

CATALOGUE

<input type="checkbox"/>	0												
<input type="checkbox"/>	1												
<input type="checkbox"/>	2												
<input type="checkbox"/>	3												
<input type="checkbox"/>	4												
<input type="checkbox"/>	5												
<input type="checkbox"/>	6												
<input type="checkbox"/>	7												
<input type="checkbox"/>	8												
<input type="checkbox"/>	9												

← codez votre numéro d'étudiant ci-contre, et inscrivez votre nom et prénom ci-dessous.

Nom et prénom : .....
-----------------------

---

**Réseaux : QCM N°1 (M1 2024–2025)**

Noircir complètement à l'encre la case cochée (une croix ne suffit pas)  
Une seule réponse correcte par question.  
Réponse correcte : 2 pts; Réponse fausse : -0.5 pt; Réponse vide : 0 pt

---

**Question [Fragmentation4]** La fragmentation d'un message en plusieurs paquets peut

- augmenter l'impact des erreurs binaires et provoquer le déséquilibrage des données
- accélérer la transmission du message mais peut la ralentir également
- permet de réduire le temps de propagation sur chaque lien traversé
- accélérer la transmission du message et permet un meilleur routage des données

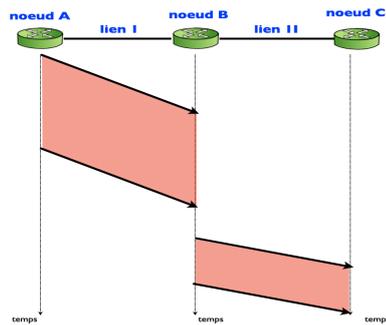
**Question [Delai5b]** Le temps d'attente d'un paquet est a priori

- dépendant de la taille du paquet et indépendant de la taille du buffer
- dépendant de la taille du paquet et dépendant de la taille du buffer
- indépendant de la taille du paquet et dépendant de la taille du buffer
- indépendant de la taille du paquet et indépendant de la taille du buffer

**Question [BBPP2]** DNS est un protocole de (1), UDP de (2), et Ethernet de (3)

- (1) bout en bout, (2) bout en bout, (3) proche en proche
- (1) proche en proche, (2) bout en bout, (3) proche en proche
- (1) proche en proche, (2) bout en bout, (3) bout en bout
- (1) bout en bout, (2) proche en proche, (3) proche en proche

**Question [Liens2]** Le schéma ci-dessous représente le déroulement du délai de bout-en-bout



d'un paquet sur 2 liens.

- La capacité d'émission du lien II et son délai de propagation sont plus grands que ceux du lien I
- La capacité d'émission du lien II et son délai de propagation sont plus petits que ceux du lien I
- La capacité d'émission du lien II est plus grande que celle du lien I mais son délai de propagation est plus petit
- La capacité d'émission du lien II est plus petite que celle du lien I mais son délai de propagation est plus grand

**Question [Boutenbout3]** Que signifie de bout-en-bout pour un protocole ?

- il est implémenté seulement sur les terminaux
- il permet d'établir un circuit entre les terminaux
- il assure un transfert de données fiable
- il assure obligatoirement un transfert bidirectionnel (dans les deux sens) entre les terminaux

**Question [Entete]** Soient  $H_x$  et  $D_x$  respectivement l'en-tête et les données associés à la couche x. D'après le principe d'encapsulation, on a :

- $D_2 = H_3 + H_4 + D_5$
- $D_2 = H_2 + H_3 + D_3$
- $D_2 = H_3 + H_4 + H_5$
- $D_2 = H_3 + H_4 + D_4$

**Question [TT4]** Le temps de transfert d'un paquet de 10 kbits sur un chemin composé de deux liens ayant chacun une capacité de 10 Mb/s et un délai de propagation de 10 ms est de

- 22 ms
- 0,11 s
- 11 ms
- 0,111 s

**Question [TT5]** Quel sera le temps de transfert total d'un paquet qui se trouve en 3e position dans le buffer de l'émetteur (impliquant que 2 autres paquets doivent être transmis avant lui), puis qui sera transmis sur un lien d'1 Mb/s ayant un délai de propagation de 5 ms sachant que tous les paquets ont tous une taille de 1kbits ?

