TD 1 — Ethernet

Exercice 1 — Envoi de données sur Ethernet

Soit une machine connectée à un réseau Ethernet IEEE 802.3 en bus à 10 Mbit/s. Un processus s'exécutant sur cette machine doit envoyer 2 922 octets de données TCP à un processus s'exécutant sur une autre machine du réseau. On fait les hypothèses suivantes :

— la connexion TCP entre les deux processus a déjà eu lieu
— aucune option (IP ou TCP) n'est utilisée lors du transfert

Q 1.1 Quel est le temps total d'envoi de ces données en supposant que la machine est la seule à émettre sur le bus ?

Correction

En utilisant TCP, chaque trame transmise contiendra :
— l'en-tête IP de 20 octets ;
— l'en tête TCP de 20 octets ;
— et des données des processus.

La trame Ethernet ne pouvant contenir plus de 1 500 octets, on aura au maximum $1\,500-20-20=1\,460$ octets de données parmi les 2 922 à envoyer.

On devra donc transmettre trois trames : les deux premières contiendront 1 460 octets de données et la troisième contiendra $2\,922-1\,460\times 2=2$ octets de données.

La troisième trame contiendra $20+20+2=42$ octets (en-tête IP + en-tête TCP + données) ce qui fait moins de 46 octets. On doit rajouter 4 octets de bourrage Ethernet pour atteindre les 46 octets.

En rajoutant le PCI Ethernet on doit donc transmettre en tout :
— 2 trames de $26+20+20+1\,460=1\,526$ octets
— 1 trame de $26+20+20+2+4=72$ octets

soit $1\,526\times 2+72=3\,124$ octets.

La machine est la seule à émettre, il n'y a donc pas aucune possibilité de collision. Le temps de transmission T_t est donc

$$T_t=\frac{3\,124\times 8}{10\times 10^6}=0,0024992s$$

Pour obtenir le temps total il faut rajouter 2 temps d'attente inter-trame (IFG) de $9,6\mu s$, ce qui donne un temps total d'envoi noté T :

$$T=T_t+2\times IFG=T_t+2\times 9,6\times 10^{-6}=0,0025184s\approx 2,5ms$$

◆

Q 1.2 Quel est maintenant le temps total d'envoi de ces données si l'on suppose que
(a) La première trame transmise subit quatre collisions successives avant d'être correctement transmise à la cinquième tentative.
(b) L'interface attend le temps maximal donné par l'algorithme BEB après la détection d'une collision.
(c) Les collisions sont détectées après la transmission de l'adresse de destination.
(d) Après l'attente après collision le bus est libre pour une transmission.

Correction

Les trames subissant des collisions font 14 octets (Préambule + SFD + DA). Après chaque collision détectée la station émet aussi 4 octets de brouillage pour prévenir les autres stations de la collision. Le temps de transmission d'une trame subissant une collision est donc :

$$T_c=\frac{18\times 8}{10\times 10^6}=14,4\mu s$$

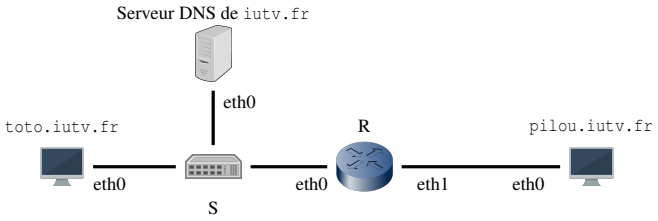
On attend $2^1-1=1\times DAR$ après la 1ère collision, $2^2-1=3\times DAR$ après la 2ère collision, $2^3-1=7\times DAR$ après la 3ère collision et $2^4-1=15\times DAR$ après la 4ère collision. Par définition $DAR=51,2\mu s$ en Ethernet. Par contre le délai inter-trame n'est plus nécessaire après chaque collision car on attend déjà un temps supérieur à ce délai. On a donc maitenant un temps d'envoi total noté T' tel que :

$$\begin{aligned}T' &= T+4\times T_c+(1+3+7+15)\times DAR\\ &= 0,0025184+4\times 14,4\times 10^{-6}+26\times 51,2\times 10^{-6}\\ &= 0,0039072\\ &\approx 3,9ms\end{aligned}$$

◆

Exercice 2 — Interactions entre Ethernet et IP

Soit l'architecture de la figure ci-dessous. Les machines sont interconnectées par un switch S et un routeur R.



