utilisation de Wireshark

Services et transport de paquets vers les applications

# Table des matières

[Table des matières 1](#_Toc185332348)

[1 : La modélisation en couche : 3](#_Toc185332349)

[1.1 : La modélisation en couche : 3](#_Toc185332350)

[1.1.1 : Le modèle TCP/IP : 3](#_Toc185332351)

[1.2 : Modèle et protocoles en couches : le processus d’encapsulation des données. 3](#_Toc185332352)

[1.3 : Exercice : 4](#_Toc185332353)

[1.1.2 : Trame numéro 1 : 4](#_Toc185332354)

[1.1.3 : Trame numéro 2 : 6](#_Toc185332355)

[2 Prise en main de Wireshark : 7](#_Toc185332356)

[1.4 : Analyse de trame : 7](#_Toc185332357)

[1.1.4 : Etude de la trame 1 : 7](#_Toc185332358)

[1.1.5 : Etude de la trame 2 : 10](#_Toc185332359)

[1.5 : Capturer et analyser les données ICMP locales : 12](#_Toc185332360)

[1.1.6 : Capture des données : 12](#_Toc185332361)

[1.1.7 : Examination des données capturées : 14](#_Toc185332362)

[1.7 : Capturer et analyser les données ICMP locales : 15](#_Toc185332363)

[1.1.8 : Capture les données de l’interface : 15](#_Toc185332364)

[1.1.9 : Examen et analyse des données à partir des hôtes distants : 16](#_Toc185332365)

[1.1.10 : Analyse les données ICMPv6 locales 16](#_Toc185332366)

[1.8 : Etthercap analyse des trames comme un attaquant simulation MAN in the midle 17](#_Toc185332367)

[3 Capture de trame liées à HTTP : 19](#_Toc185332368)

[1.9 : MTU 20](#_Toc185332369)

[4 Application pratiques de Wireshark HTTPS : 20](#_Toc185332370)

[5 NetworkMiner : 22](#_Toc185332371)

[1.10 : Première étude : 22](#_Toc185332372)

[1.11 : Etude des différents outils : 24](#_Toc185332373)

[6 Intelligence artificielle : 24](#_Toc185332374)

[1.12 : ChatGPT : 24](#_Toc185332375)

[1.13 : MistralAI le Chat : 26](#_Toc185332376)

[1.14 : Vectra AI (Vectra AI - Advanced AI Security - Stop Cyberattacks Fast): 28](#_Toc185332377)

# : La modélisation en couche :

## La modélisation en couche :

### Le modèle TCP/IP :

1)

|  |  |
| --- | --- |
| Les couches TCP/IP : | Protocoles connus |
| Couche 4 (application) | SMTP pour l’envoi de mails, HTTP(S) pour le transfert hypertexte, FTP pour le transport de fichiers, le DHCP pour coordonner l’allocation dynamique d’adresses, le DNS pour résoudre les noms de domaines en adresses IP et le SSH pour les connexions sécurisées. |
| Couche 3 (Transport) | UDP envoient des informations mais sans garantie de réception et TCP envoie des informations avec une garantit de réception. |
| Couche 2 (Internet) | IP gère l’adressage et le routage des paquets (ICMP, ARP). |
| Couche 1 (Accès réseau) | MAC identifie les équipements au niveau des réseaux locaux (câbles Ethernet, fibre, wifi, Bluetooth). |

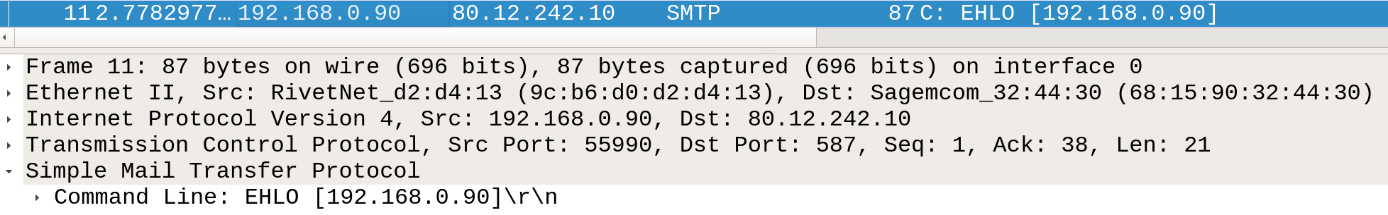
2) Les switchs utilisent les couches 1 et 2 car ils se servent que des adresses MAC. Les routeurs ont besoin des couches 1, 2 et 3 pour connecter les réseaux entre eux.

