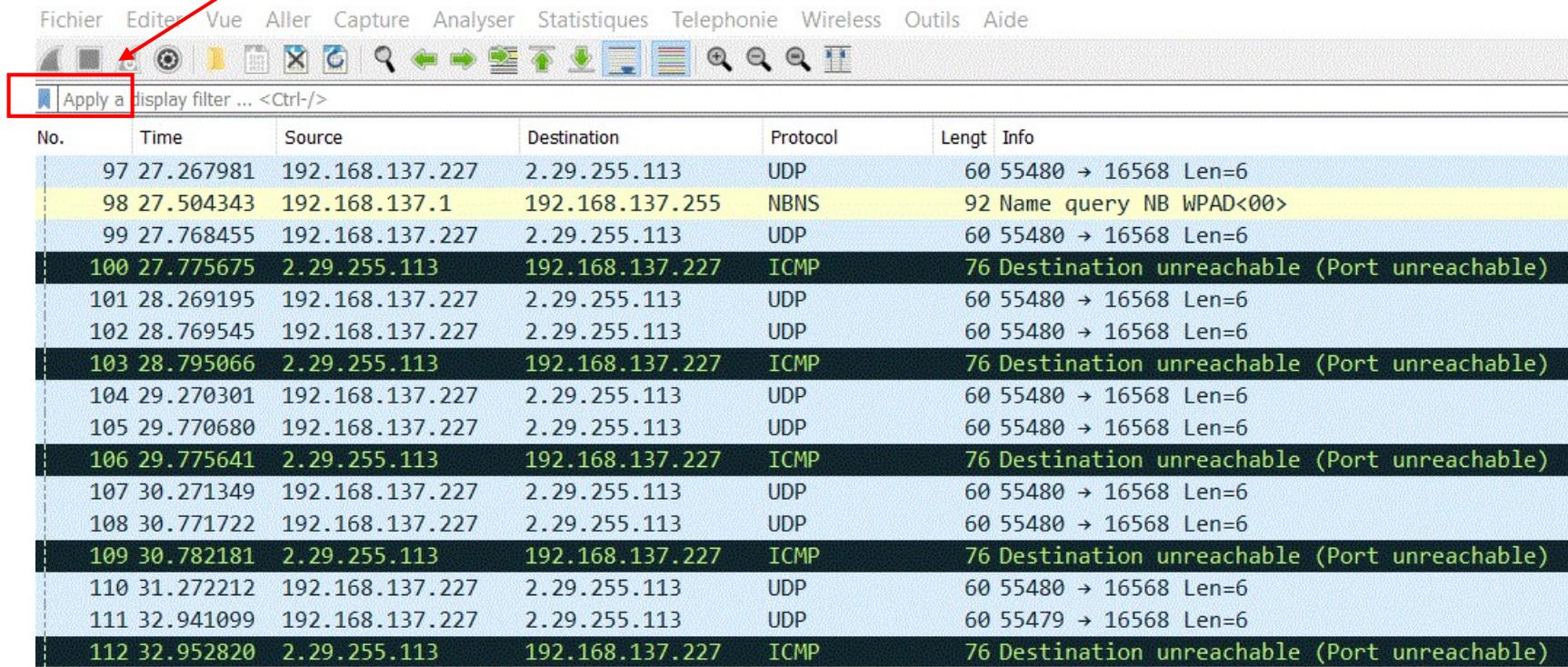
Ouvrir le fichier 1-1-2-Wireshark-bf2-2015.pcapng

- Wireshark se lance tout seul.
- Filtrer l'affichage :
 - Vous descendez pour afficher la trame 100 et les suivantes,
 - Sélectionner tout ce qui concerne le protocole ICMP (c'est celui qui sert à faire un ping en particulier)
 - Vous observez...



