CHAPITRE 4

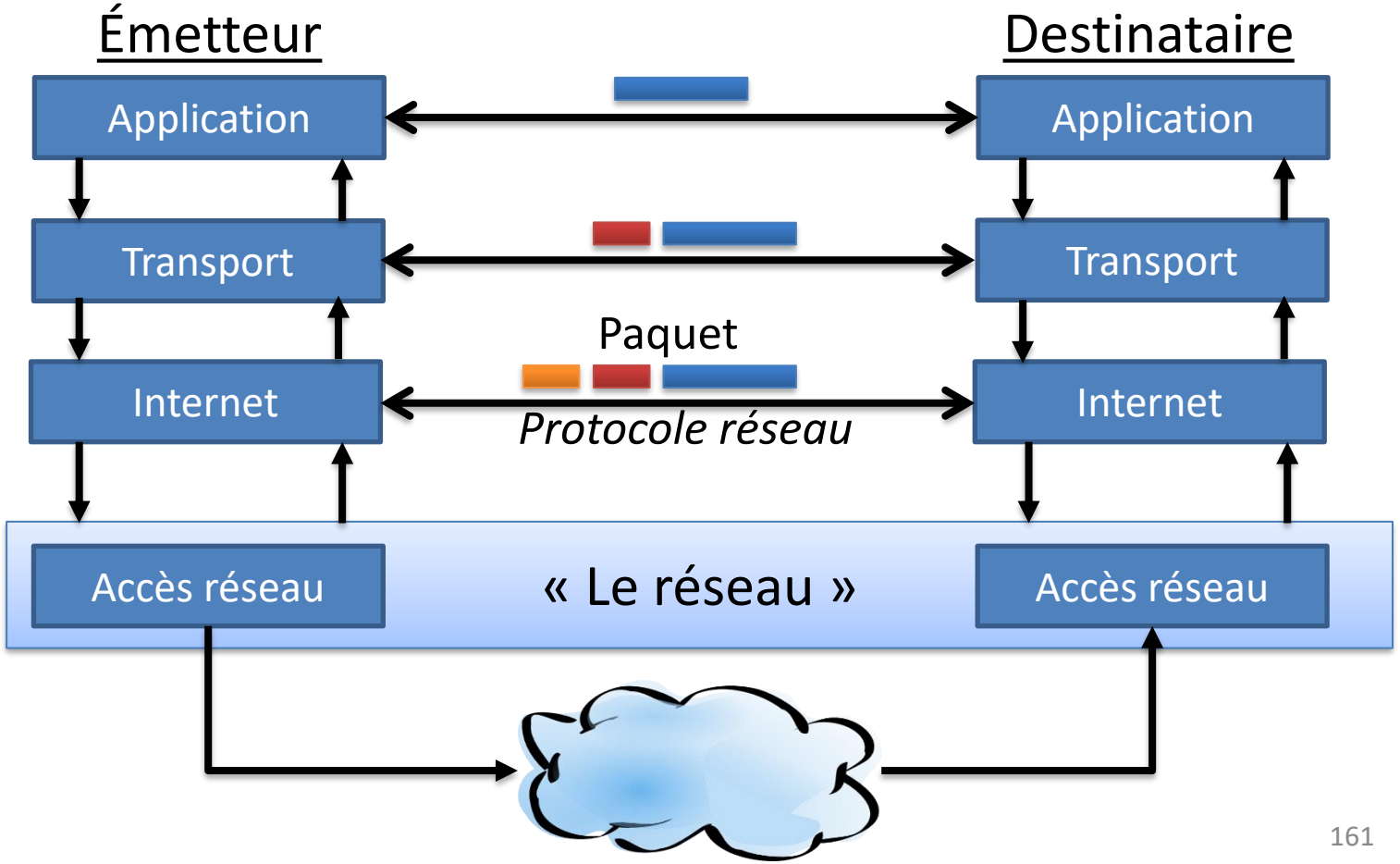
LA COUCHE INTERNET

Introduction (1)

Plan :

- **Intro**
- Routage
- IP

- Découpe en couche



SWILABUS

Introduction (2)

- Service fourni par la couche réseau
  - Le service dépend du type de couche réseau
  - *Acheminement des informations (forwarding)*
    - D'une **machine source** vers une **machine destination**
      - En utilisant la couche liaison de donnée sous-jacente
      - Indépendance entre ces couches
    - Sans connexion et de manière non fiable
      - Les informations peuvent ne pas arriver à destination
  - *Déterminer le chemin*
    - Choix de la route suivie par les paquets
    - Sur base d'une algorithme précis

Plan :
- **Intro**
- Routage
- IP

SWILABUS

Introduction (3)

– *Phase d'ouverture (facultatif)*

- Certaines architectures réseaux (comme le réseau téléphonique commuté ou le X.25) nécessite une phase d'ouverture de connexion

Plan :

- **Intro**
- Routage
- IP

Le protocole IP (en version 4 et 6) est aujourd'hui le protocole réseau de référence. Tous les autres (Novell IPX, Appe AppleTalk, Microsoft Netbui, …) sont aujourd'hui obsolètes. C'est la raison pour laquelle nous concentrerons notre étude sur IPv4 et IPv6.

Introduction (4)

Plan :
- **Intro**
- Routage
- IP

- Un routeur
  - Il est un relai au niveau de la couche réseau. Il est capable d'analyser le contenu d'un paquet
  - Il interconnecte des réseaux parfois de natures différentes (la « box » de votre ISP connecte votre réseau au réseau téléphonique [xDSL] ou de télédistribution [coaxial])
    - Il dispose de plusieurs interfaces réseaux:
      - Connexion au réseau de l'ISP
      - Connexion à votre réseau domestique
      - Couverture Wifi

Introduction (5)

Plan :

- **Intro**
- Routage
- IP

– Il détermine, **grâce à divers protocoles**, le « meilleur » chemin vers une destination.

– Il propage les paquets de la source à la destination

– Il fournit parfois un service différent en fonction du service attendu

  - Télévision
  - Téléphonique

Introduction (6)

Plan :
- **Intro**
- Routage
- IP

- Identification des machines (en IPv4)
  - Les machines sont identifiées par un entier de 32 bits, appelé adresse IPv4
    - Ex: 3 250 471 036 = C1BE407C
  - Cette adresse est souvent présentée sous la forme décimale pointée. On groupe l'adresse par octets (8 bits) que l'on présente sous forme décimale:

C1	BE	40	7C
11000001	10111110	01000000	01111100

Introduction (7)

- Identification d'un (sous-)réseau IPv4
  - Un sous-réseau est identifié par une adresse IPv4 particulière
  - Elle permet de déterminer si deux machines sont **directement connectées**.
    - Cela permet de savoir s'il faut passer par un relai (i.e. un routeur) pour atteindre la destination.
  - Comment identifier l'adresse du sous-réseau?
    - Il faut connaître le masque de (sous-)réseau
      - C'est une adresse IP particulière qui a les bits à 1 qui sont tous consécutifs

Plan :
- **Intro**
- Routage
- IP

SWILABUS

Introduction (8)

Plan :
- **Intro**
- Routage
- IP

  - Ce masque peut être représenter sous la forme d'une adresse IP en forme décimal pointée:
    - » 255.255.255.0 ou 255.255.224.0
  - Ce masque est également noté sous la forme /24 pour indiquer qu'il comporte 24 bits consécutifs à 1
  - Le masque indique combien d'adresses IPv4 différentes appartiennent à un (sous-)réseau :
    - » $2^{32-24} = 2^8 = 256$ ou $2^{32-19} = 2^{13} = 8192$

- Une fois que l'on connait le masque, il faut réaliser un ET logique entre l'IP et le masque:
  - `193.190.77.130` avec un masque `/29`

```
11000001    10111110    01001101    10000010
```

Introduction (9)

- Préfixe IPv4
  - On désigne également un (sous-)réseau par un préfixe IPv4. Il est présenté sous la forme:
    - **adresse IP réseau / masque**
    - Ex: le préfixe `193.190.77.128/29` désigne l'ensemble des machines dont les adresses sont:
      - » `193.190.77.128` (adresse réseau réservée)
      - » `193.190.77.129`
      - » `193.190.77.130`
      - » `193.190.77.131`
      - » `193.190.77.132`
      - » `193.190.77.133`
      - » `193.190.77.134`
      - » `193.190.77.135` (adresse broadcast réservée)

Plan :
- **Intro**
- Routage
- IP

$2^{32-29} = 2^3 = 8$

Introduction (10)

- Identification des machine (IPv6)
  - Une adresse IPv6 est codée sur 128 bits
    - Nombre d'adresses: 3,49282 * 10^{38} !
  - Ces adresses sont souvent représentées en hexadécimales, regroupées par blocs de 16 bits, séparés par « : »
    - Ex: `2001:0bc8:38eb:fe10:0000:0000:0000:0011`
    - La conversion entre la version hexadécimale et binaire est immédiate
    - On peut abréger les « 0 » consécutif en utilisant `::` (1 seule fois)
      - `2001:bc8:38eb:fe10::11`

Plan :
- **Intro**
- Routage
- IP

SWILABUS

Introduction (11)

Plan :
- **Intro**
- Routage
- IP

– Identification d'un (sous-)réseau IPv6
  - Comme IPv4, le sous-réseau est désigné par une adresse IPv6 particulière
  - Ce sous-réseau permet de déterminer si les machines **sont directement connectées.**
  - En IPv6, le masque est noté `/masque` (ex: `/64`, masque composé de 64 bits consécutifs à 1)
  - Comme en IPv4, le masque renseigne le nombre d'adresses IPv6 présentes

Introduction (12)

Plan :
- **Intro**
- Routage
- IP

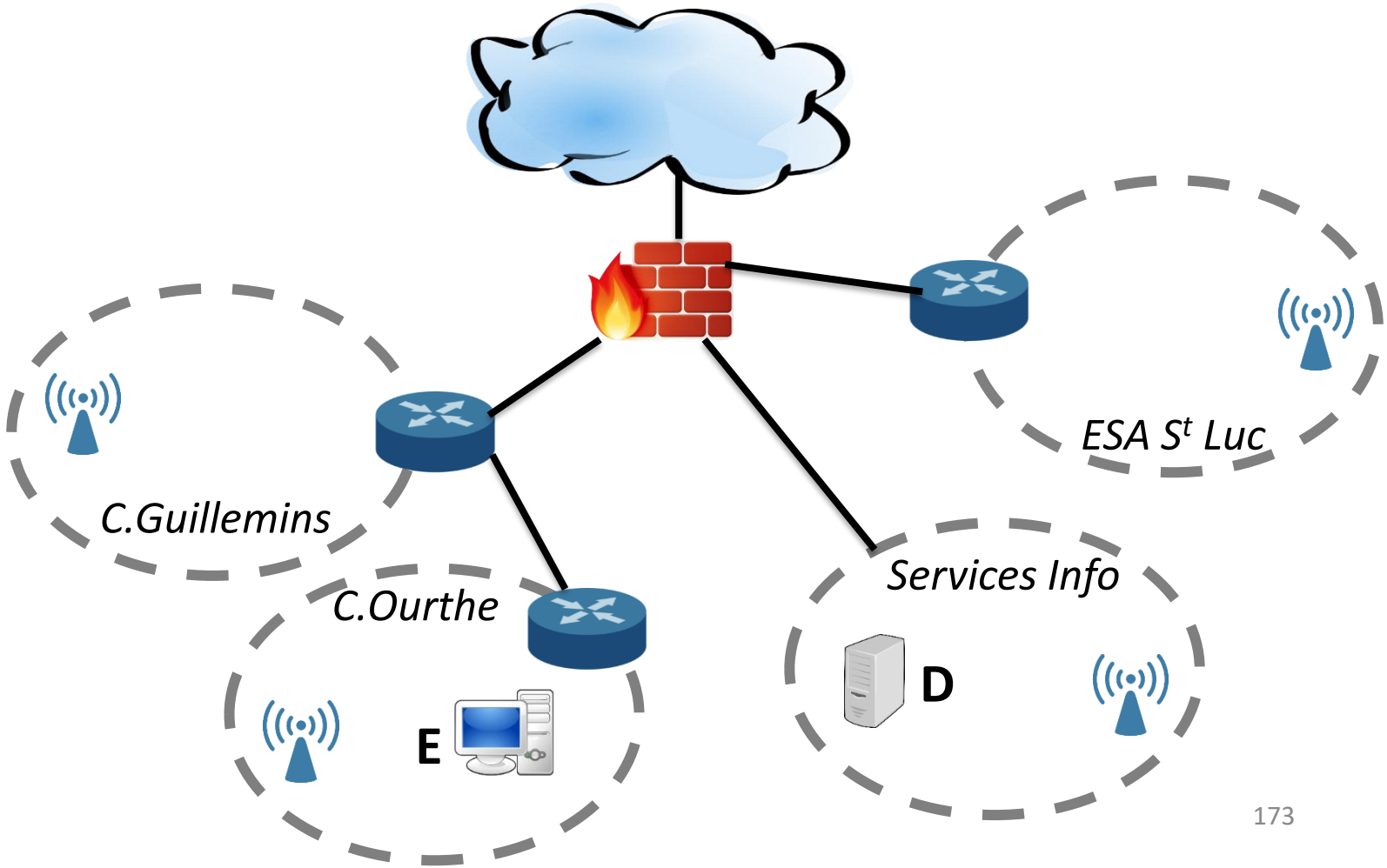
- Préfixe IPv6
  - Le nombre d'adresses IPv6 disponible permet d'avoir une adresse dédiée à la désignation du réseau
    - » Exemple: `2001:bc8:38eb:fe10::/64`
    - » Désigne toutes les adresses IPv6 allant de `2001:bc8:38eb:fe10:0:0:0:0` à `2001:bc8:28eb:fe10:ffff:ffff:ffff:ffff`
    - » Soit 2^{64} adresses IPv6 publiques disponibles, soit $1{,}89*10^{19}$ adresses IPv6 pour le client (moins l'adresse réseau).
    - » L'adresse *broadcast* est différente en IPv6 car elle commence par un préfixe particulier (nous en parlerons plus loin dans notre découverte d'IPv6).

Introduction (13)

- Comment acheminer l'information ?

Plan :

- **Intro**
- Routage
- IP



Introduction (14)

Plan :
- **Intro**
- Routage
- IP

- Deux modèles distincts
  - Mode *circuit virtuel*. 3 phases:
    - Etablissement du circuit
      - Sur base d'informations données par l'émetteur, la couche réseau va déterminer un chemin entre la source et la destination
      - Tous les éléments réseaux (routeurs, …) sont impliqués dans l'établissement du circuit
    - Transfert d'informations
      - Envoi des paquets, qui suivent le chemin établi
    - Fermeture du circuit
      - Une des entités annonce la déconnexion
      - La couche réseau interrompt le circuit

SWILABUS

Introduction (15)

Plan :
- **Intro**
- Routage
- IP

– *Mode datagramme*

- Les paquets mentionnent l'adresse destination
  - Pas d'information à retenir par flux, scalable
- Sur base de cette information, les routeurs orientent les paquets vers la destination
  - Le paquet va passer de routeur en routeur jusqu'à atteindre la destination.
  - Comment les routeurs font-ils cela ?
    - » Sur base de leur table de routage / forwarding
    - » Cette tables peut évolution en fonction des événements observés (pannes, charge, …)
    - » **Attention! 2 paquets successifs peuvent ne pas suivre le même chemin**

Introduction (16)

- Table de routage / forwarding
  - Cette table est présente sur *tous* les équipements connectés au réseau (routeurs, PC, tablettes, …)
  - Elle associe des préfixes IP à des interface réseaux. Elle permet de déterminer comment atteindre (quelle interface) une destination (un préfixe)
    - » Windows: route print
    - » Unix: netstat -r –n ou ip -6 route

Plan :

- **Intro**
- Routage
- IP

```
192.168.126.136 (masque /24)
2a02:578:85de:100:5d37:a7e9:5217:fe6d (masque /64)
```

```
2a02:578:85de:100::/64 dev eth0
fe80::/64 dev eth0
::/0 via 2a02:578:85de:100:420f:a487:19ff:fee9 dev eth0
```

Destination	Gateway	Masque	Iface
192.168.126.0	0.0.0.0	255.255.255.0	eth0
169.254.0.0	0.0.0.0	255.255.0.0	eth0
0.0.0.0	192.168.126.1	0.0.0.0	eth0

SWILABUS

Routage (1)

Plan :
- Intro
- **Routage**
- IP

- Comment constituer la table de routage?
  - Manuellement
    - Routage statique
  - Automatiquement
    - Collaboration entre les équipements connectés au réseau pour établir et distribuer les informations de routage
    - *Routage dynamique*
      - 2 méthodes très différentes:
        - » Sur base d'une carte complète du réseau
        - » Sur base d'information partielle du réseau

Routage (2)

- Routage statique
  - Configuration manuelle effectuée par l'administrateur
    - Méthode de routage très courante
    - Pour des réseaux avec une structure simple
    - L'administrateur configure chaque routeur (souvent peu nombreux) et chaque machine
      - La configuration peut parfois être « téléchargée » à partir d'une autre machine (ex. TFTP, DHCP, …)
    - Méthode ingérable dans des réseaux importants comme ceux des ISPs ou lorsqu'il faut fournir des services réseaux particuliers.