## Modèle et protocoles en couches : le processus d’encapsulation des données.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| @MAC destination | @MAC source | IP source | IP destination | Port source | Port Destination |
| 74:46:a0:9c:54:1e | 74:46:a0:9b:16 :cb | 192.168.1.2 | 192.168.1.1 | 80 | 49200 |

La trame est arrivée à destination et veut envoyer une confirmation à l’envoyeur. Il faut donc inverser les informations.

a)

La trame établit une connexion SMTP entre un poste (192.168.0.90) et un serveur (80.12.242.10) pour envoyer des mails. Il y a écrit Simple Mail Transfer Protocol, qui veut dire SMTP et le port 587 est utilisé pour envoyer des mails avec SMTP.

|  |  |
| --- | --- |
| Couches du modèle TCP/IP | Protocole associé |
| 4 - Application | SMTP |
| 3 - Transport | TCP |
| 2 - Internet | IP |
| 1 – Accès réseau | Ethernet |

b)

c)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| @MAC destination | @MAC source | IP source | IP destination | Port source | Port Destination |
| 9c :b6 :d0 :d2 :d4 :13 | 68 :15 :90 :32 :44 :30 | 192.168.0.90 | 80.12.242.10 | 55990 | 587 |

d)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| @MAC destination | @MAC source | IP source | IP destination | Port source | Port Destination |
| 68 :15 :90 :32 :44 :30 | 9c :b6 :d0 :d2 :d4 :13 | 80.12.242.10 | 192.168.0.90 | 587 | 55990 |

## Exercice :

### Trame numéro 1 :

e)

|  |  |
| --- | --- |
|  | C’est le FTP (File Transfer Protocol). |

Il sert à transporter des fichiers entre deux utilisateurs dans le protocole TCP/IP.

f)

|  |  |
| --- | --- |
|  | C’est le TCP (Transmission Control Protocol). |

Il permet une communication fiable entre les dispositifs d’un réseau.

g)

|  |  |
| --- | --- |
|  | Le port 49681 est le port client ouvert. |

h)

|  |  |
| --- | --- |
|  | La longueur de l’en-tête TCP est de 20 bytes. |

i)

|  |  |
| --- | --- |
|  | La taille de la fenêtre est de 255. |

j)

|  |  |
| --- | --- |
|  | Il a un accusé de réception (ACK). |

### Trame numéro 2 :

k)

|  |  |
| --- | --- |
|  | C’est le DNS (Domain Name System) |

12) UDP est un protocole léger qui n’a pas besoin de connexion, ce qui permet des résolutions DNS rapides.  Les requêtes DNS sont dans la plupart des cas de petite taille et n'ont donc pas besoin de mécanismes de retransmission comme TCP.

l)

|  |  |
| --- | --- |
|  | C’est le port 51530, un port éphémère qui est utilisé pour établir une connexion éphémère entre deux postes. Il est libre après la connexion terminée et peut être attribué à d’autres connexions. |

m) la longueur indiquée dans l'en-tête UDP correspond à la taille totale du segment UDP en octets. Lenght indique le nombre d'octets à lire et indique la fin du segment UDP.

# Prise en main de Wireshark :

Wireshark est un outil d’analyse de trafic réseau permettant d’observer, de filtrer et d’interpréter en temps réel les données circulant sur un réseau. Pour un administrateur réseau, son utilisation consiste principalement à diagnostiquer des problèmes de communication, identifier les causes de pannes ou de ralentissements, et vérifier la conformité des protocoles employés. En capturant et décomposant les paquets, Wireshark offre une visibilité granulaire sur les échanges, aidant ainsi à détecter d’éventuelles failles de sécurité, à optimiser les performances et à assurer une meilleure maîtrise de l’infrastructure informatique.

a)

|  |  |
| --- | --- |
|  | L’adresse physique est 00-15-5D-B9-10-53 et l’adresse IP est 172.19.48.1. |

b) L’adresse IP de ma voisine (Shayma) est 172.31.1.106.

## Analyse de trame :

### Etude de la trame 1 :

a)

Champs de la trame Ethernet :

|  |  |  |
| --- | --- | --- |
| Adresse Ethernet (MAC) Destination | ff:ff:ff:ff:ff:ff |  |
| Adresse Ethernet Source | 09 :ab :14 :d8 :05 :48 |  |
| EtherType | ARP |  |
| Données | Il recherche l’adresse MAC de l’appareil qui a 125.18.110.3 comme adresse IP. Il demande de répondre à son adresse IP |  |

L’adresse MAC de destination est une adresse broadcast, qui est destinée à toutes les machines du réseau pour soit découvrir des appareils sur ce réseau (ARP) soit demander des requêtes DHCP (pour obtenir une adresse IP). Ici c’est une requête ARP.