Avant :

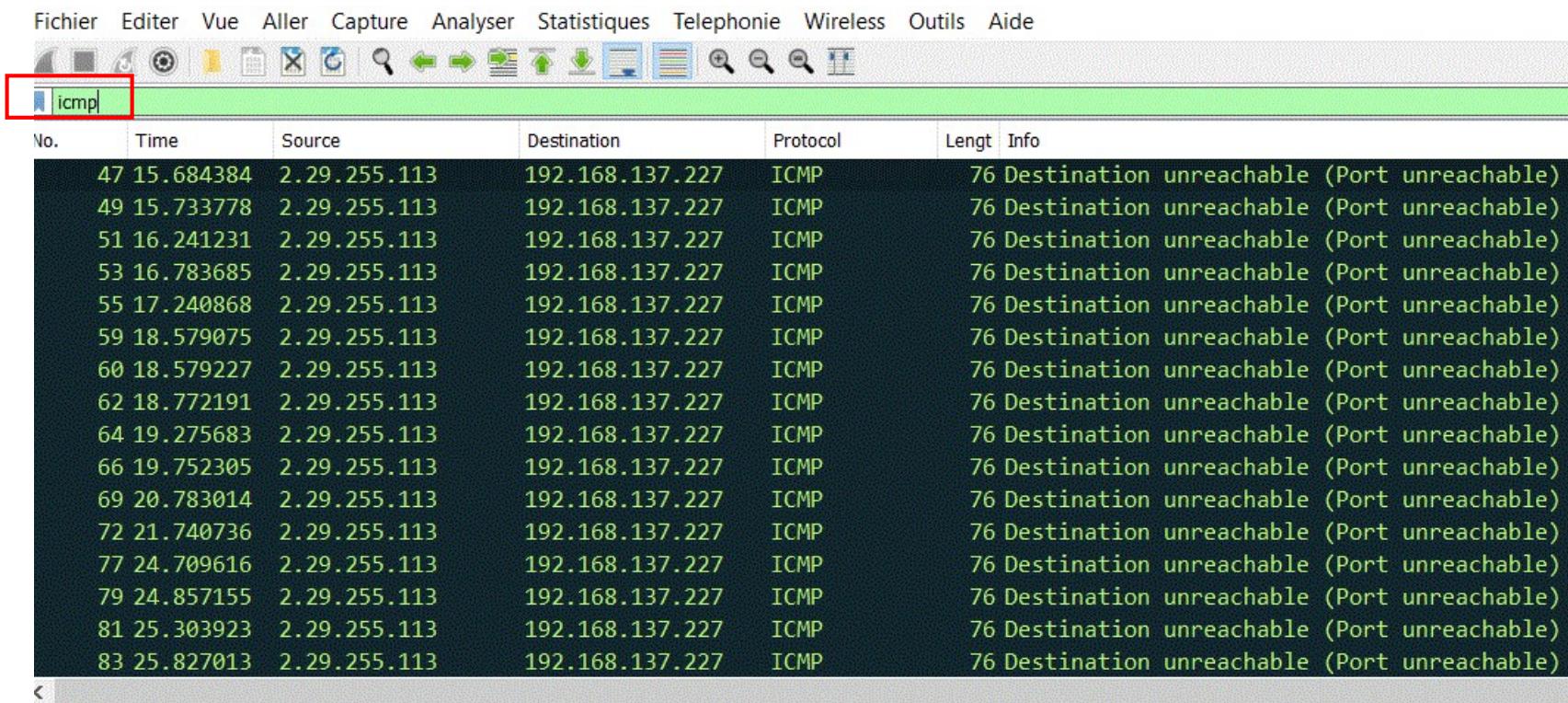
Taper "icmp" ici !



No.	Time	Source	Destination	Protocol	Length	Info
97	27.267981	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
98	27.504343	192.168.137.1	192.168.137.255	NBNS	92	Name query NB WPAD<00>
99	27.768455	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
100	27.775675	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
101	28.269195	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
102	28.769545	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
103	28.795066	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
104	29.270301	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
105	29.770680	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
106	29.775641	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
107	30.271349	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
108	30.771722	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
109	30.782181	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
110	31.272212	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
111	32.941099	192.168.137.227	2.29.255.113	UDP	60	55479 → 16568 Len=6
112	32.952820	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)



Après :



No.	Time	Source	Destination	Protocol	Lengt	Info
47	15.684384	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
49	15.733778	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
51	16.241231	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
53	16.783685	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
55	17.240868	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
59	18.579075	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
60	18.579227	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
62	18.772191	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
64	19.275683	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
66	19.752305	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
69	20.783014	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
72	21.740736	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
77	24.709616	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
79	24.857155	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
81	25.303923	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
83	25.827013	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)

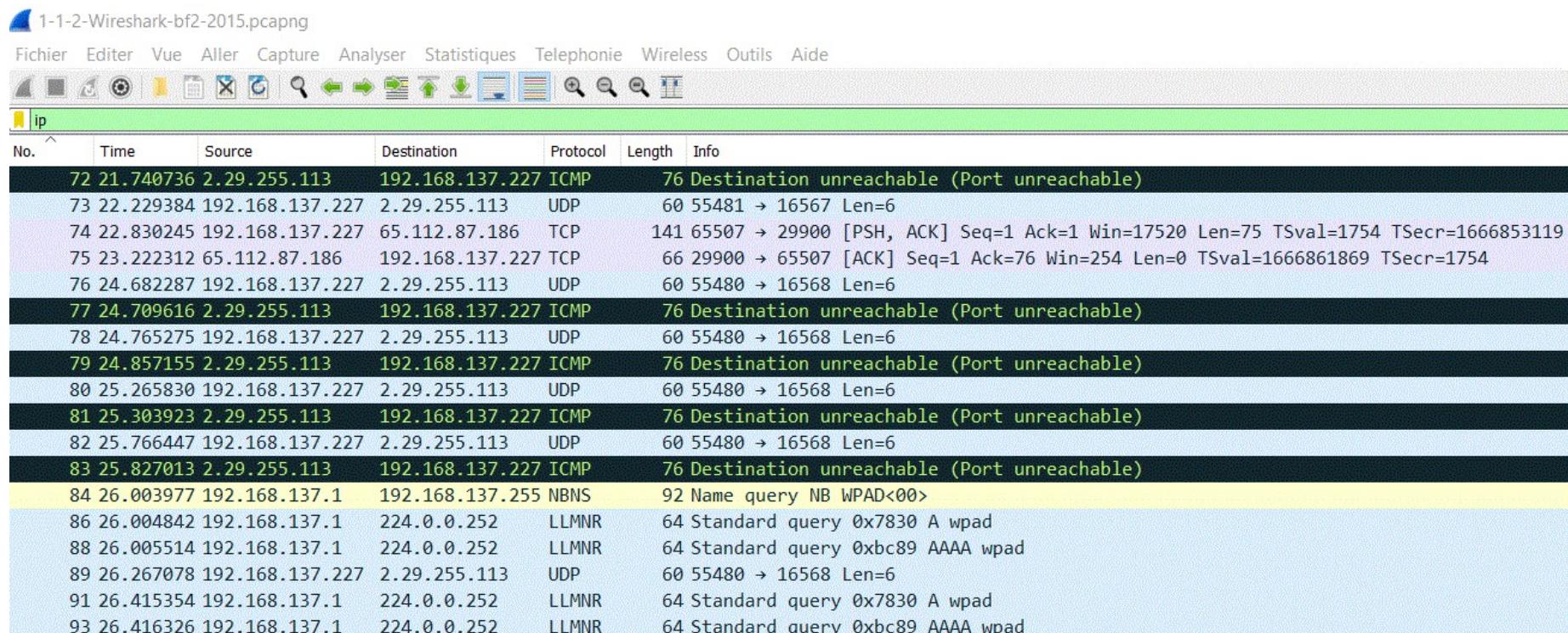


Observer les changements après avoir :

- Filtrer avec ip
- Filtrer avec ipv6
- Filtrer avec udp
- Filtrer avec tcp
- Filtrer avec "ip.addr == 192.168.137.227 and tcp.port == 65493"
en se positionnant à partir de la trame 17



On obtient pour Filtrer avec "ip" :



No.	Time	Source	Destination	Protocol	Length	Info
72	21.740736	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
73	22.229384	192.168.137.227	2.29.255.113	UDP	60	55481 → 16567 Len=6
74	22.830245	192.168.137.227	65.112.87.186	TCP	141	65507 → 29900 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=75 TSval=1754 TSecr=1666853119
75	23.222312	65.112.87.186	192.168.137.227	TCP	66	29900 → 65507 [ACK] Seq=1 Ack=76 Win=254 Len=0 TSval=1666861869 TSecr=1754
76	24.682287	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
77	24.709616	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
78	24.765275	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
79	24.857155	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
80	25.265830	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
81	25.303923	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
82	25.766447	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
83	25.827013	2.29.255.113	192.168.137.227	ICMP	76	Destination unreachable (Port unreachable)
84	26.003977	192.168.137.1	192.168.137.255	NBNS	92	Name query NB WPAD<00>
86	26.004842	192.168.137.1	224.0.0.252	LLMNR	64	Standard query 0x7830 A wpad
88	26.005514	192.168.137.1	224.0.0.252	LLMNR	64	Standard query 0xbc89 AAAA wpad
89	26.267078	192.168.137.227	2.29.255.113	UDP	60	55480 → 16568 Len=6
91	26.415354	192.168.137.1	224.0.0.252	LLMNR	64	Standard query 0x7830 A wpad
93	26.416326	192.168.137.1	224.0.0.252	LLMNR	64	Standard query 0xbc89 AAAA wpad

On obtient toutes les trames qui contiennent des protocoles qui utilisent le protocole d'acheminement IP (v4) inclus



On obtient pour Filtrer avec "ipv6" :

No.	Time	Source	Destination	Protocol	Length	Info
2	0.000760	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0x978b A wpad
4	0.001952	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0xebc8 AAAA wpad
5	0.410992	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0x978b A wpad
7	0.411935	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0xebc8 AAAA wpad
42	2.892524	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
43	3.892049	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
44	5.892201	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
45	9.892193	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
57	17.893340	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
85	26.004654	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0x7830 A wpad
87	26.005340	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0xbc89 AAAA wpad
90	26.415198	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0x7830 A wpad
92	26.416173	fe80::8141:595...	ff02::1:3	LLMNR	84	Standard query 0xbc89 AAAA wpad
116	33.893534	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893
205	65.893651	fe80::8141:595...	ff02::1:2	DHCPv6	152	Solicit XID: 0x412ba0 CID: 000100011d1dd20c74d4357ba893

> Frame 85: 84 bytes on wire (672 bits), 84 bytes captured (672 bits) on interface \Device\NPF_{049C1D03-B59D-45B0-A0F1
> Ethernet II, Src: Giga-Byt_7b:a8:93 (74:d4:35:7b:a8:93), Dst: IPv6mcast_01:00:03 (33:33:00:01:00:03)
▼ Internet Protocol Version 6, Src: fe80::8141:595b:e487:4bc3, Dst: ff02::1:3
 0110 = Version: 6
 > 0000 0000 = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
 0000 0000 0000 0000 = Flow Label: 0x000000
 Payload Length: 30
 Next Header: UDP (17)
 Hop Limit: 1
 Source Address: fe80::8141:595b:e487:4bc3
 Destination Address: ff02::1:3
 > User Datagram Protocol, Src Port: 59050, Dst Port: 5355
 > Link-local Multicast Name Resolution (query)

On obtient toutes les trames qui contiennent des protocoles qui utilisent le protocole d'acheminement IP (v6) inclus... il n'y a pas beaucoup...



On obtient pour Filtrer avec "udp" :

The screenshot shows the Wireshark interface with a list of captured network packets. The filter applied is "udp". The packet list includes:

- 73 22.229384 192.168.137.227 2.29.255.113 UDP 60 55481 → 16567 Len=6
- 76 24.682287 192.168.137.227 2.29.255.113 UDP 60 55480 → 16568 Len=6
- 77 24.709616 2.29.255.113 192.168.137.227 ICMP 76 Destination unreachable (Port unreachable)
- 78 24.765275 192.168.137.227 2.29.255.113 UDP 60 55480 → 16568 Len=6
- 79 24.857155 2.29.255.113 192.168.137.227 ICMP 76 Destination unreachable (Port unreachable)
- 80 25.265830 192.168.137.227 2.29.255.113 UDP 60 55480 → 16568 Len=6
- 81 25.303923 2.29.255.113 192.168.137.227 ICMP 76 Destination unreachable (Port unreachable)
- 82 25.766447 192.168.137.227 2.29.255.113 UDP 60 55480 → 16568 Len=6
- 83 25.827013 2.29.255.113 192.168.137.227 ICMP 76 Destination unreachable (Port unreachable)
- 84 26.003977 192.168.137.1 192.168.137.255 NBNS 92 Name query NB WPAD<00>
- 85 26.004654 fe80::8141:595.. ff02::1:3 LLMNR 84 Standard query 0x7830 A wpad
- 86 26.004842 192.168.137.1 224.0.0.252 LLMNR 64 Standard query 0x7830 A wpad
- 87 26.005340 fe80::8141:595.. ff02::1:3 LLMNR 84 Standard query 0xbc89 AAAA wpad
- 88 26.005514 192.168.137.1 224.0.0.252 LLMNR 64 Standard query 0xbc89 AAAA wpad
- 89 26.267078 192.168.137.227 2.29.255.113 UDP 60 55480 → 16568 Len=6