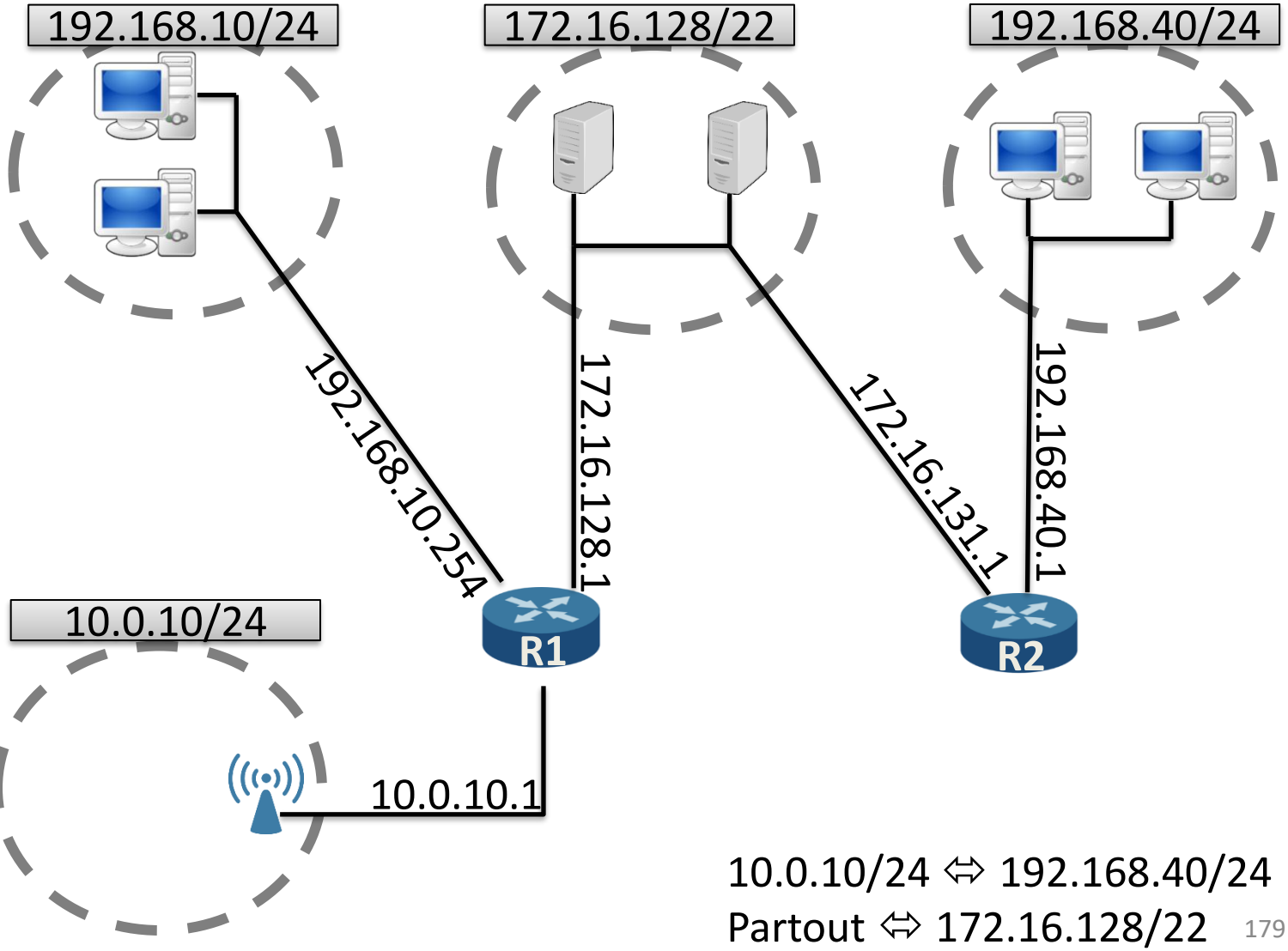
Plan :
- Intro
- **Routage**
- IP

SWILABUS

Routage (3)

Plan :

- Intro
- **Routage**
- IP



10.0.10/24 ⇔ 192.168.40/24

Partout ⇔ 172.16.128/22

SWILABUS

Routage (4)

- Routage dynamique
  - Collaboration entre les routeurs pour établir les routes vers les destinations
    - Les routeurs vont informer leurs voisins des routes qu'ils connaissent et des pannes découvertes.
    - Utilisation d'un algorithme réparti dans tous les routeurs déployés sur le réseau
  - Avantages
    - Adapté aux grands réseaux
  - Inconvénients
    - Complexité plus grande

Plan :
- Intro
- **Routage**
- IP

Routage (5)

Plan :
- Intro
- **Routage**
- IP

– 2 types d'algorithmes
  - Algorithme de routage global
    – Chaque routeur dispose de la carte complète du réseau et peut ainsi détermine les plus courts chemins vers une destination (en tenant compte d'un poids / coût configuré)
    – Cette famille d'algorithmes est appelée **algorithme à états de liaison**
  - Algorithme de routage décentralisé
    – Les routeurs ont une vue partielle du réseau. En fonction de cette vue, ils vont tenter d'acheminer les informations de la source vers la destination.
    – Cette famille d'algorithmes est appelée **algorithme par vecteur de distance**.

Routage (6)

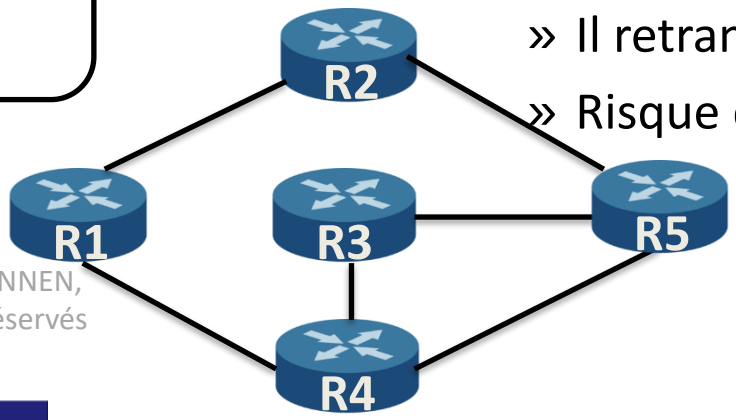
Plan :
- Intro
- **Routage**
- IP

- Algorithme à états de liaison
  - Principe
    - Chaque routeur dispose de la carte complète du réseau et, en fonction de celle-ci, peut déterminer le meilleur chemin pour atteindre la destination.
  - Comment disposer de la carte complète ?
    - Au départ, le routeur connaît les lignes auxquelles il est relié, et les coûts associés
    - Les routeurs publient ces informations aux autres
    - Tous les routeurs disposent d'une carte complète.

Routage (7)

Plan :
- Intro
- **Routage**
- IP

– Echange d'information entre les routeurs
  - Les routes sont échangées dans des LSPs (*Link State Packets*). Ils contiennent:
    – Identification du routeur annonçant ses informations
    – Paires <destination, coût associé>
  - Comment distribuer cette topologie ?
    – Que fait un routeur lorsqu'il reçoit les informations ?
      » Il retransmet l'information aux autres ?
      » Risque de bouclage ?



Routage (8)

Plan :
- Intro
- **Routage**
- IP

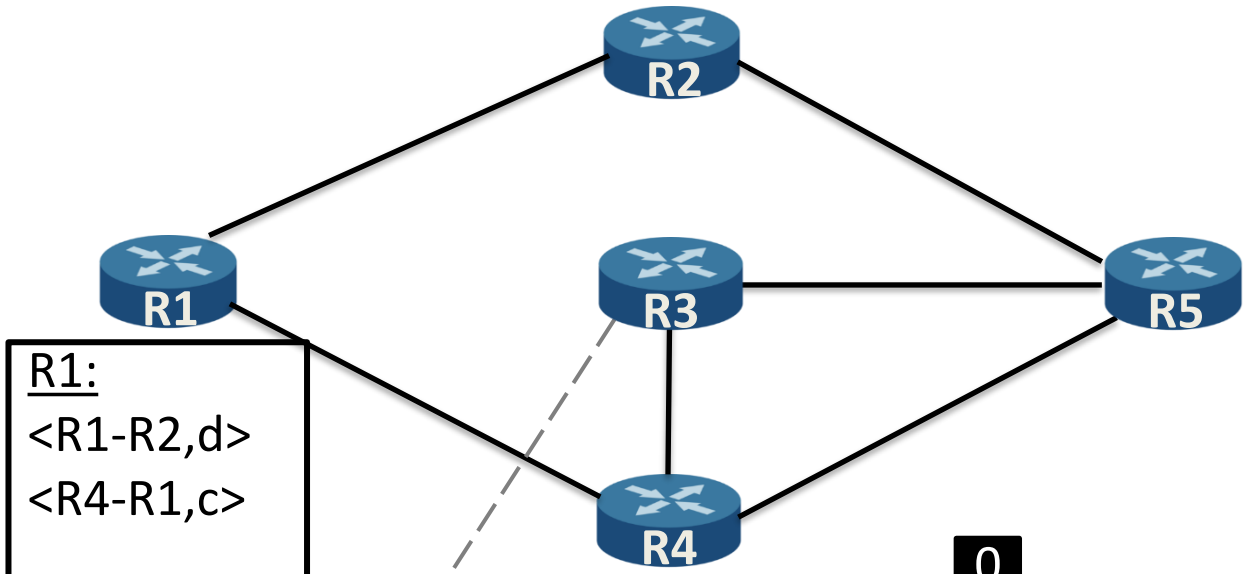
- Idée 1:
  - » Un routeur retransmet un LSP sur toutes les lignes sauf celle sur laquelle il l'a reçu
  - » Ne permet pas d'éviter les boucles
- Idée 2:
  - » Mémoriser, pour chaque routeur, le dernier LSP reçu:
    - Il est possible de comparer les LSPs et ne retransmette que les nouveaux LSPs
  - » On utilise un numéro de séquence pour déterminer la version de l'information
    - Seule la source incrémente ce numéro
    - Si un routeur reçoit plusieurs LSPs avec le même numéro ➔ pas retransmis
      - Information déjà connue.

Routage (9)

- Exemple (R3 démarre)

Plan :

- Intro
- **Routage**
- IP



R2:
<R2-R1,d>
<R2-R5,e>

2

R5:
<R5-R2,a>
<R5-R4,b>
<R5-R3,x>

1

R1:
<R1-R2,d>
<R4-R1,c>

2

R3:
<R3-R4,y>
<R3-R5,x>

0

R4:
<R4-R1,c>
<R4-R5,b>
<R4-R3,y>

1

0

1

2

SWILABUS

Routage (10)

Plan :
- Intro
- **Routage**
- IP

– Comment calculer les chemins les + courts?
  - A partir de la carte complète du réseau
    – En utilisant l'algorithme de E.W. Dijkstra (1930-2002)
      » Détermine le plus court chemin à partir d'un noeud vers tous les autres nœuds du réseau
      » Algorithme itératif: a chaque tour, une branche de l'arbre des plus courts chemins est ajoutée. L'arbre relie la source à toutes les destinations.
    – Fonctionnement de l'algorithme
      » *c(i,j):* coût de la ligne joignant le nœud *i* au nœud *j*. Par facilité, on supposera que c(i,j) = c(j,i). Si aucune ligne ne joint les nœuds i et j, c(i,j)=∞
      » *D(v):* coût du chemin à partir de la source jusqu'au nœud destination v

Routage (11)

- *p(v) :* identification du nœud précédent le nœud v et voisin de ce dernier
- *N:* ensemble des nœuds à considérer

Plan :
- Intro
- **Routage**
- IP

Calcul des routes pour le nœud source A

Pseudocode: © Pearson Education, 2013 [1]

```
1°  initialisation :
      N = {A}
      for all nodes v
        if v adjacent to A
        then D(v) = c(A,v)
        else D(v) = INF
2°  loop
      find w not in N such that D(w) is a minimum
      add w to N
      update D(v) for all v adjacent to w and not in N
        D(v) = min [ D(v), D(w) + c(w,v) ]
        /* new cost to v is either old cost to v or known
           shortest path cost to w plus cost from w to v */
    until all nodes in N
```

Routage (12)

- Exemple (calcul pour R1)

Plan :
- Intro
- **Routage**
- IP

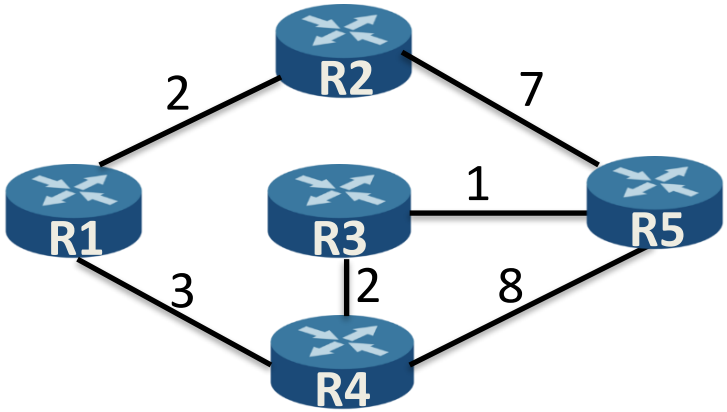
Initialisation:
N={R1}
D(R2) = 2
D(R3) = ∞
D(R4) = 3
D(R5) = ∞

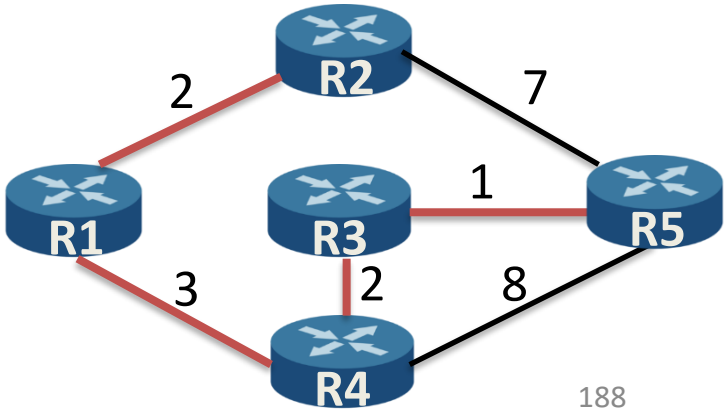
Loop 1:
N={R1,R2}
D(R3) = ∞
D(R4) = 3
D(R5) = 9

Loop 2:
N={R1,R2,R4}
D(R3) = 5
D(R5) = 9

Loop 3:
N={R1,R2,R4,R3}
D(R5) = 6

Loop 4:
N={R1,R2,R4,R3,R5}





Faire l'exercice pour R5 !

SWILABUS

Routage (13)

- Algorithme par vecteur de distance
  - Algorithme réparti dans les routeurs
  - Les routeurs ont une vue partielle du réseau et orientent les paquets suivant leur connaissance
  - Algorithme itératif qui converge, à chaque étape, vers les chemins de coûts minimaux
  - Pourquoi un tel algorithme peut-il être intéressant ?
    -

Plan :
- Intro
- **Routage**
- IP

Routage (14)

Plan :
- Intro
- **Routage**
- IP

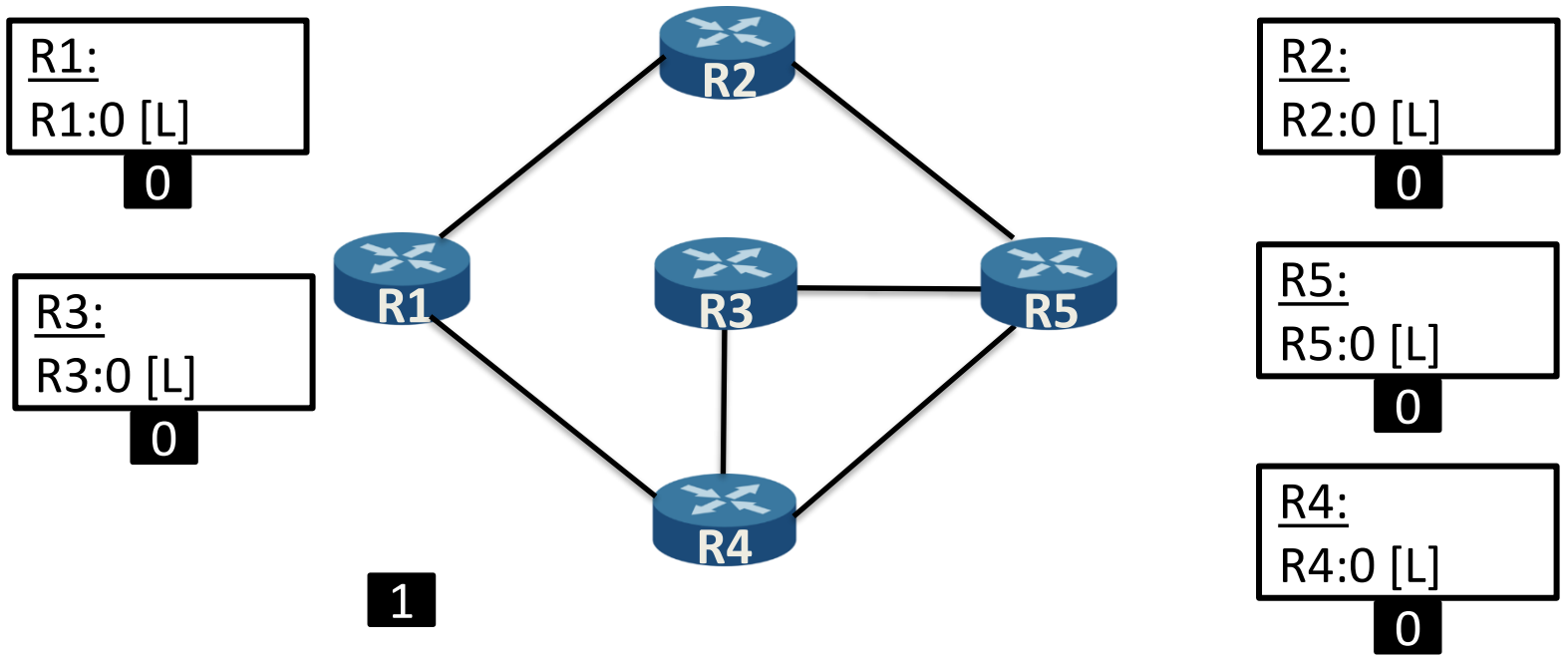
– Principes
  - Chaque routeur envoie une information appelée *vecteur* qui contient:
    – L'identification de la destination
    – Coût / distance pour atteindre celle-ci depuis ce routeur
  - Ce vecteur est envoyé régulièrement aux voisins
  - Tous les routeurs procèdent de cette manière et construisent une vue partielle du réseau, suffisante pour acheminer l'information
  - Lorsqu'un vecteur est reçu, la table de routage / forwarding intègre les informations

