TD2 : Modèle TCP/IP

L'utilisation de tout outil d'IA générative (comme ChatGPT, Copilot, etc.) est strictement interdite pour la réalisation de ce sujet de TD. Vous êtes fortement encouragé à résoudre ces exercices par vous-même afin de vous assurer de votre compréhension du cours.

Exercice 1

On considère un réseau basé sur le modèle de la pile de protocoles TCP/IP, où un protocole p de la couche application est utilisé pour transmettre des données entre deux machines. Ainsi, une machine A qui souhaite transmettre des données en utilisant ce protocole émet un ou plusieurs messages à destination de la machine B.

1. Rappelez brièvement le rôle de chacune des couches du modèle TCP/IP.
2. Qu'est-ce que l'encapsulation de données ?
3. On suppose que le protocole p utilise, au niveau des trois (03) autres couches, les protocoles indiqués dans le tableau suivant. En vous aidant des formats joints en annexe, déterminez la taille des en-têtes de chaque protocole.

Couche	**Protocole**	**En-tête (octets)**
Accès réseau	Ethernet	?
Internet	IP	?
Transport	TCP	?

4. Un analyseur de trame Ethernet a fourni la trace hexadécimale représentée par la Figure 1 et correspondant à une trame échangée lors de la transmission de données.

```
0000  00 04 76 f0 fb b5 00 06 5b c2 f5 9e 08 00 45 00
0010  01 4f 06 cf 40 00 40 06 b1 6f c0 a8 00 17 c0 a8
0020  00 03 80 09 00 50 85 e6 67 33 03 6c 42 f4 80 18
0030  16 d0 78 f1 00 00 01 01 08 0a 00 09 62 11 0b 5a
0040  6a 43 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31
0050  0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 4b 65
0060  65 70 2d 41 6c 69 76 65 0d 0a 55 73 65 72 2d 41
0070  67 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e
0080  30 20 28 63 6f 6d 70 61 74 69 62 6c 65 3b 20 4b
0090  6f 6e 71 75 65 72 6f 72 2f 32 2e 32 2d 31 31 3b
00a0  20 4c 69 6e 75 78 29 0d 0a 41 63 63 65 70 74 3a
00b0  20 74 65 78 74 2f 2a 2c 20 69 6d 61 67 65 2f 6a
00c0  70 65 67 2c 20 69 6d 61 67 65 2f 70 6e 67 2c 20
00d0  69 6d 61 67 65 2f 2a 2c 20 2a 2f 2a 0d 0a 41 63
00e0  63 65 70 74 2d 45 6e 63 6f 64 69 6e 67 3a 20 78
00fe  2d 67 7a 69 70 2c 20 67 7a 69 70 2c 20 69 64 65
0100  6e 74 69 74 79 0d 0a 41 63 63 65 70 74 2d 43 68
0110  61 72 73 65 74 3a 20 41 6e 79 2c 20 75 74 66 2d
0120  38 2c 20 2a 0d 0a 41 63 63 65 70 74 2d 4c 61 6e
0130  67 75 61 67 65 3a 20 66 72 2c 20 66 72 5f 46 52
0140  40 65 75 72 6f 2c 20 65 6e 0d 0a 48 6f 73 74 3a
0150  20 73 65 72 76 43 33 30 39 0d 0a 0d 0a
```

Figure 1: Trace hexadécimale d'une trame Ethernet.

a. Décodez le contenu de la trame précédente en vous servant du format donné en annexe.
b. Selon vous, quel est le protocole p transporté à l'intérieur du segment TCP ?

Exercice 2

Dans cet exercice, vous allez monter plusieurs mini-réseaux dans l'outil de simulation **Filius** afin d'observer le fonctionnement de différents équipements d'interconnexion : hub, switch, routeur.

Vous réaliserez plusieurs configurations puis répondrez aux questions à partir de vos observations.

Vous pouvez installer Filius depuis : https://www.lernsoftware-filius.de/Herunterladen. Il peut fonctionner sous Windows, Linux et MacOS, et nécessite la présence de l'environnement d'exécution Java sur le système (Java 17 ou version supérieure) sauf pour Windows puisque Java est inclus dans l'installateur.

1. Créez un premier réseau local LAN_1 composé de trois (03) PCs reliés à un **switch**.

   Attribuez une adresse IP à chaque PC dans le réseau **192.168.1.0/24**, puis passez en mode simulation (flèche verte ▶) pour observer l'activité du réseau.

   Pour pouvoir envoyer des messages, il vous faudra installer une **Ligne de commande** sur chaque PC (*Cliquez sur le PC* → *Installation des logiciels* → *Ligne de commande*).

   Pensez également à ajuster la **vitesse de simulation** pour faciliter l'observation des signaux dans les câbles.

   Depuis $PC1$, faites un **ping** vers $PC2$ (ou vers $PC3$).

   a. Au tout premier envoi ping, que fait le switch avec la trame ? À quel autre **équipement** ce comportement vous fait-il penser ?

   b. Lors des envois suivants, comment évolue le comportement du switch ?

