TP- Analyse de trafic réseau avec Wireshark

Objectifs pédagogiques

Au terme de ce TP, l'étudiant sera capable de :

- Capturer et analyser des trames réseau avec Wireshark
- Identifier les étapes du Three-Way Handshake TCP
- Décoder les champs d'un en-tête TCP (numéro de séquence, ACK, drapeaux, etc.)
- Analyser un échange HTTP complet (requête, réponse, fermeture)
- Comprendre les mécanismes de retransmission TCP

1. TCP vs UDP

TCP (Transmission Control Protocol) et UDP (User Datagram Protocol) sont les deux protocoles principaux de la couche Transport (couche 4 du modèle OSI).

Caractéristique	TCP	UDP
Connexion	Orienté connexion	Sans connexion
Fiabilité	Garantie (ACK, retransmission)	Non garantie
Ordre des paquets	Garanti	Non garanti
Contrôle de flux	Oui (fenêtrage)	Non
Usage typique	HTTP, HTTPS, FTP, SSH	DNS, VoIP, streaming

2. Champs de l'en-tête TCP

Les champs principaux à connaître pour ce TP :

Champ	Description
Source Port	Port de l'émetteur
Destination Port	Port du destinataire
Sequence Number	Numéro d'ordre du premier octet du segment
Acknowledgment Number	Prochain octet attendu par le récepteur
Flags (SYN, ACK, FIN, RST…)	Drapeaux de contrôle de la connexion
Window Size	Taille du tampon de réception (contrôle de flux)
Checksum	Vérification de l'intégrité du paquet
TCP Segment Length	Taille des données utiles du segment

3. Three-Way Handshake (3WH)

L'établissement d'une connexion TCP se fait en 3 étapes :

- Étape 1 — SYN : Le client envoie un paquet avec le drapeau SYN activé. Il annonce son numéro de séquence initial (ISN).
- Étape 2 — SYN-ACK : Le serveur répond avec SYN + ACK. Il accuse réception du SYN client et annonce son propre ISN.
- Étape 3 — ACK : Le client accuse réception du SYN serveur. La connexion est établie.

4. Fermeture de connexion TCP

La fermeture propre d'une connexion TCP utilise 4 étapes (Four-Way Teardown) :

- L'hôte qui ferme envoie FIN + ACK
- L'autre hôte accuse réception avec ACK
- L'autre hôte envoie à son tour FIN + ACK
- Le premier hôte répond ACK. La connexion est fermée.

- **PARTIE 1 — Prise en main de Wireshark**

Exercice 1 — Installation et découverte de l'interface

1.1 — Lancement de Wireshark

Lancez Wireshark depuis le menu Applications ou en exécutant la commande suivante dans un terminal :

```
$ wireshark &
```

- Identifiez les 3 zones principales de l'interface Wireshark : liste des paquets, détail des champs, vue hexadécimale.
- Sélectionnez votre interface réseau active (eth0, ens33 ou Wi-Fi selon la machine).
- Lancez une capture en cliquant sur la fièche bleue ou en appuyant sur Ctrl+E.

Q1	Quelle interface réseau avez-vous sélectionnée ? Quelle est son adresse IP ? (utilisez la commande ip a ou ipconfig)

Q2	Citez les 3 panneaux principaux de l'interface Wireshark et décrivez brièvement le rôle de chacun.

1.2 — Filtres de capture et d'affichage

Wireshark propose deux types de filtres :

- Filtre de capture (BPF) : sélectionne les trames avant enregistrement (ex : tcp port 80)
- Filtre d'affichage : filtre l'affichage après capture (ex : tcp.flags.syn == 1)

Filtre d'affichage	Description
tcp	Tous les paquets TCP
http	Tous les paquets HTTP
ip.addr == 192.168.1.1	Trafic depuis ou vers cette IP
tcp.flags.syn == 1	Paquets SYN uniquement
tcp.flags.fin == 1	Paquets FIN uniquement
tcp.flags.reset == 1	Paquets RST uniquement
tcp.port == 80	Trafic sur le port 80
tcp.analysis.retransmission	Retransmissions TCP

PARTIE 2 — Analyse du Three-Way Handshake

Exercice 2 — Capture et analyse du 3WH

2.1 — Génération du trafic

- Démarrez une capture Wireshark sur votre interface réseau.
- Appliquez le filtre d'affichage : tcp.flags.syn == 1 ou tcp.port == 80
- Ouvrez un navigateur web et accédez à : http://example.com (HTTP simple, sans HTTPS)
- Attendez que la page se charge, puis arrêtez la capture (Ctrl+E).

2.2 — Identification des paquets

Localisez les 3 paquets du Three-Way Handshake en utilisant le filtre : tcp.flags.syn == 1

Paquet #	Étape	Drapeaux	IP Source	IP Destination	N° séquence (Seq)
1	SYN	SYN			
2	SYN-ACK	SYN+ACK			
3	ACK	ACK			

Q3	Quelle est la valeur du numéro de séquence (Seq) du paquet SYN envoyé par le client ? Quelle est la valeur affichée par Wireshark (relative) ?