Routage (15)

- Exemple (en commençant par R4)

Plan :

- Intro
- **Routage**
- IP



0 Au commencement, le routeur ne connait pas ses voisins

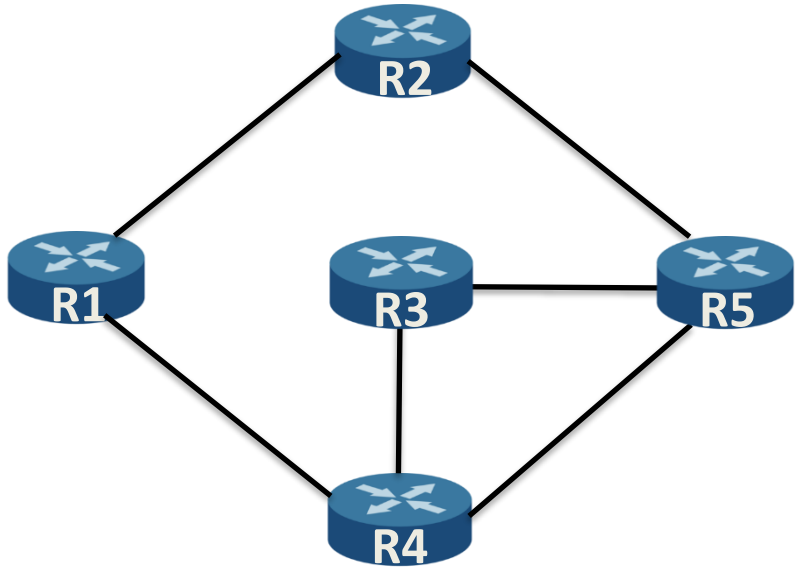
1

SWILABUS

Routage (16)

Plan :

- Intro
- **Routage**
- IP



2 Les routeurs R1, R3 et R5 intègrent le vecteur reçu dans leur table

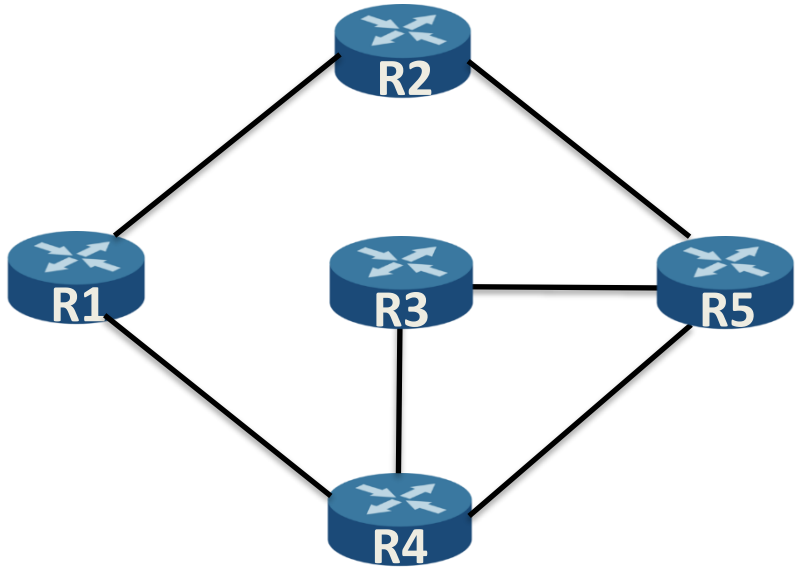
3

SWILABUS

Routage (17)

Plan :

- Intro
- **Routage**
- IP



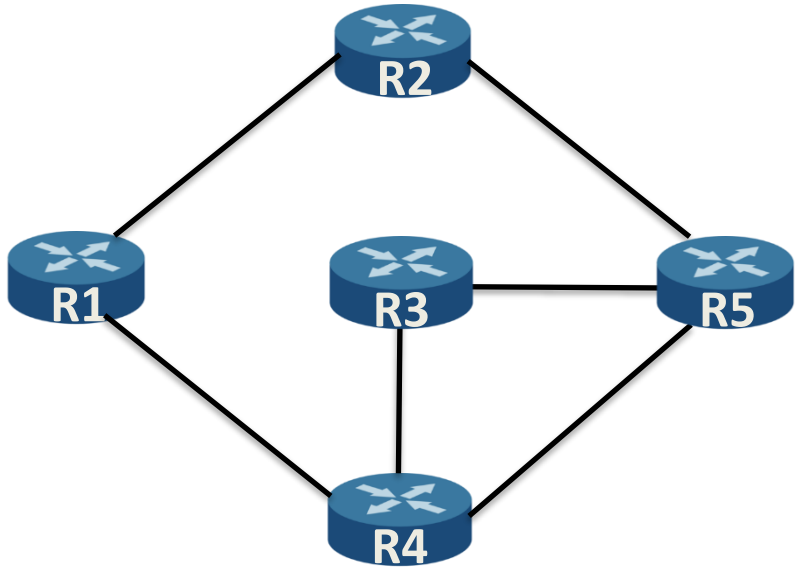
4 Les routeurs R1 et R5 intègrent le vecteur reçu dans leur table
5

SWILABUS

Routage (18)

Plan :

- Intro
- **Routage**
- IP



6 Les routeurs R2 et R4 intègrent le vecteur reçu dans leur table

7

SWILABUS

Routage (19)

Plan :

- Intro
- **Routage**
- IP

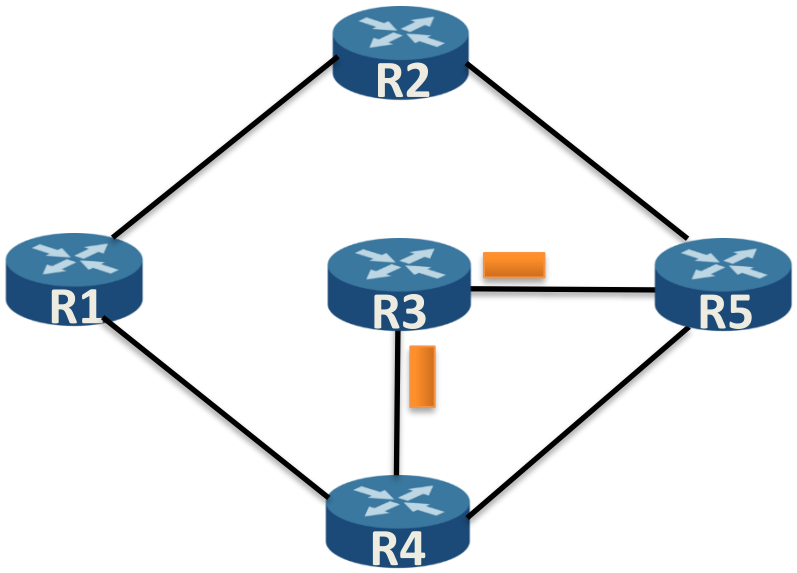
R1:
R1:0 [L]

8

R3:
R3:0 [L]

8

9



R2:
R2:0 [L]

8

R5:
R5:0 [L]

8

R4:
R4:0 [L]

8

8 Les routeurs R2, R3 et R4 intègrent le vecteur reçu

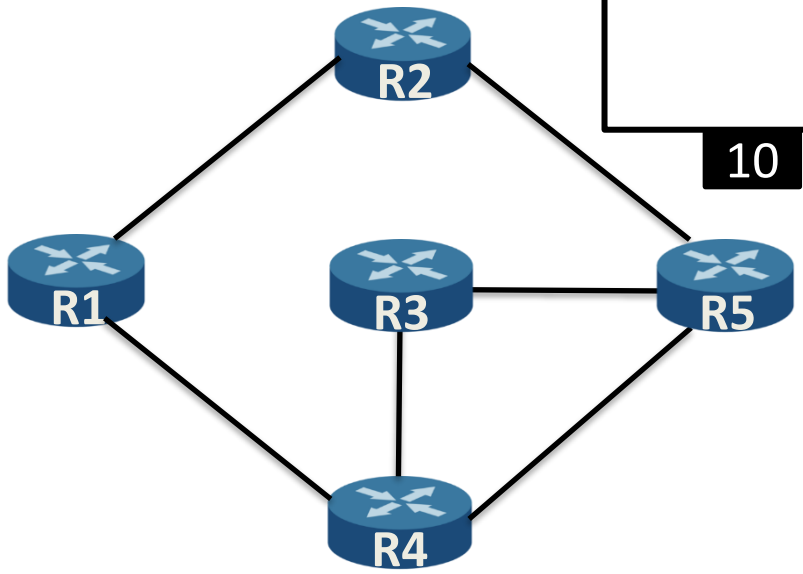
9

SWILABUS

Routage (20)

Plan :

- Intro
- **Routage**
- IP



10 Les routeurs R4 et R5 intègrent le vecteur reçu

SWILABUS

Routage (21)

Plan :
- Intro
- **Routage**
- IP

- Gestion des pannes
  - Des messages sont échangés régulièrement entre les routeurs. Une panne de la ligne peut être détectée si aucun message n'est reçu
  - Que faire en cas de panne ?
    - Réaction simple: le routeur considère la ligne en panne comme ayant un coût à l'infini
    - Un nouveau vecteur est envoyé pour avertir les voisins
    - Les autres routeurs vont tenter de trouver un autre chemin pour arriver à la destination et contourner la panne détectée.

Routage (22)

Plan :

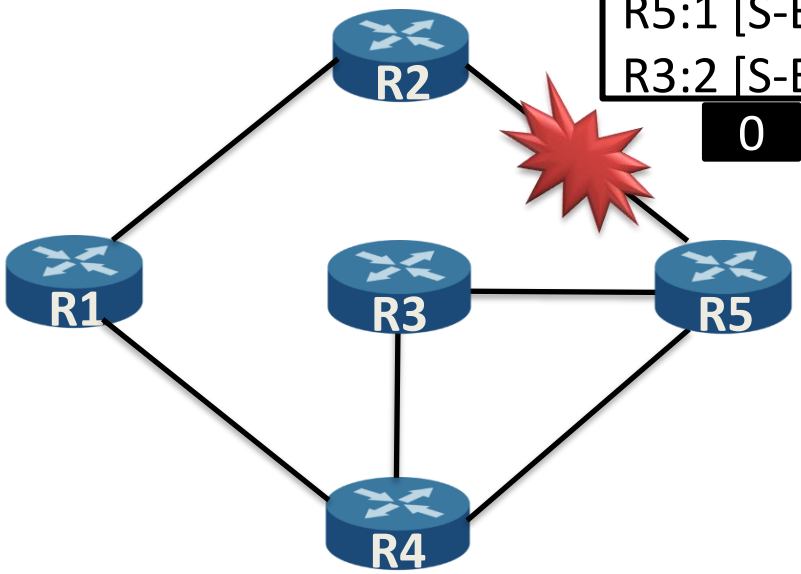
- Intro
- **Routage**
- IP

R1:
R1:0 [L]
R4:1 [S-E]
R2:1 [N-E]
R3:2 [S-E]
R5:2 [S-E]
0

R2:
R2:0 [L]
R1:1[O]
R4:2[O]
R5:1 [S-E]
R3:2 [S-E]
0

R5:
R5:0 [L]
R4:1 [S-O]
R2:1 [N-O]
R3:1 [O]
R1:2 [S-O]
0

R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:2 [E]
R1:2 [S]
0



R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]
0

0 Etat de départ: les routes sont stabilisées

1 Panne entre R2 et R5: ligne accessible

SWILABUS

Routage (23)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]
R4:1 [S-E]
R2:1 [N-E]
R3:2 [S-E]
R5:2 [S-E]

0

3

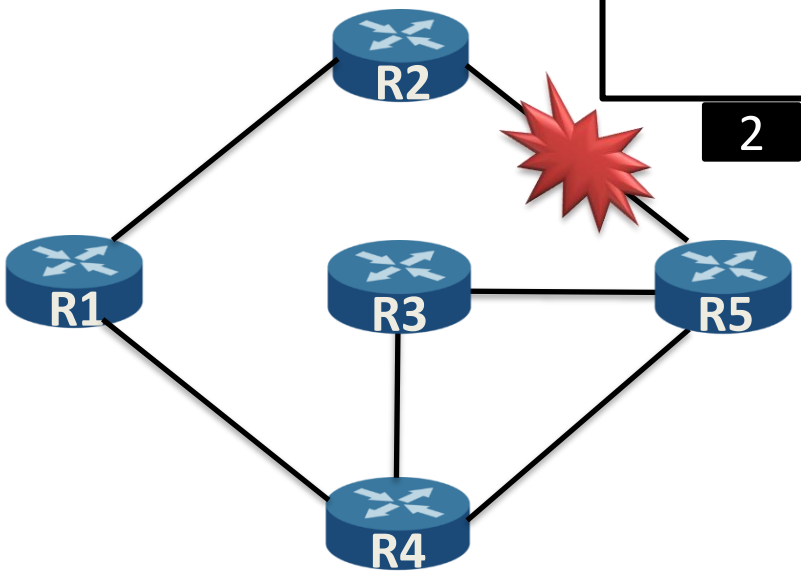
R2:
R2:0 [L]
R1:1[O]
R4:2[O]

2

R5:
R5:0 [L]
R4:1 [S-O]

R3:1 [O]
R1:2 [S-O]

2



R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:2 [E]
R1:2 [S]

0

3

R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

0

2 R2 et R5 détectent la panne

3

SWILABUS

Routage (24)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]
R4:1 [S-E]
R2:1 [N-E]
R3:2 [S-E]
R5:2 [S-E]

4

R2:
R2:0 [L]
R1:1[O]
R4:2[O]

4

R5:
R5:0 [L]
R4:1 [S-O]

R3:1 [O]
R1:2 [S-O]

4

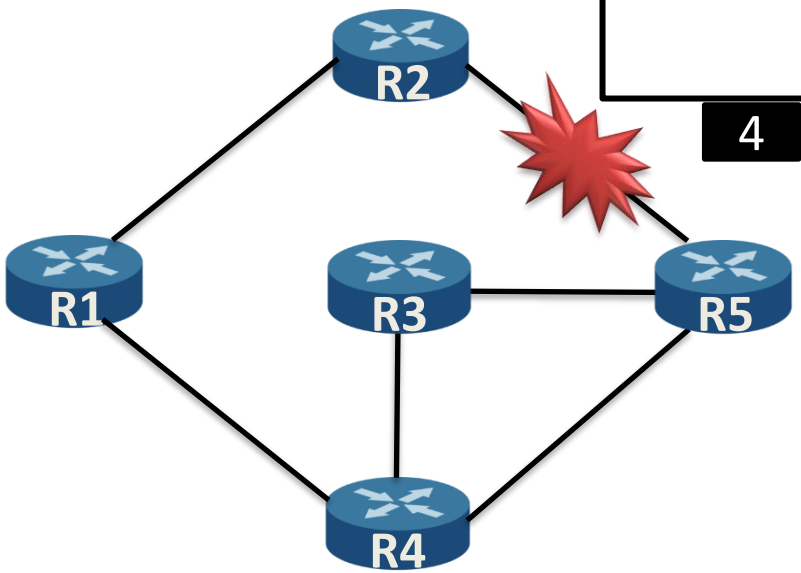
R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]

R1:2 [S]

4

R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

4



5

5

4 R1, R3 et R4 mettent à jour leur table

5

SWILABUS

Routage (25)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]
R4:1 [S-E]
R2:1 [N-E]
R3:2 [S-E]
R5:2 [S-E]

6

R2:
R2:0 [L]
R1:1[O]
R4:2[O]

6

R5:
R5:0 [L]
R4:1 [S-O]

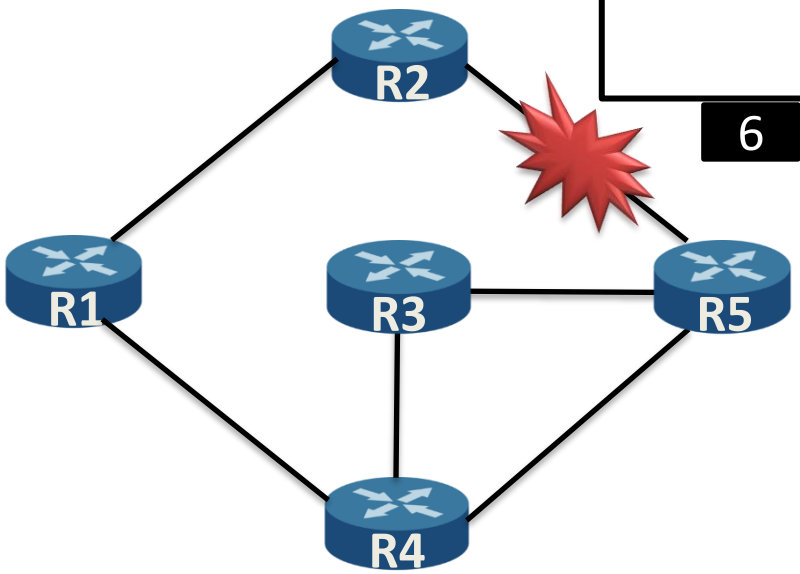
R3:1 [O]
R1:2 [S-O]

6

R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]

R1:2 [S]

6



R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

6

6 R2, R3 et R5 mettent à jour leur table

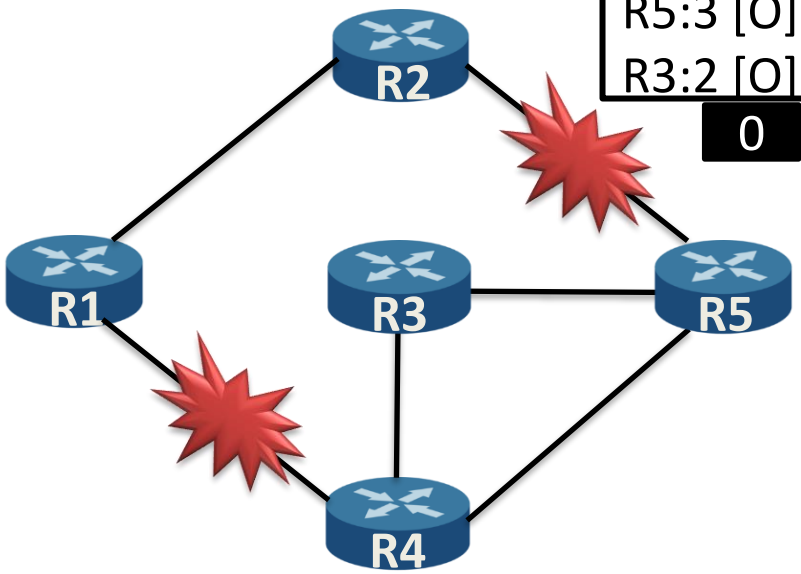
Routage (26)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]
R4:1 [S-E]
R2:1 [N-E]
R3:2 [S-E]
R5:2 [S-E]
0