Les informations sont trouvables dans la zone 1 comme dans la zone 2 (qui donne tous les détails de la trame) :

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

On voie une trame ARP avec une requête demandant qui possède l’adresse IP 125.18.110.3. L’adresse source 09: ab:14:d8:05:48 envoie une requête en broadcast. Cela signifie que l’hôte veut connaître l’adresse MAC correspondant à cette IP. Le protocole ARP est utilisé ici pour résoudre l’adresse IP en une adresse MAC. Cette opération est essentielle pour permettre la communication au niveau de la couche 2 (liaison).

Champs du datagramme ARP :

|  |  |
| --- | --- |
| Type de réseau | Ethernet |
| Protocole | IPv4 |
| Taille adresse physique | 6 |
| Taille adresse logique | 4 |
| Opération | 1 (ARP) |
| Adresse physique source | 09 :ab :14 :d8 :05 :48 |
| Adresse IP source | 125.5.48.10 |
| Adresse MAC cible | 00:00:00:00:00:00 |
| Adresse protocole destination | 125.18.110.3 |

Explication :

Une image contenant texte, logiciel, Police, capture d’écran

Description générée automatiquement

Cette capture montre une analyse détaillée de la requête ARP. On identifie le type de réseau (Ethernet), le protocole (IPv4), et les adresses physiques et IP. L’adresse MAC source 09:ab:14:d8:05:48 et l’adresse IP source 125.5.48.10 sont les identifiants de l’émetteur. L’adresse cible MAC est 00:00:00:00:00:00, car l’émetteur ne la connaît pas encore. L’adresse IP cible 125.18.110.3 est celle recherchée. Cette analyse permet de comprendre le fonctionnement de la résolution d’adresse pour établir une communication locale.

b) L’objet du message est de trouver l’adresse MAC et l’emplacement du poste 125.18.110.3. Le protocole IPv4 est cohérent car l’ARP fait le lien entre une adresse MAC et une adresse IP. IPv4 achemine les paquets avec les adresses IPv4, la trame contient l’adresse IP du destinataire, il va donc s’en servir pour acheminer les paquets vers le destinataire.

Vérification :

|  |  |  |
| --- | --- | --- |
| Type de réseau | Ethernet | Cohérent |
| Protocole | IPv4 | Cohérent |
| Taille adresse physique | 6 | Longueur standard |
| Taille adresse logique | 4 | Longueur standard |
| Opération | 1 (ARP) | Cohérent |
| Adresse physique source | 09 :ab :14 :d8 :05 :48 | Cohérent |
| Adresse IP source | 125.5.48.10 | Cohérent |
| Adresse MAC cible | 00:00:00:00:00:00 | Inconnu mais c’est le but de la trame |
| Adresse protocole destination | 125.18.110.3 | Adresse recherchée |

c) Une trame a une taille minimale de 64 octets mais la trame 1 fait 42 octets. Il manque le préambule (7 octets) , le SFD (1 octets) et le FCS (4 octets) car ils sont générés automatiquement et ne sont pas capturés par Wireshark donc c’est normal.

Le padding manque aussi, il permet à la trame d’atteindre 64 octets en en rajoutant. Il en faut donc 10 octets en plus.

La norme IEEE 802.3 exige que les trames fassent au minimum 64 octets pour qu’elles soient traitées et reconnues correctement.

SFD (Start Frame Delimiter) marque le début effectif des données utiles d'une trame Ethernet.

FCS (Frame Check Sequence) détecte les erreurs qui peuvent survenir lors de la transmission des données sur le réseau.

### Etude de la trame 2 :

a) Champs de la trame :

|  |  |
| --- | --- |
| Adresse Ethernet (MAC) Destination | 00:0f:1f:13:34:9a |
| Adresse Ethernet Source | 00 :01 :30 :4a :38 :00 |
| EtherType | Internet Control Message Protocol (ICMP). |
| Données | LE ping a atteint son destinataire et la réponse est revenue correctement à son envoyeur. |

Une image contenant texte, logiciel, Police, nombre

Description générée automatiquement

Ici, il s’agit d’une trame ICMP, spécifiquement une réponse à un ping (Echo (ping) reply). L’adresse IP source est 139.124.187.4 et l’adresse destination 172.16.203.109. Au niveau de la couche 2, l’adresse MAC source est 00:01:30:4a:38:00 et l’adresse MAC destination est 00:0f:1f:13:34:9a. Le protocole ICMP permet de diagnostiquer les problèmes de connectivité. Cette trame montre que le ping a réussi et que la réponse a bien été reçue.