2. Créez maintenant un second réseau local LAN_2 composé de deux (02) PCs reliés à un switch et ayant des adresses IP dans **192.168.2.0/24**.

   a. Essayez de relier les deux LAN avec un switch supplémentaire. Testez la communication entre un PC du LAN_1 et un PC du LAN_2. Que constatez-vous ?

   b. Remplacez maintenant le switch central par un **routeur**. Configurez ses deux interfaces et les passerelles par défaut sur les PCs des deux LAN. Veillez également à activer le **routage automatique** lors de la configuration du routeur.
   Testez à nouveau la communication entre deux PCs de chaque LAN. Quel est le rôle du routeur dans ce cas ?

   c. Déduisez pour chaque équipement dans quelle **couche** du modèle TCP/IP il opère.

Exercice 3

Soit le réseau illustré par le schéma de la Figure 2 suivante. L'objectif est d'observer les échanges de données et les protocoles mis en jeu au niveau des différentes couches du modèle TCP/IP en utilisant Filius.

Commencez par télécharger le fichier **network-archi-1.fls** sur Moodle et ouvrez-le avec Filius.

1. Effectuez un **ping** depuis la machine $PC1$ vers la machine SRV.

   a. Ouvrez la **table d'échange de données** sur la machine $PC1$. On s'intéresse ici aux deux premiers paquets échangés.

      i. Observez l'en-tête du premier paquet et vérifiez les adresses utilisées. Quelles sont les machines ayant reçu ce paquet ? Pourquoi ?

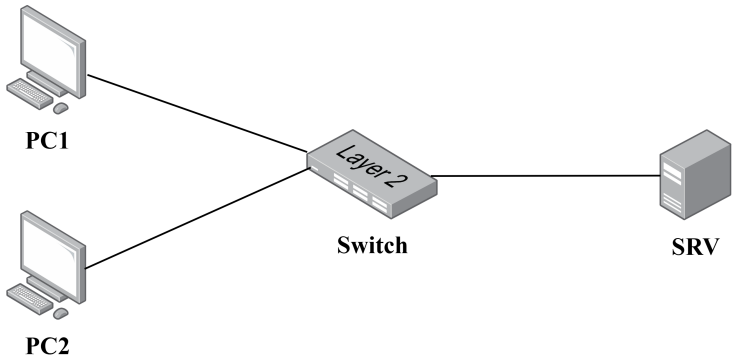


Figure 2: Première architecture réseau.

ii. Analysez maintenant l'en-tête du second paquet. Quelles sont les machines l'ayant reçu ? Pourquoi ?

iii. Faites une synthèse du fonctionnement du protocole impliqué dans ces échanges et de ses objectifs.

b. Affichez le contenu de la **table d'adresses** sur le commutateur. À quelles machines correspondent les adresses qui s'y trouvent ? En déduire le rôle de cette table.

c. Lors de l'exécution du *ping*, quatre (04) requêtes/réponses ont été envoyées.

i. Quel est le protocole impliqué dans ces échanges ?

ii. Analysez les 3^e et 4^e paquets dans la table d'échange de données sur la machine $PC1$ (ou SRV), puis indiquez, pour chacun de ces paquets, les adresses MAC et les adresses IP source et destination.

iii. Pour différencier les requêtes *ping*, un **numéro de séquence** est attribué. Vérifiez que ce numéro est différent pour chaque requête.
Ce numéro apparaît également dans les réponses, pourquoi selon vous ?

2. Tout en laissant la table d'échange de données, ouvrez le navigateur web de $PC1$ et saisissez l'adresse IP du serveur dans la barre d'URL : `http://<adresse-ip-du-serveur>`.

   On ne s'intéresse ici qu'aux messages échangés suite à la validation de l'URL.

   a. Quels paquets sont envoyés avant la première commande **GET** ? À quoi servent-ils ? Noter les numéros de ports source et destination utilisés.

   b. À quoi correspondent les quatre derniers paquets de l'échange ?

   c. Déduire quels sont les protocoles impliqués dans ces échanges.

3. Que pouvez-vous dire sur les **couches** du modèle TCP/IP mises en œuvre par les activités réseau des parties 1 et 2 ?

Annexe

Format d'une trame Ethernet

Une trame Ethernet est un ensemble de données structuré composé de plusieurs champs, chacun ayant un rôle spécifique. Une trame est composée d'une partie **en-tête** et d'une partie **données**, que l'on peut aussi décoder si on connaît le **protocole** correspondant.

Le format d'une trame Ethernet est le suivant :

Adresse destination (6 octets)	**Adresse source** (6 octets)	**Type** (2 octets)	**Données** (46 à 1500 octets)	**Code correcteur** (4 octets)

Le champ **Données** représente les données à transmettre, généralement encapsulées dans des protocoles comme IP ou ARP. Si les données sont inférieures à 46 octets, un remplissage est ajouté pour atteindre la taille minimale de 46 octets.