- Problème en cas de plusieurs pannes



R2:
R2:0 [L]
R1:1[O]
R4:2[O]
R5:3 [O]
R3:2 [O]
0

R5:
R5:0 [L]
R4:1 [S-O]
R2:3 [S-O]
R3:1 [O]
R1:2 [S-O]
0

R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:3 [S]
R1:2 [S]
0

R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]
0

0 Cas de départ

1 Panne de la ligne R1-R4

SWILABUS

Routage (27)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]

R2:1 [N-E]

3

0

R2:
R2:0 [L]
R1:1[O]
R4:2[O]
R5:3 [O]
R3:2 [O]

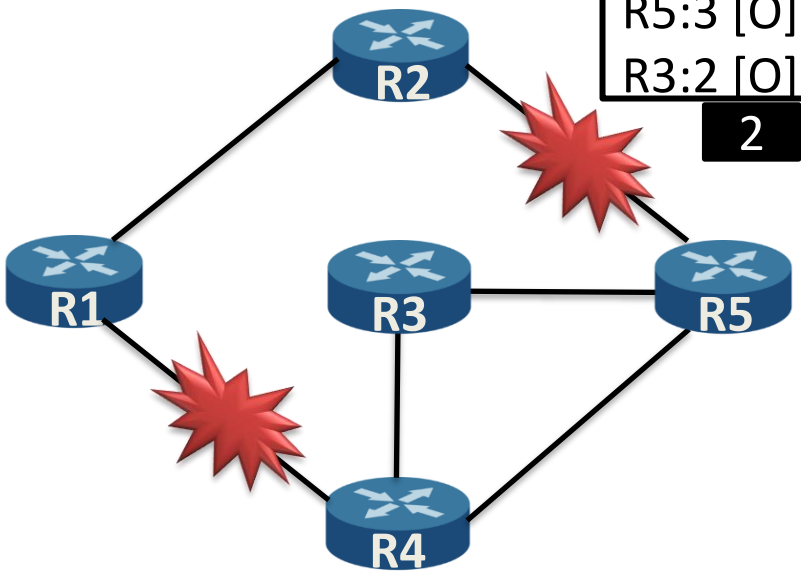
2

R5:
R5:0 [L]
R4:1 [S-O]
R2:3 [S-O]
R3:1 [O]
R1:2 [S-O]

0

R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:3 [S]
R1:2 [S]

0



R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

0

2 R1 détecte la panne (on ne tient pas compte de R4)

3

SWILABUS

Routage (28)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]

R2:1 [N-E]

5

4

R2:
R2:0 [L]
R1:1[O]
R4:2[O]
R5:3 [O]
R3:2 [O]

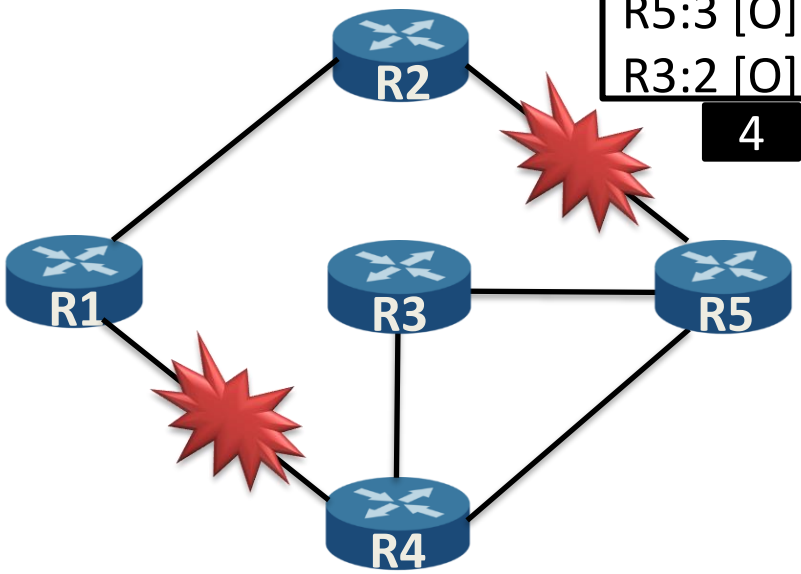
4

R5:
R5:0 [L]
R4:1 [S-O]
R2:3 [S-O]
R3:1 [O]
R1:2 [S-O]

0

R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:3 [S]
R1:2 [S]

0



R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

0

4 R1 met à jour sa table

5

Routage (29)

Plan :

- Intro
- **Routage**
- IP

R1:
R1:0 [L]

R2:1 [N-E]

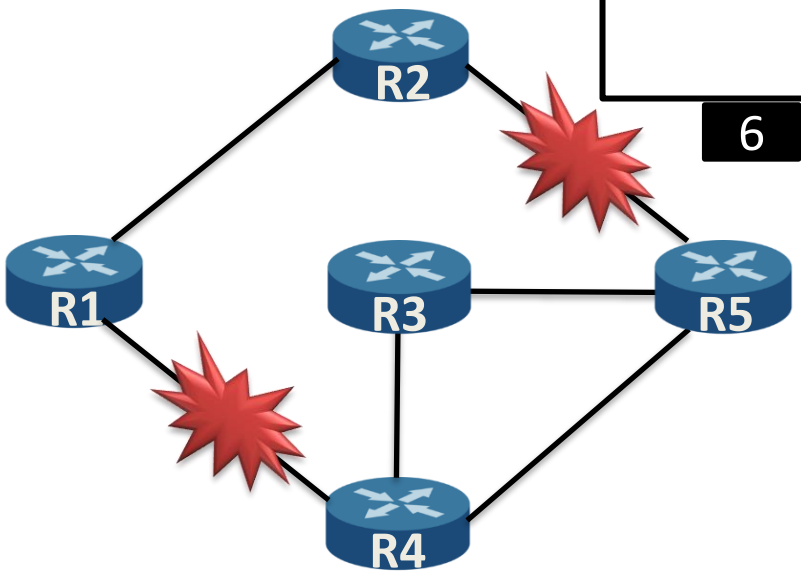
6

R2:
R2:0 [L]
R1:1[O]

6

R5:
R5:0 [L]
R4:1 [S-O]
R2:3 [S-O]
R3:1 [O]
R1:2 [S-O]

0



R3:
R3:0 [L]
R4:1 [S]
R5:1 [E]
R2:3 [S]
R1:2 [S]

0

R4:
R4:0 [L]
R1:1[O]
R2:2[O]
R5:1 [N-E]
R3:1 [N]

0

6 R2 met à jour sa table

SWILABUS

Routage (30)

- Comment régler ce problème ?
  - Origine: un routeur annonce sur une ligne les routes qu'il a appris par cette même ligne
  - Solution: empêcher qu'un routeur annonce sur une ligne les routes qu'il a appris par cette ligne.
    - » Horizon partagé: consiste à créer un vecteur différent et spécifique à chaque ligne
  - Mais le problème réapparait lorsque plus de routeurs sont impliqués
    - » Pas de solution à ce problème pour l'instant
    - » Un mécanisme sera mis en place dans le protocole réseau implémentant cet algorithme.

Plan :

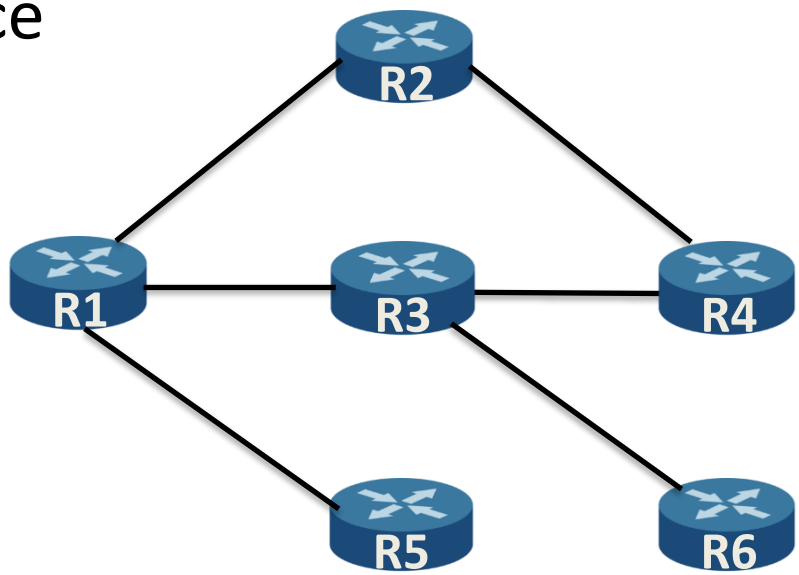
- Intro
- **Routage**
- IP

Routage (31)

Plan :

- Intro
- **Routage**
- IP

– Exercice



1. Ordre d'envoi des vecteurs: R3, R1, R6, R4, R5, R2, R3
2. Que faudrait-il faire pour que les routes soient stabilisées ?
3. Que se passe-t-il en cas de panne sur R3-R4 ?

IP (1)

Plan :
- Intro
- Routage
- **IP**

- Internet Protocol
  - Protocole de la couche réseau
    - Assure l'identification des machines
    - Permet l'envoi d'information sur le réseau
    - Fonctionne en mode datagramme
    - Deux versions majeures:
      - IPv4 – RFC 791
      - IPv6 – RFC 2373 et 2460
    - Ce protocole est déployé sur Internet. Majoritairement IP version 4
    - Service sans connexion et non fiable

IP (2)

Plan :
- Intro
- Routage
- **IP**

– Adressage des machines (rappel)
  - L'adresse IP se décompose en 2 parties:

    – La longueur de chaque partie est définie par le masque de sous-réseau. En IPv4: si 24 bits à 1, identification réseau comporte 24 bits et l'identification de la machine 8 bits (32 – 24)
    – L'adresse réseau est obtenir en remplissant tous les bits de la partie *identification machine* par des 0. Elle est également obtenue en faisant un ET LOGIQUE entre l'IP ou le masque
    – L'adresse réseau permet de déterminer si 2 machines sont *directement connectées*.

SWILABUS

IP (3)

Plan :
- Intro
- Routage
- **IP**

- Si 2 machines **sont** directement connectées, elles peuvent s'échanger des informations sans relai intermédiaire
- Si 2 machines **ne sont pas** directement connectées et souhaitent communiquer, il faut qu'elles utilisent un ou plusieurs relais intermédiaires pour dialoguer entre-elles.
  - Les relais ont pour but d'acheminer les informations de la source à la destination
- L'adresse de multi-diffusion (ou broadcast) en IPv4 est obtenue en plaçant tous les bits à 1 dans la partie *identification machine*
  - Cela permet d'envoyer une informaiton à toutes les machines du réseau / sous-réseau

IP (4)

Plan :
- Intro
- Routage
- **IP**

- IPv4: Adresses et masques courants
  - Historiquement, les adresses étaient données suivant des *classes* prédéfinies :
    - Classe A: 1.0.0.0 à 127.255.255.255, masque : /8
    - Classe B: 128.0.0.0 à 191.255.255.255, masque: /16
    - Classe C: 192.0.0.0 à 223.255.255.255, masque: /24
    - Classe D: 224.0.0.0 à 239.255.255.255, utilisé pour le multicast
    - Classe E: 240.0.0.0 à 255.255.255.255, réservé pour une utilisation future.
  - C'est pourquoi, les institutions universitaires belges disposent d'un /16 (65 536 adresses IP)
    - FUNDP: 138.48.*.* - UCL: 130.104.*.* - **MIT: 18.*.*.***
    - ULg: 139.165.*.* - ULB: 164.15.*.*

IP (5)

Plan :
- Intro
- Routage
- **IP**

- Aujourd'hui, utilisation du CIDR (Classless InterDomain Routing) [RFC 1519]
  - Granularité plus fine, possibilité d'avoir des (sous-)réseaux plus précis
  - Ex: 193.190.64.112/28 (@HELMo)
- Types d'adresse
  - Adresses publiques
    - » Adresses IP allouées par l'IANA (Internet Assigned Number Authority), souvent achetées
    - » Uniquement sur l'internet
  - Adresse boucle-locale (loopback)
    - » Adresse désignant la machine courante
    - » 127.0.0.1 (IPv4), ::1 (IPv6) ou *localhost*
    - » Permet d'accéder à un serveur local

SWILABUS

IP (6)

Plan :
- Intro
- Routage
- **IP**

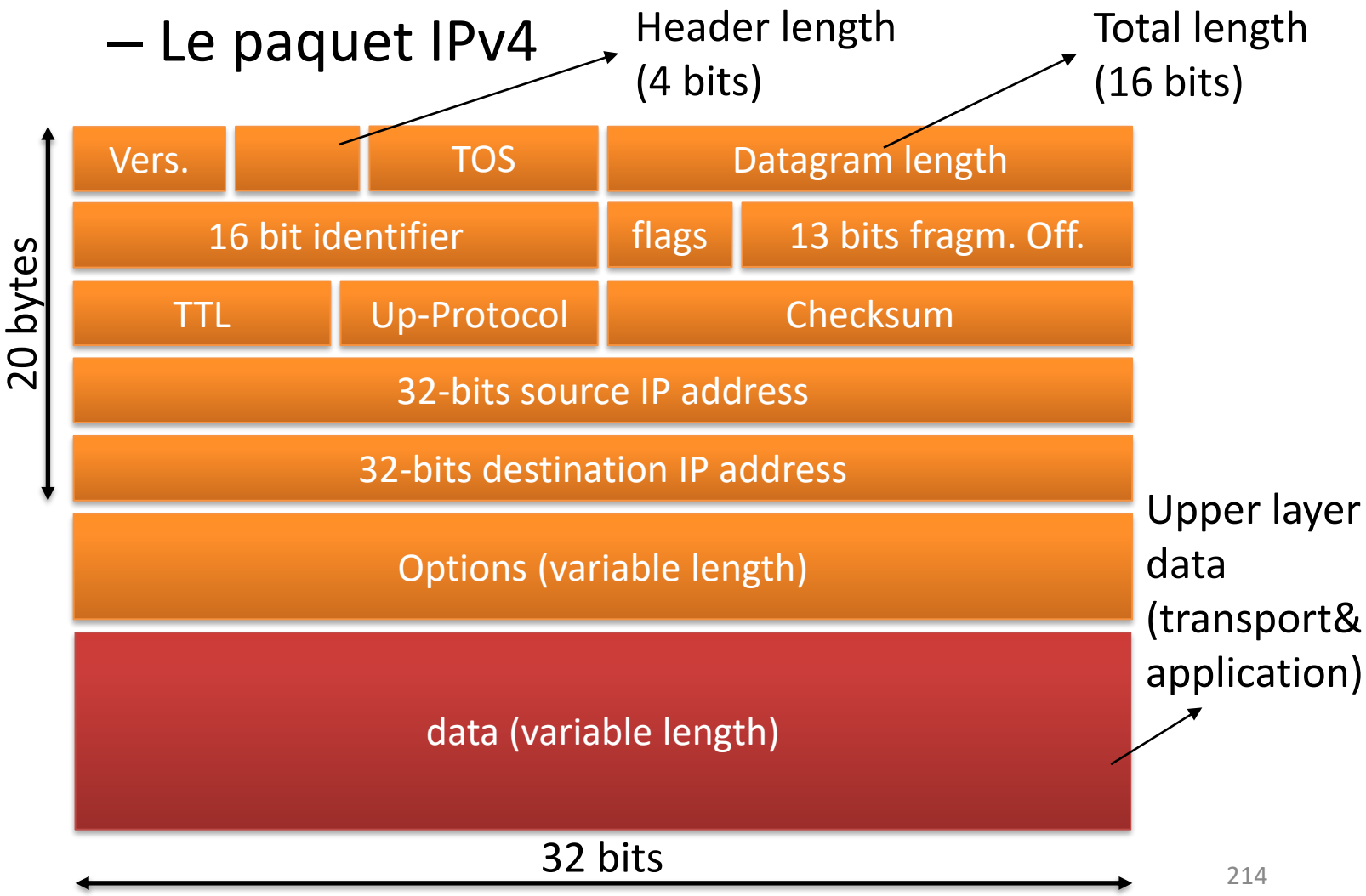
– Adresses privées (pour les réseaux locaux)
  » 10/8
  » 172.16/12
  » 192.168/24
  » FC00::/7 (IPv6)
  » Ces adresses ne doivent jamais être propagées sur l'internet. Normalement, les routeurs sont configurés pour jeter les paquets en provenance / à destination de ces adresses privées
– Adresse du réseau | NETWORK | 0 … 0 |
– Adresse broadcast | NETWORK | 1 … 1 |

IP (7)

Plan :

- Intro
- Routage
- **IP**

– Le paquet IPv4



SWILABUS

IP (8)

Plan :
- Intro
- Routage
- **IP**

- *Vers:* Version IP, soit IPv4, soit IPv6
- *Header Length:* longueur de l'entête (options comprises)
- *TOS (Type of Service):* priorité du paquet (généralement ignoré)
- *Datagram Length:* Longueur totale du paquet codé sur 16 bits donc taille maximale: 65536 o
- *Identifier*: Identifiant placé par la source
- *Flags*: Ensemble de *drapeaux* pouvant être positionnés à 0 ou 1.
  - *DF: Don't Fragment – Ne pas fragmenter*
  - *MF: More Fragment – Des fragments suivent*
- *Fragmentation offset:* déplacement par rapport au paquet initial (fragmentation)

SWILABUS

IP (9)

Plan :
- Intro
- Routage
- **IP**

- *TTL – Time-to-live:* temps de survie d'un paquet dans le réseau
- *Up-protocol:* Indique qui est le destinataire des données (TCP, UDP, …) → quelle couche transport ? (Ex: 6➔ TCP; 17 ➔ UDP)
- *Checksum*: Détection d'erreur sur l'entête du paquet
- *Source address:* Adresse IP source
- *Destination address:* Adresse IP destination
- *Options:* Option pouvant être ajoutée à l'information (*record-route, source-routing,…)*
- *Data:* données placées par la couche transport.