Champs du datagramme :

|  |  |
| --- | --- |
| Type de réseau | Ethernet II |
| Protocole | ICMP |
| Taille adresse physique | Inconnu, il est de 6 octets |
| Taille adresse logique | Inconnu, il est de 4 octets. |
| Opération | Echo Reply (ping) |
| Adresse physique source | 00 :01 :30 :4a :38 :00 |
| Adresse IP source | 139.124.187.4 |
| Adresse MAC cible | 00 :0f :1f :13f34 :9a |
| Adresse protocole destination | 172.16.203.109 |

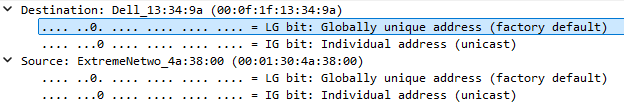
Cette capture détaille encore la trame ICMP vue précédemment. On voit les adresses MAC source 00:01:30:4a:38:00 et destination 00:0f:1f:13:34:9a au niveau Ethernet II. L’adresse IP source 139.124.187.4 et l’adresse IP destination 172.16.203.109 sont clairement identifiées. Le protocole ICMP est utilisé avec le type 0, ce qui correspond à une réponse Echo (ping) reply. Cette capture permet de vérifier la connectivité réseau en confirmant que le paquet ICMP a bien été transmis et reçu.

Une image contenant texte, capture d’écran, diagramme, Police

Description générée automatiquement

Sur cette capture, nous avons une analyse du protocole Ethernet et IP. L'adresse MAC source est celle de "ExtremeNetwo" et celle de destination est "Dell", ce qui correspond à des équipements sur le réseau. En dessous, on remarque le protocole utilisé qui est ICMP (protocole de contrôle de messages Internet). L'adresse IP source et destination sont visibles, de même que l'opération effectuée, qui est un "Echo request" (ping). Cette capture permet de diagnostiquer un test de connectivité entre deux équipements à travers un réseau, en utilisant ICMP pour vérifier si un appareil répond à une requête de ping.

Les tailles d’adresses MAC et IP sont inconnues car la trame est mal configurée.



Wireshark n’arrive pas à analyser correctement les adresses IP et MAC.

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

De plus il y a des données (data) dans une trame de connectivité réseau, ce qui n’a rien à faire là. La trame essaie peut-être de faire passer des informations pour injecter du code malveillant. M Drogue n’aurait aucun objectif à infecter les postes de son réseau, celle-là est donc sans danger.

b) L’adresse MAC source change à chaque saut. Pour arriver à destination, elle a forcément dû passer par un routeur ou un switch. C’est donc l’adresse MAC de l’appareil qui a transmis le paquet à l’hôte destinataire et non celle de l’hôte source.

## Capturer et analyser les données ICMP locales :

### Examination des données capturées :

J’ai ping Shayma et je vois une série de requêtes ICMP entre deux adresses IP : 172.31.1.67 (source) et 172.31.1.106 (destination). Ces échanges montrent une communication de type ping pour vérifier la connectivité entre ces deux hôtes. La source et la destination alternent, indiquant que des requêtes et réponses ICMP se succèdent correctement. Cela permet de diagnostiquer le bon fonctionnement du réseau entre ces deux machines.

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Adresse IP de Shayma :172.31.1.106

a) Une image contenant texte, Police, nombre, capture d’écran

Description générée automatiquementSur cette capture, on voit bien que l’adresse MAC source e0:73:e7:b2:64:d7 correspond à mon adresse MAC source. Cela est logique puisque la requête ICMP (ping) provient de ma machine. Wireshark capture et analyse cette trame en direct depuis mon interface réseau, ce qui explique pourquoi l’adresse MAC source affichée est la mienne. Cela confirme que le ping est émis correctement depuis mon poste vers l’adresse 172.31.1.106.

|  |  |
| --- | --- |
|  |  |

L’adresse MAC e0:73:e7:b2:66:5f correspond à l’interface réseau de la machine 172.31.1.106. Cette information est vérifiée via la commande ipconfig /all,

## Capturer et analyser les données ICMP locales :

### Capture les données de l’interface :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

On voit une commande ping. Les adresses IP et les temps de réponse sont affichés, avec une perte de paquets de 0%. Les adresses IP 145.239.37.162, 51.210.99.219, 151.80.233.222 correspondent aux serveurs ciscomadesimple.be, [www.developpez.com](http://www.developpez.com) et [www.reseaucerta.org](http://www.reseaucerta.org) , ce qui montre une vérification de la connectivité vers des ressources extérieures à travers le réseau.

