Dersim Besiktas BTS SIO 1

TP 3 – Les Ports Logiciels

SOMMAIRE :

Objectif :

L’objectif de ce TP est de **comprendre et analyser le fonctionnement des échanges réseau entre deux machines**, en utilisant des **outils d’administration et d’analyse réseau**, ainsi que l’étude de **captures de trames HTTP** réalisées avec Wireshark

1 . Conexion Bureau à Distance

La commande **netstat -no** est utilisée sous Windows pour afficher les connexions réseau actives et les ports utilisés sur la machine, avec des informations détaillées sur les processus qui les exploitent

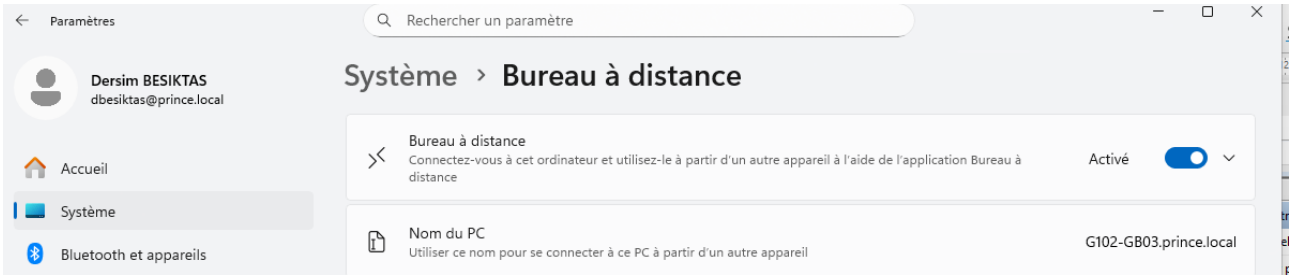
```
C:\Windows\System32>netstat -no

Connexions actives

  Proto  Adresse locale         Adresse distante       État
  TCP    172.17.2.8:57242       104.75.232.13:80       TIME_WAIT       0
  TCP    172.17.2.8:57247       20.42.65.89:443        TIME_WAIT       0
  TCP    172.17.2.8:57260       172.17.254.5:445       ESTABLISHED     4
  TCP    172.17.2.8:57261       95.100.133.25:443      ESTABLISHED     15188
  TCP    172.17.2.8:57262       40.126.32.140:443      ESTABLISHED     1800
  TCP    172.17.2.8:57264       102.133.96.237:443     ESTABLISHED     15188
  TCP    172.17.2.8:57266       172.17.2.21:7680       SYN_SENT        11468
  TCP    172.17.2.8:57267       13.107.4.254:443       ESTABLISHED     15188
  TCP    172.17.2.8:57268       150.171.44.254:443     ESTABLISHED     15188
  TCP    172.17.2.8:57269       204.79.197.222:443     ESTABLISHED     15188
  TCP    172.17.2.8:59945       98.66.133.186:443      ESTABLISHED     4340
  TCP    172.17.2.8:59962       172.17.254.5:445       ESTABLISHED     4
  TCP    172.17.2.8:60248       23.200.86.235:443      CLOSE_WAIT      8360
  TCP    172.17.2.8:60501       172.17.2.3:7680        ESTABLISHED     11468
  TCP    172.17.2.8:60506       172.17.2.23:7680       ESTABLISHED     11468
  TCP    172.17.2.8:60507       172.17.2.19:7680       ESTABLISHED     11468
  TCP    172.17.2.8:60521       172.17.2.19:7680       ESTABLISHED     11468
  TCP    172.17.2.8:60553       172.17.2.23:7680       ESTABLISHED     11468
  TCP    172.17.2.8:60554       172.17.2.9:7680        ESTABLISHED     11468
  TCP    172.17.2.8:60557       172.17.2.19:7680       ESTABLISHED     11468
  TCP    172.17.2.8:60558       172.17.2.1:7680        ESTABLISHED     11468
  TCP    172.17.2.8:60560       172.17.2.3:7680        ESTABLISHED     11468
```

L’adresse IP de la machine de mon voisin est 172.17.2.14.