Q4	Dans le paquet SYN-ACK (paquet 2), quelle est la valeur du champ Acknowledgment Number ? Expliquez ce que cela signifie.

Q5	Le numéro de séquence initial (ISN) est-il égal à 0 en valeur brute (raw) ? Pourquoi Wireshark affiche-t-il 0 en valeur relative ?

2.3 — Détail des champs TCP

Cliquez sur le paquet SYN (paquet 1) et développez la section « Transmission Control Protocol » dans le panneau du bas.

Champ	Valeur observée	Signification
Source Port		
Destination Port		
Sequence Number (raw)		
Header Length		
Window Size		
Checksum		
TCP Segment Length		

PARTIE 3 — Analyse de l'échange de données HTTP

Exercice 3 — Requête et réponse HTTP

Après le 3WH, les données applicatives sont échangées. Appliquez le filtre : http pour n'afficher que les paquets HTTP.

3.1 — La requête HTTP GET

- Cliquez sur le paquet HTTP GET.
- Développez la section « Hypertext Transfer Protocol ».
- Relevez les informations demandées ci-dessous.

Q6	Quelle est la méthode HTTP utilisée ? Quelle ressource est demandée (chemin) ? Quelle est la version HTTP ?

Q7	Quel est le numéro de séquence TCP (Seq) du paquet GET ? Quelle est la taille du segment TCP (TCP Segment Length) ?

3.2 — L'ACK du serveur

Juste après le GET, le serveur envoie un ACK pour accuser réception de la requête.

Q8	Quelle est la valeur du champ Acknowledgment Number dans l'ACK envoyé par le serveur ? Comment est-elle calculée ?

3.3 — La réponse HTTP

Le serveur envoie ensuite la réponse HTTP contenant le contenu de la page.

Q9	Quel est le code de statut HTTP renvoyé par le serveur (ex : 200 OK) ? Quelle est la taille du contenu (Content-Length) ?

Q10	Comparez le Sequence Number et le Acknowledgment Number entre le paquet GET du client et la réponse du serveur. Schématisez l'échange ci-dessous.

PARTIE 4 — Fermeture de connexion et cas particuliers

Exercice 4 — Analyse de la fermeture TCP

Appliquez le filtre : tcp.flags.fin == 1 pour identifier les paquets de fermeture.

4.1 — Tableau de fermeture

Paquet	Drapeaux	Émetteur	Seq / Ack	Signification
#8	FIN+ACK	Serveur		Le serveur demande la fermeture
#9	ACK	Client		
#10	FIN+ACK	Client		
#11	ACK	Serveur		

Q11	Pourquoi le drapeau FIN incrémente-t-il le numéro d'acquittement de 1, comme le drapeau SYN ?

Exercice 5 — Cas du drapeau RST (Reset)

Le drapeau RST indique une fermeture abrupte de connexion. Pour l'observer :

- Appliquez le filtre : tcp.flags.reset == 1
- Si aucun RST n'est visible dans votre capture, essayez : telnet 8.8.8.8 80 (connexion vers un port fermé).

Q12	Avez-vous observé des paquets RST dans votre capture ? Dans quel(s) contexte(s) apparaissent-ils ? Donnez au moins 2 causes possibles d'un RST.

PARTIE 5 — Retransmissions TCP

Exercice 6 — Analyse des retransmissions

Appliquez le filtre : tcp.analysis.retransmission pour visualiser les retransmissions dans votre capture.

6.1 — Rappel théorique

Il existe deux types de retransmissions TCP :

Type	Mécanisme
Retransmission RTO	L'émetteur attend l'expiration d'un minuteur (RTO). Si aucun ACK n'est reçu avant, il retransmet le paquet.
Retransmission rapide	Si l'émetteur reçoit 3 ACK dupliqués consécutifs (dupthresh = 3), il retransmet immédiatement sans attendre le RTO.

Q13	Avez-vous observé des retransmissions dans votre capture ? Si oui, s'agit-il de retransmissions RTO ou rapides ? Comment Wireshark les signale-t-il visuellement ?

Q14	Qu'est-ce qu'un ACK dupliqué (Duplicate ACK) ? Dans quel cas le TCP passe-t-il en mode retransmission rapide ?

6.2 — Ajout de colonnes dans Wireshark

Pour faciliter le suivi des numéros de séquence, ajoutez les colonnes suivantes dans Wireshark :

- Sequence Number (Seq)
- Acknowledgment Number (Ack)
- TCP Segment Length

Procédure : Clic droit sur un champ dans le panneau de détail → « Apply as Column »

BILAN — Synthèse et questions de réflexion

Questions de synthèse

Q15	Résumez en 5 à 7 lignes le fonctionnement d'une connexion TCP complète (de l'établissement à la fermeture), en vous appuyant sur les observations de ce TP.
Q16	Pourquoi TCP est-il considéré comme un protocole « fiable » ? Quels mécanismes assurent cette fiabilité ?
Q17	Quelle est la différence entre le Sequence Number (raw) et le Sequence Number (relatif) affiché par Wireshark ? Pourquoi Wireshark normalise-t-il cette valeur ?
Q18	Expliquez le rôle du champ Window Size dans TCP. Que se passe-t-il si ce champ atteint 0 ?