### Examen et analyse des données à partir des hôtes distants :

|  |  |  |
| --- | --- | --- |
| Emplacement | Adresse IP | Adresse MAC |
| 1 | 145.239.37.162 | 00 :0c :29 :b8 :25 :53 |
| 2 | 51.210.99.219 | 00 :0 c :29 :b8 :25 :53 |
| 3 | 151.80.233.222 | 00 :0 c :29 :b8 :25 :53 |

b) L’adresse MAC est toujours la même car c’est celle de la passerelle par défaut. Lorsqu’on envoie un ping à des adresses IP externes, le trafic doit passer par le routeur pour atteindre ces destinations. Le routeur relaie ensuite le paquet vers sa destination finale. C’est donc l’adresse MAC du routeur qui est utilisée comme adresse de destination dans chaque trame Ethernet.

### Analyse les données ICMPv6 locales

a) ICMPv6 est utilisé pour diagnostiquer et signaler les erreurs dans les réseaux IPv6

Une image contenant texte, Police, ligne, nombre

Description générée automatiquement

On peut observer l'adresse MAC source et de destination associée à des adresses IP en IPv6. Ici, le protocole ICMPv6 est utilisé pour le rapport multicast, ce qui est typiquement utilisé pour l'échange d'informations de groupe, comme les communications en temps réel ou le multicast vidéo. Le paquet montre un type "Multicast Listener Report Message v2", et les informations de transport sont également visibles dans l'analyse.

b) Le préfixe link-local des adresses IPv6 est FE80 ::/10 car il a été défini par la RFC 4291 pour identifier les adresses link-local. Elles permettent aux appareils de communiquer dès qu’ils sont connectés à un réseau sans avoir besoin d’une configuration manuelle ou d’un serveur d'adresse.

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

## Etthercap analyse des trames comme un attaquant simulation MAN in the midle

Une image contenant texte, Appareils électroniques, capture d’écran, logiciel

Description générée automatiquement

Dans cette capture, on utilise ettercap pour scanner le réseau. Le scanne commence avec une plage d'adresses IP (172.22.21.121) et génère un message d'attente pour 4095 hôtes à analyser. Cette opération est utilisée pour détecter des dispositifs et services sur un réseau local.

Une image contenant texte, Appareils électroniques, capture d’écran, logiciel

Description générée automatiquement

Dans cette capture, ettercap montre le trafic réseau en temps réel qu'il surveille. Les paquets UDP envoyés entre des adresses IP sont capturés. Cela indique que l'outil capture des requêtes multicast sur le réseau. Il est aussi possible de voir des pourcentages indiquant l'avancement de l'analyse des paquets. Cette analyse aide à surveiller la communication sur le réseau et à identifier des services actifs.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Dans cette troisième image, **ettercap** est à nouveau exécuté avec l'option -T -I pour lister les interfaces réseau disponibles sur le système. On voit que l'interface eth0 est listée comme étant disponible pour l'analyse. Cela montre comment l'outil permet à l'utilisateur de sélectionner et d'interagir avec différentes interfaces réseau.

Lancer une attaque ARP Spoofing entre le serveur DHCP (172.31.1.4) et une machine cliente (par exemple, 172.31.1.59) :

* + -T : Mode texte.
  + -i eth0 : Utiliser l'interface eth0.
  + -M arp:remote : Attaque ARP MITM.
  + /172.31.1.4/ : Première cible (le serveur DHCP).
  + /172.31.1.59/ : Deuxième cible (la machine cliente).

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Ici on a remplacé le dhcp originel par le nôtre nous somme devenue le dhcp principale en remplaçant 172.31.1.4.

# Capture de trame liées à HTTP :

a)

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

Mon adresse IP L’adresse IP du site internet

b) Dans Statistiques puis conversation dans la barre de menu en haut, on voit apparaitre les statistiques de la trame :

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

Il y a donc 4 paquets qui sont échangés.

c) Il y a plusieurs trames car les données sont trop lourdes pour une seule trame, elles sont donc fragmentées en plusieurs trames.

Le protocole TCP a besoin de plusieurs trames, une de SYN (demande de connexion), une SYN-ACK (accepte de la demande) et une ACK (confirmation de l’acceptation).

Des paquets peuvent être perdus, le protocole demande de les re-envoyer.

d) On voit dans statistiques puis taille maximale des trames reçues dans le menu en haut :

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

La taille maximale est de 16450.