J’effectue un ping vers cette adresse afin de vérifier la connectivité réseau entre ma machine physique et celle de mon voisin

```
C:\Windows\System32>ping 172.17.2.14

Envoi d’une requête 'Ping'  172.17.2.14 avec 32 octets de données :
Réponse de 172.17.2.14 : octets=32 temps=2 ms TTL=128
Réponse de 172.17.2.14 : octets=32 temps=2 ms TTL=128
Réponse de 172.17.2.14 : octets=32 temps=1 ms TTL=128
Réponse de 172.17.2.14 : octets=32 temps=1 ms TTL=128

Statistiques Ping pour 172.17.2.14:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 1ms, Maximum = 2ms, Moyenne = 1ms
```

Depuis les paramètres du système, j'ai autorisé l'accès au Bureau à distance afin de permettre la connexion à ma machine depuis un autre poste du réseau (mon voisin)

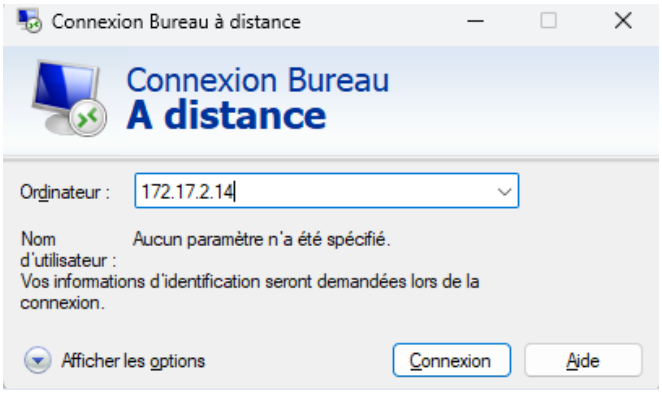


J'exécute la commande **`netstat -an`** depuis l'invite de commandes afin d'obtenir un **aperçu détaillé des connexions TCP** ainsi que des **ports ouverts** en cours d'utilisation sur le poste

```
C:\Windows\System32>netstat -an

Connexions actives

  Proto  Adresse locale         Adresse distante       État
  TCP    0.0.0.0:80             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:2179           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:3306           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:3307           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49669          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49670          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49677          0.0.0.0:0              LISTENING
  TCP    127.0.0.1:27017        0.0.0.0:0              LISTENING
  TCP    172.17.2.8:139         0.0.0.0:0              LISTENING
  TCP    172.17.2.8:57424       20.42.65.89:443        TIME_WAIT
  TCP    172.17.2.8:57425       172.17.254.1:135       TIME_WAIT
  TCP    172.17.2.8:57426       172.17.254.1:49666     TIME_WAIT
```

Question : Quel est le port d'écoute du serveur Terminal Serveur ?
Réponse : Le port d'écoute est le port 3389

J’ouvre l’application **Bureau à distance** sur mon poste, puis je saisis **l’adresse IP de la machine de mon voisin** afin d’établir une connexion à distance