- 8 ms
- 0,08 s
- 5,3 ms
- 0,053 s

**Question [Cookies1]** Où sont stockés les cookies utilisés par HTTP ?

- dans la carte réseau de votre machine
- sur votre machine
- sur le serveur distant
- chez votre épicier préféré

**Question [CWeb]** Le cache Web

- vérifie la validité de toutes les requêtes HTTP reçues et retransmet toutes les requêtes HTTP valides aux serveurs Web concernés
- peut envoyer des réponses HTTP sans solliciter les serveurs Web concernés s'il contient les données demandées dans les requêtes HTTP
- contient toutes les données de tous les serveurs Web existants
- peut fournir les adresses IP des serveurs Web contenant les pages Web demandées dans les requêtes HTTP

**Question [DNS3]** DNS

- est un protocole applicatif, qui fonctionne en mode client-serveur et qui utilise un seul serveur
- est un protocole applicatif, qui repose toujours UDP et qui permet de retrouver l'adresse IP d'une machine hôte
- est un protocole applicatif, qui fonctionne en mode client-serveur et qui utilise une hiérarchie de serveurs DNS
- est un protocole applicatif dans lequel toutes les requêtes DNS sont envoyées au serveur racine -root DNS server-

**Question [Ordre]** Dans quel ordre se font les actions suivantes sur une machine cliente qui n'a pas d'adresse IP et qui va chercher à accéder à du contenu Web ?

- i) échange DHCP ; ii) envoi d'une requête HTTP ; iii) échange DNS ; iv) réception de la réponse HTTP
- i) échange DNS ; ii) échange DHCP ; iii) envoi d'une requête HTTP ; iv) réception de la réponse HTTP
- i) échange DHCP ; ii) échange DNS ; iii) envoi d'une requête HTTP ; iv) réception de la réponse HTTP
- i) échange DHCP ; ii) envoi d'une requête HTTP ; iii) réception de la réponse HTTP ; iv) échange DNS

**Question [MultiplexageSC]** Dans le multiplexage de la couche Transport sans connexion (par exemple UDP),

- le numéro de port source permet de sélectionner la bonne socket sur la machine destination
- plusieurs nœuds ne peuvent pas émettre sur la même socket d'une machine destination
- le numéro de port source n'est jamais utilisé
- l'adresse IP source et le numéro de port source peuvent être utilisés pour le trajet retour si des paquets doivent être envoyés de la machine destination vers la machine source

**Question [Socket5]** Les segments TCP envoyés par un nœud A et destinés à un même nœud B

- sont envoyés vers des sockets différentes selon leur numéro de port source
- sont envoyés vers des sockets différentes selon leur numéro de port destination
- sont envoyés vers des sockets différentes selon leurs numéros de port source et destination
- sont toujours envoyés vers la même socket car les adresses IP source et destination sont identiques pour tous les paquets

**Question [UDP3]** Le protocole de transport UDP assure principalement les services (fonctions) suivants

- le multiplexage et la détection d'erreur bien que la détection d'erreur ne soit pas un service obligatoire
- la détection d'erreur et le contrôle de flux
- le multiplexage et le contrôle de congestion
- le multiplexage et la correction d'erreur car ce sont deux services obligatoires pour UDP

## CATALOGUE

**Question [Multiplexage2bis]** La couche Transport permet le multiplexage des flux applicatifs en se basant

- uniquement sur les adresses IP source et destination
- uniquement sur numéros de port source et destination
- sur les adresses MAC (de la couche 2) source et destination et les numéros de port source et destination
- sur certaines de ces informations : adresses IP source et destination et numéros de port source et destination

**Question [SocketHTTP]** On suppose qu'à  $t_0$ , un nœud A vient d'envoyer une requête à un serveur Web via le protocole Application HTTP. Juste après, à  $t_1$ , le nœud A ouvre une nouvelle connexion avec ce serveur et y envoie une nouvelle requête HTTP tandis que deux autres nœuds (B et C) envoient chacun 1 requête HTTP à ce même serveur Web. Combien de sockets seront ouvertes sur le serveur dues aux requêtes initiées à  $t_0$  et  $t_1$ ? On suppose qu'initialement (avant  $t_0$ ) aucune socket n'est ouverte sur le serveur.

- 4
- 5
- 2
- 3