La taille des trames est de 1518octets pour les trames classiques et 9000octets pour les trames jumbo, des trames pour des environnements spécifiques (centre de données). La plus grande est donc une trame jumbo.

## MTU

a)

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

 Le paramètre -f est utilisé pour activer le flag "Do Not Fragment" (DF), ce qui signifie que le paquet ne doit pas être fragmenté lors de son envoi. Le message d'erreur « Le paquet doit être fragmenté mais paramétré DF » indique que le paquet est trop volumineux pour passer dans un seul paquet IP et qu'il devrait être fragmenté, le MTU 1500 est donc trop grand pour passer, il faut le baisser.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Le paquet 1452 est plus petit que celui de 1500 octets et, en conséquence, il peut être transmis sans nécessiter de fragmentation, malgré le flag DF. Le ping réussit sans perte. Le paquet de 1453 octets ne peut pas être fragmenté en raison du paramètre DF et le MTU est trop grand, et donc une nouvelle perte de paquet se produit.

Le MTU optimal est donc 1452.

# Application pratiques de Wireshark HTTPS :

[10 Minute Mail - Free Anonymous Temporary email](https://10minutemail.com/) (un site sécurisé inconnu pour moi)

Une image contenant texte, Appareils électroniques, capture d’écran, nombre

Description générée automatiquement

Cette capture montre le Client Hello dans un processus de handshake TLS. Le client initie la connexion avec le serveur en utilisant le protocole TLS 1.2 et inclut dans le message une liste de suites de chiffrement qu'il est prêt à accepter. Le paquet indique que le client supporte plusieurs suites de chiffrement, dont la suite de chiffrement TLS\_ECDHE\_RSA\_WITH\_AES\_256\_GCM\_SHA384, qui est un choix populaire en raison de sa robustesse et de ses performances. Cela fait partie de l'échange qui permet d'établir une connexion sécurisée entre le client et le serveur.

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquement

Dans cette capture, le serveur répond avec un message Server Hello. Le serveur confirme qu'il utilise également TLS 1.2 et a choisi une suite de chiffrement qui correspond à celle envoyée par le client. Ce message de réponse contient également un certificat numérique pour authentifier l'identité du serveur. Le certificat est vérifié pour établir une connexion sécurisée.

Une image contenant texte, Police, nombre, capture d’écran

Description générée automatiquement

 Ici, on voit un extrait de données capturées qui est la signature digitale TCP avec des données cryptées. Le contenu visible est une séquence de données illisibles car il est chiffrée.

# NetworkMiner :

NetworkMiner est un outil d’analyse réseau open-source qui analyse les hôtes et les sessions, récupère les fichiers transmis sur le réseau et visualise les certificats. Il est intuitif, analyse passivement (discrète) et extrait automatiquement les fichiers et mots de passe en clair. Il ne permet pas de capturer les données en temps réel, il faut entrer un fichier. pcap.

Installation : Une image contenant texte, Police, logiciel, Page web

Description générée automatiquement

J’ai choisi la version gratuite sur le site officiel et sécurisé de netresec, le groupe qui a créé NetworkMiner. Les fonctionnalités que nous avons besoin sont comprises dedans.

## Première étude :

a)

|  |  |
| --- | --- |
|  | La trame [Bloc1\_sem1-atelier\_05\_trames\stp.pcap](file:///F:\Travail\btssio_nf\Bloc1_sem1\Ateliers\05_capture_de_trame\Bloc1_sem1-atelier_05_trames\stp.pcap) n’existe pas, j’ai ouvert la trame\_3 |

|  |  |
| --- | --- |
|  | Nous n’avons pas de trame comme sur Wireshark, mais des adresses IP avec les détails. C’est plus facile à comprendre pour quelqu’un qui ne connait rien ou qui a du mal dans l’analyse de trame. |

Il y a une section pour voir les machines (hosts), une pour voir les fichiers (files), une pour les images, une pour les messages, une pour voir les identifiants d’authentification (Credentials), une pour les sessions, une pour les DNS, une pour les paramètres réseaux (Parameters), une pour des mots clés spécifique (Keywords) et une pour voir les anomalies.

b)

Une image contenant texte, ligne, Police, capture d’écran

Description générée automatiquement

Il y a un changement inattendu d’adresse MAC, ce qui peut être une attaque ARP spoofing. L’adresse MAC multicast est remplacée par une autre adresse sur le réseau (un poste dont l’hackeur à accès).

Pendant une attaque ARP spoofing, l’hackeur usurpe une adresse MAC dans la table ARP pour intercepter et rediriger le trafic réseau ou faire une attaque de type MITM.

