Wireshark

Wireshark est un logiciel d'analyse réseau (sniffer) qui permet de **visualiser l'ensemble des données transitant sur la machine qui l'exécute, et d'obtenir des informations sur les protocoles applicatifs utilisés**. Les octets sont capturés en utilisant la librairie réseau PCAP, puis regroupés en blocs d'informations et analysés par le logiciel.

Le principe fonctionne comme tout autre outil d'analyse qui existe, on **capture** les paquets d'une interface réseau et on **applique** des filtres pour capture et trouver ce qui nous intéresse.
WireShark permet d'appliquer des filtres directement à la **capture** ou sur les **résultats** de la capture.

Bien entendu, en ce qui concerne le contenu des paquets (les données), **seules les connexions non chiffrées pourront être analysées**.
On voit donc l'importance d'utiliser les connexions sécurisées surtout lors des authentification (échange de mot de passe) par exemple en utilisant des sites **HTTPs ou un VPN**.

Notez qu'il existe une autre multitude d'outils pour capturer les paquets qui passent par une interface. Notamment sous **GNU/Linux** (**ngrep, tshark, tcpdump, etc**). Mais tous ne sont pas équivalents.

Ses **"dissectors" ou décodeurs de protocoles** permettent d'interpréter le trafic du réseau.

Conçu en 1997-1998 par Gerald Combs sous le nom historique de "Ethereal", il est repris en 2006 sous le nom moderne de "Wireshark". En 2008, Wireshark sort en version 1.0 et en 2015 en version 2.0 avec une nouvelle interface graphique.

Quelle est la différence entre une trame et un paquet ? Qu'est-ce que le format pcap/pcapng ?

Fondamentalement, une trame est utilisée pour envoyer des données entre un seul réseau. Un paquet est utilisé pour envoyer des données d'un réseau à un autre, puis vers un périphérique spécifique sur ce réseau.

Partie 1

Installation

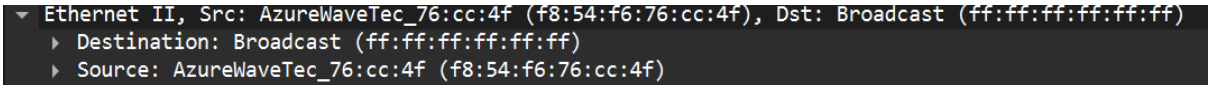
```
Paramétrage de libqt5multimediawidgets5:amd64 (5.15.8-2) ...
Paramétrage de libqt5multimediagsttools5:amd64 (5.15.8-2) ...
Paramétrage de libqt5multimedia5-plugins:amd64 (5.15.8-2) ...
Paramétrage de libqt5quick5:amd64 (5.15.8+dfsg-3) ...
Paramétrage de libqt5svg5:amd64 (5.15.8-3) ...
Paramétrage de libqt5waylandcompositor5:amd64 (5.15.8-2) ...
Paramétrage de wireshark-qt (4.0.11-1~deb12u1) ...
Paramétrage de wireshark (4.0.11-1~deb12u1) ...
Paramétrage de qtwayland5:amd64 (5.15.8-2) ...
Traitement des actions différées (« triggers ») pour man-db (2.11.2-2) ...
Traitement des actions différées (« triggers ») pour shared-mime-info (2.2-1) ..
.
Traitement des actions différées (« triggers ») pour mailcap (3.70+nmu1) ...
Traitement des actions différées (« triggers ») pour desktop-file-utils (0.26-1)
 ...
Traitement des actions différées (« triggers ») pour hicolor-icon-theme (0.17-2)
 ...
Traitement des actions différées (« triggers ») pour gnome-menus (3.36.0-1.1) ..
.
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u7) .
..
root@Wireshark:~# S
```

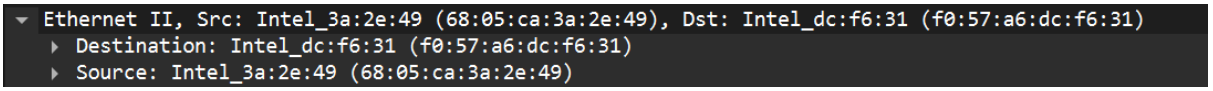
Réception des paquets

541	455.997483	VMware_c0:00:08	Broadcast	ARP	42 Who has 192.168.159.2? Tell 192.168.159.1
542	457.260432	VMware_c0:00:08	Broadcast	ARP	42 Who has 192.168.159.2? Tell 192.168.159.1
543	457.995044	VMware_c0:00:08	Broadcast	ARP	42 Who has 192.168.159.2? Tell 192.168.159.1
544	458.989707	VMware_c0:00:08	Broadcast	ARP	42 Who has 192.168.159.2? Tell 192.168.159.1
2526	31.109633	142.250.200.234	10.10.17.46	UDP	201 443 → 61666 Len=159
2527	31.109633	142.250.200.234	10.10.17.46	UDP	65 443 → 61666 Len=23
2528	31.112538	10.10.17.46	142.250.200.234	UDP	75 61666 → 443 Len=33
2529	31.129865	fe80::6ba:d6ff:fe2d…	ff02::1	UDP	328 52824 → 62976 Len=266
2670	33.192782	10.10.17.46	10.10.0.1	TCP	54 64957 → 53 [ACK] Seq=1 Ack=1 Win=131328 Len=0
2671	33.192852	10.10.17.46	10.10.0.1	TCP	54 64958 → 53 [ACK] Seq=1 Ack=1 Win=131328 Len=0
2672	33.193128	10.10.17.46	10.10.0.1	TCP	56 64958 → 53 [PSH, ACK] Seq=1 Ack=1 Win=131328 Len=2 [TCP segment of a reassembled PDU]
2673	33.193206	10.10.17.46	10.10.0.1	DNS	111 Standard query 0xd43d HTTPS global.telemetry.insights.video.a2z.com
2674	33.193293	10.10.17.46	10.10.0.1	TCP	56 64957 → 53 [PSH, ACK] Seq=1 Ack=1 Win=131328 Len=2 [TCP segment of a reassembled PDU]
2675	33.193335	10.10.17.46	10.10.0.1	DNS	111 Standard query 0xe60a A global.telemetry.insights.video.a2z.com
2676	33.197906	10.10.0.88	10.10.17.46	AJP13	171 AJP13 Error?
2677	33.200460	10.10.0.1	10.10.17.46	TCP	54 53 → 64958 [ACK] Seq=1 Ack=3 Win=64256 Len=0

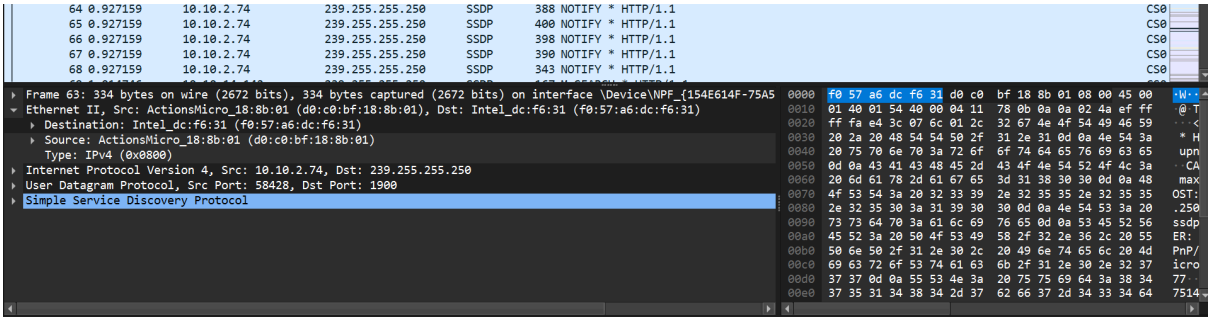
ARP : Couche de Liaison de données (Ethernet II).
UDP : Couche de Liaison de données (Ethernet II) + Couche de transport (Internet Protocol + User Data Protocol).
TCP : Couche de Liaison de données (Ethernet II) + Couche de transport (Internet Protocol).

Quelles sont les adresses MAC sources, les IP sources et les adresses MAC sources, les IP destinations des données capturées ?

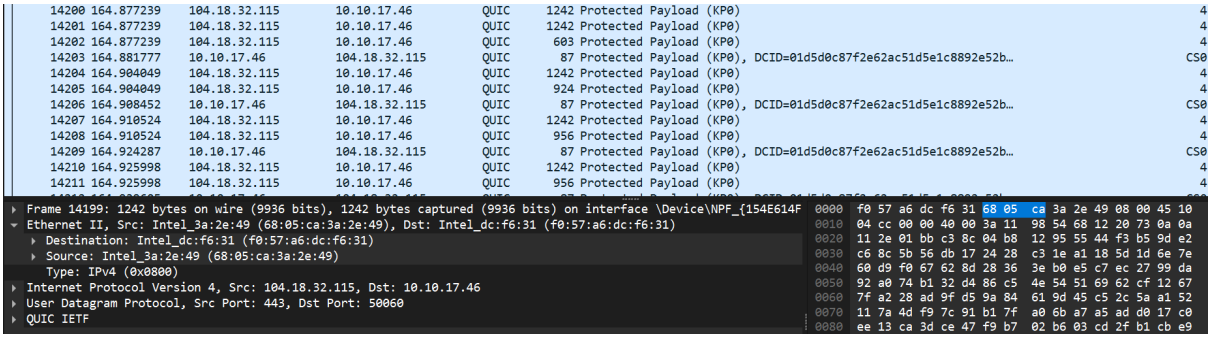




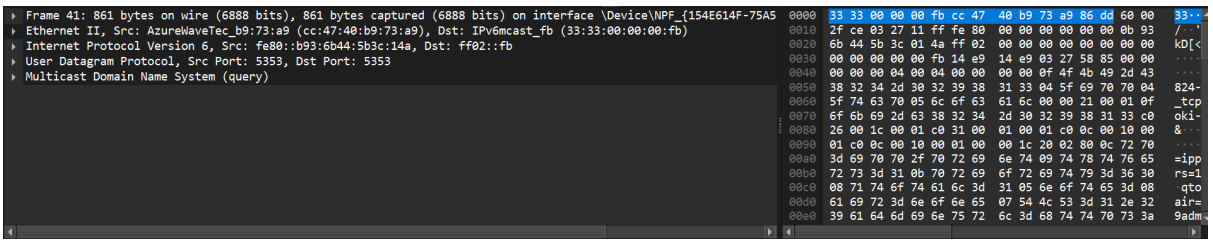
Référencez d'autres trames ou paquets circulants sur le réseau. Identifiez leurs protocoles et leur fonction.



SSDP : Protocole réseau basé sur la suite de protocoles Internet pour la diffusion et la découverte de services de réseau et d'informations de présence. **Couche de Liaison de données (Ethernet II) + Couche de transport (Internet Protocol + User Data Protocol).**



QUIC : QUIC est un protocole expérimental, créé par le moteur de recherche Google et présenté au public en 2013. Le nom signifie « Quick UDP Internet Connections » (connexions Internet UDP rapides), car il permet l'envoi rapide de paquets simples via le protocole UDP (User Datagram Protocol) sans connexion. La raison du développement de QUIC était le désir de fournir une alternative aux solutions de sécurité TCP, HTTP/2 et TLS/SSL en développant la même protection mais avec un délai de connexion et de transport réduit, et en permettant des connexions de multiplexage.

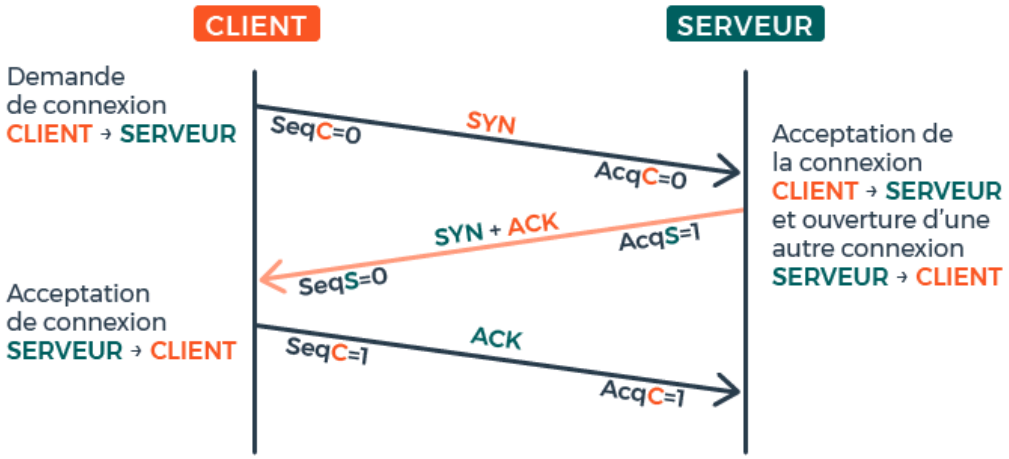


MDNS : Multicast DNS (mDNS) est un service conçu pour aider à la résolution de noms dans les petits réseaux. Toutefois, le mDNS utilise une méthode différente de celle du DNS traditionnel : au lieu de solliciter un serveur de noms, tous les participants au réseau sont directement adressés. Le client correspondant envoie un multicast dans le réseau et demande à quel participant du réseau le nom d’hôte correspond. Le multicast est une forme de communication spécifique dans laquelle un seul message est adressé à un groupe de destinataires. Le groupe peut par exemple être constitué de l’ensemble du réseau ou d’un sous-réseau.

```
232 3.072304    QNAP_7e:1f:2b    Broadcast    ARP    60 Who has 10.10.3.204? Tell 10.10.0.200

Frame 232: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface
Ethernet II, Src: QNAP_7e:1f:2b (24:5e:be:7e:1f:2b), Dst: Broadcast (ff:ff:ff:ff:
Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 24 5e  be 7e 1f 2b 08 06 00 01   ......$^ .~.+....
0010  08 00 06 04 00 01 24 5e  be 7e 1f 2b 0a 0a 00 c8   ......$^ .~.+....
0020  00 00 00 00 00 00 0a 0a  03 cc 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

```
233 3.072842    10.10.0.107    255.255.255.255    UDP    292 39282 → 62976 Len=250

Frame 233: 292 bytes on wire (2336 bits), 292 bytes captured (2336 bits) on inter
Ethernet II, Src: DLinkInterna_b8:82:40 (00:ad:24:b8:82:40), Dst: Broadcast (ff:f
Internet Protocol Version 4, Src: 10.10.0.107, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 39282, Dst Port: 62976
Data (250 bytes)
    Data [truncated]: 760c486839314e6b6234333238436c346c3244732f716f762f2f434e3841
    [Length: 250]

0000  ff ff ff ff ff ff 00 ad  24 b8 82 40 08 00 45 00   ........ $..@..E.
0010  01 16 0f fa 40 00 40 11  1f 69 0a 0a 00 6b ff ff   ....@.@. .i...k..
0020  ff ff 99 72 f6 00 01 02  f7 30 76 0c 48 68 39 31   ...r.... .0v.Hh91
0030  4e 6b 62 34 33 32 38 43  6c 34 6c 32 44 73 2f 71   Nkb4328C l4l2Ds/q
0040  6f 76 2f 2f 43 4e 38 41  6f 41 43 74 4a 4c 69 43   ov//CN8A oACtJLiC
0050  51 41 6f 4b 41 47 73 41  41 41 41 46 41 4a 49 41   QAoKAGsA AAAFAJIA
0060  41 51 41 4b 52 43 31 4d  53 55 35 4c 49 45 46 51   AQAKRC1M SU5LIEFQ
0070  41 41 41 43 41 41 31 42  59 32 4e 6c 63 33 4d 67   AAACAA1B Y2Nlc3Mg
0080  55 47 39 70 62 6e 51 41  41 41 51 41 43 44 51 75   UG9pbnQA AAQACDQu
0090  4e 79 34 7a 4c 6a 41 41  41 41 34 41 42 67 43 74   Ny4zLjAA AA4ABgCt
00a0  4a 4c 69 43 51 41 41 46  41 41 4e 42 4d 51 41 41   JLiCQAAF AANBMQAA
00b0  43 77 41 4f 55 7a 4e 47  4e 6a 46 4a 51 6a 41 77   CwAOUzNG NjFJQjAw
00c0  4d 44 41 7a 4d 67 41 41  43 51 41 4f 52 45 78 4a   MDAzMgAA CQAORExJ
00d0  54 6b 73 74 56 30 78 42  54 69 31 42 55 41 41 41   TkstV0xB Ti1BUAAA
00e0  44 77 41 45 43 67 6f 41  61 77 41 52 41 41 54 2f   DwAECgoA awARAAT/
00f0  41 41 41 41 41 42 30 41  42 55 78 42 54 6a 45 41   AAAAAB0A BUxBTjEA
0100  41 42 41 41 45 41 41 41  41 41 41 41 41 41 41 41   ABAAEAAA AAAAAAAA
0110  41 41 41 41 41 41 41 41  41 41 41 41 47 41 41 42   AAAAAAAA AAAAGAAB
0120  41 41 3d 3d                                        AA==
```

```
234 3.110935    10.10.17.46    162.159.130.234    TCP    54 61153 → 443 [ACK] Seq=1 Ack=394 Win=509 Len=0

Frame 234: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface
Ethernet II, Src: Intel_dc:f6:31 (f0:57:a6:dc:f6:31), Dst: Intel_3a:2e:49 (68:05:
Internet Protocol Version 4, Src: 10.10.17.46, Dst: 162.159.130.234
Transmission Control Protocol, Src Port: 61153, Dst Port: 443, Seq: 1, Ack: 394,

0000  68 05 ca 3a 2e 49 f0 57  a6 dc f6 31 08 00 45 00   h..:.I.W ...1..E.
0010  00 28 75 1c 40 00 80 06  00 00 0a 0a 11 2e a2 9f   .(u.@... ........
0020  82 ea ee e1 01 bb 32 f4  08 b4 e1 31 ca d9 50 10   ......2. ...1..P.
0030  01 fd 40 dc 00 00                                  ..@...
```



Partie 2

```
11 8.076081      0.0.0.0              255.255.255.255      DHCP      342 DHCP Discover - Transaction ID 0x3edf3c3a
12 8.076309      192.168.159.254      192.168.159.147      DHCP      342 DHCP Offer    - Transaction ID 0x3edf3c3a
13 8.076727      0.0.0.0              255.255.255.255      DHCP      342 DHCP Request  - Transaction ID 0x3edf3c3a
14 8.076907      192.168.159.254      192.168.159.147      DHCP      342 DHCP ACK      - Transaction ID 0x3edf3c3a
2 0.008084     192.168.159.2        192.168.159.147      DNS        87 Standard query response 0x6996 A example.com A 93.184.215.14
3 0.008772     192.168.159.147      192.168.159.2        DNS        71 Standard query 0x5673 AAAA example.com
4 0.016855     192.168.159.2        192.168.159.147      DNS        99 Standard query response 0x5673 AAAA example.com AAAA 2606:280…
 5 5.341900    192.168.159.147      224.0.0.251          MDNS       87 Standard query 0x0000 PTR _ipps._tcp.local, "QM" question PTR…
 6 5.354463    192.168.159.2        224.0.0.251          MDNS      792 Standard query response 0x0000 PTR _ipp._tcp.local, "QU" ques…
 14 1.762867   192.168.159.146      192.168.159.147      FTP       124 Response: vsftpd: refusing to run with writable root inside c…
 15 1.762947   192.168.159.146      192.168.159.147      FTP        68 Response:
8737 146.305322    192.168.159.144      192.168.159.146      SMB2      138 KeepAlive Request
8738 146.305739    192.168.159.146      192.168.159.144      SMB2      138 KeepAlive Response
 8981 201.755570   192.168.159.144      128.31.0.62          TLSv1.3    93 Application Data
 8982 201.755663   192.168.159.144      128.31.0.62          TLSv1.3    78 Application Data
```

Partie 3

Installation de Tshark

```
root@Wireshark:~# apt install tshark
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
```

Captures

```
Capturing on 'ens33'
 ** (tshark:5300) 09:37:33.026651 [Main MESSAGE] -- Capture started.
 ** (tshark:5300) 09:37:33.026759 [Main MESSAGE] -- File: "captured_traffic.pcap"
```

```
root@Wireshark:~# tshark -r captured_traffic.pcap -Y "bootp || dns || mdns || ssl || ftp || smb || tls"
Running as user "root" and group "root". This could be dangerous.
    1 0.000000000 192.168.159.1 → 224.0.0.251  MDNS 85 Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QU" question
    2 0.000889885 fe80::7fbc:7450:c8c:bdf1 → ff02::fb     MDNS 105 Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QU" question
    3 1.008075061 192.168.159.1 → 224.0.0.251  MDNS 85 Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QM" question
    4 1.008959561 fe80::7fbc:7450:c8c:bdf1 → ff02::fb     MDNS 105 Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QM" question
    5 14.306240106 fe80::20c:29ff:fe4f:ba96 → ff02::fb     MDNS 180 Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR _nfs._tcp.local, "QM" questi
on PTR _afpovertcp._tcp.local, "QM" question PTR _smb._tcp.local, "QM" question PTR _sftp-ssh._tcp.local, "QM" question PTR _webdavs._tcp.local, "QM" questio
n PTR _webdav._tcp.local, "QM" question
    6 14.306391331 192.168.159.144 → 224.0.0.251  MDNS 188 Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR _nfs._tcp.local, "QM" question PTR _a
fpovertcp._tcp.local, "QM" question PTR _smb._tcp.local, "QM" question PTR _sftp-ssh._tcp.local, "QM" question PTR _webdavs._tcp.local, "QM" question PTR _we
bdav._tcp.local, "QM" question PTR YOUR_HOSTNAME._smb._tcp.local
    7 14.370408858 192.168.159.2 → 224.0.0.251  MDNS 301 Standard query response 0x0000 PTR _smb._tcp.local, "QU" question PTR MacBook Pro de Edouard._smb._t
cp.local SRV 0 0 445 MacBook-Pro-de-Edouard.local TXT TXT AAAA fe80::f7:eedf:e543:fa7f A 10.10.36.3
    8 14.388150569 192.168.159.2 → 224.0.0.251  MDNS 289 Standard query response 0x0000 PTR _smb._tcp.local, "QU" question PTR MacBook Air de Ali._smb._tcp.l
ocal SRV 0 0 445 MacBook-Air-de-Ali.local TXT TXT AAAA fe80::874:9880:3211:c33d A 10.10.35.179
    9 20.149485823 192.168.159.146 → 192.168.159.254 DHCP 326 DHCP Request  - Transaction ID 0xdeaf7d26
   10 20.149486016 192.168.159.254 → 192.168.159.146 DHCP 342 DHCP ACK      - Transaction ID 0xdeaf7d26
   11 142.332084972 fe80::20c:29ff:fe4f:ba96 → ff02::fb     MDNS 180 Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR _nfs._tcp.local, "QM" quest
ion PTR _afpovertcp._tcp.local, "QM" question PTR _smb._tcp.local, "QM" question PTR _sftp-ssh._tcp.local, "QM" question PTR _webdavs._tcp.local, "QM" questi
on PTR _webdav._tcp.local, "QM" question
   12 142.332466674 192.168.159.144 → 224.0.0.251  MDNS 188 Standard query 0x0000 PTR _ftp._tcp.local, "QM" question PTR _nfs._tcp.local, "QM" question PTR _
afpovertcp._tcp.local, "QM" question PTR _smb._tcp.local, "QM" question PTR _sftp-ssh._tcp.local, "QM" question PTR _webdavs._tcp.local, "QM" question PTR _w
ebdav._tcp.local, "QM" question PTR YOUR_HOSTNAME._smb._tcp.local
   13 142.361629705 192.168.159.2 → 224.0.0.251  MDNS 291 Standard query response 0x0000 PTR _smb._tcp.local, "QU" question PTR MacBook Air de Léa._smb._tcp.
local SRV 0 0 445 MacBook-Air-de-Lea.local TXT TXT AAAA fe80::14e5:9f60:2c80:de22 A 10.10.35.121
   14 142.391472298 192.168.159.2 → 224.0.0.251  MDNS 301 Standard query response 0x0000 PTR _smb._tcp.local, "QU" question PTR MacBook Pro de Edouard._smb._
tcp.local SRV 0 0 445 MacBook-Pro-de-Edouard.local TXT TXT AAAA fe80::f7:eedf:e543:fa7f A 10.10.36.3
   15 142.490635362 192.168.159.2 → 224.0.0.251  MDNS 289 Standard query response 0x0000 PTR _smb._tcp.local, "QU" question PTR MacBook Air de Ali._smb._tcp.
local SRV 0 0 445 MacBook-Air-de-Ali.local TXT TXT AAAA fe80::874:9880:3211:c33d A 10.10.35.179
   16 144.301049324 192.168.159.1 → 224.0.0.251  MDNS 77 Standard query 0x0000 ANY DylanCapron.local, "QM" question
```