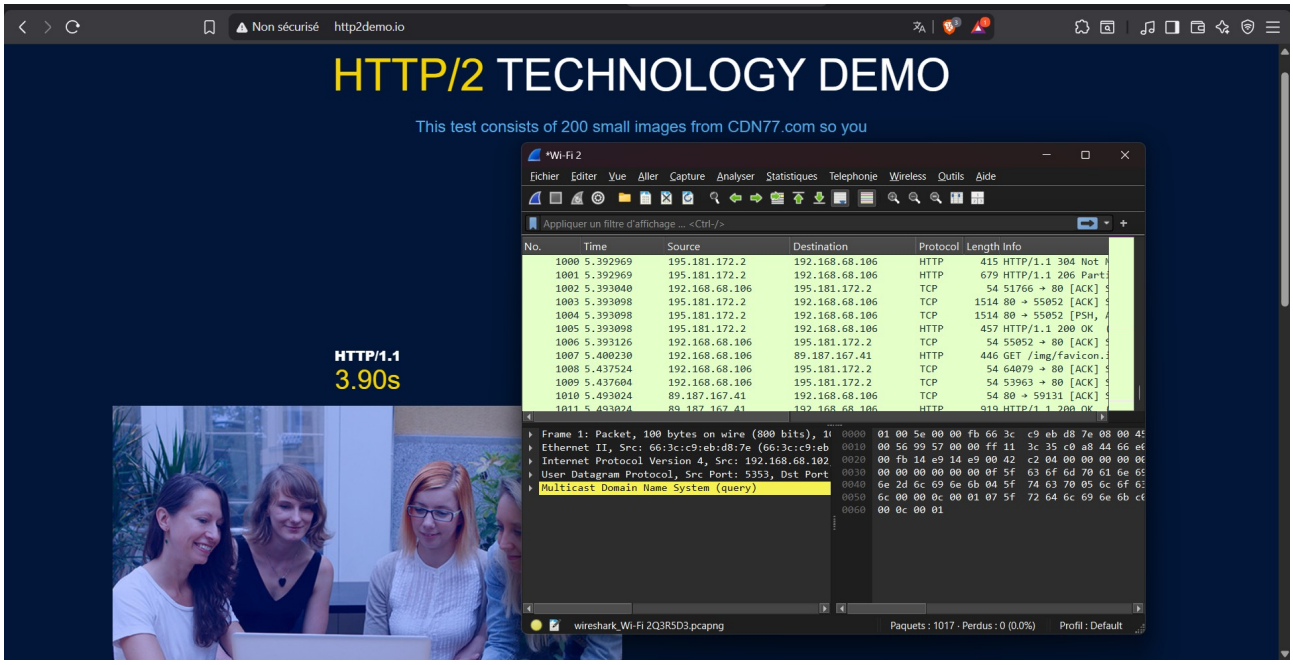


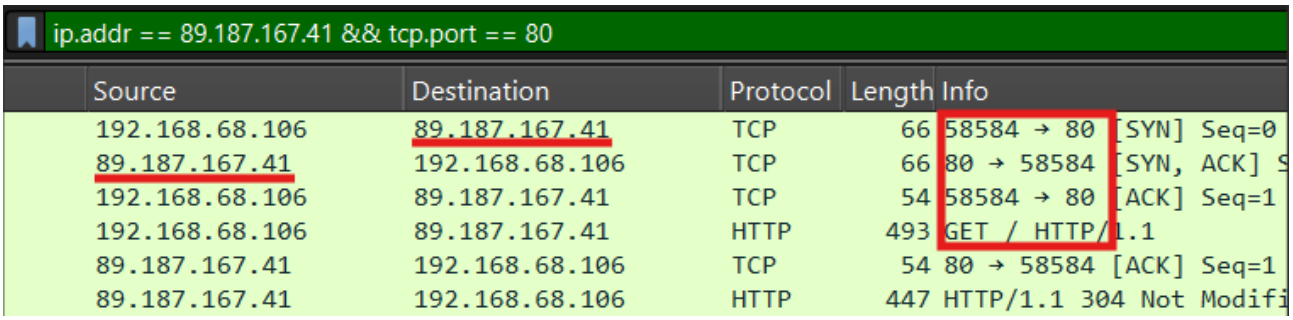
Une fois connecté à la machine distante via le Bureau à distance, je lance la commande **`ipconfig`**. Le résultat confirme que l’adresse IP de la machine correspond bien à **celle de mon voisin (172.17.2.14)**

```
C:\Users\dbesiktas>ipconfig

Configuration IP de Windows


Carte Ethernet vEthernet (Default Switch) :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6 de liaison locale. . . . .: fe80::2e35:c7bd:7baf:f0e%21
   Adresse IPv4. . . . . . . . . . . . . .: 172.19.0.1
   Masque de sous-réseau. . . . . . . . . : 255.255.240.0
   Passerelle par défaut. . . . . . . . . :

Carte Ethernet Ethernet :

   Suffixe DNS propre à la connexion. . . : prince.local
   Adresse IPv6 de liaison locale. . . . .: fe80::9072:83:49b8:adf%12
   Adresse IPv4. . . . . . . . . . . . . .: 172.17.2.14
   Masque de sous-réseau. . . . . . . . . : 255.255.0.0
   Passerelle par défaut. . . . . . . . . : 172.17.250.3
```

2. Capture de trames HTTP

Je lance **Wireshark en tant qu'administrateur** afin de pouvoir **capturer les trames réseau** générées lors de la navigation sur le site **http://www.http2demo.io/**