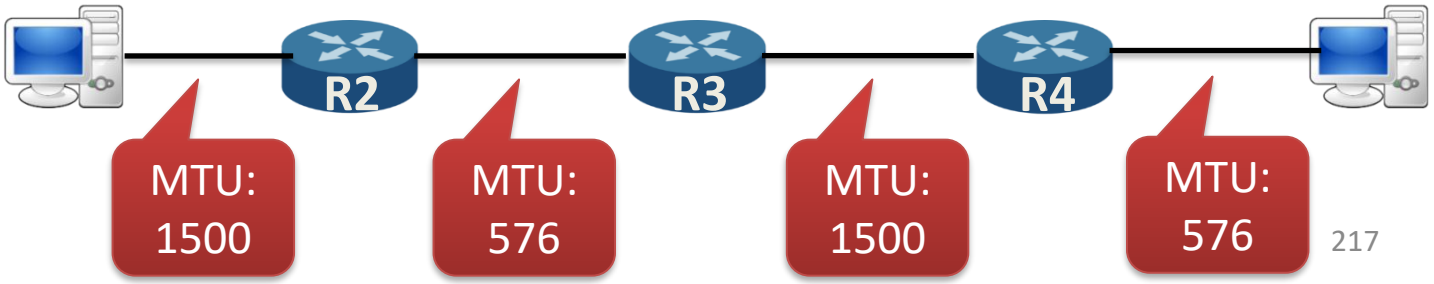
IP (10)

Plan :

- Intro
- Routage
- **IP**

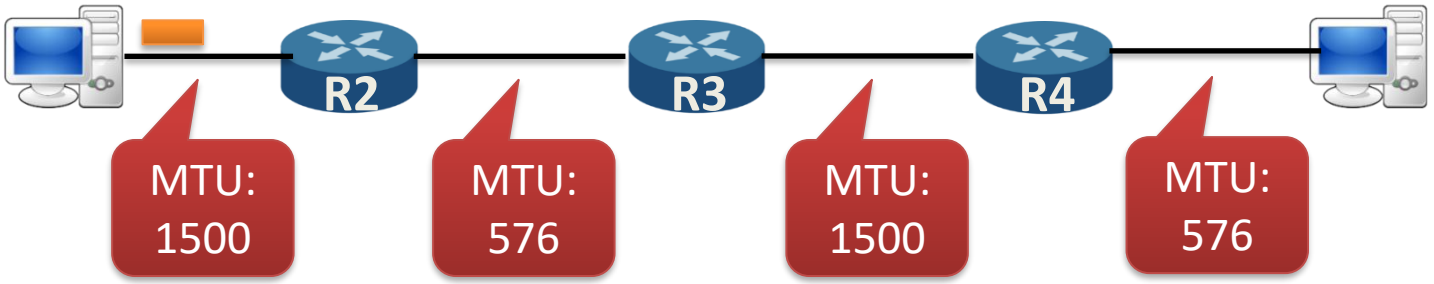
– Fragmentation IPv4

- MTU: Maximum Transfert Unit
  - Taille macimale supportée d'un paquet IP
  - Cette taille est souvent imposée par la couche accès réseau
  - SI des réseaux de nature différente, le MTU peut ne pas être le même
- MSS vs MTU
  - MSS au niveau de la couche transport
  - MSS <= MTU



SWILABUS

IP (11)



Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (12)

– Comment cela se met-il en place ?

-
  -
-
  -
  -
-

Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (13)

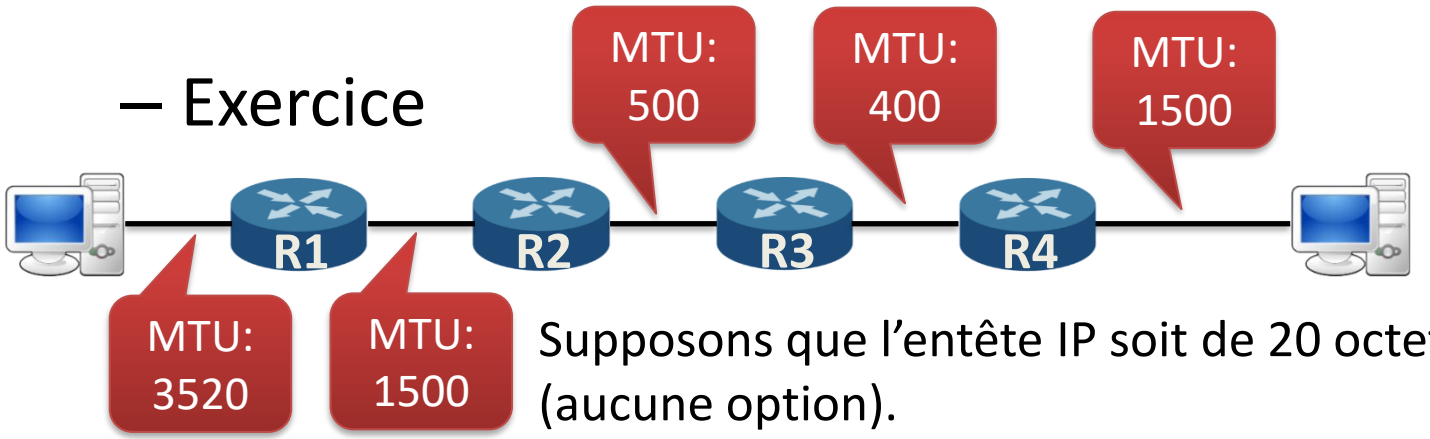
- Comment ne pas mélanger les fragments de plusieurs paquets ?
  - –
- Comment reconstituer les paquet original ?
  - –
  - Comment déterminer si tous les fragments sont arrivés ?
    - »
- Quand reconstituer le paquet ?
  - –

Plan :

- Intro
- Routage
- **IP**

IP (14)

– Exercice



Supposons que l'entête IP soit de 20 octets (aucune option).

Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (15)

Plan :
- Intro
- Routage
- **IP**

– Comment empêcher qu'un paquet ne boucle sur le réseau ?
  - Il y a plusieurs chemins pour atteindre une destination
  - Les tables de routage ne sont pas toujours stables (dynamique du réseau, problèmes, initialisation, etc.)
  - Utilisation du TTL (Time-To-Live)
    –
      »
      »
    –

IP (16)

Plan :
- Intro
- Routage
- **IP**

– ICMP (Internet Control Message Protocol)
  - Protocole de signalisation d'événements / exceptions
  - RFC 792
  - Utilisé entre les couches réseaux
  - Quelques messages courants:
    – **Destination unreachable:** destination inaccessible
    – **echo-request:** demande d'état d'une machine
    – **echo-reply:** réponse d'état d'une machine
    – **ttl-expired:** le TTL d'un paquet passe à 0
    – **ip header bad:** erreur dans l'entête

IP (17)

- Manipulation d'ICMP
  - PING
    - » L'application *ping* permet de savoir si une machine / routeur est capable de recevoir et répondre au niveau IP
    - » Utile pour vérifier les connexions réseaux
    - » Souvent limité par les firewalls
  - TRACEROUTE
    - » L'application *traceroute* permet de déterminer le chemin d'une source vers une destination.
    - » Envoi de paquets IP avec un TTL croissant (en commençant par 1)
      - A chaque message ICMP **time-exceeded**, l'indentification du routeur / station qui a jeté le paquet est à l'intérieur du message ICMP
      - Trace entre la source ➔ destination

Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (18)

Plan :
- Intro
- Routage
- **IP**

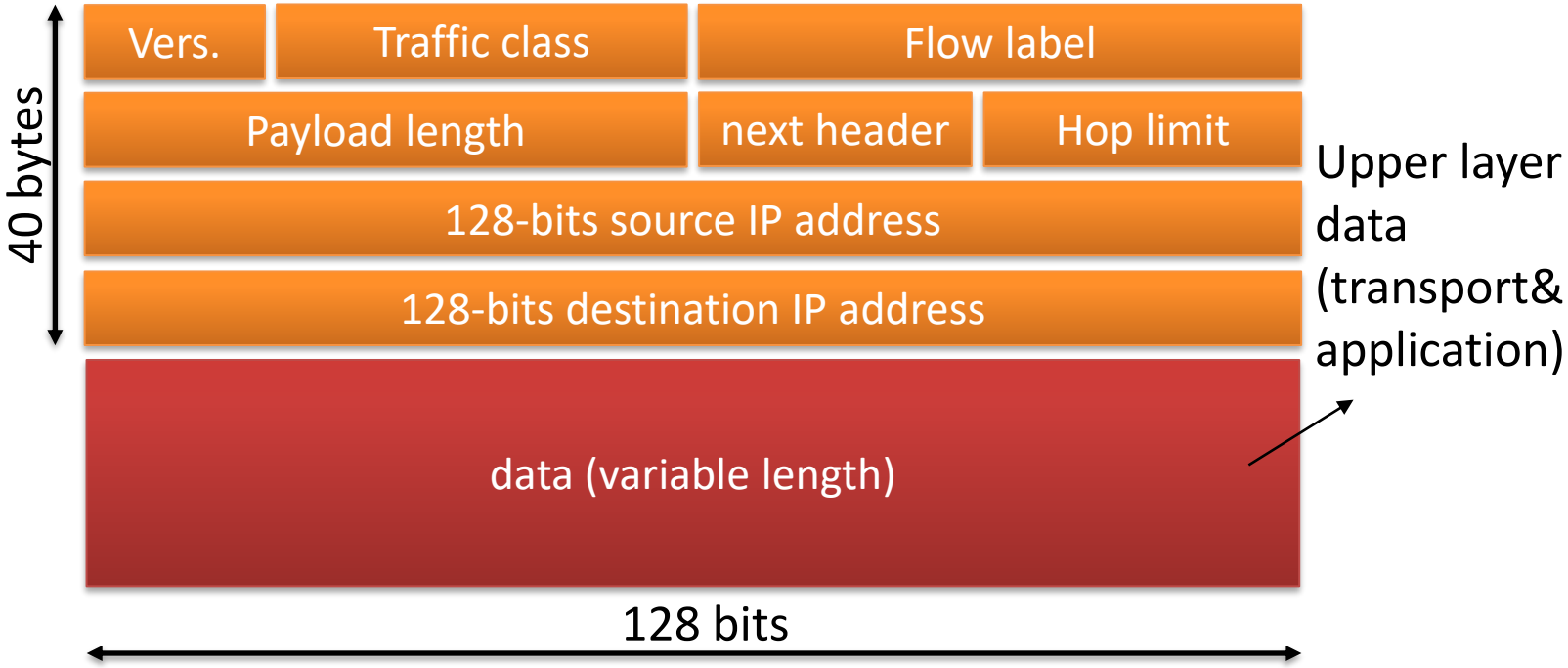
– IPv6
  - Objectifs
    – **Augmenter le nombre d'adresses IP**, pour faire face aux besoins futurs
    – **Améliorer les performances** et le traitement des paquets par les routeurs
    – **Intégrer la sécurité** directement dans le protocole en définissant les aspects à l'intérieur. Il ne s'agit pas d'un ajout (à l'inverse d'IPv4 et Ipsec)
    – **Supporter les nouvelles applications** comme les applications temps réelles utilisant des flux de données
    – **Faciliter la configuration** en intégrant des mécanismes faciles d'auto-configuration et de découverte.

IP (19)

- Paquet IPv6

Plan :

- Intro
- Routage
- **IP**



IP (20)

Plan :

- Intro
- Routage
- **IP**

- **Version:** Indique la version du protocole (IPv6 ici)
- **Traffic class:** permet de marquer les paquets en vue de fournir une qualité de service particulière
- **Flow label:** Permet d'étiqueter un paquet, identifier un contexte utilisé par les routeurs pour transmettre le paquet, identifier les paquets d'un flux temps réel, …
- **Payload length**: taille des données, si celles-ci sont inférieures à 65 535.
- **Next header:** permet 2 choses: mentionner la couche supérieure qui doit recevoir les données (TCP, UDP, …) ou mentionne une option à traiter
- **Hop limit:** nombre total de routeur que ce paquet peut traverser. Dès qu'il atteint 0, le paquet est jeté.
- **128 bits addresses**: adresse source et destination codée sur 128 bits, ce qui donne 2^{128} adresses possibles
- **Data:** données qui viennent de la couche supérieure.

IP (21)

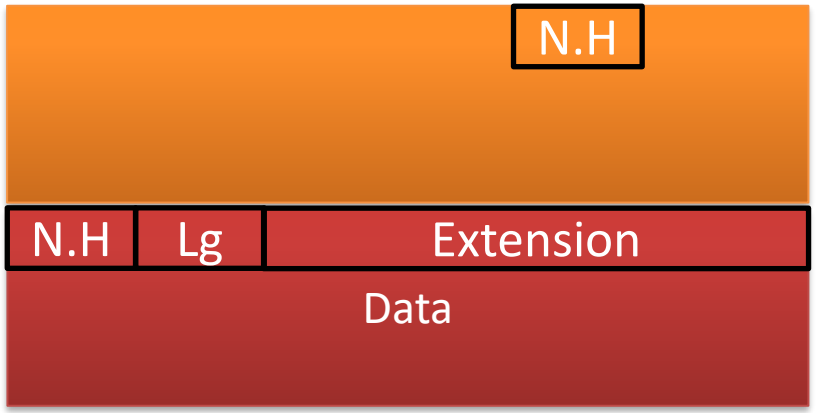
- Différence IPv6 ⇔ IPv4
  - **Disparition du checksum:** le paquet IPv6 n'embarque plus de mécanisme de contrôle, jugé redondant par rapport à la couche transport et/ou la couche accès réseau.
    - » Cette disparition améliore les performances car les routeurs ne doivent plus vérifier cette valeur
  - **Disparition des options de taille variable:** des options peuvent se placer entre les données via une entête particulière
    - » L'entête du paquet étant de taille fixe, les routeurs peuvent traiter ceux-ci plus rapidement
  - **Disparition de la fragmentation:** IPv6 ne supporte plus la fragmentation réalisée par les routeurs. Cependant, la source peut fragmenter l'information via une entête particulière
    - » Amélioration des performances car la fragmentation était coûteuse pour les routeurs

Plan :
- Intro
- Routage
- **IP**

IP (22)

Plan :
- Intro
- Routage
- **IP**

- Le champ Next-Header (N.H.)
  - Il mentionne l'entête suivante (ce que contiennent les données) comme *up-protocol* en IPv4



Valeurs de ce champ:
- 4 IPv4
- 6 TCP
- 17 UDP
- 41 IPv6
- 44 Fragmentation
- 50 Confidentialité
- …

Donc si ce champ vaut 6, les données contiennt un TPDU TCP. Si la Valeur est 44, cela signifie qu'au début des données, une entête Particulière décrit le fragment. Un nouveau champ Next-Header Renseigne alors sur la suite (autres options, nature des données)

SWILABUS

IP (23)

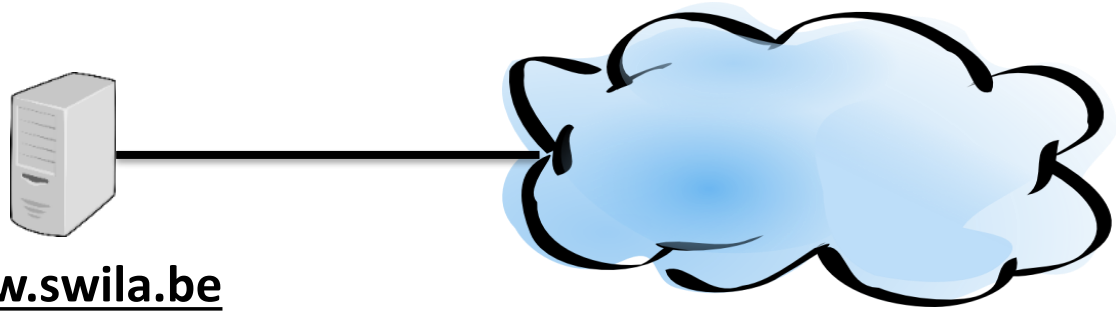
Plan :
- Intro
- Routage
- **IP**

- Les adresses IPv6
  - **Adresse locale-lien:** c'est une adresse attachée à l'interface. Elle a une portée limitée au LAN (ne traverse pas les routeurs).
    - » Préfixe: `FE80::/10`, configurée souvent automatiquement.
  - **Adresse locale-unique**: c'est une adresse attachée à une interface. Elle n'est pas utilisable sur Internet
    - » C'est un *range privé* non routable sur Internet. Le préfixe est: `FC00::/7`
  - **Adresse globale-unique**: C'est une adresse globale (unique sur Internet). Elle est attribuée par un ISP ou un fournisseur de service.
    - » Préfixe attribué: `2000::/3`

IP (24)

Plan :
- Intro
- Routage
- **IP**

– **Adresse multicast:** c'est une adresse désignant un groupe de receveurs
  » Préfixe attribué: `FF00::/8`
– **Adresse anycast**: n'importe quelle machine d'un groupe (utilisé pour les DNS mondiaux)
– En IPv6, une machine **a plusieurs adresses !**



www.swila.be
`195.154.39.227` (IPv4 globale)
`2001:bc8:38eb:fe10::11` (IPv6 globale-unique)
`fe80::20c:29ff:febc:1f57` (IPv6 locale-lien)

SWILABUS

IP (25)

Plan :
- Intro
- Routage
- **IP**

- ICMPv6
  - Evolution du protocole ICMP existant en IPv4
  - Regroupe
    - La gestion des groupes multicast (IGMP en IPv4)
    - Le mécanisme permettant de connaître l'adresse physique en fonction d'une IP (ARP en IPv4)
    - Neighbor discovery qui permet de déterminer les adresses locale-lien des voisins, le routeur, savoir si un voisin est accessible ou non
    - Mobile IPv6 (support des mobiles)
  - 2 types de message:
    - Signalisation des erreurs
    - Message d'information

IP (26)

Plan :
- Intro
- Routage
- **IP**

- Messages d'erreur
  - **Destination unreachable :** destination inaccessible
  - **Packet too big:** paquet trop grand (aucune fragmentation supportée par les routeurs)
  - **Time Exceeded:** le paquet a atteint la limite hop-limit
  - **Parameter problem:** Problème dans les paramètres
- Quelques messages d'information
  - [Ping] Echo request/Reply: demande/réponse [ping6]
  - [group] Multicast Listener Query/Report/Done : gestion des groupes multicast
  - [ND] Router solicitation/advertisement, Neighbor solicitation/advertisement, Redirect Message : échange concernant les voisins.

