Réseaux IP

Introduction aux Réseaux TCP/IP

David TILLOY < d.tilloy@nnx.com>

Table des matières

Le matériel	3
Une normalisation	4
Sécurité des transmissions (fiabilité)	5
Encapsulation	7
Liaisons	7
Address Resolution Protocol ou ARP	8
Internet Protocol (IP)	9
Le transport : TCP et UDP	10
Ports TCP/UDP : Multiplexage/Démultiplexage	13
Les adresses IP	14
Adressage IP	15
Informations de configuration TCP/IP	15
Calcul du masque de sous-réseau (netmask)	16
Calcul de l'adresse de diffusion (broadcast)	16
Station particulière : la passerelle (gateway)	17
Routage IP	18

2

Introduction

Faire communiquer les systèmes.

Lorsque les systèmes multi-utilisateurs ont connu une stabilité satisfaisante, les recherches se sont orientées vers un protocole de communication de ces systèmes, entraînant diverses expérimentations dans le monde. C'est l'armée américaine qui accélèrera ce processus, avec le financement de programmes de recherches universitaires pour le développement de voies de télécommunications redondantes (ARPANet). En cas de conflit militaire, la bonne circulation de l'information et sa confidentialité sont majeures, et les liaisons de communication étaient les premières cibles militaires.

Le standard des implémentations TCP/IP est issu des centres de Recherches en Systèmes Informatiques de l'Université de Berkeley (Californie, USA). Sa diffusion a été assurée par le système 4.X BSD. Aujourd'hui, les implémentations TCP/IP (*Transmission Control Protocol/Internet Protocol*) sont caractérisées par des versions majeures Net/X¹ du BSD Networking Software.

Les étapes de la communication

Le matériel

Toute communication demande un support. L'informatique n'échappe pas à cette règle, et il a donc été nécessaire de mettre au point des interfaces capable de traduire le langage binaire d'un système digital en signal adapté à un support (paire de cuivre, câble coaxial, fibre optique, etc.). Ces interfaces comportent des circuits électroniques permettant d'écouter ou d'émettre sur un support. Chaque adaptateur dispose d'une petite quantité de mémoire, qui peut être accédée par le système hôte (PC, etc.).

Une première phase de développement a donc consisté à mettre au point ces adaptateurs, ainsi que de fournir en vue de développements à venir une documentation précise sur les registres et adresses à utiliser pour exploiter les fonctions de communication.

Ces opérations forment la première couche du modèle de référence OSI. C'est la couche physique. Elle permet une communication entre un système (digital) et un support (analogique) de transmission (ondes, laser, cuivre, fibre optique, etc.).

Les cartes les plus répandues sont des cartes **Ethernet 802.3** (RJ45, BNC, AUI) et peuvent supporter des débits de 10Mb/s ou 100Mb/s. Elles permettent de transcrire une donnée digitale (ex. **0011 0100**) en une tension adaptée au support (amplitude, codage, etc.).

Mais les adaptateurs réseaux gèrent également les états du support, et savent détecter un certain nombre d'erreurs sur ce support. Cette partie est transparente aux développeurs, mais elle est toutefois assurée par chaque NIC (*Network Interface Card*).

¹ A l'heure actuelle (septembre 2000), les noyaux Linux et *BSD implémentent NET/4.

Toutes les cartes Ethernet disposent d'une adresse physique unique². Cette adresse, également appelée adresse MAC (Media Access Control), est utilisée pour les dialogues entre deux cartes. Elle est codée sur 6 octets³, dont les 3 premiers décrivent le constructeur (ex. 00:0a:24 désigne le constructeur 3COM). Les adresses MAC (Figure 1 format des adresses MAC) sont généralement représentées sous forme hexadécimale, chaque octet étant séparé par le symbole ':' (ex. 00:40:05:61:71:EC).

00	0a	24	12	3b	CO
(8 bits)	(8 bits)	(8 bits)	(8 bits)	(8 bits)	(8 bits)
Id.	construct	eur	Numé	ro désiré ι	unique

Figure 1 - format des adresses MAC

Une normalisation

Pour communiquer, chaque adaptateur doit être informé de toute demande de communication lui étant destinée. Il est également nécessaire d'informer les autres utilisateurs du réseau d'une volonté de communiquer avec un tiers. C'est dans ce cadre que les interfaces utilisent leur adresse MAC.

Mais les adresses MAC ne peuvent être utilisées que sur un réseau local (LAN⁴), et ne sont pas transportées sur les réseaux distants (MAN⁵ ou WAN⁶). Ces adresses étant généralement universelles (c'est à dire unique sur un segment physique donné), la gestion d'un parc informatique s'en trouve complexifiée au même titre que l'adressage (ex. filtrages de sécurité extrêmement complexes et fastidieux dans ces conditions).

Il a donc été décidé de mettre au point une méthode d'adressage logique (donc informatique et non matériel), qui permettrait à un ou plusieurs organismes de régulation de distribuer des adresses IP à ceux qui en feront la demande. Cette normalisation des adresses se nomme **Internet Protocol** (IP), et connaît aujourd'hui sa version 4, alors que la version 6 s'apprête à remplacer la version 4 aujourd'hui obsolète et frein de l'évolution d'internet.

D'autres protocoles autres que IP (IPX, DECNet, Appletalk, etc.) utilisent également les adresses MAC pour leurs fonctions d'adressages et/ou de routage.

Le protocole IP va donc permettre de communiquer simplement en indiquant l'adresse IP de l'émetteur et l'adresse IP du destinataire. Son seul rôle est d'émettre et recevoir les données, mais également de les orienter si nécessaire entre différentes interfaces réseaux (généralement des routeurs remplissent cette fonction, seuls équipement destinés à recevoir plusieurs interfaces de communication).

L'Internet Protocol (IP) se place donc comme couche réseau de la pile IP, puisqu'elle fournit fonctions et informations nécessaires au routage et adressage. Un équipement IP

² Par définition. Voir les adresses localement et universellement administrées

³ Bien que que protocole Ethernet 802.3 autorise les adresses MAC de 16 bits, on ne rencontre généralement que des adresses MAC de 48 bits (6 octets).

⁴Local Area Network

⁵ Metropolitan Area Network

⁶ Wide Area Network

⁷ Pour l'EUROPE, les demandes d'adresses IP doivent être transmises pour affectation au Réseau IP Européen (RIPE, http://www.ripe.net/).

n'a pas connaissance nécessairement de la topologie du réseau auquel il est relié, mais sait comment router les paquets (connaissance de la prochaine passerelle pour une destination donnée). Ce processus se résume à « la donnée émise vers la destination xxx.xxx.xxx existe dans ma table de routage et doit aller sur telle interface, ou n'existe pas est doit aller sur l'interface par défaut (si elle existe) ».

Ce processus se rapproche de la méthode de distribution du courrier postal. Le destinataire n'est pas dans ma ville, je l'envoie à la ville de destination, etc...

Ainsi, les paquets seront routés de proche en proche jusqu'à la destination, par le jeu des routages par défaut (les grosses infrastructures utilisant plusieurs liens de transits et/ou peering disposent de protocoles plus complexes pour la gestion du routage comme BGP⁸).

Les données seront donc «encapsulées » dans une trame IP. C'est à dire que chaque donnée fera l'objet de l'émission d'un datagramme IP comportant une entête fixe de quelques octets (16) et éventuellement d'un champ de contrôle en fin de données.

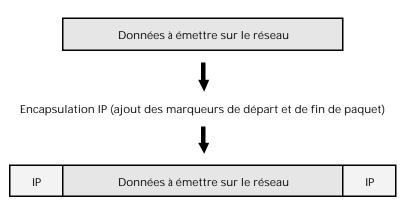


Figure 2 - processus d'encapsulation de données

Sécurité des transmissions (fiabilité)

La couche réseau permet donc de joindre un destinataire relié au réseau (directement ou indirectement d'où les fonctions de routage), mais prend en charge également des opérations de bases sur la gestion du service⁹. Des trames ICMP peuvent être échangées entre les routeurs ou les stations pour signaliser un évènement sur le réseau (perte de paquets, filtrage, taille de trame trop grande et fragmentation IP nécessaire, etc.).

Mais il est nécessaire de disposer d'une partie logicielle capable d'assurer la bonne émission/réception des données. Lorsqu'un paquet (ou datagramme) ne parvient pas au destinataire, sans intervention logicielle, le paquet n'est pas arrivé, mais n'est jamais réémis automatiquement.

Deux protocoles viennent combler ce manque: **UDP** (*User Datagramme Protocol*) et **TCP** (*Transmission Control Protocol*).

L'UDP ne contrôle pas la perte des paquets, chaque paquet et émis sans numérotation vers le destinataire, et sans acquiescement.

⁸ Border Gateway Protocol.

⁹ IPv4 apporte un minimum de fonctions pour la gestion des liens IP, IPv6 fournira quant à lui l'ensemble des fonctions nécessaires à la gestion de la qualité de service.

Le protocole TCP quant à lui assure un transfert plus fiable des données, en ouvrant une session de communication avant tout dialogue, puis en numérotant les paquets pour la reconstruction, en ré-émettant les paquets perdus ou erronés...

QUESTION: Quel protocole (UDP ou TCP) choisiriez-vous pour les applications suivantes et pourquoi?

- Une application de courrier électronique,
- Une radio en temps réel (diffusion « live » d'un serveur vers x usagers),
- Une application d'administration à distance,
- Des requêtes DNS (nombreux paquets IP mais de faible taille),
- Une application de téléphonie sur IP.

Les protocoles **TCP** et **UDP** se placent donc comme couche de transport dans la pile IP, car ce sont eux qui assurent la transmission des données d'un point à l'autre d'un réseau, en gérant les nécessaires ré-émissions (ou non) des paquets perdus ou altérés, etc.

Les modèles OSI et TCP/IP

Les communications entre systèmes ne sont possibles que si chaque système comprend son destinataire (un français ne parle pas espagnol, et vice-versa). Il a donc été nécessaire de définir une norme pour permettre à chacun de communiquer avec un réseau existant.

C'est en cela que TCP/IP est appelé réseau ouvert. Les protocoles utilisés sont normalisés et disponibles pour le monde entier. Chacun peut donc adapter son système propriétaire pour communiquer en TCP/IP, en écrivant les différents composants logiciels répondant aux normalisations TCP/IP (la majorité des OS disposent aujourd'hui d'une implémentation TCP/IP).

L'Open Systems Interconnection Reference Model a donc normalisé un modèle de référence OSI-RM, utilisant 7 couches distinctes. TCP/IP s'inscrit dans ce modèle, mais n'utilise pas systématiquement l'ensemble des 7 couches.

Application	7			
Présentation	6	Application (FTP, HTTP, DNS, SMTP, etc.)		
Session	5			
Transport	4	Transport (TCP)		
Réseau	3	Internet (IP)		
Liaison de données	2	PPP/SLIP/PLIP		
Physique	1	Couche physique		

Modèle de référence OSI

Modèle TCP/IP

Figure 3 - modèle de référence OSI et modèle TCP/IP

Le rôle de chaque couche est de permettre à la couche supérieure de lui passer des données qui seront émises, ainsi que de transmettre les données de la couche inférieure à la couche supérieure (données reçues).

On voit donc que pour une seule communication entre deux systèmes, il est nécessaire d'utiliser plusieurs protocoles.

QUESTION 1 : Quel avantage propose un modèle en couche plutôt qu'un modèle horizontal englobant toutes les fonctions ?

QUESTION 2: L'adresse IP est indépendante de l'adresse MAC, contrairement à d'autres protocoles comme IPX (Novell) où l'adresse MAC conditionne l'adresse du réseau IPX. Quelle technique d'adressage logiciel pourra-t-on utiliser avec IP qui est impossible avec IPX?

Documentation TCP/IP

- RFCs 1583, 1812, 1122, et 1123.
- Sites web de: APNIC (Asie), RIPE (Europe), INTERNIC (U.S.)

Encapsulation

Seule la couche supérieure (Application) contient les données et uniquement les données à émettre ou reçues. Chaque couche ajoute ses propres entêtes, encapsulant les paquets de données dans de plus grands paquets, ou enlevant les entêtes dans le cas d'une réception.

Lorsqu'un paquet de données demande à être émis par une application, ces données vont donc recevoir plusieurs entêtes fonction des protocoles utilisés.

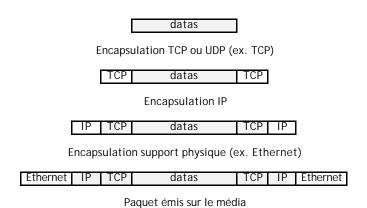


Figure 4 - encapsulation TCP/IP

Dans le cas d'une réception, chaque couche prendra les informations nécessaires et retirera ensuite ses entêtes pour donner le bloc de données restant à la couche de niveau immédiatement supérieur.

Liaisons

Il existe certains cas d'utilisation qui demandent une couche supplémentaire. Dans le cas d'un accès par modem (donc par liaison série), il n'y a pas d'adresse matérielle (adresse MAC) sur le modem. Cette adresse étant utilisée par les couches physiques et de réseau¹⁰, il ne peut en théorie y avoir de communication. Le modem n'étant pas une interface réseau mais série (la communication se fait par un port COM), elle ne dispose pas en standard d'adresse matérielle, ni de ROM fournissant une interface de communication IP.

¹⁰ Voir le protocole ARP

Il faut donc une couche logicielle supplémentaire, afin de simuler et fournir une alternative à l'utilisation des adresses MAC. Dans le cas d'une liaison TCP/IP avec un modem, on utilisera généralement le protocole **PPP** (*Point to Point Protocol*), qui se placera entre la couche réseau et la couche physique. Ce protocole fournira une solution logicielle aux communications IP nécessitant une adresse MAC.

Couches et protocoles utilisés

Chaque couche de la pile IP fait appel à un ou plusieurs protocoles pour remplir certaines fonctions (la couche transport utilise TCP ou UDP). Par cette méthode, les couches permettent de normaliser les flux de données entrants et sortants. Chaque couche (et donc chaque implémentation) est donc indépendante des couches supérieures et/ou inférieures.

On appelle **protocole**, un dialogue connu par les deux parties, entre deux couches de même niveau. Une couche de niveau (n) ne sera capable de dialoguer qu'avec une autre couche de même niveau qu'elle.

On appelle service l'ensemble des fonctions que doit absolument remplir une couche, fournissant l'interface pour transmettre des données de la couche (n) à la couche (n+1) ou (n-1).

Address Resolution Protocol ou ARP

Lors de dialogue entre deux stations, il est nécessaire que les adaptateurs réseaux soient en mesure de prendre les données qui leur sont adressées, sans traiter celles qui ne les concernent pas (d'où un gain de temps CPU et réseau). Certains réseaux fonctionnant sous forme de bus (Ethernet non switché, raccordement coaxiaux, etc.), toutes les données transitent sur le support, et donc tous les adaptateurs réseaux doivent analyser les trames pour ne prendre en compte que celles qui leurs sont destinées.

Les seules adresses disponibles et utilisables au niveau de l'interface physique (Couche 1 du modèle TCP/IP) sont les adresses MAC. Sans ces adresses, chaque adaptateur devrait décoder chaque trame jusqu'au niveau 3 (IP) pour savoir si cette donnée lui est adressée ou non.

Dans le cas d'un dialogue entre deux stations 10.23.23.2 et 10.23.23.254, la première étape consiste donc à trouver l'adresse matérielle de la station destinatrice, de manière à envoyer les données à cette station (en précisant son adresse matérielle plutôt qu'IP). C'est là qu'intervient le protocole ARP (niveau 3, couche réseau). Ce protocole va permettre à une station de découvrir l'adresse matérielle d'une autre station.

Pour cela, si 10.23.23.2 cherche à contacter 10.23.23.254, la station va, avant tout dialogue, diffuser (*broadcaster*) à l'ensemble des stations du réseau une requête ARP. Chaque station va alors recevoir cette requête ARP, composée du message suivant :

Station 10.23.23.2 d'adresse matérielle xx:xx:xx:xx:xx cherche l'adresse matérielle de la station 10.23.23.254.

Toutes les stations reliées à ce segment analysent alors cette demande, mais seule la station 10.23.23.254 va répondre à cette demande¹¹, en renvoyant le message suivant :

Station 10.23.23.254 a pour adresse matérielle yy:yy:yy:yy:yy.

¹¹ La réponse est destinée à l'émetteur et non plus à l'ensemble des hôtes du réseau.

Les deux stations 10.23.23.2 et 10.23.23.254 stockeront alors le couple (adresse IP, adresse MAC) obtenu dans un cache (dit cache ARP, cf. Figure 5 - exemple de table ARP) pour ne plus reposer cette question dans le cas d'une nouvelle communication dans un faible délai (quelques minutes avant que le cache ARP n'efface ce couple s'il n'est plus utilisé).

```
% arp -a gate. victim.com (192. 168. 16. 254) at 00: 50: 04: 34: EC: 84 [ether] on eth0 truc. victim.com (192. 168. 16. 74) at 00: 80: C8: 75: B0: 65 [ether] on eth0 dyn-23. victim.com (192. 168. 16. 151) at 00: 40: 05: 61: 72: 1F [ether] on eth0 al pha. victim.com (192. 168. 16. 34) at 00: 50: BA: C8: 20: E1 [ether] on eth0 dyn-32. victim.com (192. 168. 16. 160) at 00: C0: B6: 01: 92: A0 [ether] on eth0 triton. victim.com (192. 168. 16. 51) at 00: 80: C8: 6B: AC: EE [ether] on eth0 hp4050. victim.com (192. 168. 16. 93) at 00: 10: 83: BB: 7F: 4E [ether] on eth0
```

Figure 5 - exemple de table ARP

0.670265	004009011110	030000000000	Dotte	security oneck (0x03)
3.735371	0040056171EC	FFFFFFFFFFF	ARP_RARP	ARP: Request, Target IP: 10.23.23.254
3.745385	0040F9127042	0040056171EC	ARP_RARP	ARP: Reply, Target IP: 10.23.23.2 Target Hdwr Addr: 0040056171EC
2 245205	004005617176	004080122042	TOWN	Paka- Pram 10 90 90 00 Ta 10 90 90 954

Figure 6 - émission et réponse d'une requête ARP sur un segment local

On voit dans cette capture (Figure 6 - émission et réponse d'une requête ARP sur un segment local) que la station 10.23.23.2 a demandée l'adresse matérielle associée à l'adresse IP 10.23.23.254. L'adresse MAC de destination est une adresse de broadcast (toutes les stations du réseau), puisque tous les bits de l'adresse MAC sont à 1, à savoir 0xFFFFFFFFFF. L'ensemble des stations connectées au réseau devront donc traiter cette trame. La trame suivante montre la réponse de la station 10.23.23.254 à cette demande. Cette fois la réponse n'est envoyée qu'à la station émettrice (ce qui est indiqué comme adresse matérielle de réponse).

```
16: 08: 13. 422549 eth0 > arp who-has 10. 23. 23. 2 tell 10. 23. 23. 34 (0: 50: ba: c8: 20: e1) 16: 08: 13. 422915 eth0 < arp reply 10. 23. 23. 2 is-at 0: 40: 5: 36: 53: 3d (0: 50: ba: c8: 20: e1)
```

Figure 7 - une émission/réception ARP vue par tcpdump

Internet Protocol (IP)

Sur un segment Ethernet, il n'est pas nécessaire d'utiliser une couche matérielle ou logicielle pour remplir les fonctions de liaison. Les protocoles de niveau 2 sont utilisées uniquement sur des liens séries ou parallèles, ou tout autre interface ou équipement ne disposant pas d'adresse MAC (ex. le PPP ou SLIP pour les accès IP via modem).

Chaque trame circulant sur le réseau possède par le jeu des encapsulations successives plusieurs entêtes. Une trame de données a donc au minimum une entête liée au média utilisé (généralement Ethernet). C'est le cas de l'ARP. Les trames utilisant les adresses IP auront en plus des informations d'entête IP. Les informations contenues dans cette entête fixe de 20 octets (au minimum, voire plus si des options IP sont utilisées) renseignent sur la station émettrice (adresse IP), l'adresse du destinataire, le checksum (somme de contrôle), le protocole, la version, etc.

```
Frame: Base frame properties
◆ETHERNET: ETYPE = 0x0800 : Protocol = IP: DOD Internet Protocol
\rightleftharpoonsIP: ID = 0x179F; Proto = ICMP; Len: 60
   IP: Version = 4 (0x4)
   IP: Header Length = 20 (0x14)
   IP: Precedence = Routine
   IP: Type of Service = Normal Service
   IP: Total Length = 60 (0x3C)
   IP: Identification = 6047 (0x179F)
 \PhiIP: Flags Summary = 0 (0x0)
   IP: Fragment Offset = 0 (0x0) bytes
   IP: Time to Live = 128 (0x80)
   IP: Protocol = ICMP - Internet Control Message
   IP: Checksum = 0xDFF4
   IP: Source Address = 10.23.23.2
   IP: Destination Address = 10.23.23.254
   IP: Data: Number of data bytes remaining = 40 (0x0028)
ICMP: Echo: From 10.23.23.02 To 10.23.23.254
00000000
          00 40 F9 12 70 42 00 40 05 61 71 EC 08 00 45 00
                                                              .0".pB.0.aqý.
00000010
          00 3C
                17 9F 00 00 80 01 DF F4 0A 17 17 02 0A
          17 FE 08 00 42 5C 02 00 09 00 61 62 63 64 65 66
00000020
                                                                ..B\....abcdef
00000030
          67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76
                                                              ghijklmnopqrstuv
00000040 77 61 62 63 64 65 66 67 68 69
                                                              wabcdefghi
```

Figure 8 - décodage d'une entête de datagramme IP

Version	Longueur	Type de service	vice Taille totale			
	Identification		Flags	Offset pour données		
Time To Li	ve	Protocol	CRC d	'entête		
		Adresse II	P source			
	Adresse IP destination					
	0	ptions IP		Bourrage (padding)		

Figure 9 - définition des entêtes IP

Il existe 3 sortes d'adresses IP:

- Unicast qui ne s'adressent qu'à une station particulière,
- **Broadcast** qui s'adressent à toutes les stations,
- Multicast qui s'adressent à un groupe Multicast (prédéfini).

Le transport : TCP et UDP

La couche de transport (couche n°4 dans la pile IP) assure le bon transfert des données. C'est cette couche qui par exemple va numéroter les trames TCP avant de les émettre sur le réseau, pour que le destinataire puisse reconstruire l'ensemble des données dans le bon ordre (ce n'est pas le cas d'UDP).

Deux protocoles sont couramment utilisés dans un environnement TCP/IP. TCP et UDP.

Port source							Port destination	
				Nι	umé	éro (de s	séquence TCP
				Nι	umé	éro (d'ad	cquiescement
offset					R S T	'	F I N	Fenêtre
Somme de contrôle					Pointeur d'urgence			
Options							Bourrage	
Données TCP				es TCP				

Figure 10 - entête TCP

TCP assure la numérotation des paquets, et le destinataire acquiesce chaque paquet. Il est donc nécessaire pour les deux parties d'établir une négociation du dialogue. C'est pour cela qu'une communication TCP débute toujours par une synchronisation des deux protagonistes. L'émetteur demande au récepteur si ce dernier est prêt à recevoir les données, ce dernier acquiesce donc la demande, que valide l'émetteur. Les transferts de données peuvent alors commencer.

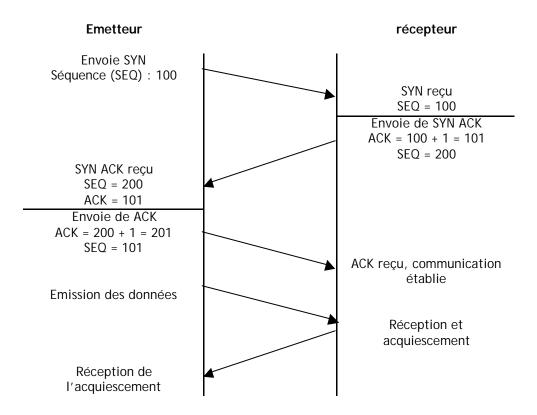


Figure 11 - négociation d'une échange TCP (établissement de la connexion)

```
♣Frame: Base frame properties
◆ETHERNET: ETYPE = 0x0800 : Protocol = IP: DOD Internet Protocol
♣IP: ID = 0x1E; Proto = TCP; Len: 44
⇒TCP: .A..S., len: 0, seq:1113022464-1113022464, ack:1661500826, win: 2048, src: 23 (TELNET)
   TCP: Source Port = Telnet
   TCP: Destination Port = 0x05C3
   TCP: Sequence Number = 1113022464 (0x42576000)
   TCP: Acknowledgement Number = 1661500826 (0x63087D9A)
   TCP: Data Offset = 24 (0x18)
   TCP: Reserved = 0 (0x0000)
  ➡TCP: Flags = 0x12 : .A..S.
    TCP: ..0.... = No urgent data
    TCP: ...l... = Acknowledgement field significant
    TCP: ....O... = No Push function
    TCP: .....O.. = No Reset
    TCP: .....l. = Synchronize sequence numbers
    TCP: .....0 = No Fin
   TCP: Window = 2048 (0x800)
   TCP: Checksum = 0xC7C8
   TCP: Urgent Pointer = 0 (0x0)
  TCP: Options
   ➡TCP: Maximum Segment Size Option
      TCP: Option Type = Maximum Segment Size
      TCP: Option Length = 4 (0x4)
      TCP: Maximum Segment Size = 512 (0x200)
00000000
         00 40 05 61 71 RC 00 40 F9 12 70 42 08 00 45 00
                                                             .@.aqý.@".pB..E.
000000010
         00 2C 00 1K 00 00 FK 06 79 80 0A 17 17 FK 0A 17
                                                             .,.D..;.yÇ...;..
00000020
         17 02 00 17 05 C3 42 57 60 00 63 08 7D 9A 60 12
                                                             .....+BW`.c.}Ü`.
         08 00 C7 C8 00 00 <mark>02 04 02 00 </mark>00 00
00000030
```

Figure 12 -décodage d'une entête TCP.

UDP est plus tolérant, plus rapide, mais également moins fiable dans sa technique de transmission. Les données sont émises sans aucune assurance que l'émetteur puisse les recevoir. Chaque paquet est émis sur le réseau (sans numérotation), au maximum de la vitesse possible (fonction de la station et de l'état du média). Si des paquets sont perdus, il

n'est pas possible pour l'émetteur de le détecter (ni pour le destinataire), de même que les données peuvent parvenir au destinataire dans un désordre complet suivant la complexité de la topologie du réseau.

```
Frame: Base frame properties
 $\displaystyle=\text{Conditions}$ \displaystyle=\text{Conditions}$ \displaystyle=\text{Conditions}$
$\displaystyle=\text{Conditions}$
$\d
 IP: ID = 0x17B6; Proto = UDP; Len: 57
  =UDP: Src Port: Unknown, (1476); Dst Port: DNS (53); Length = 37 (0x25)
            UDP: Source Port = 0x05C4
            UDP: Destination Port = DNS
            UDP: Total length = 37 (0x25) bytes
           UDP: UDP Checksum = 0x2A15
            UDP: Data: Number of data bytes remaining = 29 (0x001D)
00000000
                                       00 40 F9 12 70 42 00 40 05 61 71 EC 08 00 45 00
                                                                                                                                                                                                                                         .0".pB.0.aqý..E.
 00000010
                                       00 39 17 B6 00 00 80 11 AD 8E 0A 17 17 02 D4 55
                                                                                                                                                                                                                                         .9.Â..Ç.;Ä....ÈU
                                      80 01 05
                                                                                                                                                                                                                                                             5.%*§.ê....
00000020
                                                                                                35
                                                                                                           00 25
                                                                                                                                                15 00 88 01 00 00 01
                                                                                     00
00000030
                                     00 00 00 00 00 00 03 6E 6E 78 03 6E 6E 78 03 63
                                                                                                                                                                                                                                         .....nnx.nnx.c
00000040 6F 6D 00 00 01 00 01
                                                                                                                                                                                                                                         om....
```

Figure 13 - décodage d'une entête UDP

Ports TCP/UDP: Multiplexage/Démultiplexage

Une station peut émettre et recevoir simultanément plusieurs flux de données TCP et UDP. Pour cela, il est donc nécessaire que chaque extrémité (qui peuvent être différentes pour chaque communication établie) sachent à quel processus (système) rattacher une trame arrivant sur une interface. Pour cela, les protocoles TCP et UDP utilisent des numéros de ports. Ces numéros sont **PRIMORDIAUX** dans toute communication TCP ou UDP, et permettent d'associer une communication à un processus. Chaque donnée transitant sur le réseau s'est donc vu associer deux numéros de ports : le premier *côté émetteur*, le second *côté récepteur*. Chaque communication est donc caractérisée par deux couples (adresse IP, port utilisé) relatif à chaque extrémité.

Les ports TCP et UDP sont totalement indépendants. Il est donc possible d'avoir simultanément une communication sur le port 25/TCP et le port 25/UDP.

Cette technique se rapporte au multiplexage/démultiplexage, par décodage du numéro de port dans la trame, la donnée est envoyée à l'un ou l'autre des processus du système.

Par convention, les systèmes implémentent la logique suivante :

- Les numéros de ports inférieurs à 1024 ne peuvent être utilisés que par le super-utilisateur,
- Une application cliente utilisant TCP ou UDP utilisera un numéro de port supérieur à 1024 (même si l'utilisateur est le super utilisateur). Il existe toutefois certaines exceptions volontaires comme les r-services...

L'adressage IP

Chaque système souhaitant discuter sur le réseau mondial IP (Internet) doit disposer d'une adresse IP. Ces adresses, affectées par des organismes de régulation, sont classifiées et normalisées.

Une station de l'Internet ne peut être localisée (joignable) que par son **couple unique** (*Adresse IP*, *Masque de sous-réseau*).

Les adresses IP

Une adresse IP est composée de deux champs : l'adresse réseau et l'adresse machine. L'adresse réseau est placée sur les bits de poids forts, alors que l'adresse de machine est calculée sur les bits de poids faible.

Il existe plusieurs classes d'adresses. On parle des classes A, B, C, D et E. Elles sont différenciées par les bits de poids forts qui les compose.

A	0000	Identifiant du réseau	Identifiant de la machine
В	1000	Identifiant du réseau	Identifiant de la machine
C	1100	Identifiant du réseau	Identifiant de la machine
D	1110	Identifiant du réseau	Identifiant de la machine
E	1111	Non utilisé	Non utilisé

Figure 14 - classes d'adressage IP

Une adresse IP est toujours de la forme a.b.c.d. Dans le cas d'une classe A, on peut librement fixer les valeurs b, c et d. On pourra donc adresser théoriquement un maximum de 16 777 216 (2^{3x8}=2²⁴) machines.

Une classe B laisse libre les valeurs de c et d. On pourra alors adresser 65 536 $(2^{2x8}=2^{16})$ machines.

Une classe C laisse uniquement libre la valeur de d. On pourra donc adresser $256\ (2^8)$ machines.

La classe D est une classe quelque peu différente, puisqu'elle est réservée à une utilisation particulière : le multicasting (diffusion temps réel vers plusieurs destinations).

La classe E est quant à elle une classe non usitée à ce jour, et réservées pour des utilisation à des fins d'expérimentation.

On dispose donc en théorie des plages d'adresses suivantes :

Classe	Pla	ge
A	0.0.0.0	127.255.255.255
В	128.0.0.0	191.255.255.255
С	192.0.0.0	223.255.255.255
D	224.0.0.0	239.255.255.255
Е	240.0.0.0	247.255.255.255

Figure 15 - représentation des réseaux IP de classes A, B, C, D et E

Il existe quelques adresses dites non routables. Ces adresses sont réservées à un usage interne, ou dans le cas de réseaux privés. Elles ne sont en théorie jamais routées sur l'Internet¹². Il existe 3 classes d'adresses IP:

Classe A: 10.0.0.0

Classe B: 172.16.0.0 à 172.31.0.0 Classe C: 192.168.0.0 à 192.168.255.0

127.0.0.0 est aussi une classe A particulière, puisqu'elle ne sera jamais routée sur un réseau. Elle est réservée pour un usage interne. On l'appelle aussi interface **loopback** (*interface de bouclage*).

Adressage IP

Dans un réseau TCP/IP, chaque machine est configurée avec une adresse IP et un masque de sous-réseau adapté. Une configuration TCP/IP comprend aussi une adresse de passerelle le cas échéant (réseaux ouverts ou interconnectés).

L'adresse IP est codée sur 2³² bits, soit 4 octets. **Chaque adresse IP est unique** dans le réseau Internet, ce qui est une première condition nécessaire pour le bon adressage des machines connectées au réseau (l'adressage par bloc est ainsi réalisable). Le réseau est en fait composé d'une multitude de sous-réseaux (blocs d'adresses IP contiguës), chaque sous-réseau possédant un ou plusieurs équipements de routage et d'interconnexion IP (généralement, des routeurs).

Remarque : Les organismes de normalisation ont défini l'écriture d'une adresse IP sous trois formes :

192.168.1.16 (base 10), 0300.0250.01.020 (base 8, préfixe '0'), 0xc0a80110 (base 16, préfixe '0x').

Informations de configuration TCP/IP.

Pour configurer une machine destinée à un réseau IP, on fournit plusieurs informations :

Son adresse IP, son masque de sous-réseau et son adresse de passerelle. Ces informations vont servir à correctement orienter les dialogues avec les autres machines.

Certaines adresses IP sont réservées à un usage particulier :

- La première adresse d'un sous-réseau (ayant donc tous les bits à 0, sauf ceux identifiant le sous-réseau) est appelée **adresse de réseau**, et est **réservée**. Elle ne pourra en aucun cas être utilisée.
- La dernière adresse d'un sous-réseau (ayant donc tous les bits à 1, sauf ceux identifiant le sous-réseau) est appelée **adresse de broadcast**, ou adresse de diffusion. Elle ne pourra en aucun cas être utilisée pour une station.

¹² Voir RFC 1918.

```
cisco#show interfaces Ethernet 1/0
Ethernet1/0 is up, line protocol is up
Hardware is AmdP2, address is 0010.7be1.3a10 (bia 0010.7be1.3a10)
Description: Interface du réseau TEST_000
Internet address is 10.0.0.190/26
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 3/255
Encapsulation ARPA, loopback not set, keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 16 drops; input queue 0/75, 2174 drops
5 minute input rate 198000 bits/sec, 65 packets/sec
901818032 packets input, 2032766659 bytes, 48 no buffer
Received 3619945 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
917472472 packets output, 2690304656 bytes, 0 underruns
139 output errors, 27445026 collisions, 1 interface resets
0 babbles, 0 late collision, 15827800 deferred
139 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out

Figure 16 - Informations sur l'interface Ethernet d'un routeur CISCO
```

Calcul du masque de sous-réseau (*netmask*)

Imaginons un sous-réseau de classe C d'adresse 192.168.16.x (0 < x < 255). La première chose à faire est de déterminer le masque de sous-réseau (netmask). Le netmask permet de définir le réseau dans lequel vous vous trouvez. Dans le cas du réseau 192.168.16.x, nous avons donc un réseau de 254 (256 - 2) machines. On va donc chercher le masque de sous-réseau d'un réseau IP dont le dernier octet varie entre 0 et 255, soit une variation des 8 derniers bits de poids faibles.

```
On écrit donc n = 0x000000FF = 255 (les bits variables sont mis à 1), Netmask = NON (n) Donc, ici, le netmask est de NON(0.0.0.255), soit (255.255.255.0).
```

Le netmask est donc utile pour connaître le nombre de machines présentent dans le même sous-réseau que le nôtre. Cette option est intéressante, car elle va permettre de diviser une classe d'adresses (C ou autre) en plusieurs sous-réseaux, par exemple :

Adresse de réseau	Netmask	Nombre d'IP disponibles
192.168.16.0	255.255.255.128	126
192.168.16.128	255.255.255.192	62
192.168.16.192	255.255.255.224	30
192.168.16.224	255.255.255.224	30

Figure 17 - découpage d'une classe C en 4 sous-réseaux IP

Calcul de l'adresse de diffusion (broadcast)

L'adresse de diffusion, ou adresse de broadcast, est calculée à partir du netmask et de l'adresse du sous-réseau. Si ${\bf R}$ est l'adresse du sous-réseau, et ${\bf N}$ le netmask associé, on connaît donc B, l'adresse de broadcast par la formule suivante :

```
B = (R)OU(NON(N))
```

```
Exemple pour le réseau 192.168.16.0/255.255.255.128 : Netmask N=255.255.255.128 Sous-réseau R=192.168.16.0
```

```
B = 192.168.16.0 OU (NON (255.255.255.128))
```

= 192.168.16.0 OU 0.0.0.127 B = 192.168.16.127

L'adresse de broadcast permet d'adresser toutes les machines du même réseau (logique ou IP) que le vôtre d'une seule opération. Imaginons que vous souhaitiez envoyer la commande **ping** à toutes les machines appartenant au même réseau que vous, vous pourrez alors entrer: **ping** 192.168.16.127 et toutes les machines de ce réseau vous répondront (en théorie toutefois).

Il est maintenant clair qu'une adresse de sous-réseau ou de broadcast ne peut pas être attribuée à une machine, du fait de son usage quelque peu particulier.

Station particulière : la passerelle (gateway)

L'adresse de passerelle indique si nécessaire à quelle machine doit-on s'adresser lorsqu'une requête n'est pas destinée à une machine de notre réseau. La passerelle est chargée de correctement transmettre les paquets de notre réseau aux autres passerelles des autres réseaux, mais elle doit aussi nous transmettre les paquets des autres réseaux à destination de notre réseau.

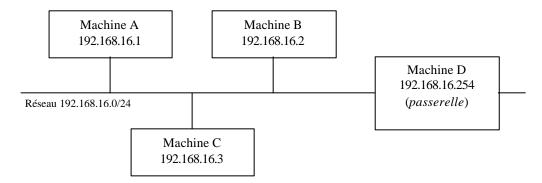


Figure 18 - réseau local interconnecté au réseau mondial

Dans la Figure 18 - réseau local interconnecté au réseau mondial, on a trois machines A, B et C, respectivement d'adresses IP 192.168.16.1, 192.168.16.2 et 192.168.16.3. La quatrième machine D est généralement un routeur, d'adresse IP 192.168.16.254. Chacune de ces machines appartient à la classe C 192.16.168.0, qui permet donc à l'administrateur de ce site d'utiliser selon sa volonté les adresses de 0 à 255. Le netmask de ce réseau est donc 255.255.255.0.

Un routeur est un ordinateur ne disposant pas d'écran ni de clavier, mais de deux interfaces réseau. Par exemple, on peut envisager le routeur comme un PC muni du logiciel adéquat et de deux cartes réseau (eth0 et eth1), configurées chacune sur des sous-réseaux (physiques et/ou logiques) différents.

Lorsqu'une requête est émise, si l'adresse de destination ne concerne pas le réseau IP dans lequel l'émetteur se trouve, alors le paquet sera dirigé vers le routeur pour être routé sur un autre tronçon (physique) du réseau. Si par contre la machine de destination est dans le même réseau que la machine émettrice, la passerelle n'est pas utilisée, car elle serait inutile

Pour déterminer si la machine émettrice se trouve dans le même réseau que la machine qu'elle souhaite contacter, l'opération suivante est effectuée : $V = IP_{dest}$ ET Netmask. Si le résultat V de cette opération donne la même adresse de sous-réseau que la machine

émettrice, alors la machine de destination se trouve dans le même réseau que la machine émettrice, et la passerelle n'est pas sollicitée. Elle l'est dans tous les cas contraires.

Remarque: Certains réseaux non connectés (réseaux locaux) ne disposent pas de passerelle. Lors de la configuration, on omet alors l'adresse de passerelle.

Routage IP

Le routage est primordial pour l'interconnexion des réseaux. Le réseau Internet est en fait composé d'une multitude de petits réseaux interconnectés entre eux. Chaque réseau envoie et reçoit ses informations par le biais de passerelles.

Chaque réseau connecté comprend au minimum une passerelle. Chaque passerelle est obligatoirement connectée à une autre passerelle, appartenant à un autre réseau. Les passerelles sont généralement des routeurs, appareils dédiés au routage de paquets.

Un routeur est donc nécessaire pour relier deux réseaux entre eux, car il n'est pas concevable de relier tous les réseaux par liaison Ethernet (ou tout autre technologie adaptée aux réseaux locaux). En effet, si on prend le cas d'un établissement universitaire, il est totalement inconcevable de le relier à un autre site distant de plusieurs kilomètres par une liaison Ethernet. Les limites de transmission physique par liaison Ethernet seraient largement dépassées (distance, données et contraintes techniques, électriques et électroniques), mais aussi à cause du coût que cela engendrerait.

On fait donc appel aux liaisons louées, qui sont fournies par les opérateurs télécoms (TELCO). Chaque ligne dispose donc de son propre protocole et de son propre débit (exemple : E1 = 2Mb/s, E3 = 34Mb/s, etc.).

Les routeurs ne décodent pas les trames au-delà de la couche 3 du modèle OSI¹³. Par contre, comme les routeurs retranscrivent les trames d'un protocole dans un autre, il faut que le logiciel intégré dans le routeur soit capable de router ce protocole.

Si on utilise un protocole non routable (ARP par exemple, ou encore NetBIOS), le routeur ne fait que transporter le paquet d'un point à un autre, et on parle alors de pontage (ou de proxy). Ainsi, un pont ne fait que bêtement transcrire des trames entre deux réseaux reliés par une interface autre que celle du réseau local, alors qu'un routeur sera capable d'orienter les paquets selon leur destination. Un pont n'est pas capable de comprendre un protocole au-delà du niveau 2 du modèle OSI¹⁴.

¹³ En théorie, car de nombreux constructeurs proposent des fonctions de filtrage ou de redirection qui demandent un décodage jusqu'au niveau 4 ou 5 de la pile IP.

¹⁴ En théorie, car comme pour les routeurs, certains constructeurs proposent des fonctions de filtrage demandant un décodage plus haut dans les couches.

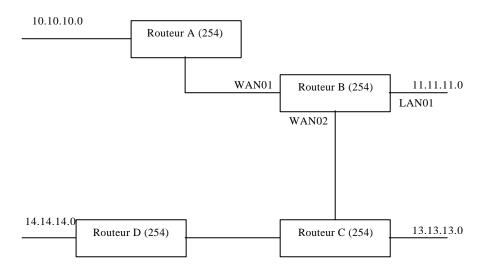


Figure 19 - réseaux locaux interconnectés par routeurs

Dans l'exemple ci-dessus, on a 4 réseaux (10.10.10.x, 11.11.11.x, 14.14.14.x et 13.13.13.x). Chaque réseau dispose d'un routeur (respectivement A, B, C et D) qui leur permet de dialoguer entre réseaux IP distants. On a donc configuré les routeurs de manière à permettre les échanges entre chaque réseau.

On a donc ce que l'on appelle des tables de routage pour chacun des routeurs.

Exercice : Complétez la table de routage suivante :

	Table de routage de la station E	}
destination	netmask (en bits)	interface
	24	WAN01
	24	WAN02
	24	LAN01
0.0.0.0	0	

Chaque paquet à destination du réseau ____._____/___ sera donc orienté vers l'interface WAN01, et ainsi de suite pour tous les paquets.

Par défaut, la destination $0.0.0.0/0^{15}$ est utilisée, et donc tous les paquets ne trouvant pas leur destination dans une table de routage seront orientés vers l'interface _____.

Remarque: un routeur ne route pas les messages de broadcast, et «isole » ainsi un réseau local des autres réseaux. De la même manière, il ne diffuse pas les requêtes ARP, mais les émet lui-même à d'autres routeurs auxquels il peut être relié.

¹⁵ Cette représentation symbolise la route « par défaut ».

Figure 20 -Exemple de table de routage d'un routeur CISCO

cisco#show i	p route summary			
Route Source	e Networks	Subnets	Overhead	Memory (bytes)
connected	0	8	448	1564
static	6	12	1064	3492
rip	0	4	224	752
bgp 9036	77755	14736	5179496	18501448
External:	92491 Internal:	0 Local: 0		
internal	1036			149184
Total	78797	14760	5181232	18656440

Figure 21 - exemple des informations sur le routage contenu sur un routeur CISCO

```
#show ip bgp summary
BGP table version is 35309724, main routing table version 35309724
92488 network entries (264118/277464 paths) using 19920816 bytes of memory
79576 BGP path attribute entries using 11151228 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
24416 BGP filter-list cache entries using 390656 bytes of memory

Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
1.1.1.1 4 1000 5172836 182306 35309720 0 0 4w2d 86134
2.2.2.2 4 2000 3527854 84029 35309720 0 0 2w5d 91187
3.3.3.3 4 3000 18329674 182130 35309724 0 0 1d02h 86796
```

Figure 22 - informations BGP contenues en RAM sur un CISCO multi-homing BGP

Standardisation du protocole TCP/IP

L'Internet Society (ISOC) est une société américaine privée, dont le but est de promouvoir l'usage et de privilégier la croissance d'Internet pour la recherche en communications.

L'Internet Architecture Board (IAB) est composée de volontaires internationaux. Cette structure (dépendante de l'ISOC) supervise et coordonne le développement d'Internet.

L'Internet Engineering Task Force (IETF) veille à la standardisation des protocoles, en définissant les spécifications qui deviennent les standards Internet.

L'Internet Research Task Force (IRTF) s'occupe des recherches et projets à long termes.

Tous les standards en usage sur Internet font l'objet d'une normalisation, et donc d'un document nommé **RFC** (*Request For Comments*). Les RFC sont disponibles sur de nombreux sites FTP publics, comme celui de l'Internic.

De nombreux ouvrages traitent de TCP/IP. La source d'informations la plus complète se trouve dans les ouvrages suivants :

TCP/IP illustré, Volume 1, *Les protocoles*. W. Richard Stevens, Vuibert. ISBN 2-7117-8639-0

TCP/IP illustré, Volume 2, *La mise en oeuvre*. Gary R. Wright et W. Richard Stevens, Vuibert. ISBN 2-7117-8613-7

Acronymes

ACK Acknowledgment (entête TCP) API Application Program Interface ARP Address Resolution Protocol

ARPANET Advanced Research Projects Agency Network

BGP Border Gateway Protocol **BSD** Berkeley Software Distribution **CRC** Cyclic Redundancy check **FDDI** Fiber Distributed Data Interface FIN Finish flag (entête TCP) **FQDN** Fully Qualified Domain Name

Internet Architecture Board IAB **IANA** Inernet Assigned Number Authority **ICMP** Internet Control Message Protocol

Institute of Electrical ans Electronic Engineers IEEE

IESG Internet Engineering Steering Group **IETF** Internet Engineering Task Force

IP Internet Protocol **ISN** Initial Sequence Number

ISO International Organization for Standardisation

ISOC Internet SOCiety LAN Local Area Network **MAN** Metropolitan Area Network **MBONE** Multicast backBONE **MILNET** Military Network **MSS** Maximum Segment Size Maximum Transmission Unit **MTU** NIC Network Interface Card

NOP No OPeration

OSI Open Systems Interconnection **OSPF** Open Shortest Path First **PDU** Protocol Data Unit

POSIX Portable Operating System Interface

PPP Point-to-Point Protocol **PSH** Push flags (entête TCP)

RARP Reverse Address Resolution Protocol

RFC Request for comment

RIP **Routing Information Protocol RST** Reset flag (entête TCP)

RTT Round-Trip time

SACK Selective Acknowledgment **SLIP** Serial Line Internet Protocol SYN SYNchronize (entête TCP) Transmission Control Protocol **TCP**

TOS Type Of Service TTL Time To Live

UDP User Datagram Protocol URG URGent flag (entête TCP) WAN Wide Area Network

Travaux Pratiques I

- I.1 Calculez le nombre maximum d'adresses IP disponibles et utilisables dans les classes A, B et C.
- I.2 Obtenez une copie de la RFC «Assigned Numbers ». Quel port utilise le protocole « quote of the day » ? Quelle RFC définit ce protocole ?
- I.3 Connectez-vous sur une station distante via un réseau IP. Quelle est la configuration IP de cette station ? Est-elle multi-interfaces ? Est-elle d'adressage privée ou public ?
- I.4 En utilisant la commande netstat, déterminez les MTU de vos interfaces. Que représentent ces valeurs ?
- I.5 L'interface loopback doit-elle obligatoirement utiliser la valeur 127.0.0.1 ?
- I.6 Le masque de sous-réseau 255.255.0.255 est-il valide pour une adresse de classe A?
- I.7 Un protocole **RARP** (*Reverse Address Resolution Protocol*) est parfois utilisé. Quel peut-être l'intérêt de ce protocole ? Peut-on utiliser ce protocole sur Internet (expliquer) ?
- I.8 En observant le résultat de la commande ifconfig a ci-dessous, qu'apprenez-vous sur l'interface eth0?

eth0	Link encap: Ethernet HWaddr 00: D0: B7: 2D: 17: 6C
	inet addr: 10. 0. 0. 14 Bcast: 10. 0. 0. 127 Mask: 255. 255. 255. 128
	UP BROADCAST RUNNING MULTICAST MTU: 1500 Metric: 1
	RX packets: 2805431 errors: 0 dropped: 0 overruns: 0 frame: 0
	TX packets: 3113072 errors: 12 dropped: 0 overruns: 0 carrier: 0
	collisions: 0 txqueuelen: 100 Interrupt: 19 Base address: 0x9800
	Interrupt: 19 Base address: 0x9800
lo	Link encap: Local Loopback inet addr: 127. 0. 0. 1 Mask: 255. 0. 0. 0
	UP LOOPBACK RUNNING MTU: 8020 Metric: 1
	RX packets: 395592 errors: 0 dropped: 0 overruns: 0 frame: 0 TX packets: 395592 errors: 0 dropped: 0 overruns: 0 carrier: 0
	TX packets: 395592 errors: 0 dropped: 0 overruns: 0 carrier: 0
	collisions: 0 txqueuelen: 0

I.9 Un routeur filtrant détecte et rapporte la trame suivante. Complétez à l'aide de cette capture les informations demandées ci-dessous.

cisco-1485 229470: .Aug 20 04:04:15.631: %SEC-6-IPACCESSLOGP: list acl-in denied tcp 194.149.170.139(61217) -> 212.85.128.1(53), 1 packet

IP source	
IP destination	
Port source	
Port destination	
Protocole couche 3	
Protocole couche 4	
Nombre de segments enregistrés	
Nom du routeur	
Heure d'enregistrement de l'incident	