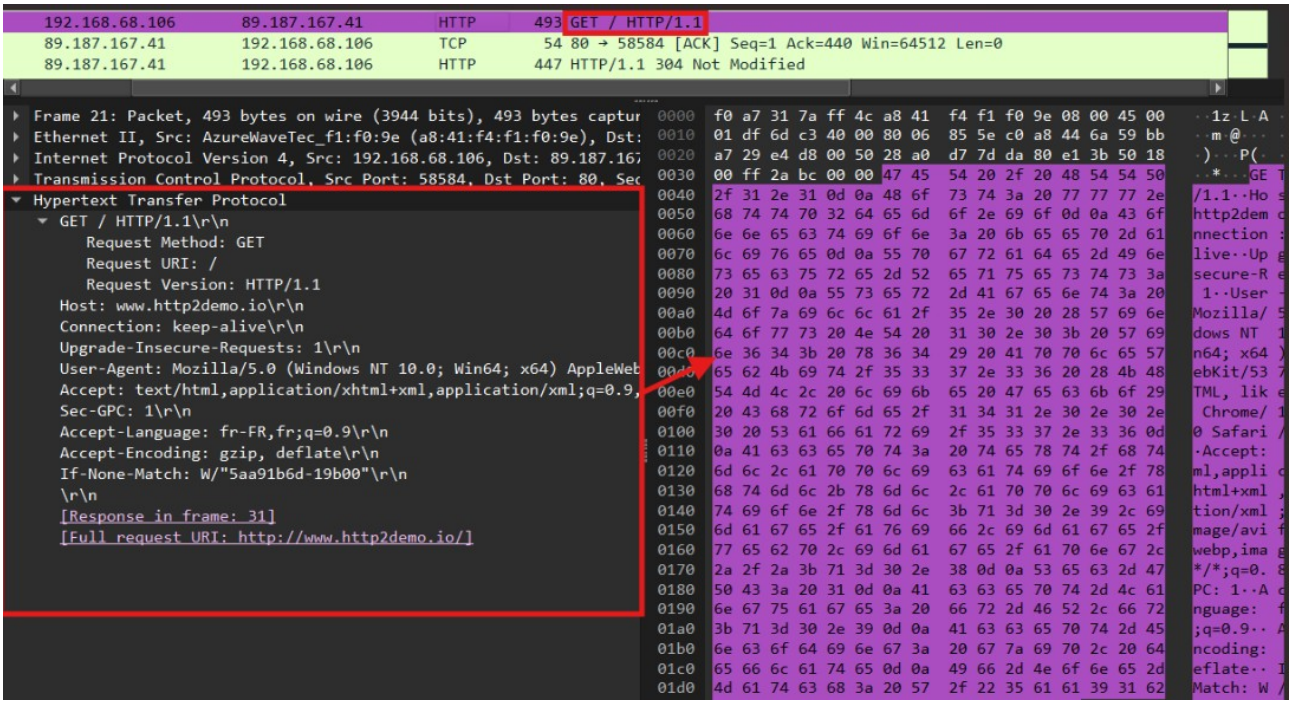
Ce filtre permet de **n'afficher que les trames HTTP** échangées entre ma machine (**192.168.68.106**) et le serveur **89.187.167.41** sur le **port TCP 80** (HTTP).

On observe dans la capture les **étapes de la communication TCP** :

1. **SYN → SYN, ACK → ACK** : établissement de la connexion TCP (Three-Way Handshake).
2. **GET / HTTP/1.1** : envoi de la requête HTTP du client vers le serveur.

```
C:\Windows\System32>nslookup www.http2demo.io
Serveur :   bbox.lan
Address:  192.168.1.254

Réponse ne faisant pas autorité :
Nom :    1906714720.rsc.cdn77.org
Addresses:  2a02:6ea0:ca00::52
          2a02:6ea0:ca00::13
          2a02:6ea0:ca00::7
          2a02:6ea0:ca00::8
          89.187.167.39
          89.187.167.41
          84.17.50.8
          84.17.50.23
Aliases:  www.http2demo.io
```

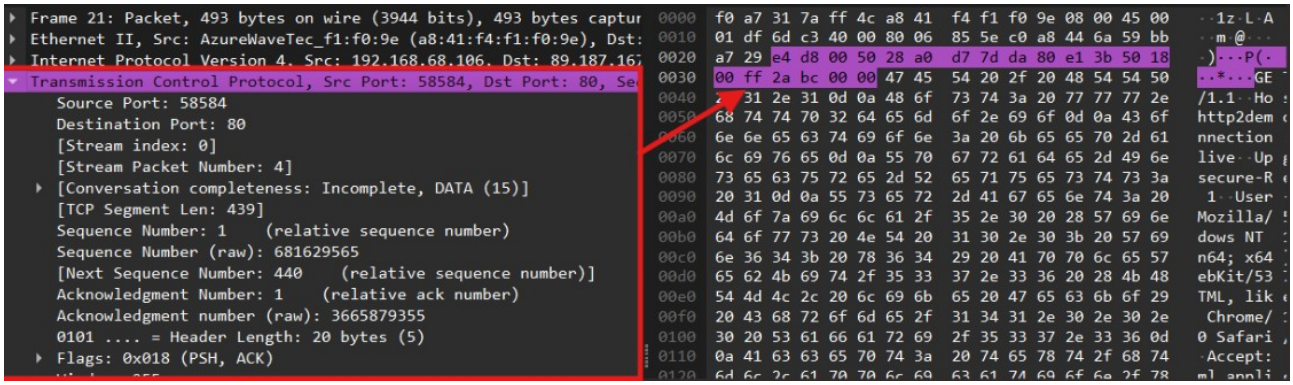
La commande **nslookup** www.http2demo.io permet d'effectuer une **résolution DNS**, c'est-à-dire de connaître **les adresses IP associées à un nom de domaine**



Sur cette capture, on observe une **communication HTTP** entre le client (**192.168.68.106**) et le serveur web (**89.187.167.41**) sur le **port 80**.