C’est rassurant car c’est sur une adresse multicast et pas une passerelle par défaut, elle reste sur le réseau local sans en sortir. L’hackeur peut perturber seulement les appareils sur le réseau. Le réseau ne sera pas totalement coupé. Il faut impérativement surveiller le réseau et identifier l’adresse MAC 08D23EE9067E pour régler ce problème.

## Etude des différents outils :

Synthese\_FTP.docx

# Intelligence artificielle :

## ChatGPT :

Je vais utiliser GPT-4o (Chat-GPT) qui traite des fichiers et des outils Python qui permet d’analyser les fichiers. pcap (qui contienne les trames).

|  |  |
| --- | --- |
| Une image contenant texte, Appareils électroniques, capture d’écran, logiciel  Description générée automatiquement | Chat GPT va utiliser Python pour analyser et extraire les données contenues dans la trame trame\_1.pcap |
|  | Il n’arrive pas à analyser le fichier et me redirige vers Wireshark |

J’ai donc modifié le fichier pour qu’il soit lisible par Chat GPT, en fichier .txt :

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

J’ai eu les informations suivantes :

**Explication des champs**

1. **Adresse de destination** : ffff ffff ffff
   * **Broadcast** : la requête est envoyée à toutes les machines du réseau.
2. **Adresse MAC source** : 09:ab:14:d8:05:48
   * L'adresse MAC de l'expéditeur de la requête ARP.
3. **Type de protocole** : 0806
   * Ce champ indique que le protocole utilisé est **ARP**.
4. **Format matériel (Hardware Type)** : 0001
   * Indique qu'il s'agit d'une adresse MAC Ethernet.
5. **Format du protocole (Protocol Type)** : 0800
   * Indique le protocole IPv4.
6. **Longueur de l'adresse matérielle** : 06
   * La longueur de l'adresse MAC est de 6 octets.
7. **Longueur de l'adresse protocole** : 04
   * La longueur de l'adresse IPv4 est de 4 octets.
8. **Type d'opération** : 0001
   * C'est une requête ARP (1 = Request).
9. **Adresse MAC source** : 09:ab:14:d8:05:48
   * Répété pour le champ d'adresse source.
10. **Adresse IP source** : 125.5.48.10
    * L'adresse IP de l'expéditeur.
11. **Adresse MAC cible** : 00:00:00:00:00:00
    * Indique que l'adresse MAC de la cible est inconnue (d'où la requête ARP).
12. **Adresse IP cible** : 125.18.110.3
    * L'adresse IP pour laquelle l'expéditeur cherche l'adresse MAC correspondante.

**Conclusion**

Il s'agit d'une **requête ARP** où l'émetteur demande l'adresse MAC associée à l'adresse IP 125.18.110.3.

Il a très bien analysé la trame, c’est l’intelligence artificielle la plus puissante et la plus développé aujourd’hui.

## Gamma :

## MistralAI le Chat :

MistralAI le Chat est une i intelligence artificielle française open-source et transparente mais elle est limitée et a moins de capacités que Chat GPT. Il ne peut pas lire les fichiers .pcap ni .txt, j’ai donc pris la trame\_1.txt et copier-coller le texte qu’il y avait dedans :

Une image contenant texte, Appareils électroniques, capture d’écran, Police

Description générée automatiquement

J’ai eu cette réponse :

**Analyse des segments**

1. ffff ffff ffff
   * Ces valeurs peuvent représenter des marqueurs ou des valeurs de remplissage.
2. 09ab 14d8 0548
   * Ces valeurs peuvent être des identifiants, des adresses ou des données spécifiques.
3. 0806 0001
   * Ces valeurs peuvent représenter des commandes ou des paramètres.
4. 0800 0604 0001
   * Ces valeurs peuvent également représenter des commandes ou des paramètres.
5. 09ab 14d8 0548
   * Réapparition des mêmes valeurs, ce qui suggère une répétition ou une structure récurrente.
6. 7d05 300a
   * Ces valeurs peuvent être des données spécifiques ou des marqueurs de fin de segment.
7. 0000 0000 0000
   * Ces valeurs peuvent représenter des zéros de remplissage ou des valeurs par défaut.
8. 7d12 6e03
   * Ces valeurs peuvent être des données spécifiques ou des marqueurs de fin de trame.