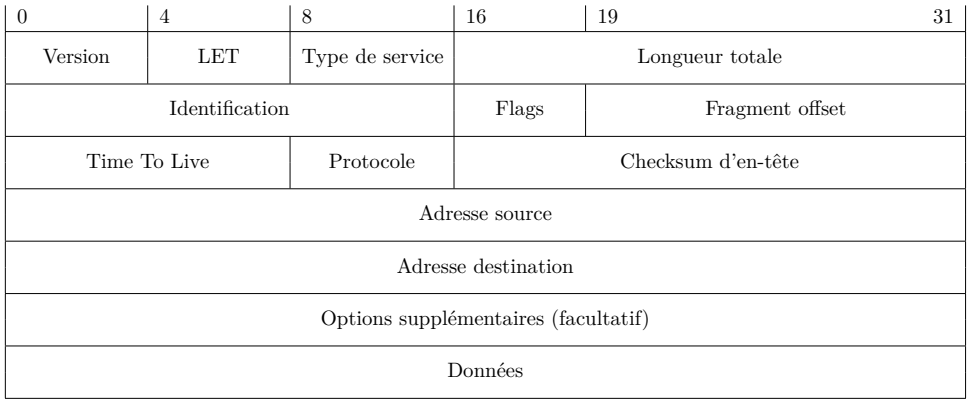
Le champ **Type** renseigne sur la manière de lire le contenu du champ **Données**.

- Si Type = 08 00, alors il s'agit d'un paquet IP.
- Si Type = 08 06, alors il s'agit d'un paquet ARP.

Le **code correcteur** n'est souvent pas affiché lors d'une capture de trames.

Format d'un paquet IP

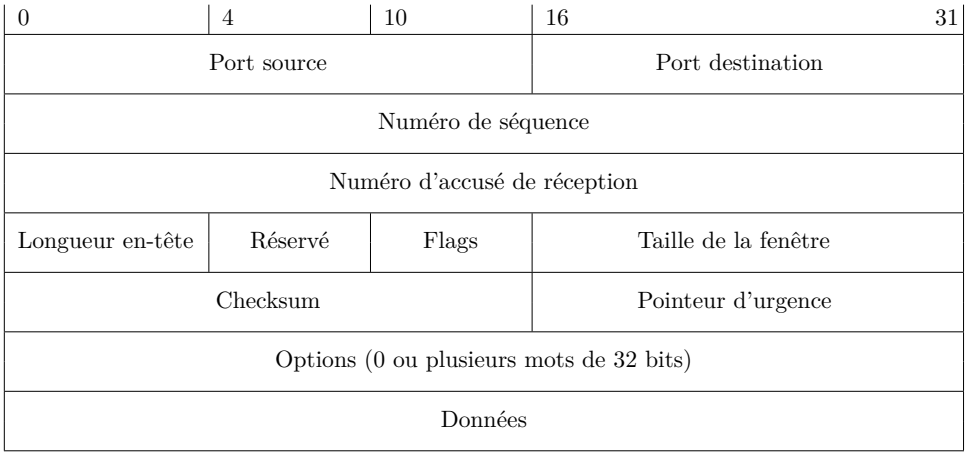
Un paquet IP est composé d'un **en-tête** et de **données** provenant de la couche supérieure.



- **Version :** indique la version d'IP (4 pour IPv4).
- **LET :** longueur de l'en-tête en mots de 32 bits (minimum 5, soit 20 octets).
- Type de service : définit la priorité et la qualité de service (QoS).
- **Longueur totale :** taille totale du paquet en octets (en-tête + données).
- Identification : numéro d'identification unique pour les fragments de paquet.
- Flags : contrôle la fragmentation.
- Fragment offset : position du fragment dans le paquet initial.
- Time To Live : temps maximal que le paquet peut rester dans le réseau.
- **Protocole :** indique le protocole de la couche supérieure (6 pour TCP, 17 pour UDP).
- Checksum d'en-tête : code de contrôle d'erreur pour l'en-tête.
- **Adresse source :** adresse IP de l'expéditeur.
- **Adresse destination :** adresse IP du destinataire.
- Options : options supplémentaires pour le routage ou le contrôle (rarement utilisé).

Format d'un segment TCP

Un segment TCP est composé d'un **en-tête** et de **données** provenant de la couche application.



- **Port source :** indique quelle est l'application à l'origine de la demande de transmission.
- **Port destination :** indique à quelle application sont destinées les données transmises.
- Numéro de séquence : c'est le numéro du premier octet de données transmis dans le segment.
- Numéro d'accusé de réception : indique le numéro de séquence du prochain octet attendu.
- **Longueur en-tête :** spécifie la longueur de l'en-tête TCP en mot de 4 octets.
- Flags : indique des bits de contrôle spécifiques liés à la connexion ou aux données.
- Taille de la fenêtre : indique le nombre d'octets que le destinataire peut accepter.
- Checksum : vérifie l'intégrité des données du segment (y compris l'en-tête).
- Pointeur d'urgence : indique des octets qui doivent être traités en priorité (flag URG à 1).