Le type de message est une requête GET

La trame sélectionnée correspond à une **requête HTTP GET**

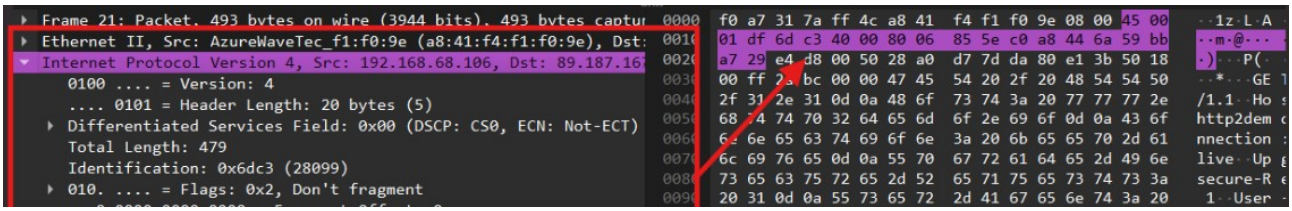


Quel est le nom du protocole de transport utilisé par une trame HTTP ?
→ Le protocole de transport utilisé par une trame HTTP est **TCP (Transmission Control Protocol)**.
HTTP repose sur TCP pour assurer une transmission fiable des données entre le client et le serveur.
Quel est le nom du PDU encapsulant les données applicatives HTTP ?
→ Le **PDU** de la couche transport qui encapsule les données applicatives HTTP est appelé un **segment TCP**.
Les données HTTP sont donc encapsulées dans un segment TCP avant d'être envoyées sur le réseau.

Quelle est la longueur de l'en-tête de transport ?
→ La longueur de l'en-tête TCP est de **20 octets**, ce qui correspond à un en-tête TCP sans options.

Quelles sont les valeurs décimale et hexadécimale correspondant aux ports source et destination ?
→ **Port source :** 58584 (valeur hexadécimale : E4 D8)

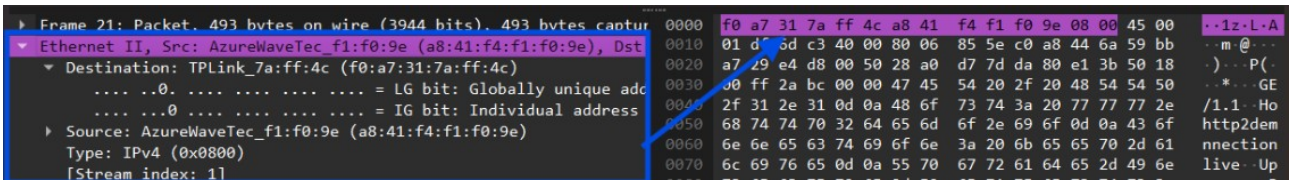
→ **Port destination :** 80 (valeur hexadécimale : 00 50)
Le port 80 correspond au service **HTTP** utilisé par le serveur web.



Quelle est la longueur de l'en-tête de réseau ?
→ La longueur de l'en-tête IP est de **20 octets**.
Cette valeur est indiquée par le champ **Header Length = 5**, ce qui correspond à **5 × 4 = 20 octets**.

Repérez le champ Protocole figurant dans l'en-tête Réseau. Quelle est la valeur présente ?
→ Le champ **Protocole** contient la valeur **6 en décimal** (soit **0x06 en hexadécimal**).
Que signifie-t-elle ?
→ Cette valeur identifie le protocole **TCP (Transmission Control Protocol)** comme protocole de transport.

Quelles sont les valeurs décimales et hexadécimales des adresses IP source et destination ?
→ **Adresse IP source :** 192.168.68.106 = **C0 A8 44 6A**
→ **Adresse IP destination :** 89.187.167.41 = **59 BB A7 29**



Repérez le champ EtherType. Quelle est la valeur contenue ? Que signifie-t-elle ?

→ Le champ **EtherType** contient la valeur **0x0800**.

Cette valeur indique que la trame transporte un **paquet IPv4** dans la couche supérieure (couche réseau).

Quelles sont les valeurs des adresses MAC destination et source ?