**Hypothèses possibles**

1. En-tête et queue de trame
   * Les segments **ffff ffff ffff** et **0000 0000 0000** pourraient être des en-têtes ou des queues de trame.
2. Identifiants ou adresses
   * Les segments **09ab 14d8 0548** pourraient être des identifiants ou des adresses récurrents.
3. Commandes ou paramètres
   * Les segments **0806 0001** et **0800 0604 0001** pourraient être des commandes ou des paramètres spécifiques.
4. Données spécifiques
   * Les segments **7d05 300a** et **7d12 6e03** pourraient être des données spécifiques ou des marqueurs de fin de segment.

**Conclusion**

Sans contexte supplémentaire, il est difficile de fournir une analyse précise de la trame. Les segments identifiés peuvent représenter différentes parties d'un protocole de communication, mais une compréhension plus approfondie nécessiterait des informations sur le contexte d'utilisation de cette trame (par exemple, le type de protocole, l'application spécifique, etc.).

Si vous avez des informations supplémentaires sur le contexte ou l'application de cette trame, cela aiderait à fournir une analyse plus détaillée.

Ce qui est largement insuffisant et qui est un grand cran en dessous de Chat GPT.

## Vectra AI ([Vectra AI - Advanced AI Security - Stop Cyberattacks Fast](https://www.vectra.ai/)):

C’est une plateforme spécialisée dans la détection et la réponse aux menaces. Elle est basée sur l’intelligence artificielle pour automatiser ça.

Elle a une détection en temps réel, des alertes plus pertinentes et une visibilité complète du réseau mais a un cout élevé et est compliqué à mettre en place.

Une image contenant capture d’écran, texte, graphisme

Description générée automatiquement

Nouvelle méthode trouver

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Prompte utiliser :

**Prompt :**  
Analyse cette trame Wireshark en détaillant les informations suivantes : identifie les adresses MAC présentes et surveille les changements éventuels. Indique les adresses IP source et destination en les associant aux adresses MAC correspondantes. Décris concrètement ce qui semble se passer dans cette trame en fonction du protocole utilisé, des ports, et des champs pertinents. Fournis une analyse précise et complète de l’activité réseau observée.

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Résulat

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Une image contenant texte, capture d’écran, Police

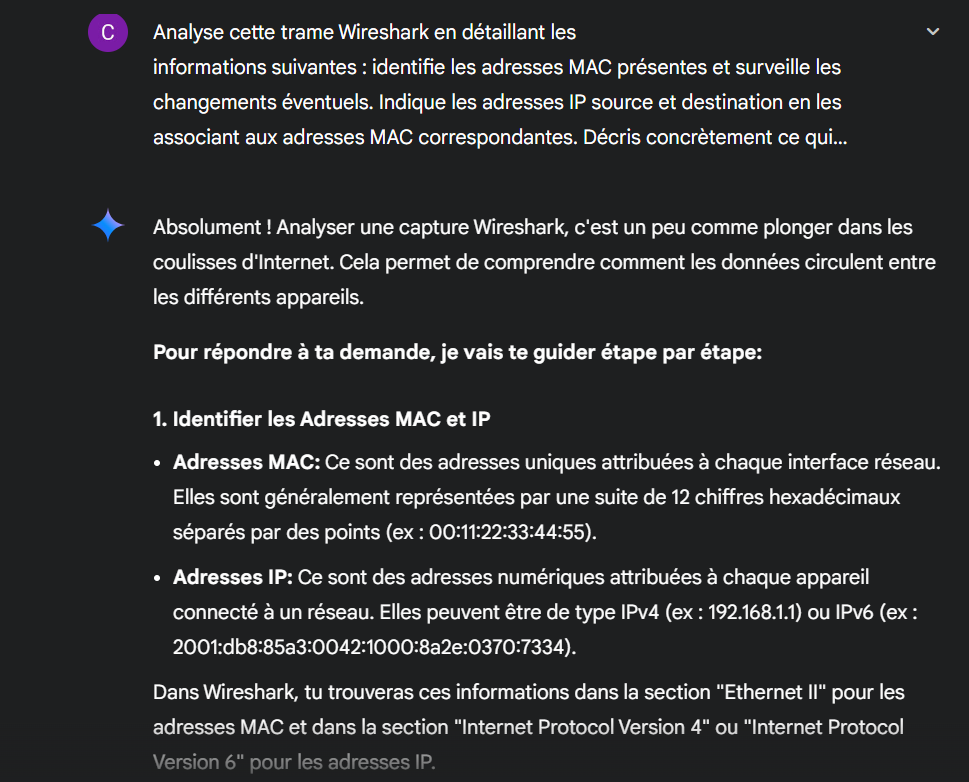
Description générée automatiquement

On lui donne en premier la trame car trop grosse pour allez avec un prompte :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Puis on lui donne prompte :



Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Github :

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement