Enoncé du TP 2 Réseaux

Adresses IP et routage (partie 1)

C. Pain-Barre

INFO - IUT Aix-en-Provence

version du 18/2/2013

Ce TP est à faire depuis Linux. Démarrer les PC sur Debian.

1 Premiers pas sur l'adressage IP et le routage

Une adresse IP, telle que `139.124.132.69`, désigne un unique ordinateur appartenant à Internet. En l'occurrence il s'agit d'un serveur Web de l'université. Si un ordinateur d'Internet envoie un datagramme IP à destination de `139.124.132.69`, il sera acheminé par les routeurs pour passer de réseau en réseau jusqu'à parvenir à ce serveur.

Pour router un datagramme, les routeurs consultent leur table de routage. Chaque routeur possède sa propre table. Elle contient, pour chaque destination *D* connue, un entrée (*D*, *R*) indiquant l'adresse IP du routeur *R* qui doit servir de relais (*next hop*) pour atteindre *D*. Ce relais *R* et le routeur doivent partager une liaison sur laquelle ils s'échangent des datagrammes. En d'autres termes, ils sont connectés à un réseau commun.

La consultation des tables des routeurs traversés par un datagramme ralentit considérablement son parcours dans l'Internet. Pour qu'elle soit rapide, la décision de routage ne peut pas se prendre sur la totalité de l'adresse de destination. Si tel était le cas, les 32 bits de l'adresse joueraient un rôle et il y aurait potentiellement 2^{32} entrées par table !

Pour réduire la taille des tables et leur consultation, des ensembles de destinations doivent être regroupés pour ne former qu'une entrée. Dans ce cas, seule une partie de l'adresse de destination est utilisée pour trouver une entrée qui corresponde. Cette méthode est utilisée depuis le début, dans le routage historique d'IP dit par classe d'adresse, et s'est affinée avec l'introduction du subnetting puis de l'adressage hors classe (CIDR).

1.1 Routage IP historique et classes d'adresses

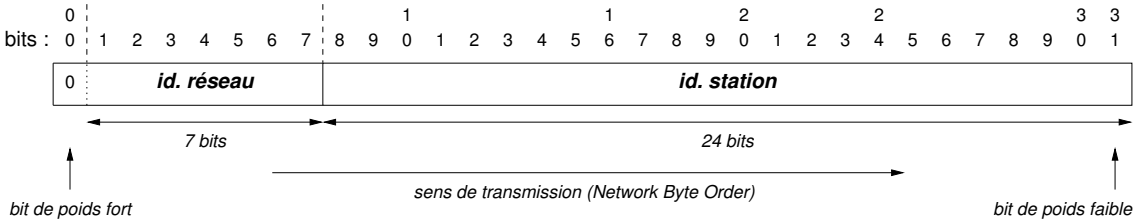
Le routage IP historique se base sur la notion de classes d'adresse IP pour n'utiliser que des **adresses IP de réseau** comme entrées dans les tables. Le principe est qu'un réseau physique, comprenant un ensemble d'hôtes pouvant communiquer entre eux, possède une adresse IP de réseau. Ses hôtes possèdent chacun une adresse IP qui **appartient** à ce réseau. L'adresse IP de réseau représente l'ensemble des adresses IP de ses hôtes et identifie ce réseau dans l'inter-réseau (Internet).

1.1.1 Classes d'adresse

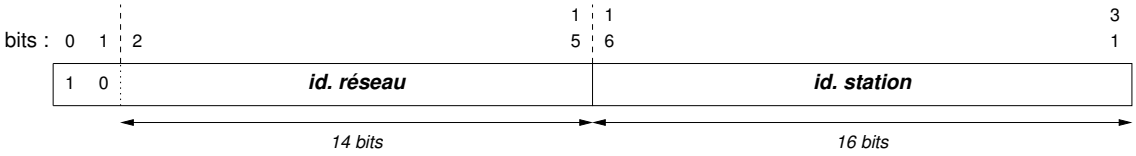
Pour mettre en œuvre ce mécanisme, les adresses IP attribuables aux réseaux/hôtes sont partitionnées en 3 classes : A, B et C. Les classes permettent de tenir compte de la diversité en taille des réseaux. La classe de l'adresse IP affectée à un réseau physique d'un administrateur qui en fait la demande dépend de la taille du réseau et de son évolution prévue.