→ **Adresse MAC destination :** `f0:a7:31:7a:ff:4c`

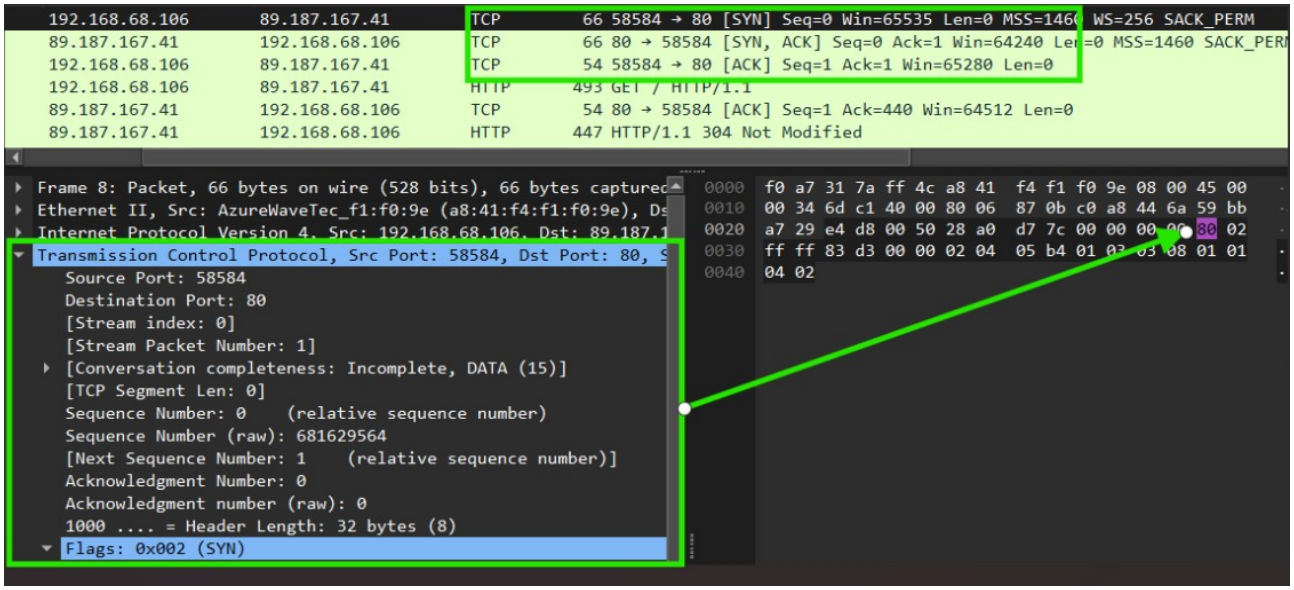
→ **Adresse MAC source :** `a8:41:f4:f1:f0:9e`

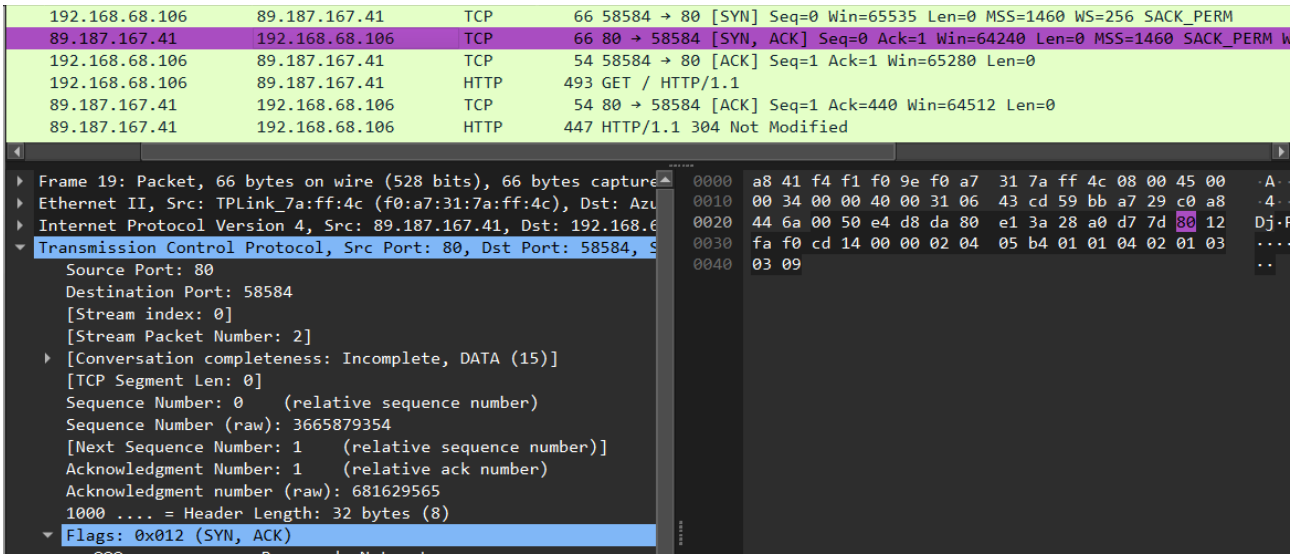
Ces adresses identifient respectivement la **carte réseau du routeur (TP-Link)** et celle de la **machine cliente (AzureWave)**.