Les adresses MAC et IP associées aux interfaces sont les suivantes :

	MAC	IP
eth0 de `toto.iutv.fr`	01:01:01:01:01:01	10.0.1.1
eth0 du serveur DNS de `iutv.fr`	02:02:02:02:02:02	10.0.1.2
eth0 de R	03:03:03:03:03:03	10.0.1.254
eth1 de R	04:04:04:04:04:04	10.0.2.254
eth0 de `pilou.iutv.fr`	05:05:05:05:05:05	10.0.2.1

Sur la machine `toto.iutv.fr` un utilisateur exécute la commande suivante :

```
$ ping -c1 pilou.iutv.fr
```

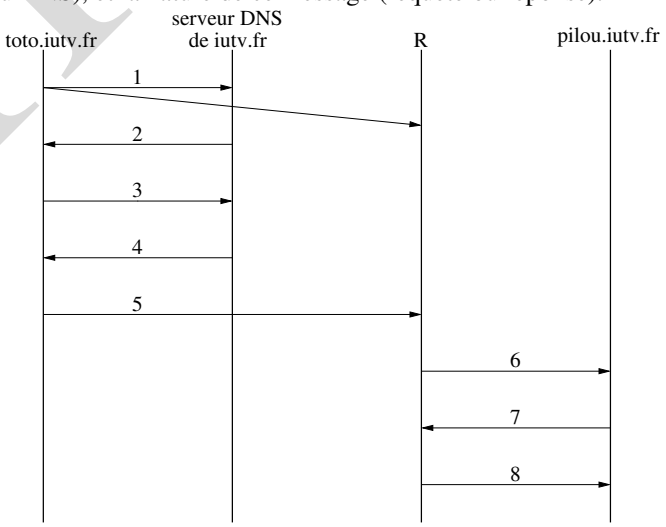
La commande ping utilise le protocole ICMP (qui repose sur IP) : elle envoie un message ICMP de type "*demande d'écho*" à la machine destinataire qui répond par un message ICMP de type "*envoi d'écho*". On suppose que DNS utilise UDP comme protocole de transport.

Enfin avant l'exécution du ping, les tables ARP contenaient les entrées suivantes :

— Table de `toto.iutv.fr` : (10.0.1.254,03:03:03:03:03:03)
— Table du serveur DNS de `iutv.fr` : (10.0.1.1, 01:01:01:01:01:01)
— Tables de R et `pilou.iutv.fr` : vides

En capturant les trames sur le réseau on arrive à reproduire le chronogramme ci-dessous sur lequel figurent toutes les trames envoyées pour que la demande d'écho envoyée par ping arrive à `pilou.iutv.fr`.

Q 2.1 Donner la signification des trames capturées en indiquant pour chacune d'elles : le protocole du message encapsulé dans la trame (ARP, ICMP ou DNS), et la nature de ce message (requête ou réponse).



Correction

Dans un premier temps toto doit convertir le nom `pilou.iutv.fr`. Il s'adresse pour cela au serveur DNS de la zone `iutv.fr` qui est sur le même réseau que lui. Cependant toto ne connaît pas l'adresse MAC de ce serveur. Il envoie donc une requête ARP (1) sur tout le réseau pour découvrir l'adresse MAC du serveur DNS. Seul le serveur lui répond (2). Il peut alors demander au serveur DNS l'adresse IP de pilou (3) qui lui donne (4). toto peut alors envoyer son message ping (5). Le routeur doit le relayer mais il ne connaît pas l'adresse MAC de pilou. Il envoie donc une requête ARP pour la découvrir (6). pilou lui donne son adresse MAC dans (7). Le routeur peut alors lui relayer le ping de toto.

En résumé :

— [1] ARP : Quelle est l'adresse MAC correspondant à l'adresse IP 10.0.1.2 ?
— [2] ARP : L'adresse MAC correspondant à l'adresse IP 10.0.1.2 est 02:02:02:02:02:02.
— [3] DNS : Quelle est l'adresse IP correspondant à `pilou.iutv.fr` ?
— [4] DNS : L'adresse IP correspondant à `pilou.iutv.fr` est 10.0.2.1.
— [5] ICMP : Demande d'écho pour 10.0.2.1.
— [6] ARP : Quelle est l'adresse MAC correspondant à l'adresse IP 10.0.2.1 ?
— [7] ARP : L'adresse MAC correspondant à l'adresse IP 10.0.2.1 est 05:05:05:05:05:05.

— [8] ICMP : Demande d'écho pour 10.0.2.1. ◆

Q 2.2 Quelles entrées supplémentaires contiendront les tables ARP après l'envoi de la demande d'écho ?

Correction

— Pour `toto.iutv.fr` : (10.0.1.2, 02:02:02:02:02:02)
— Pour R : (10.0.2.1, 05:05:05:05:05:05) ◆

Q 2.3 Donner la structure des trames échangées en faisant apparaître les PCI des protocoles utilisées.

Correction

— Pour les messages ARP : Ethernet - ARP
— Pour les messages DNS : Ethernet - IP - UDP - DNS
— Pour les messages ICMP : Ethernet - IP - ICMP ◆

Q 2.4 Donner pour chacune de ces trames les adresses MAC source et destination apparaissant dans les PCI Ethernet et, pour les trames encapsulant des paquets IP, les adresses IP source et destination apparaissant dans les PCI IP.