IP (27)

- Neighbor Discovery Protocol (ND)
  - Ce protocole est utilisé pour découvrir les voisins et réaliser l'auto-configuration des interfaces réseaux
    - » Déduire l'adresse locale-lien, vérifier son unicité, construire des adresses globales uniques sur un préfixe donné, distribuer la route par défaut
    - » On voit que ND reprend certaines caractéristiques du mécanisme DHCP (sauf que l'adresse n'est pas distribuée mais construite). S'il faut distribuer des adresses, on peut utiliser DHCPv6
  - Ce protocole prévoit 5 types de messages: sollicitation et annonce d'un routeur, sollicitation et annonce de voisin et le message de redirection

Plan :

- Intro
- Routage
- **IP**

IP (28)

Plan :
- Intro
- Routage
- **IP**

– Router solicitation / advertisement
  » Le routeur envoie régulièrement ses annonces (avertisement) et lorsqu'il est sollicité pour le faire.
  » Ces annonces peuvent mentionner :
    - S'il s'agit du routeur par défaut
    - Le préfixe à utiliser pour l'auto-configuration
    - Si la configuration est de type *stateful* (utilisant DHCPv6) ou *stateless* (construction de l'adresse)
    - Le temps d'accessibilité des voisins (IPv6 simplifie la renumérotation du réseau)
    - Des routes particulières à distribuer aux périphériques réseaux

IP (29)

- Neighbor solicitation / advertisement
  - » Ces messages permettent **de résoudre une adresse** (obtenir l'adresse physique correspondant à l'adresse réseau) mais aussi **de déterminer si le voisin est accessible**.
  - » Nous y reviendrons lorsque nous aborderons la couche inférieure : la couche accès réseau.

Plan :

- Intro
- Routage
- **IP**

IP (30)

Plan :
- Intro
- Routage
- **IP**

– Transition vers IPv6
  - La transition entre IPv4 et IPv6 doit être progressive (on ne peut pas imposer un changement brutal)
  - Les coûts pour la transition peuvent être importants (le logiciel et/ou le matériel doit être adapté)
    – Lors de vos développements, veillez à être compatible avec IPv6 !
  - Plusieurs techniques existent pour permettre une transition vers IPv6 de manière graduelle.

IP (31)

- Techniques possibles:
  - Particulier: tunnel-brokers
  - Entreprise: 6to4, tunnel-brokers
  - ISP: déploiement complet (fonctionnement en dual-stack), 6rd
    - » A titre d'exemple, il faut remarquer que le fournisseur français **free.fr** a déployé IPv6 en 5 semaines à plus d'1,5 millions de clients (en développant 6rd).

Plan :

- Intro
- Routage
- **IP**

IP (32)

Plan :
- Intro
- Routage
- **IP**

– Tunnel-brokers
  - » Idée: créer un tunnel entre le réseau du client et un fournisseur de service IPv6
  - » Le fournisseur de service attribue un sous-réseau à chaque client (sixxs.net, he.net, …)
  - » Enregistrement de la part du client
  - » Tunnel: Placer des paquets IP comme données d'autres paquets. Par exemple, le client encapsule des paquets IPv6 dans des IPv4

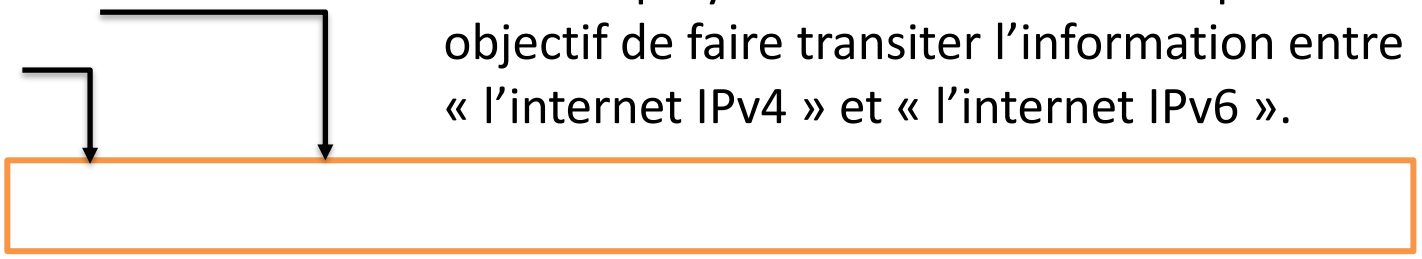
Image: SonicOS 6.2.1 Administration Guide, © Dell Documentation

SWILABUS

IP (33)

Plan :
- Intro
- Routage
- **IP**

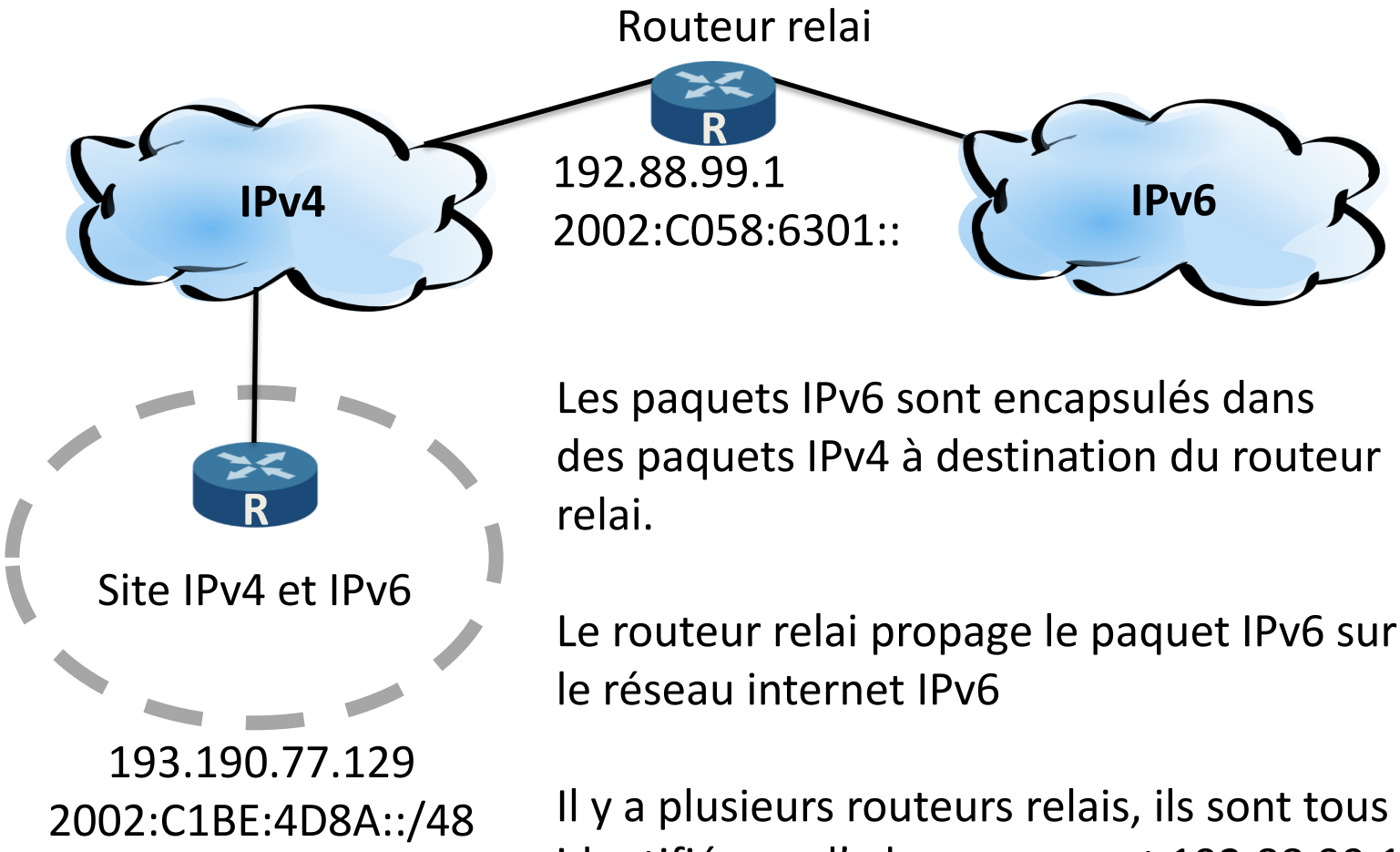
– Tunnel automatique et 6to4
  - » Idée: on peut voir le réseau IPv4 comme un immense réseau point-à-point permettant la communication entre 2 entités données (sorte de tunnel automatique).
  - » On pourrait créer un lien direct entre l'internet IPv6 et IPv4 **en utilisant l'adresse IPv4 globale du client**.
  - » Plus besoin d'établir des tunnels et de s'enregistrer auprès d'un fournisseur. Plus besoin de distribuer des préfixes IPv6
  - » Il faut déployer des routeurs-relais qui ont comme objectif de faire transiter l'information entre « l'internet IPv4 » et « l'internet IPv6 ».



IP (34)

Plan :

- Intro
- Routage
- **IP**



Les paquets IPv6 sont encapsulés dans des paquets IPv4 à destination du routeur relai.

Le routeur relai propage le paquet IPv6 sur le réseau internet IPv6

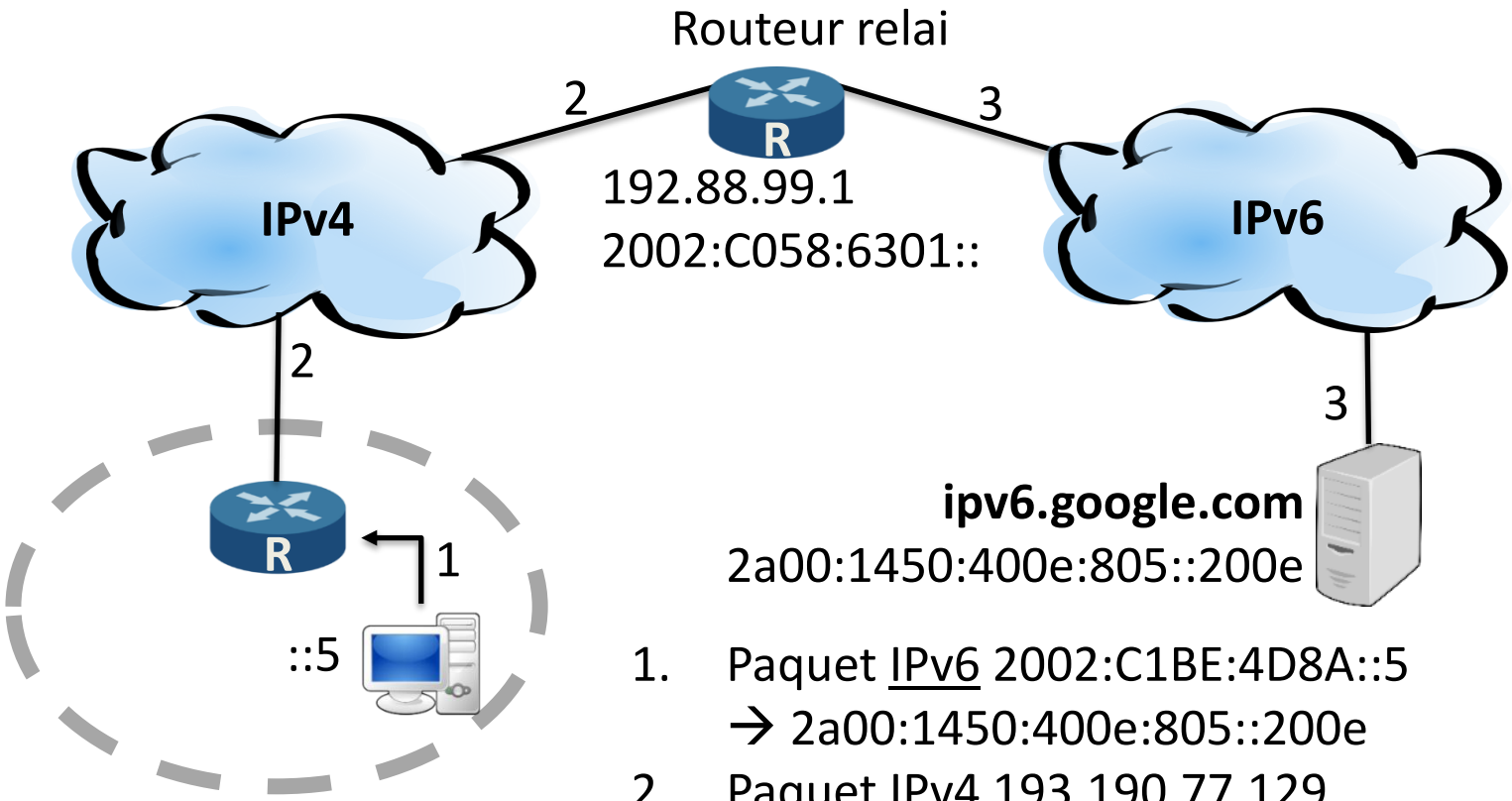
Il y a plusieurs routeurs relais, ils sont tous identifiés par l'adresse anycast 192.88.99.1.

SWILABUS

IP (35)

Plan :

- Intro
- Routage
- **IP**



1. Paquet IPv6 2002:C1BE:4D8A::5
   → 2a00:1450:400e:805::200e
2. Paquet IPv4 193.190.77.129
   → 192.88.99.1 contenant le paquet IPv6 original
3. Paquet IPv6 2002:C1BE:4D8A::5
   → 2a00:1450:400e:805::200e

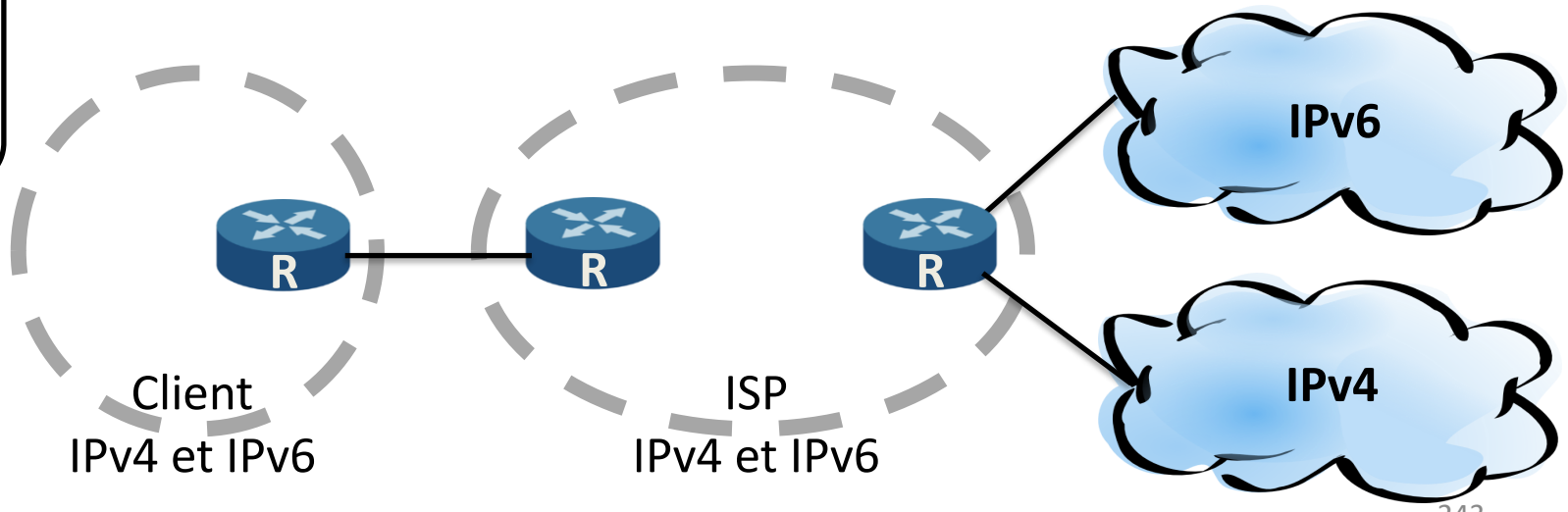
SWILABUS

IP (36)

Plan :

- Intro
- Routage
- **IP**

– Déploiement (et dual-stack)
  - » Le fournisseur a transformé son réseau pour supporter nativement IPv6
  - » Il fournit à ses client un accès réseau IPv6 et IPv4
  - » Il distribue des adresses IPv6 et IPv4 à ses clients
  - » Les serveurs et routeurs disposent d'une adresse IPv6 et d'une adresse IPv4 (dual stack)



SWILABUS

IP (37)

Plan :
- Intro
- Routage
- **IP**

» Le déploiement complet impose que toute l'infrastructure soit compatible

» Dans certains types de réseau (i.e. xDSL), l'ISP n'a pas nécessairement la maîtrise sur l'ensemble de l'équipement
  - L'opérateur historique (Proximus, Orange / France Télécom, Voo, Télénet, …) reste responsable de la connectivité vers le client (paires torsadées, câble de télédistribution, …)
  - Il est donc parfois difficile de réaliser un déploiement complet d'IPv6 jusqu'au client.

