Nom : RALAIKOTO
Prénom : Mamilalao Sitraka Nadia
Classe : L3

THREE-WAY HANDSHAKE :

L'établissement d'une connexion avec TCP (Transmission Control Protocol) nécessite un processus appelé "handshake en trois étapes" (three-way handshake). Voici les trois étapes :

1. **SYN (Synchronize)** : Le client envoie un segment TCP au serveur avec le drapeau SYN (synchronize) activé. Cela indique que le client souhaite établir une connexion et fournit un numéro de séquence initial.

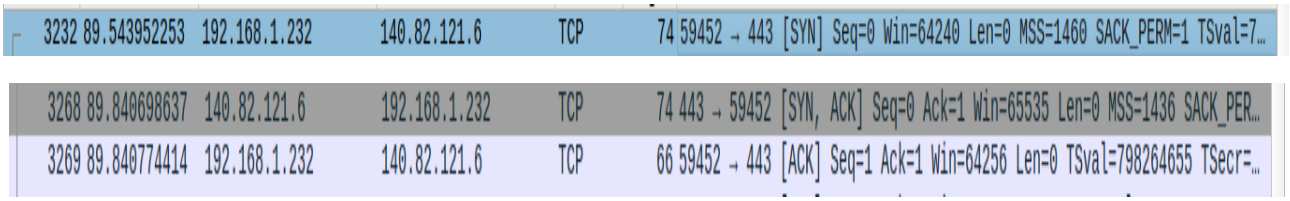
   Hôte A --------> SYN (Seq = x)

2. **SYN-ACK (Synchronize-Acknowledge)** : Le serveur répond en envoyant un segment TCP au client avec les drapeaux SYN et ACK (acknowledge) activés. Cela signifie que le serveur accepte la demande de connexion et fournit également son propre numéro de séquence initial.

   Hôte B --------> SYN-ACK (Seq = y, Ack = x + 1)

3. **ACK (Acknowledge)** : Le client envoie un segment TCP au serveur avec le drapeau ACK activé, confirmant la réception du numéro de séquence du serveur. À ce stade, la connexion est établie et les deux parties peuvent commencer à échanger des données.

   Hôte A --------> ACK (Seq = x + 1, Ack = y + 1)



DATA TRANSMISSION :

Une fois la connexion établie, les données sont envoyées sous forme de segments TCP. Chaque segment comporte plusieurs éléments :

- **Numéro de séquence** : Chaque segment est numéroté, ce qui permet au récepteur de réassembler les données dans le bon ordre, même si les segments arrivent dans un ordre différent.
- **Numéro d'accusé de réception (ACK)** : Utilisé pour confirmer la réception des segments.
- **Taille de la fenêtre** : Indique combien de données le récepteur est prêt à accepter. Cela aide à gérer le flux de données et à éviter la congestion.
- **Checksum** : Utilisé pour vérifier l'intégrité des données. Si le checksum ne correspond pas, le segment est considéré comme corrompu et doit être retransmis.

- **Options** : Permet des fonctionnalités supplémentaires, comme la négociation de la taille de la fenêtre ou des options de sécurité.

Contrôle d'erreurs et de flux

- **Contrôle d'erreurs** : Chaque segment TCP est vérifié à l'aide d'un **checksum**. Si le récepteur détecte une erreur, il ignore le segment et n'envoie pas d'accusé de réception (ACK) pour ce segment. L'expéditeur retransmet alors le segment après un délai d'attente.
- **Contrôle de flux** : Utilise la taille de la fenêtre pour indiquer combien de segments peuvent être envoyés avant de recevoir un ACK. Cela permet d'adapter le taux d'envoi de données à la capacité du récepteur.
- **Contrôle de congestion** : TCP intègre des mécanismes pour gérer la congestion du réseau. Si le réseau est congestionné, TCP réduit la vitesse de transmission des données pour éviter de surcharger le réseau.

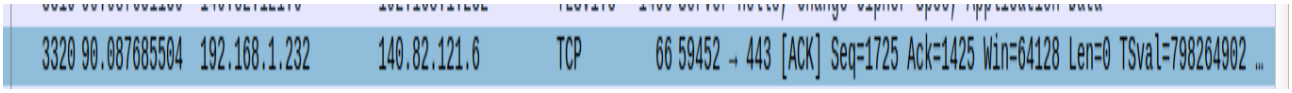
Confirmation de réception :

Chaque fois qu'un segment est reçu avec succès, le récepteur envoie un segment d'accusé de réception (ACK) pour informer l'expéditeur qu'il a reçu le segment. Si l'expéditeur ne reçoit pas l'ACK dans un certain délai, il retransmet le segment.

Fenêtre glissante (Sliding Window) :

TCP utilise une technique appelée **fenêtre glissante** pour gérer l'envoi de données :

- La taille de la fenêtre détermine combien de segments peuvent être envoyés avant qu'un ACK ne soit requis.
- Lorsque l'expéditeur reçoit un ACK, il déplace la fenêtre pour permettre l'envoi de nouveaux segments.



FOUR-WAY HANDSHAKE :

La terminaison d'une connexion TCP se déroule en quatre étapes :

1. **Demande de fermeture (FIN)** : L'hôte qui souhaite terminer la connexion envoie un segment TCP avec le drapeau **FIN** pour indiquer qu'il ne va plus envoyer de données.
2. **Confirmation de fermeture (ACK)** : L'autre hôte confirme la réception de la demande de fermeture en envoyant un segment avec le drapeau **ACK**. Cela signifie qu'il a reçu la demande, mais il peut encore avoir des données à envoyer.
3. **Envoi de données restantes (si applicable)** : Si l'autre hôte a des données à envoyer, il les envoie maintenant. Une fois les données envoyées, il envoie également un segment avec le drapeau **FIN** pour signaler qu'il ne va plus envoyer de données.

4. **Accusé de réception final (ACK)** : Le premier hôte confirme la réception du segment FIN de l'autre hôte en envoyant un segment **ACK**. À ce stade, la connexion est complètement fermée.

Hôte A Hôte B

| FIN ----------> |

| ACK <---------- |

| (envoi de données restantes)

| <------ FIN ---------- |

| <------ ACK ---------- |