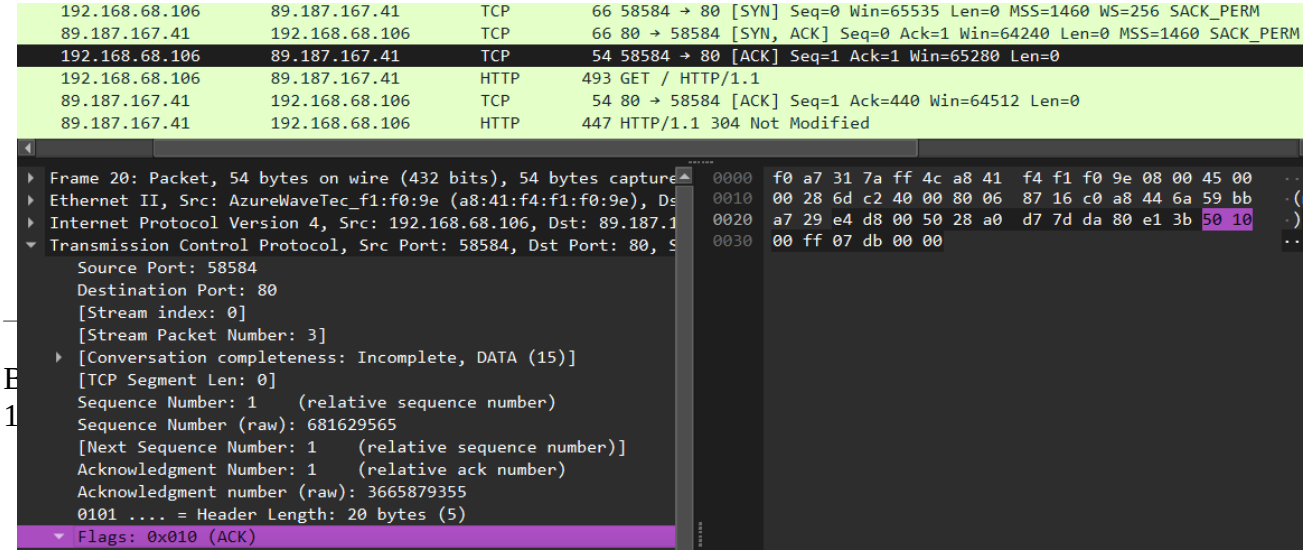
Repérez les trames associées à la mise en place de la connexion TCP entre le client et le serveur (Three-Way Handshake). Pour chacune d'entre-elles, identifiez le champ Flags dans l'en-tête de segment :

1. **1re trame (SYN)** : le client initie la connexion → **Flags = SYN (synchronisation)**
2. **2e trame (SYN, ACK)** : le serveur répond et confirme la synchronisation → **Flags = SYN, ACK**
3. **3e trame (ACK)** : le client accuse réception → **Flags = ACK**

Ces trois trames forment le **Three-Way Handshake**, qui établit une **connexion TCP fiable** avant le transfert des données HTTP.


192.168.68.106 89.187.167.41 TCP 66 58584 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
89.187.167.41 192.168.68.106 TCP 66 80 → 58584 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
192.168.68.106 89.187.167.41 TCP 54 58584 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
192.168.68.106 89.187.167.41 HTTP 493 GET / HTTP/1.1
89.187.167.41 192.168.68.106 TCP 54 80 → 58584 [ACK] Seq=1 Ack=440 Win=64512 Len=0
89.187.167.41 192.168.68.106 HTTP 447 HTTP/1.1 304 Not Modified
Frame 8: Packet, 66 bytes on wire (528 bits), 66 bytes captured
Ethernet II, Src: AzureWaveTec_f1:f0:9e (a8:41:f4:f1:f0:9e), Ds
Internet Protocol Version 4, Src: 192.168.68.106, Dst: 89.187.1
Transmission Control Protocol, Src Port: 58584, Dst Port: 80, S
Source Port: 58584
Destination Port: 80
[Stream index: 0]
[Stream Packet Number: 1]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 0 (relative sequence number)
Sequence Number (raw): 681629564
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 0
Acknowledgment number (raw): 0
1000 .... = Header Length: 32 bytes (8)
Flags: 0x002 (SYN)
0000 f0 a7 31 7a ff 4c a8 41 f4 f1 f0 9e 08 00 45 00
0010 00 34 6d c1 40 00 80 06 87 0b c0 a8 44 6a 59 bb
0020 a7 29 e4 d8 00 50 28 a0 d7 7c 00 00 00 00 80 02
0030 ff ff 83 d3 00 00 02 04 05 b4 01 03 03 08 01 01
0040 04 02