Correction

	MAC source	MAC destination	IP source	IP destination
1	01:01:01:01:01:01	FF:FF:FF:FF:FF:FF	—	—
2	02:02:02:02:02:02	01:01:01:01:01:01	—	—
3	01:01:01:01:01:01	02:02:02:02:02:02	10.0.1.1	10.0.1.2
4	02:02:02:02:02:02	01:01:01:01:01:01	10.0.1.2	10.0.1.1
5	01:01:01:01:01:01	03:03:03:03:03:03	10.0.1.1	10.0.2.1
6	04:04:04:04:04:04	FF:FF:FF:FF:FF:FF	—	—
7	05:05:05:05:05:05	04:04:04:04:04:04	—	—
8	04:04:04:04:04:04	05:05:05:05:05:05	10.0.1.1	10.0.2.1

Remarques :
— Une requête ARP est toujours envoyée à toutes les machines du réseau local (adresse MAC de destination = FF:FF:FF:FF:FF:FF).
— Les adresses MAC identifient toujours les machines sur le même réseau local (l'émettrice et la destinataire de la trame) alors que les adresses IP identifient toujours l'émetteur et le destinataire final des paquets circulant sur Internet. Par exemple, pour les messages ICMP 5 et 8, les adresses MAC changent mais pas les adresses IP car elles identifient `toto.iutv.fr` et `pilou.iutv.fr`. ◆

Q 2.5 Dans quel ordre les tables ARP et de routage sont-elles consultées avant l'envoi d'une trame ? Détailler comment ces tables sont consultées avant l'envoi des trames 1 et 5.

Correction

On consulte d'abord la table de routage pour connaître l'IP du prochain sur la route vers le destinataire final : soit l'IP du prochain routeur sur la route dans le cas d'une remise indirecte, soit l'IP du destinataire final dans le cas d'une remise directe.
On consulte ensuite la table ARP pour trouver l'adresse MAC correspondant à l'IP du destinataire de la trame (soit le routeur, soit le destinataire final du paquet). Si cette adresse n'est pas dans la table ARP, on envoie une requête ARP pour la découvrir.
Opérations faites par `toto.iutv.fr` pour la trame 1 :
(a) consultation de la table de routage : recherche de 10.0.1.2 ⇒ remise directe via eth0
(b) consultation de la table ARP : recherche de 10.0.1.2 ⇒ adresse MAC non trouvée
(c) [⇒] transmission de la trame 1
Opérations faites par `toto.iutv.fr` pour la trame 5 :
(a) consultation de la table de routage : recherche de 10.0.2.1 ⇒ remise indirecte au routeur 10.0.1.254 via eth0
(b) consultation de la table ARP : recherche de 10.0.1.254 ⇒ adresse MAC trouvée : 03:03:03:03:03:03
(c) [⇒] transmission de la trame 5 ◆

Exercice 3 — Codage de trames Ethernet 802.3

On considère dans cet exercice une machine M sur laquelle on observe les résultats suivants :

```
$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:25:64:C4:2D:95
inet adr:10.10.0.1  Bcast:10.10.255.255  Masque:255.255.0.0
eth1      Link encap:Ethernet  HWaddr 00:25:64:AA:E1:DF
inet adr:10.20.0.1  Bcast:10.10.255.255  Masque:255.255.0.0

$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.10.0.97               ether   00:14:22:2E:60:81   CM                    eth0
10.10.0.254              ether   00:50:56:BE:7F:9D   CM                    eth0
10.20.0.254              ether   00:50:56:A8:A4:E6   CM                    eth1
10.20.0.11               ether   00:B0:D0:F9:C3:BA   CM                    eth1
```

```
$ route -n
Table de routage IP du noyau
Destination     Passerelle      Genmask         Indic Metric Ref    Use Iface
0.0.0.0         10.10.0.254     0.0.0.0         UG    0      0        0 eth0
10.10.0.0       *               255.255.0.0     U     0      0        0 eth0
10.20.0.0       *               255.255.0.0     U     0      0        0 eth1
1.2.3.0         10.20.0.254     255.255.255.0   UG    0      0        0 eth1
```

Un processus s'exécutant sur M envoie les 6 octets de données suivants (en hexadécimal) à la machine d'adresse IP 1.2.3.4/24 : `AA BB CC DD EE FF`. Ce processus utilise le port 10 et le processus sur la machine 1.2.3.4/24 utilise le port 20. Aucune option IP n'est utilisée. Le protocole de transport utilisé est UDP.

Donner le contenu de la trame Ethernet (en hexadécimal) envoyée par le processus. Le préambule et le champ SFD ne devront pas apparaître. On utilisera le caractère X pour les champs qui ne peuvent pas être déduits de l'énoncé.

Correction

Le paquet est destiné à 1.2.3.4. Dans un premier temps M consulte sa table de routage pour savoir sur quelle interface et à qui envoyer ce paquet. C'est la dernière ligne de la table de routage (commande route) qui s'applique. La consultation de la table de routage donne donc : *remise indirecte à* 10.20.0.254 *sur eth1*. Le paquet doit donc être encapsulé dans une trame destinée à 10.20.0.254 en utilisant l'interface eth1. La commande arp indique ensuite l'adresse MAC associée à 10.20.0.254 : 00:50:56:A8:A4:E6. Enfin la commande ifconfig nous donne les adresses MAC et IP de M associées à eth1 : 00:25:64:AA:E1:DF et 10.20.0.1. En résumé, les adresses qui doivent apparaître dans les PCI Ethernet et IP sont :

— MAC source = 00:25:64:AA:E1:DF
— MAC destination = 00:50:56:A8:A4:E6
— IP source = 10.20.0.1
— IP destination = 1.2.3.4

On retrouve les PCI suivants dans la trame : Ethernet — IP — UDP.
Le contenu de la trame est donc le suivant :

En-tête Ethernet	00	50	56	A8	A4	E6	00	25
	64	AA	E1	DF	08	00	45	XX
En-tête IP	00	22	XX	XX	XX	XX	XX	11
	XX	XX	0A	14	00	01	01	02
	03	04	00	0A	00	14	00	0E
En-tête UDP	XX	XX	AA	BB	CC	DD	EE	FF
Données	00	00	00	00	00	00	00	00
Bourrage Ethernet (les 00)	00	00	00	00	XX	XX	XX	XX
FCS Ethernet (les XXX)								

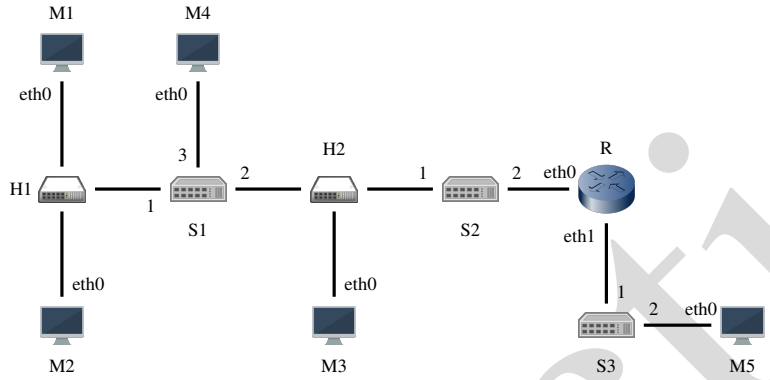
Dans l'en-tête Ethernet :
— `08 00` = DL/Etype. `08 00` est le code Etype qui identifie le protocole IP.
Dans l'en-tête IP :
— `45` = numéro de version d'IP utilisée (4) + longueur de l'en-tête IP en unités de 4 octets (5)
— `XX` = type de service
— `00 22` = longueur totale du paquet IP. Le paquet IP contient : 20 octets d'en-tête IP + 8 octets d'en-tête UDP + 6 octets de données = 34 octets en tout soit 22 en hexadécimal.
— `XX XX XX XX XX` = Identification, DF, MF, Position, Durée de vie
— `11` = code du protocole UDP
— `XX XX` = total de contrôle
— `0A 14 00 01` = adresse IP source
— `01 02 03 04` = adresse IP destination
Dans l'en-tête UDP :
— `00 0A` = port source
— `00 14` = port destination
— `00 0E` = longueur total du paquet UDP. Le paquet UDP contient : 8 octets d'en-tête UDP + 6 octets de données = 14 octets en tout, soit E en hexadécimal

La trame encapsule un paquet IP de 34 octets. On doit donc avoir, après le paquet IP, $46 - 34 = 12$ octets de bourrage Ethernet, et les 4 octets du FCS Ethernet. ◆

TD 2 — Interconnexion de réseaux Ethernet

Exercice 1 — Commutation

Soit le réseau ci-dessous. S1, S2 et S3 sont des switchs, H1 et H2 des hubs, et R un routeur.



Machine	Interface	Adresse MAC
M1	eth0	01:01:01:01:01:01
M2	eth0	02:02:02:02:02:02
M3	eth0	03:03:03:03:03:03
M4	eth0	04:04:04:04:04:04
M5	eth0	05:05:05:05:05:05
R	eth0	06:06:06:06:06:06
	eth1	07:07:07:07:07:07

Q 1.1 Donner les tables de commutation de S1, S2 et S3 en supposant qu'elles sont pleines.

Correction

S1	
Adresse MAC	Port
01:01:01:01:01:01	1
02:02:02:02:02:02	1
03:03:03:03:03:03	2
04:04:04:04:04:04	3
06:06:06:06:06:06	2

S2	
Adresse MAC	Port
01:01:01:01:01:01	1
02:02:02:02:02:02	1
03:03:03:03:03:03	1
04:04:04:04:04:04	1
06:06:06:06:06:06	2

S3	
Adresse MAC	Port
05:05:05:05:05:05	2
07:07:07:07:07:07	1

Remarque. Le routeur R sépare les deux réseaux Ethernet. C'est pourquoi S1 et S2 ne recevront jamais de trame de M5 et que l'adresse MAC de M5 n'est pas dans leurs tables. ◆

Q 1.2 On suppose maintenant que les tables de commutation des trois switchs sont vides et que les trois trames ci-dessous sont successivement envoyées.

— trame 1 : SA = 04:04:04:04:04:04 et DA = 03:03:03:03:03:03
— trame 2 : SA = 02:02:02:02:02:02 et DA = 04:04:04:04:04:04
— trame 3 : SA = 03:03:03:03:03:03 et DA = 02:02:02:02:02:02

Donner, pour chacune de ces trames, la liste des machines qui la recevront. Quels sont les états des tables de commutation de S1, S2 et S3 à la fin de cette séquence ?

Correction

	M1	M2	M3	M4	M5	R
trame 1	×	×	×			×
trame 2	×			×		
trame 3	×	×				×

Au début les tables sont vides. Par conséquent, les switchs retransmettront la trame 1 sur tous leurs ports (hormis le port de réception). M5 est la seule machine à ne pas recevoir la trame car elle est séparée du reste du réseau par R (qui ne retransmet pas la trame). Après l'envoi de la trame 1, S1 et S2 ont inséré l'adresse MAC source de la trame dans leurs tables.
La trame 2 est reçue par S1 sur son port 1. S1 la retransmet uniquement sur son port 3 car l'adresse du destinataire est dans sa table. Après l'envoi de la trame 2, seul S1 a inséré l'adresse MAC source de la trame dans sa table.
La trame 3 est reçue par S1 sur son port 2 et par S2 sur son port 1. S1 a l'adresse MAC du destinataire dans sa table. Il ne la retransmet donc que sur son port 1. Par contre, S2 n'a pas cette adresse MAC dans sa table. Il retransmet donc la trame sur tous ses ports hormis le port de réception, donc sur son port 2. Après l'envoi de la trame 3, S1 et S2 ont inséré l'adresse MAC source de la trame dans leurs tables.
Après l'envoi de ces trames, les tables de commutation de contiendront les entrées suivantes :

S1	
Adresse MAC	Port
02:02:02:02:02:02	1
03:03:03:03:03:03	2
04:04:04:04:04:04	3

S2	
Adresse MAC	Port
03:03:03:03:03:03	1
04:04:04:04:04:04	1

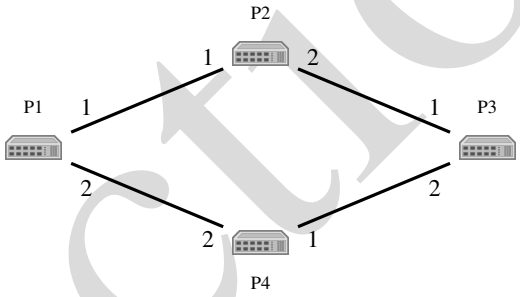
La table de S3 est vide.

Exercice 2 — Utilité de STP

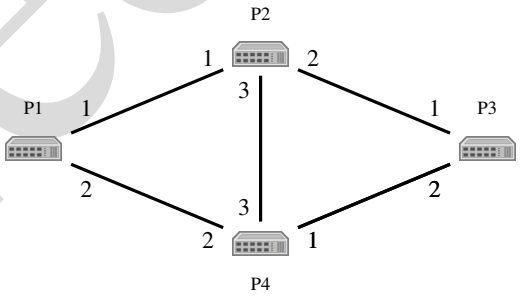
Dire, pour chacun des trois réseaux de switchs ci-dessous, si le protocole STP est nécessaire et donner le résultat de son exécution (les configurations finales des ponts et les ports bloqués) en supposant que les adresses MAC sont ordonnées ainsi : P1<P2<P3<P4.

Réseau 1

Réseau 2



Réseau 3



Correction

Réseau 1 On n'observe pas de cycle ⇒ STP n'est pas nécessaire car il n'est pas besoin de bloquer des ports.
La configuration finale sera :

Pont	Configuration finale			Ports
	racine	distance	portRacine	bloqués
P1	P1	0	-1	aucun
P2	P1	1	1	aucun
P3	P1	2	1	aucun

Réseau 2 On observe 1 cycle : P1→P2→P3→P4→P1 ⇒ STP va permettre de bloquer un port
La configuration finale sera :

Pont	Configuration finale			Ports
	racine	distance	portRacine	bloqués
P1	P1	0	-1	aucun
P2	P1	1	1	aucun
P3	P1	2	1	2
P4	P1	1	2	aucun

P3 est a une distance de 2 de la racine P1. Il a deux chemins le menant à P1 : un passant par P2 et un passant par P4. Parmi ces deux-là il bloque celui le menant au pont qui a la plus grande adresse MAC, soit P4.

Réseau 3 On observe des cycles :
— P1→P2→P3→P4→P1
— P1→P2→P4→P1
STP va donc permettre de bloquer des ports.

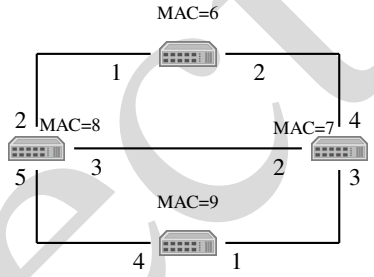
La configuration finale sera :

Pont	Configuration finale			Ports
	racine	distance	portRacine	bloqués
P1	P1	0	-1	aucun
P2	P1	1	1	aucun
P3	P1	2	1	2
P4	P1	1	2	3

Pour P3 c'est le même port bloqué que dans le cas précédent. P2 et P4 sont à la même distance de la racine P1 et sont tous les deux connectés. Dans ce cas c'est celui qui a la plus grande adresse MAC (P4) qui bloque son port qui le mène à l'autre. ◆

Exercice 3 — Fonctionnement de STP

Appliquer le protocole STP au réseau de la figure ci-dessous. À chaque étape vous indiquerez dans un tableau les trames BPDU reçues par chaque pont, la meilleure BPDU reçue, l'ancienne et la nouvelle (avant et après réception des BPDU) configuration du pont et enfin, le(s) port(s) éventuellement bloqués par le pont. On s'arrêtera lorsque le réseau ne contiendra plus de cycle.



Correction

Pont	Configuration	BPDU reçues	Meilleure BPDU reçue	Nouvelle configuration	Ports bloqués
Étape 1					
6	(6, 0, -1)	(8, 0, 8, 2) sur le port 1 (7, 0, 7, 4) sur le port 2	(7, 0, 7, 4)	(6, 0, -1)	aucun
7	(7, 0, -1)	(8, 0, 8, 3) sur le port 2 (9, 0, 9, 1) sur le port 3 (6, 0, 6, 2) sur le port 4	(6, 0, 6, 2)	(6, 1, 4)	aucun
8	(8, 0, -1)	(6, 0, 6, 1) sur le port 2 (7, 0, 7, 2) sur le port 3 (9, 0, 9, 4) sur le port 5	(6, 0, 6, 1)	(6, 1, 2)	aucun
9	(9, 0, -1)	(7, 0, 7, 3) sur le port 1 (8, 0, 8, 5) sur le port 4	(7, 0, 7, 3)	(7, 1, 1)	aucun
Étape 2					
6	(6, 0, -1)	aucune	—	(6, 0, -1)	aucun
7	(6, 1, 4)	(6, 1, 8, 3) sur le port 2 (6, 0, 6, 2) sur le port 4	(6, 0, 6, 2)	(6, 1, 4)	aucun
8	(6, 1, 2)	(6, 0, 6, 1) sur le port 2 (6, 1, 7, 2) sur le port 3 (7, 1, 9, 4) sur le port 5	(6, 0, 6, 1)	(6, 1, 2)	3
9	(7, 1, 1)	(6, 1, 7, 3) sur le port 1 (6, 1, 8, 5) sur le port 4	(6, 1, 7, 3)	(6, 2, 1)	4

Rappels et commentaires :

— Un pont O avec une nouvelle configuration (racine,distance,portRacine) bloque un port ($\neq$ de portRacine) sur lequel il a reçu une BPDU (R,D,E,P) si une des deux conditions est remplie :
C1 R = racine ET D < distance
C2 R = racine ET D = distance ET E<O

— À l'étape 2 la condition C2 est remplie pour le port 3 du pont 8 car il a reçu sur ce port la BPDU (R=6,D=1,E=7,P=2) : on a bien R=racine=6, D=distance=1 et E=7<O=8.
De son côté, 7 reçoit la BPDU (R=6,D=1,E=8,P=3) sur son port 2. Il effectue le même test que 8. On a bien R=racine=6 et D=distance=1 mais par contre E=8$\not<$O=7.
La condition C2 exprime donc le fait que quand deux ponts à égale distance du pont racine (R = racine ET D = distance) s'échangent des BPDU c'est le pont qui a la plus grande adresse MAC (E<O) qui bloque son port qui le relie à l'autre.

— À l'étape 2, la meilleure BPDU reçue par 9 est (6,1,7,3) sur le port 1. 1 devient donc son port racine. Sa configuration, après changement, devient :
  — racine = 6
  — distance = 2
  — portRacine = 1

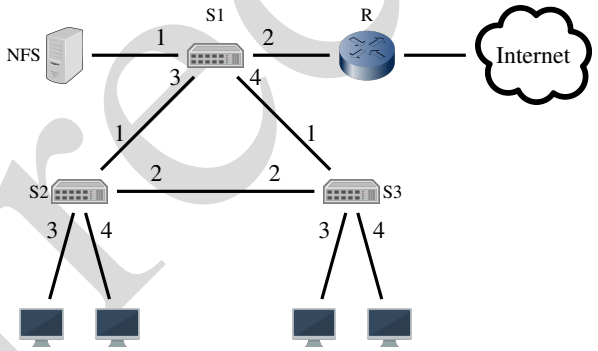
  Il a aussi reçu la BPDU (R=6,D=1,E=8,P=5) sur son port 4 (≠ portRacine). Comme R=racine=6 et D=1<distance=2, 9 bloque son port 4. 9 avait donc deux chemins de longueur 2 le menant au pont racine : un chemin passant par 7 et un chemin passant par 8. Il a gardé le chemin passant par le plus petit pont pour déterminer son port racine et bloqué l'autre chemin.
  Pour le port 4 de 9, c'est donc la condition de blocage C1 qui est remplie.
— Un pont ne retransmet jamais de BPDU sur son port racine ou sur un port bloqué. ⇒ À l'étape 2, 6 ne reçoit aucune BPDU et 7 ne reçoit pas de BPDU venant de 9. ◆

Exercice 4 — Choix de la racine

Soit le réseau de la figure ci-dessous. On a observé que les machines connectées au switch S2 et S3 accèdent très fréquemment au serveur NFS ainsi qu'à Internet via le routeur R connecté au switch S1.

Dans STP, il existe un système de priorité qui permet à l'administrateur de choisir le switch qui deviendra la racine. L'administrateur associe une priorité à chaque switch et c'est le switch ayant la plus petite priorité (plutôt que la plus petite adresse MAC) qui deviendra la racine.

Avec l'architecture ci-dessous et vu les observations faites, à quel switch est-il préférable d'attribuer la priorité la plus faible ?



Correction

Si S2 devient la racine, la liaison entre S1 et S3 sera coupée car un de ces deux switchs (celui qui a la plus grande adresse MAC) bloquera son port qui le mène à l'autre. Symétriquement, si S3 devient la racine la liaison entre S1 et S2 sera coupée.

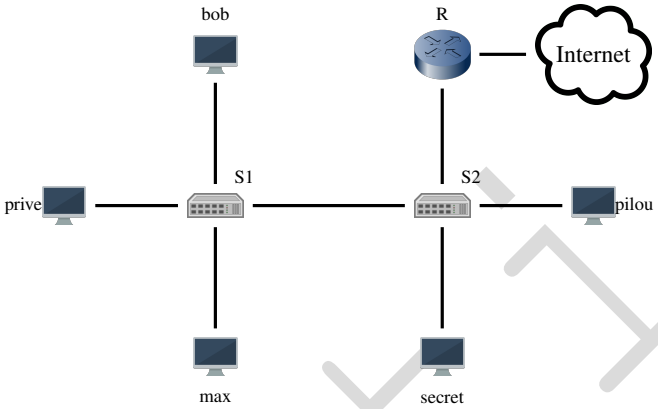
Dans ces deux cas de figure, le nombre moyen de ponts traversés pour atteindre le serveur NFS et le routeur (depuis les machines) sera de 2,5 : 2 pour les machines reliées à la racine et 3 pour les autres.

Par contre si S1 est la racine, il faudra toujours passer par 2 ponts pour atteindre le serveur NFS et le routeur (depuis les machines) car seule la liaison entre S2 et S3 sera coupée.

Par conséquent, il est préférable de choisir S1 comme racine en lui donnant la priorité la plus faible. C'est dans ce cas de figure que l'on observera le moins de communications et donc les meilleures performances. ◆

TD 3 — Réseaux locaux virtuels

On considère le réseau de la figure ci-dessous.



On suppose que les machines prive et secret s'échangent des données confidentielles et on souhaite que ces données ne puissent pas être interceptées par une autre machine du réseau.

Q 1 Dans quelle situation une trame émise par prive en direction de secret pourra-t-elle être reçue par une autre machine du réseau ?

Correction

La trame passera par les switchs S1 et S2. Si un de ces switchs n'a pas l'adresse MAC de secret dans sa table de commutation il retransmettra cette trame sur tous ces ports, et elle sera donc reçue par une machine à laquelle elle n'est pas destinée. ◆

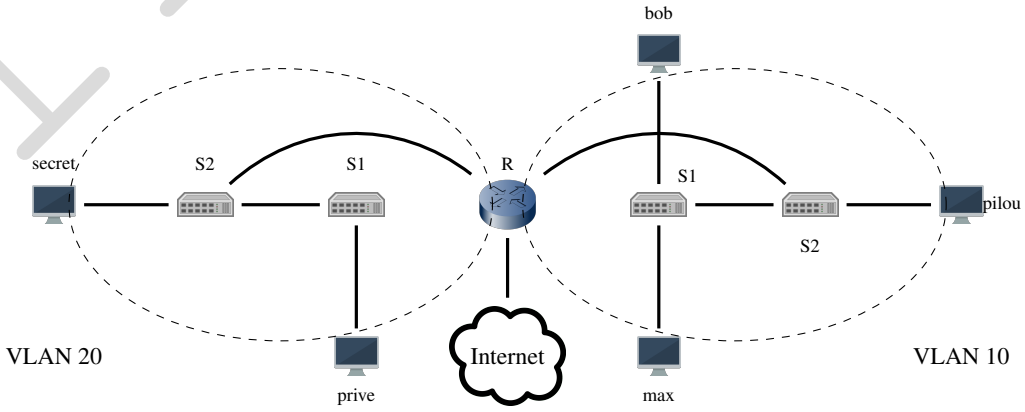
Pour répondre à ces contraintes de confidentialité, on décide de segmenter le réseau en 2 VLAN :

— Les machines bob, max et pilou sont placées sur le VLAN 10.
— Les machines secret et prive sont placées sur le VLAN 20.
— R est sur les deux VLAN.

Q 2 Dessiner la topologie logique (sans VLAN) du réseau.

Correction

D'un point de vue logique, la réseau avec VLAN est équivalent à deux réseaux Ethernet séparés par le routeur R.