IP (38)

Plan :
- Intro
- Routage
- **IP**

– 6rd (6-rapid deployment)
  » Proposition de l'opération **free.fr** pour la fourniture d'IPv6 sur son réseau
  » Chaque client dispose d'une *freebox®* qui est un routeur controlé par **free.fr**
  » 6rd est une modification de 6to4, qui est utilisable par un ISP
  » 6to4 a les faiblesses suivantes:
    • Le dimensionnement du routeur relai pose problème: il doit être accessible à tous les utilisateurs (combien de flux, combien de sessions, quelle bande passante, …)
    • La connectivité IPv6 dépend de nombreux prestataires (tous ceux disposant ces routeurs relais)

IP (39)

Plan :
- Intro
- Routage
- **IP**

- Peut difficilement être retenu pour déployer IPv6 à des clients !

» **Free.fr** a modifier le mécanisme et créé le 6rd

- Chaque client a un routeur 6rd qui place les paquets IPv6 dans des paquets IPv4
- Les routeurs relais utilisés par la freebox sont ceux de **Free.fr** exclusivement
  - Dimensionnement possible
- Les adresses IPv6 attribuées sont dans le réseau de **Free.fr** (plus d'utilisation du préfixe 6to4 possible).
- Il n'est pas nécessaire de modifier le réseau entre la *freebox* et les routeurs de **Free.fr** (ainsi **Free.fr** est indépendant de l'opérateur historique)

IP (40)

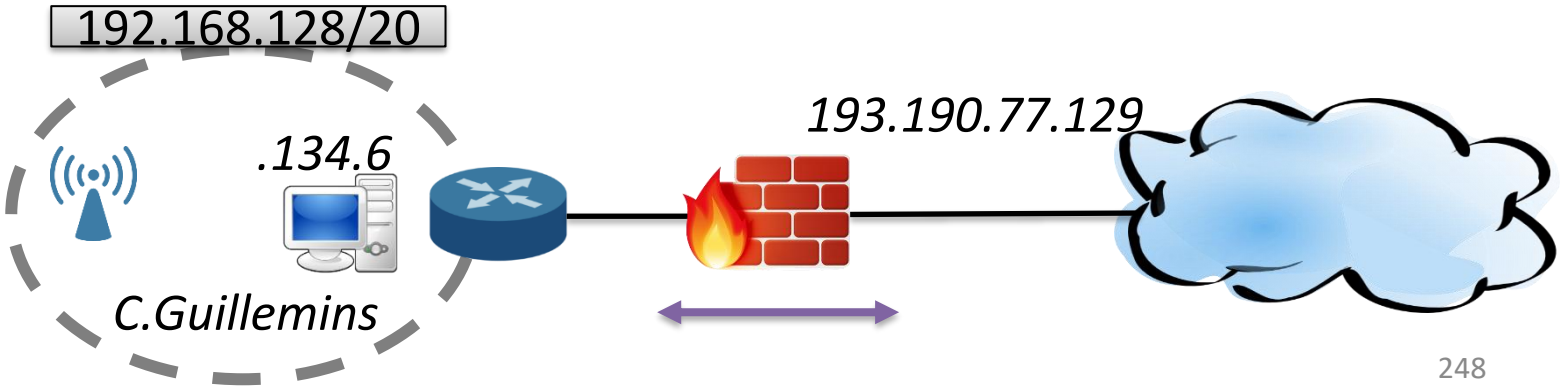
Plan :
- Intro
- Routage
- **IP**

– Adoption d'IPv6
- IPv6 est intéressant pour le nombre d'adresses
- Cependant, beaucoup de mécanismes sont mis en œuvre pour économiser les adresses IP
  – L'adoption de CIDR pour affiner les *range IP*
  – Le NAT (Network Addresss Translator) permet de *partager* une adresse IP publique entre plusieurs (centaines de) machines
  – Le DHCP (Dynamic Host Configuration Protocol) permet de distribuer les adresses au moment de la connexion
- L'adoption d'IPv6 en Belgique est très bonne
- Vérifiez la compatibilité de vos applications avec IPv6

IP (41)

- Fonctionnement du NAT
  - En modifiant le port source, le NAT peut réaliser une translation d'adresse dans les deux sens
  - Il permet de connecter un réseau privé à Internet
  - Doit modifier le paquet avant de le propager
  - Le routeur fait parfois office de firewall

Plan :
- Intro
- Routage
- **IP**



SWILABUS

IP (42)

Plan :
- Intro
- Routage
- **IP**

- Routage sur Internet
  - Topologie: Internet = réseau des réseaux
    - Ensemble de réseaux interconnectés entre-eux
    - Un système autonome (AS) est un (ou plusieurs) réseau **d'un même opérateur**
      - Exemple: Belnet, Proximus, AT&T, Level3, Telia, BT, France-Télécom
    - Les AS sont connectés entre-eux afin de permettre une connexion d'un point à l'autre
    - Les AS sont numérotés, ces numéros sont attribués par l'IANA (ou un délégué : RIPE)
    - On distingue 2 zone: intra et interdomaine.

IP (43)

Plan :
- Intro
- Routage
- **IP**

- Le routage sur Internet se fait suivant 2 niveaux:
  - » Intra-domaine: utilisation d'un protocole IGP (Interior Gateway Protocol) comme RIP et OSPF
    - Connaît la carte complète du réseau
  - » Inter-domaine: utilisation d'un protocole EGP (Exterior Gateway Protocol) comme BGP
    - Connaissance partielle du réseau
    - Utilisé entre les AS pour l'échange des informations de routage
    - La topologie d'un AS (la carte du réseau interne) n'est pas publiée à l'extérieur
      - Information stratégique
      - Volonté politique !

IP (44)

Plan :
- Intro
- Routage
- **IP**

– Routage intra-domaine
  - Protocoles courants
    – RIP (Routing Information Protocol) – RFC 1058/2453
      » Basé sur les vecteurs de distance
    – OSPF (Open Shorthest-Path First) – RFC 2328
      » Basé sur les états de liaison
    – IS-IS (Intermediate System – Intermediate System)
      » Basé sur les états de laisons
      » Non vu ici – Comparatif avec OSPF
  - RIP – protocole assez simple
    – Mesure de la qualité d'une route = nombre de sauts
    – Maximum de sauts: 15 (➔ limite la taille de l'AS)
    – Utilise UDP, port 5120

IP (45)

Plan :
- Intro
- Routage
- **IP**

- Les vecteurs sont échangés toutes les 30 s
  - » Messages envoyés avec un TTL de 1 à tous les routeurs RIP (via multicast)
- Le vecteur est envoyé également lors d'un changement
  - » Le message reprend les modifications
  - » Attention au **flapping**: le temps minimum entre 2 mises à jour d'une même route est 5s
- Deux types de message:
  - » Requêtes: permet de demander à un voisin les informations de routage
  - » Réponse:
    - Message envoyé toutes les 30 secondes
    - Peut contenir jusqu'à 25 routeurs destinations
  - » Une destination= IP + Coût

IP (46)

Plan :
- Intro
- Routage
- **IP**

- OSPF
  - Basé sur les états de liaison
  - Utilise l'algorithme de Dijkstra
  - Construit une carte de l'AS dynamiquement
    - » Au démarrage, le routeur envoie le message HELLO à ses voisins
    - » Les routeurs s'échangent ensuite des LSPs décrivant la carte complète du réseau
    - » Chaque routeur sauvegarde la dernière version de chaque lien (LSP et numéro de séquence)
    - » Les routeurs s'échangent des LSPs au moins toutes les 30 minutes et en cas de modification
    - » Un routeur peut interroger un voisin en lui transmettant un LSR (Link-State Request)

IP (47)

Plan :
- Intro
- Routage
- **IP**

– Support pour les grands réseaux:
  » Si le réseau est conséquent, la carte sera complexe et difficile à maintenir
  » Solution: **diviser pour régner !** Le réseau est découpé en zones distinctes (area)
    - Les routeurs connaissent uniquement les autres routeurs de leur zone
    - La topologie d'une zone n'est pas connue des autres
    - Une zone spécifique est définie: *le backbone* qui va assurer que toutes les zones sont joignables
      - Chaque zone doit avoir un routeur dans le backbone
      - Toutes les zones sont joignables en passant par le backbone

IP (48)

Plan :
- Intro
- Routage
- **IP**

– Routage inter-domaine
  - BGP (Border Gateway Protocol) version 4
  - Utilise TCP – port 179
  - RFC 1771, 1772, 1773
  - Protocole basé sur les vecteurs de distance. Interconnecte plusieurs AS entre-eux
  - Le routage entre les AS est un choix politique
  - Problème de comptage à l'infini réglé
    – BGP liste tous les AS traversés
    – Détection possible des boucles

IP (49)

Plan :
- Intro
- Routage
- **IP**

- Choix politiques:
  - 2 politiques de routages sont généralement connues:
    - » Customer-provider
      - Le client paie sa connectivité à Internet auprès d'un fournisseur (Level3, Telia, …)
      - Le client recevra de la part de son fournisseur les informations qui lui sont destinées (et celles de ses clients)
    - » Share-cost peering
      - Les deux AS (AS1 et AS2) sont souvent de taille équivalente
      - AS1 accepte de recevoir les informations d'AS2 *qui lui sont destinées* (ou pour un de ses clients)
      - AS2 accepte de recevoir les informations d'AS1 *qui lui sont destinées* (ou pour un de ses clients)

IP (50)

- Principes
  - Un routeur communique uniquement avec ses voisins
    - » Connexion point-à-point
    - » Appelé « sessions BGP »
    - » Les routeurs s'échangent des informations concernant **des destinations** (préfixes IP) en fonction de leur configuration (choix politiques)
    - » Sur base de sa connaissance du réseau, le routeur BGP détermine la route à utiliser pour atteindre une destination
    - » Les routes annoncées aux autres routeurs BGP se font sur base d'une sélection:
      - Toutes les routes ne sont pas nécessairement annoncées à tous les voisins
      - Si la relation est cutomer-provider, le client annonce uniquement ses propres routes

Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (51)

Plan :

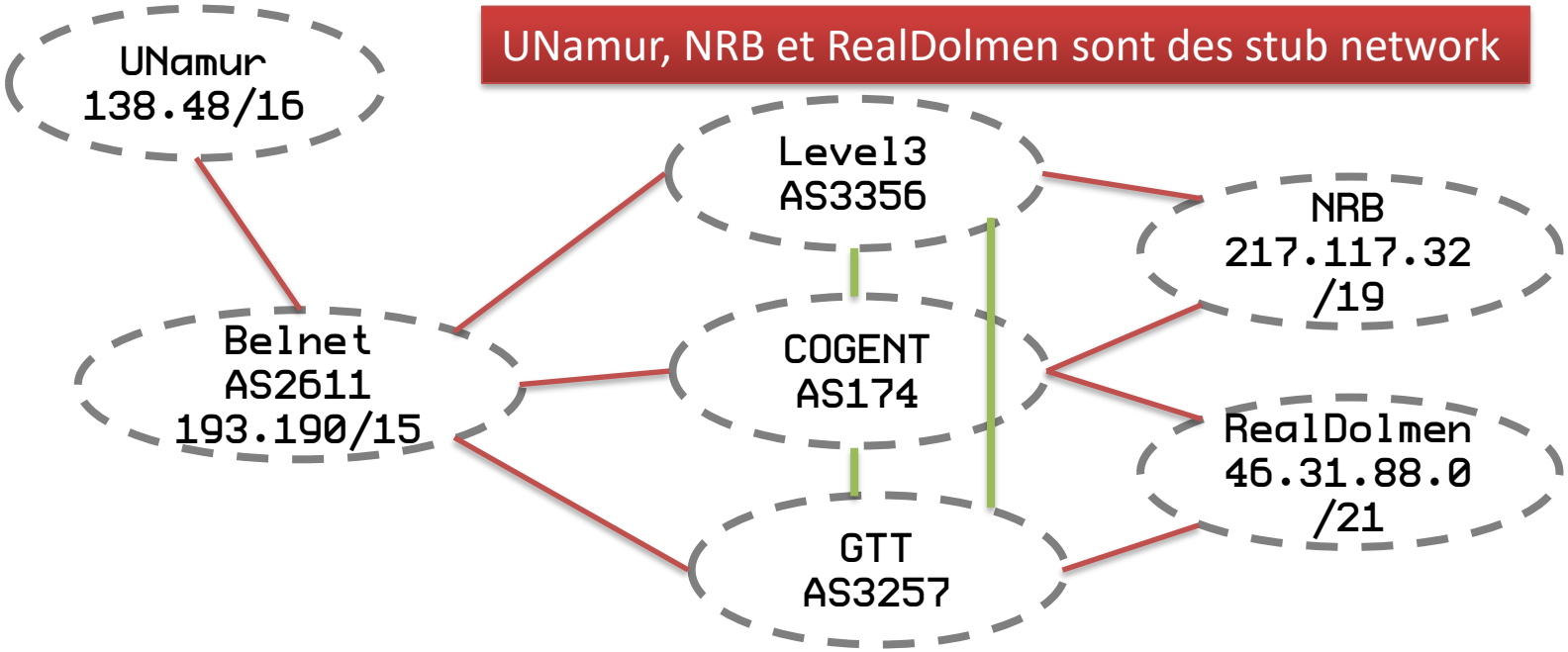
- Intro
- Routage
- **IP**

- Fonctionnement
  - Réception et filtrage des routes reçues:
    - » La réception d'une route BGP peut être considérée **comme une promesse:** si un AS A envoie une route dont la destination est l'AS B, il promet de joindre l'AS B si des informations lui sont transmises
    - » L'AS Path étant attaché à la route, il est possible de faire du routage politique
  - Sélection de la route
    - » Plusieurs routes peuvent mener à une même destination
    - » BGP ne spécifie par comment un AS doit déterminer la route à utiliser. C'est une configuration de l'administrateur.

IP (52)

Plan :
- Intro
- Routage
- **IP**

– Envoi des routes au voisinage
  » Un routeur BGP peut choisir les routes propagées à un voisin donné (promesse !!)
  » Exemple



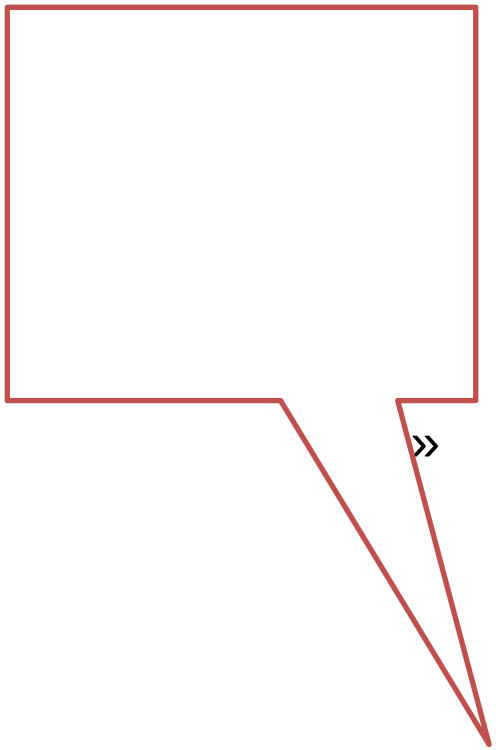
Belnet, **NRB** et **Realdolmen** sont connectés à plusieurs fournisseurs

IP (53)

» Comment Belnet peut-il éviter de servir de transit entre Level3 et Cogent ?

Plan :

- Intro
- Routage
- **IP**



IP (54)

Plan :

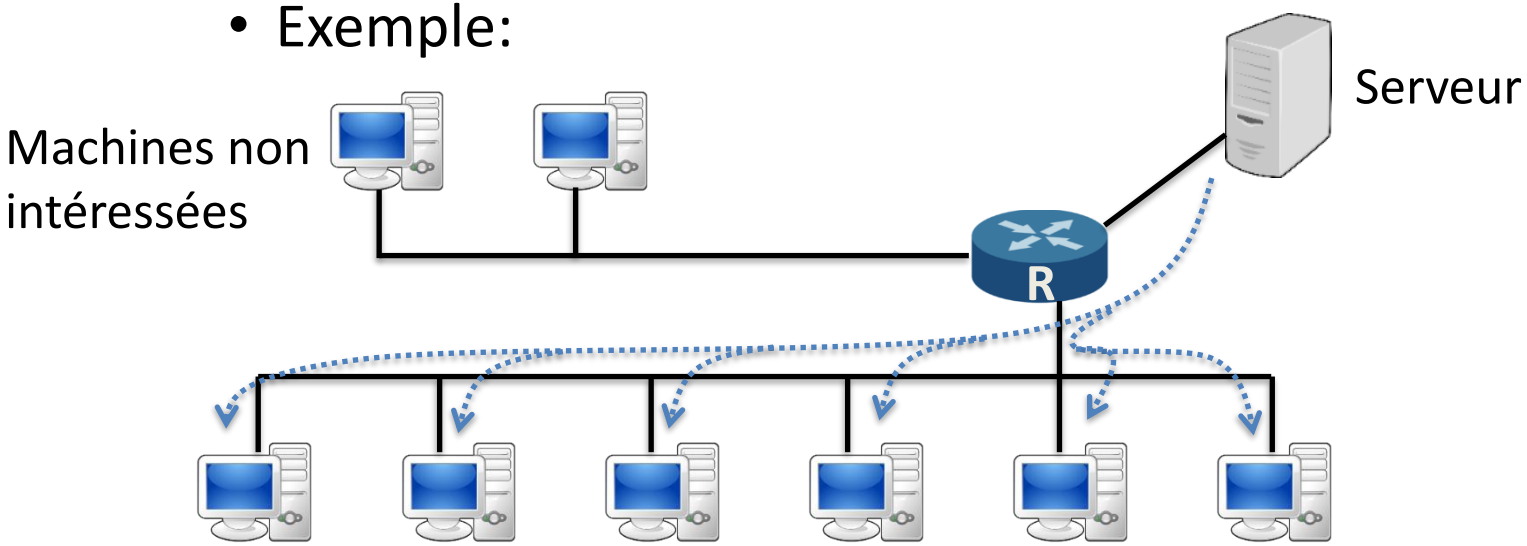
- Intro
- Routage
- **IP**

- Messages échangés
  - **OPEN**: Etablissement de la session BGP entre deux routeurs voisins (identification et authentification). Si le message est accepté, réponse par **KEEPALIVE**
  - **UPDATE:** Message utilisé pour annoncer ou retirer une route (i.e. une destination)
  - **KEEPALIVE:** Utilisé pour maintenir une session BGP active lorsque aucune nouvelle information n'est annoncée. Sert également d'acquit pour le message **OPEN**
  - **NOTIFICATION:** Message de signalisation utilisé pour notifier des problèmes / erreurs. Exemple: session BGP interrompue.