192.168.68.106 89.187.167.41 TCP 66 58584 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
89.187.167.41 192.168.68.106 TCP 66 80 → 58584 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
192.168.68.106 89.187.167.41 TCP 54 58584 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
192.168.68.106 89.187.167.41 HTTP 493 GET / HTTP/1.1
89.187.167.41 192.168.68.106 TCP 54 80 → 58584 [ACK] Seq=1 Ack=440 Win=64512 Len=0
89.187.167.41 192.168.68.106 HTTP 447 HTTP/1.1 304 Not Modified
Frame 19: Packet, 66 bytes on wire (528 bits), 66 bytes capture
Ethernet II, Src: TPLink_7a:ff:4c (f0:a7:31:7a:ff:4c), Dst: Azu
Internet Protocol Version 4, Src: 89.187.167.41, Dst: 192.168.6
Transmission Control Protocol, Src Port: 80, Dst Port: 58584, S
Source Port: 80
Destination Port: 58584
[Stream index: 0]
[Stream Packet Number: 2]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 0 (relative sequence number)
Sequence Number (raw): 3665879354
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 681629565
1000 .... = Header Length: 32 bytes (8)
Flags: 0x012 (SYN, ACK)
0000 a8 41 f4 f1 f0 9e f0 a7 31 7a ff 4c 08 00 45 00
0010 00 34 00 00 40 00 31 06 43 cd 59 bb a7 29 c0 a8
0020 44 6a 00 50 e4 d8 da 80 e1 3a 28 a0 d7 7d 80 12
0030 fa f0 cd 14 00 00 02 04 05 b4 01 01 04 02 01 03
0040 03 09


192.168.68.106 89.187.167.41 TCP 66 58584 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
89.187.167.41 192.168.68.106 TCP 66 80 → 58584 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
192.168.68.106 89.187.167.41 TCP 54 58584 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
192.168.68.106 89.187.167.41 HTTP 493 GET / HTTP/1.1
89.187.167.41 192.168.68.106 TCP 54 80 → 58584 [ACK] Seq=1 Ack=440 Win=64512 Len=0
89.187.167.41 192.168.68.106 HTTP 447 HTTP/1.1 304 Not Modified
Frame 20: Packet, 54 bytes on wire (432 bits), 54 bytes capture
Ethernet II, Src: AzureWaveTec_f1:f0:9e (a8:41:f4:f1:f0:9e), Ds
Internet Protocol Version 4, Src: 192.168.68.106, Dst: 89.187.1
Transmission Control Protocol, Src Port: 58584, Dst Port: 80, S
Source Port: 58584
Destination Port: 80
[Stream index: 0]
[Stream Packet Number: 3]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 681629565
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 3665879355
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
0000 f0 a7 31 7a ff 4c a8 41 f4 f1 f0 9e 08 00 45 00
0010 00 28 6d c2 40 00 80 06 87 16 c0 a8 44 6a 59 bb
0020 a7 29 e4 d8 00 50 28 a0 d7 7d da 80 e1 3b 50 10
0030 00 ff 07 db 00 00

Que signifie le contenu de ce champ pour chacun des 3 segments TCP ?

1. **Segment 1 – SYN** :
   Le bit **SYN (synchronize)** est activé → il sert à **initialiser une connexion** TCP et à **synchroniser** entre le client et le serveur.
2. **Segment 2 – SYN, ACK** :
   Le serveur répond avec **SYN, ACK** --> cela signifie qu'il **accepte la demande de connexion** et **envoie à son tour une demande de connexion**, tout en **accusant la réception** du SYN du client.
3. **Segment 3 – ACK** :
   Le client envoie un **ACK** pour **confirmer la réception** du SYN, ACK du serveur.
   À ce moment, la connexion TCP est **établie** et les deux parties peuvent échanger des données.

Quelle est la raison de la mise en place de ce mode connecté ?

TCP fonctionne en **mode connecté** pour garantir une **communication fiable** entre le client et le serveur.
Ce mécanisme (Three-Way Handshake) permet :

- **Établir une session** stable avant tout échange
- **Synchroniser** pour assurer l'ordre des données
- **Éviter la perte** de paquets.