Details for the ICMP destination unreachable frame (Frame 77):

- Type: 3 (Destination unreachable)
- Code: 3 (Port unreachable)
- Checksum: 0xc0bf [correct]
- [Checksum Status: Good]
- Unused: 00000000

Details for the NBNS name query frame (Frame 84):

- Internet Protocol Version 4, Src: 192.168.137.227, Dst: 2.29.255.113
- User Datagram Protocol, Src Port: 55480, Dst Port: 16568
- Data (6 bytes)

On obtient toutes les trames qui contiennent des protocoles qui sont en relation avec le transport UDP (NetBios Name Service, Link Local Multicast Name Resolution...) mais de façon surprenante aussi ICMP quand c'est un message d'erreur relatif à un message UDP... mais cela peut reposer sur de l'IPv4 ou de l'IPv6 !



On obtient pour Filtrer avec "tcp" :

The Wireshark interface shows a list of network frames. A specific frame is highlighted in red, indicating it was captured using a TCP filter. The details pane at the bottom provides the following information for the selected frame:

- > Frame 11: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{049C1D03-B59D-45B0-A0F1-744EB1530D0D}, id 0
- > Ethernet II, Src: Giga-Byt_7b:a8:93 (74:d4:35:7b:a8:93), Dst: SonyInte_fc:79:1d (00:15:c1:fc:79:1d)
- > Internet Protocol Version 4, Src: 65.112.87.186, Dst: 192.168.137.227
- > Transmission Control Protocol, Src Port: 28910, Dst Port: 65494, Seq: 0, Ack: 1, Len: 0

On obtient toutes les trames qui contiennent des protocoles qui sont en relation avec le transport TCP. Ici, c'est de l'applicatif Client/Serveur pour le jeu, pas d'HTTP, de FTP...



On obtient pour Filtrer avec "ip.addr==192.168.137.227 and tcp.port==65493" en se positionnant à partir de la trame 17:

Fichier Editer Vue Aller Capture Analyser Statistiques Telephonie Wireless Outils Aide							
No.	Time	Source	Destination	Protocol	Lengt	Info	
<input checked="" type="checkbox"/> ip.addr ==192.168.137.227 and tcp.port == 65493							
17	1.772338	192.168.137.227	65.112.87.186	TCP	74	65493 → 28910 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 WS=1 TSval=1712 T	
21	1.990615	65.112.87.186	192.168.137.227	TCP	74	28910 → 65493 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1380 WS=256 T	
22	1.991485	192.168.137.227	65.112.87.186	TCP	66	65493 → 28910 [ACK] Seq=1 Ack=1 Win=17520 Len=0 TSval=1712 TSeср=1025	
23	2.004544	192.168.137.227	65.112.87.186	TCP	419	65493 → 28910 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=353 TSval=1712 TSeср=1025	
30	2.201302	65.112.87.186	192.168.137.227	TCP	710	28910 → 65493 [PSH, ACK] Seq=1 Ack=354 Win=65536 Len=644 TSval=102575	
31	2.204355	192.168.137.227	65.112.87.186	TCP	75	65493 → 28910 [PSH, ACK] Seq=354 Ack=645 Win=17520 Len=9 TSval=1713 T	
32	2.205287	192.168.137.227	65.112.87.186	TCP	84	65493 → 28910 [FIN, PSH, ACK] Seq=363 Ack=645 Win=17520 Len=18 TSval=102575	
33	2.404672	65.112.87.186	192.168.137.227	TCP	66	28910 → 65493 [ACK] Seq=645 Ack=382 Win=65536 Len=0 TSval=1025754726	
34	2.452812	65.112.87.186	192.168.137.227	TCP	888	28910 → 65493 [PSH, ACK] Seq=645 Ack=382 Win=65536 Len=822 TSval=102575	
35	2.453601	192.168.137.227	65.112.87.186	TCP	60	65493 → 28910 [RST] Seq=382 Win=0 Len=0	
36	2.480121	65.112.87.186	192.168.137.227	TCP	1434	28910 → 65493 [PSH, ACK] Seq=1467 Ack=382 Win=65536 Len=1368 TSval=102575	
37	2.480909	192.168.137.227	65.112.87.186	TCP	60	65493 → 28910 [RST] Seq=382 Win=0 Len=0	
38	2.507773	65.112.87.186	192.168.137.227	TCP	375	28910 → 65493 [PSH, ACK] Seq=2835 Ack=382 Win=65536 Len=309 TSval=102575	
39	2.508632	192.168.137.227	65.112.87.186	TCP	60	65493 → 28910 [RST] Seq=382 Win=0 Len=0	
40	2.537938	65.112.87.186	192.168.137.227	TCP	66	28910 → 65493 [FIN, ACK] Seq=3144 Ack=382 Win=65536 Len=0 TSval=102575	
41	2.538798	192.168.137.227	65.112.87.186	TCP	60	65493 → 28910 [RST] Seq=382 Win=0 Len=0	