IP (55)

Plan :
- Intro
- Routage
- **IP**

- Le multicast IPv4
  - Idée: Partager une information avec un groupe de receveurs
  - Exemple:



Installation d'un pool de machine: distribution d'une Image disque en multicast

IP (56)

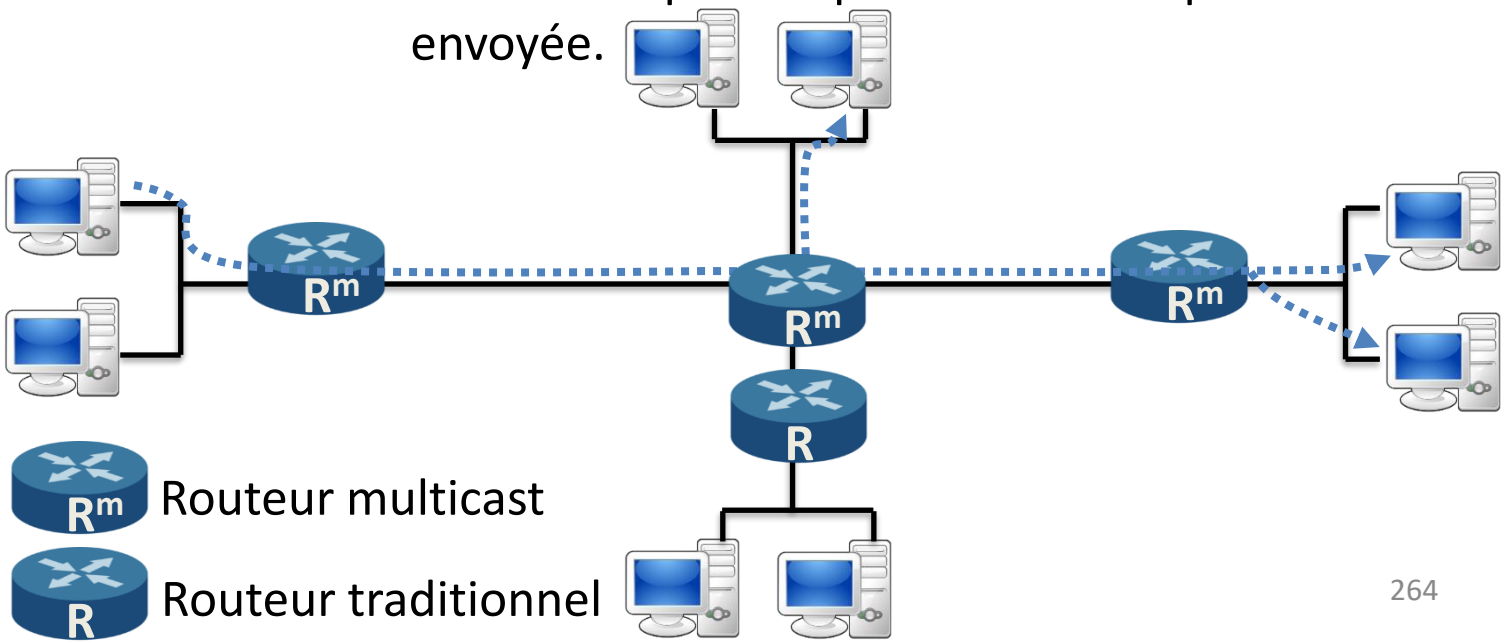
Plan :
- Intro
- Routage
- **IP**

- Mise en place du multicast
  - One-to-all unicast
    - Utilisation de connexions séparées entre l'émetteur et chaque destinataire
    - Utile lorsque la couche transport supporte uniquement l'unicast
    - Il y a autant de connexions qu'il y a de destinataires
  - Application-level multicast
    - Utilisation d'un destinataire pour propager l'information aux suivants
    - La source envoie l'information à un membre du groupe, chargé de la propager.
    - Fonctionne également quand il n'y a pas de support multicast.

IP (57)

Plan :
- Intro
- Routage
- **IP**

- Explicit multicast
  - Si la couche réseau le permet
  - Envoi d'**un seul** paquet qui est répliqué et acheminé à tous les destinataires
  - Qui s'occupe de répliquer un paquet ?
    - » Le routeur qui sait qui est intéressé par l'information envoyée.



SWILABUS

IP (58)

- Choix de la solution
  - La dernière (explicit multicast)
    - » Cette solution propose une utilisation efficace du réseau. Un seul paquet est envoyé à plusieurs destinataires
    - » La couche réseau doit supporter le multicast
  - Le seconde (application level)
    - » Plus efficace que la 1ère solution
    - » L'application doit être construite en conséquence. Pas simple à mettre en œuvre.
  - La première (one-to-all unicast)
    - » Implémentation assez simpliste du multicast
    - » Efficacité mauvaise (un même paquet sera envoyé autant de fois qu'il n'y a de destinataires intéressés.

Plan :

- Intro
- Routage
- **IP**

IP (59)

Plan :
- Intro
- Routage
- **IP**

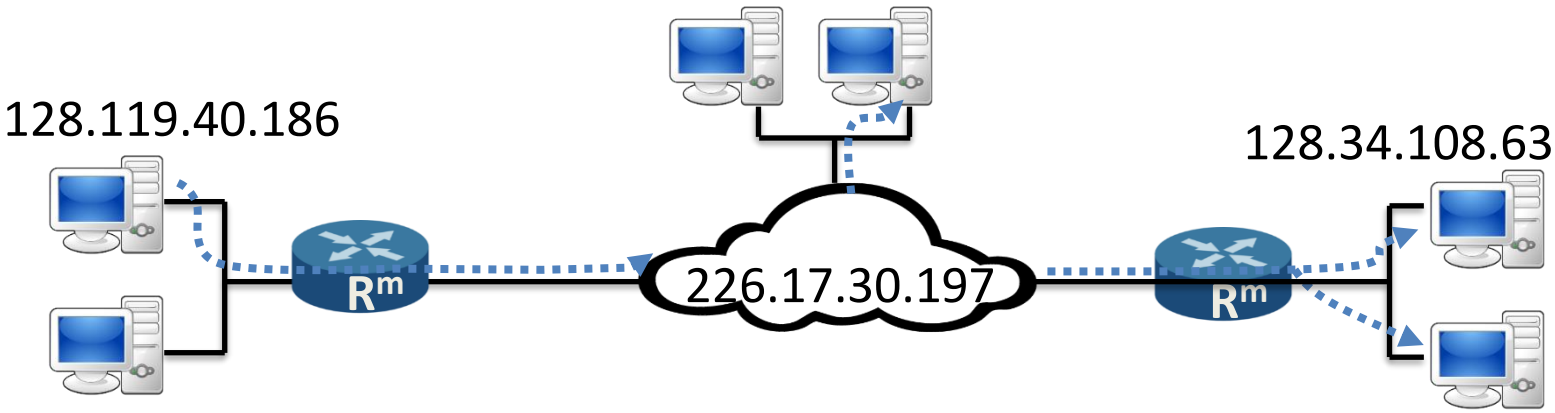
- Problèmes à résoudre
  - Qui est destinataire d'une information multicast ?
    - » Tout le monde
    - » Comment un routeur sait-il qu'une machine est intéressée par recevoir un trafic multicast ?
  - Comment différencier plusieurs informations multicast ?
    - » Si 2 sources transmettent des informations multicast, comment sont-elles différenciées ?
  - Dans le cas *unicast*, l'adresse IP permet de déterminer qui est destinataire d'une information et donc de distinguer deux informations de sources distinctes.
    - » Idée 1: Placer, dans le paquet, toutes les adresses IP des destinataires
      - OK si peu de destinataires
      - Impossible sinon

IP (60)

Plan :
- Intro
- Routage
- **IP**

- Il faut aussi que la source connaissent toutes les adresses des destinataires … pas si évident

» Idée 2: Utiliser un identifiant global pour le groupe (ex: 226.17.30.197)
- Comment choisir l'adresse ?
- Comment démarrer/arrêter un groupe ?
- Comment ajouter une nouvelle machine ?



SWILABUS

IP (61)

Plan :
- Intro
- Routage
- **IP**

– Qui peut joindre le groupe ? Qui fait un contrôle d'accès le cas échéant ?
  » Quelqu'un connaît-il l'identité de tous les membres du groupe ?
  » Comment les routeurs vont-ils coopérer pour délivrer l'information muliticast ?
  » Un protocole particulier existe :
    - IGMP (Internet Group Multicast Protocol)
    - Version 2
    - RFC 2236
      - Il faut noter que IGMPv3 [RFC 3376] existe également, il permet surtout de filtrer les messages en fonction d'une source donnée.

IP (62)

Plan :
- Intro
- Routage
- **IP**

- IGMPv2
  - Opère entre l'ordinateur et le routeur directement connecté à lui
  - Il offre le moyen, pour un hôte, d'avertir le routeur qu'il souhaite recevoir un trafic multicast donné et rejoindre le groupe.
  - Il est uniquement utilisé pour **signaler** au routeur qu'un hôte est (ou n'est plus) intéressé.
  - L'algorithme de routage multicast
    - » Sert à router l'information vers les routeurs intéressés
    - » PIM (Protocol Independant Multicast)
    - » DVMRP (Distance Vector Multicast Routing Protocol
    - » MOSPF (Multicast OSPF)

IP (63)

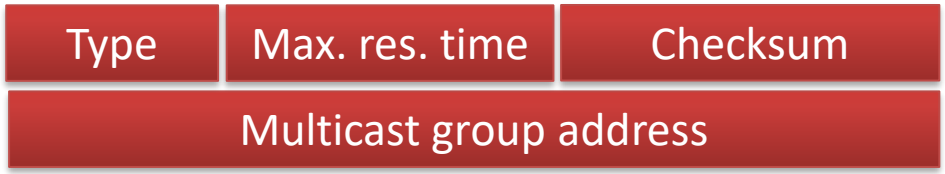
Plan :
- Intro
- Routage
- **IP**

– Fonctionnement:
  » Utilise 3 types de messages :
    - *membership query (router)*: envoyé par un routeur à tous les hôtes pour déterminer quels groupes doivent être joints
    - *membership report (host):* envoyé par un hôte pour indiquer qu'il s'abonne au groupe multicast. Le routeur sait ainsi qu'il doit propager les informations de ce groupe
      - *Optimisation: la machine ne répond pas directement, au cas où une autre répondrait pour le même groupe*
    - *Membership leave (host):* envoyé par un hôte pour indiquer qu'il se désabonne du groupe multicast donné. Le trafic de ce groupe ne doit plus lui être transmis.

SWILABUS

IP (64)

– Format d'un message IGMP



» Numéro du protocole dans le paquet IP: 2

» Le champ type identifie le type de message IGMP reçu / envoyé

» Multicast group address: adresse IP entre 224.0.0.0 et 239.255.255.255 (voir http://www.iana.org/assignments/multicast-addresses/multicast-addresses.xhtml)

» Le champ Max. Res. Time indique le temps d'attente maximum pour la réponse

» Le champ checksum permet de contrôler la cohérence de l'information

Plan :

- Intro
- Routage
- **IP**

SWILABUS

IP (65)

Plan :
- Intro
- Routage
- **IP**

– Contrôle d'accès et sécurité
  » Dans ce modèle, on comprend maintenant comment il est possible de joindre un groupe multicast
  » Qui peut joindre un groupe multicast
    • N'importe qui ! En effet, aucun mécanisme au niveau réseau n'est implémenté pour contrôler l'accès.
  » Qui connaît l'ensemble des destinataires ?
    • Personne ! En effet, il suffit d'indiquer son intérêt pour un groupe pour que le flux soit propagé.
  » Qui peut envoyer des informations multicast ?
    • Tous les membres du groupe
  » Comment assurer un contrôle d'accès ?
    • Doit être réalisé par l'application

IP (66)

Plan :

- Intro
- Routage
- **IP**

- Le routage multicast
  - Nous avons vu comment les hôtes pouvaient marquer leur intérêt pour un groupe.
  - Il faut maintenant étudier comment les routeurs vont pouvoir acheminer ce trafic multicast vers leur destination
  - Il faut déterminer l'ensemble des liens qui connectent tous les routeurs qui ont un hôte impliqué dans le groupe
  - Il faut parfois utiliser des routeurs qui ne sont pas impliqués dans le groupe, uniquement pour faire transiter l'information
  - 2 méthodes:
    - » Utiliser un arbre partagé, connectant tous les membres du groupe
    - » Construire un arbre pour chaque membre.

IP (67)

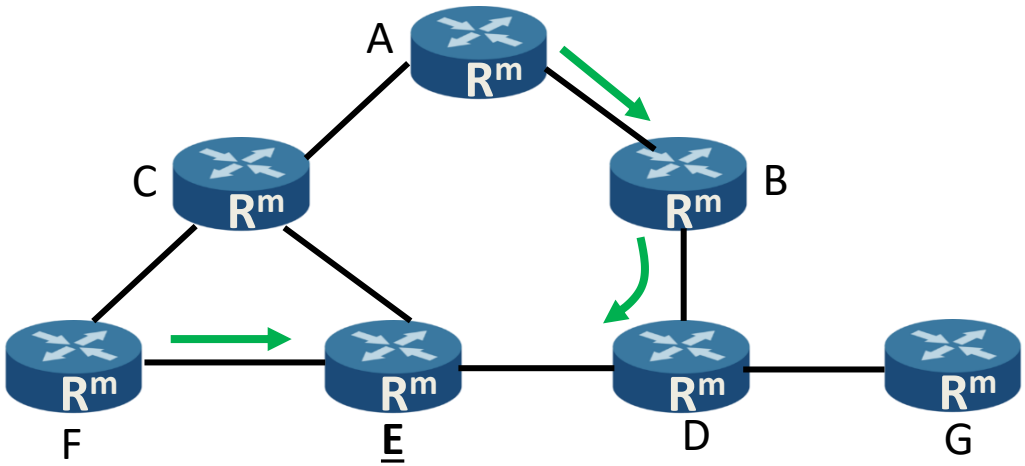
Plan :
- Intro
- Routage
- **IP**

– L'arbre partagé
  » Tous les paquets suivent le chemin partagé
  » Idée 1: il « suffit » de trouver un arbre à l'intérieur du réseau qui relie tous les routeurs nécessaires (ie. Intéressés) et de coût minimum
    - Ce problème est NP-Complet (https://fr.wikipedia.org/wiki/Probl%C3%A8me_de_l'arbre_de_Steiner)
      - Pas de solution en un temps fini
    - Cette méthode est peu utilisée en réseau pour router l'information
      - Il faut disposer d'une cartographie complète
      - Il faut recalculer l'arbre en cas de changement des routes / coûts.

IP (68)

Plan :
- Intro
- Routage
- **IP**

» Idée 2: Le point de rendez-vous
  - On définit un nœud central
  - Chaque membre du groupe envoie un message *join* vers ce nœud
  - Ce message est transmis en utilisant les routes déjà connues
  - L'arbre est augmenté des liens



Exemple: © Pearson Education, 2013 [1]

SWILABUS

IP (69)

Plan :
- Intro
- Routage
- **IP**

– Arbre pour chaque membre
  » Utilisation de RPF (Reverse Path Forwarding)
  » Idée simple:
    - Un routeur qui reçoit un paquet multicast le retransmet sur toutes ses interfaces sauf celle par laquelle le paquet est arrivé
    - **Uniquement** si le paquet arrive par l'interface que ce routeur aurait utilisé pour joindre la source. **Sinon, le paquet est ignoré**
    - Le routeur ne doit pas connaître tout le réseau pour pouvoir envoyer l'information, il doit juste connaître le routeur suivant.
    - Un routeur peut également informer son voisin qu'il n'est plus intéressé par le trafic multicast (message *prune*).

IP (70)

Plan :
- Intro
- Routage
- **IP**

- Le routage multicast dans Internet
  - Le routage multicast nécessite l'utilisation et le déploiement de routeurs multicast
  - Si un routeur multicast est entouré de routeurs non-multicast, cela ne sert pas à grand-chose.
    - » Il est possible d'établir des tunnels pour les échanges multicast, à l'instar d'IPv6
  - Différents algorithmes sont prévus:
    - » DMVRP (Distance Vector Multicast Routing Protocol) – RFC 1075
      - Algorithme implémentant un arbre pour chaque membre avec RPF et le *« pruning »*
      - Il calcule également les routeurs qui sont dépendants de lui. Ainsi, il peut déterminer quand un trafic ne doit plus lui être propagé.

IP (71)

Plan :
- Intro
- Routage
- **IP**

» PIM (Protocol Independent Multicast) – RFC 2362
- 2 modes différents :
  - Mode « dense »: beaucoup de routeurs localisés à un endroit sont impliqués dans le trafic multicast
  - Mode « sparse »: peu de routeurs sont impliqués à cet endroit
- En mode *dense*, il est raisonnable d'utiliser RPF pour transmettre le trafic multicast (avec le *pruning* pour informer son désintérêt).
- En mode *sparse*, le routage par un point de rendez-vous est utilisé (méthode plus centralisée).