◆

Q 3 Est-il possible, avec ce découpage, que des trames échangées par secret et prive puissent être reçues par d'autres machines ?

Correction

Non. R ne retransmet aucune trame du VLAN 20 sur le VLAN 10 (et inversement). ◆

Q 4 Quelles seront alors les liaisons trunks ? et les liaisons standard ?

Correction

Liaison trunk ⇔ liaison sur laquelle peuvent circuler des trames de différents VLAN (donc étiquettées). Donc, ici : la liaison S1↔S2 et la liaison S2↔R. ◆

Après la segmentation en VLAN, bob émet la trame ci-dessous, codée en hexa-décimale :

```
ff ff ff ff ff ff 00 14 22 5f 8b 7f 08 06 00 01
08 00 06 04 00 01 00 14 22 5f 8b 7f 0a 0a 00 01
00 00 00 00 00 00 0a 0a 00 02 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 ae d1 02 48
```

Les champs Préambule et SFD de l'en-tête Ethernet n'y apparaissent pas.

Q 5 Analyser cette trame.

Correction

Champ de l'en-tête Ethernet :
- `ff ff ff ff ff ff` = adresse MAC destination
- `00 14 22 5f 8b 7f` = adresse MAC source
- `08 06` = DL/Etype = code du protocole ARP ⇒ La trame contient un message ARP.

Champ du message ARP :
- `00 01` = Type de réseau physique = Ethernet (type d'adresse physique qu'on veut obtenir)
- `08 00` = Type de protocole = IP (type d'adresse logique sur lequel la requête se fait)
- `06` = Lg. adresse physique = longueur des adresses physiques (6 octets pour Ethernet)
- `04` = Lg. adresse protocole = longueur des adresses logiques (4 octets pour IP)
- `00 01` = type d'opération = requête
- `00 0a 22 5f 8b 7f` = adresse physique de la machine qui fait la demande
- `0a 0a 00 01` = adresse IP de la machine qui fait la demande = 10.10.0.1
- `00 00 00 00 00 00` = adresse physique de la machine qui a l'adresse IP demandée (à 0 car c'est une requête)
- `0a 0a 00 02` = adresse IP de la machine à laquelle on demande son adresse MAC = 10.10.0.2

Contenu de la dernière ligne :
- `00 00 00 00 00 00 00 00 00 00 00 00` = octets de bourrage Ethernet
- `ae d1 02 48` = champs FCS de la trame Ethernet

En résumé :

En-tête Ethernet	`ff ff ff ff ff ff 00 14 22 5f 8b 7f 08 06 00 01`	ARP
	`08 00 06 04 00 01 00 14 22 5f 8b 7f 0a 0a 00 01`	
	`00 00 00 00 00 00 0a 0a 00 02 00 00 00 00 00 00`	Bourrage + FCS Ethernet
	`00 00 00 00 00 00 00 00 00 00 00 00 ae d1 02 48`	

Cette trame est envoyée par la machine 10.10.0.1 d'adresse MAC 00:14:22:5f:8b:7f. La trame encapsule le message ARP suivant : quelle est l'adresse MAC de la machine dont l'adresse IP est 10.10.0.2 ? ◆

Q 6 Que fera S1 à la réception de cette trame ? Comment la trame sera-t-elle modifiée après retransmission sur la liaison entre S1 et S2 ?

Correction

S1, voyant que c'est une trame de diffusion la retransmet sur toutes les liaisons trunk ainsi que sur toutes les liaisons standard associées au VLAN du port de réception de la trame (le VLAN 10) : donc vers max et S2.
Avant la retransmission sur la liaison trunk, S1 doit insérer dans la trame l'étiquette de VLAN qui indiquera à S2 que cette trame circule sur le VLAN 10.
La trame, après modification, devient (les 4 octets soulignés forment l'étiquette) :

```
ff ff ff ff ff ff 00 14 22 5f 8b 7f 81 00 20 0a
08 06 00 01 08 00 06 04 00 01 00 14 22 5f 8b 7f
...
```

Les quatre octets de l'étiquette se décompose ainsi :
- `81 00` = 2 octets à la place du DL/Etype Ethernet indiquant que la trame contient une étiquette de VLAN
- `2` = 0010_2 se décompose en :
  - 001_2 = priorité
  - 0 = DEI (Drop Eligible Indicator) = 0 ⇒ la trame ne peut pas être supprimée en cas de congestion
- `0 0a` = VLAN sur lequel circule la trame = 10 en décimal

Les valeurs de la priorité et du champ DEI ne peuvent pas être déduites de l'énoncé. Ce sont des suppositions. ◆

Q 7 Que fera S2 à la réception de cette trame ?

Correction

S2 voit dans l'étiquette de la trame qu'il reçoit que la trame circule sur le VLAN 10 et que c'est une trame de diffusion. Il la retransmet donc sur toutes les liaisons trunk ainsi que sur toutes les liaisons standard associées au VLAN 10 : donc vers pilou et R. Avant de retransmettre la trame, S2 retire l'étiquette de VLAN. La trame reçue par pilou et R est donc la même que celle émise par bob. ◆