<
> Frame 41: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{049C1D03-B59D-45B0-A0F1-744EB1530D0D}, id 0
> Ethernet II, Src: SonyInte_fc:79:1d (00:15:c1:fc:79:1d), Dst: Giga-Byt_7b:a8:93 (74:d4:35:7b:a8:93)
> Internet Protocol Version 4, Src: 192.168.137.227, Dst: 65.112.87.186
> Transmission Control Protocol, Src Port: 65493, Dst Port: 28910, Seq: 382, Len: 0

C'est tout un échange Client-Serveur avec le protocole TCP (colonne "Protocol") qui va de la trame 17 à la trame 41... est-ce que c'est le même résultat qu'avec "Filtrer tcp" ?



Liens pour aller plus loin, c'est une vive recommandation :

Chris Geer. Wireshark Masterclass. En anglais mais bien fait et plus compréhensible que les séries américaines en VO si vous en suivez sur Netflix !!! Ce cours a la vertu de vous faire travailler votre anglais, ce n'est pas un mal car les informations les plus fiables en réseaux et sur Internet sont souvent en anglais sur le Web ou dans les documents de référence. Mais attention, toujours s'interroger sur ses sources Web. Par exemple Wikipedia, même si ce n'est pas une référence absolue, reste plus fiable en anglais qu'en français pour ce qui concerne le domaine des réseaux.

Les vidéos de cette Masterclass sont courtes, moins de 20mn pour ce que j'ai un peu exploré.

- **"Intro to Wireshark Tutorial"** :
 - 25 Avril 2021, // Lesson 1 // Wireshark Setup Free Tutorial, <https://www.youtube.com/watch?v=OU-A2EmVrKQ> (29/08/2021)
 - 5 Avril 2021, // Lesson 2 // How to Capture Network Traffic, <https://www.youtube.com/watch?v=nWvscuxqais> (29/08/2021)
 - 29 Avril 2021, // Lesson 3 // Capturing Packets with Dumpcap, (tshark est souvent mentionné comme outil en lignes de commande) <https://www.youtube.com/watch?v=DAtyzE1TUII> (29/08/2021)
 - 11 Mai 2021, // Lesson 4 // Where do we capture network traffic? How?
https://www.youtube.com/watch?v=Atde35_9AAc (29/08/2021)
 - 25 Mai 2021, // Lesson 5 // How To filter Traffic, https://www.youtube.com/watch?v=-HDpYR_QSFw (29/08/2021)
- Les vidéos suivantes sont intéressantes, mais un peu plus avancés. Le nom change en "**Wireshark Tutorial**" :
 - 21 Juillet 2021, // Lesson 6 // Name Resolution, <https://www.youtube.com/watch?v=gfxxCBCKvMU> (29/08/2021)
 - 3 Août 2021, // Lesson 7 // Using the Time Column, <https://www.youtube.com/watch?v=SIIJu5MdkAg> (29/08/2021)

Chris Geer aborde d'autres sujets sur les réseaux en s'aidant de Wireshark, globalement, c'est très bien ce qu'il fait. C'est un avis personnel.





Merci pour votre attention !!!