Le format des classes A, B et C est le suivant :

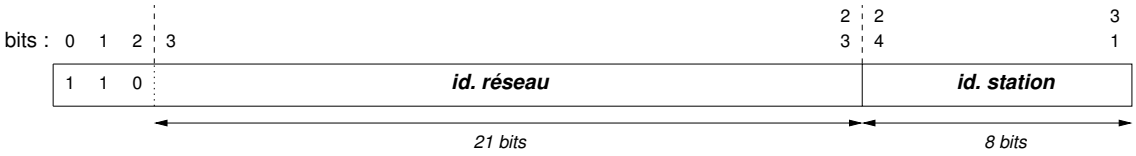
- Classe A : réseaux de très grande taille, peu nombreux



- Classe B : réseaux de taille moyenne, plus fréquents



- Classe C : réseaux de petite taille, typique des PME, très nombreux



Ces adresses contiennent toutes une partie *id. réseau* et *id. station* qui occupent une position dépendant de la classe d'adresse. **La partie id. réseau (comprenant les bits de classe) est fixée par le RIR** auprès duquel l'adresse IP du réseau a été obtenue. **Tous les hôtes du réseau la partagent.** Les hôtes d'un réseau IP se distinguent par la partie *id. station* dont les instances (valeurs des bits) sont attribuées librement par l'administrateur à ses hôtes. Deux valeurs pour *id. station* sont toutefois réservées et ne sont pas attribuables :

- tous les bits à 0 : réservée à l'adresse du réseau lui-même ;
- tout à 1 : réservée à l'adresse de diffusion dans ce réseau.

Exercice 1 (Taille de réseau et classes d'adressage)

L'administrateur d'un réseau de 200 hôtes, dont une extension d'une cinquantaine d'hôtes supplémentaires est prévue, fait une demande auprès d'un RIR pour obtenir une adresse de réseau :

1. Quelle est la classe de l'adresse qu'il doit obtenir ?
2. En reprenant ses fiches prévisionnelles, il se rend compte qu'au final son réseau contiendra 255 hôtes (tout compris). Que peut-il en déduire ?

[Corrigé]

1.1.2 Réseau IP et remise directe

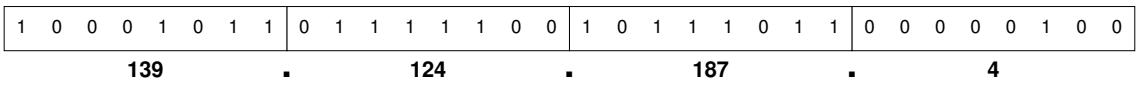
Dans la philosophie IP, un réseau IP regroupe un ensemble d'hôtes (stations/routeurs) situés sur un même réseau physique. Autrement dit, les hôtes d'un réseau IP doivent pouvoir communiquer directement (sans passer par un routeur).

Ainsi, un hôte devant envoyer un message (datagramme IP), peut l'envoyer directement à son destinataire s'il se trouve sur le même réseau. On dit alors que **la destination est directement accessible** et que **la remise est directe**. Si la destination se trouve sur un autre réseau, l'hôte doit confier son message à un routeur afin qu'il soit dirigé vers sa destination. On parle alors de **remise indirecte**.

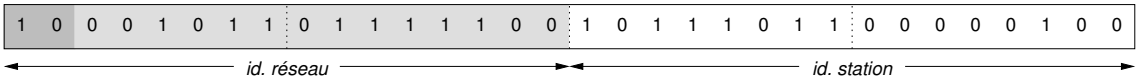
Pour vérifier si la destination appartient au même réseau que lui, l'hôte source compare l'*id. réseau* de sa propre adresse IP à celui de l'IP destination. S'ils correspondent, ils sont sur le même réseau. Cette opération se base sur la classe des adresses.

Exemple

- Soit l'adresse d'allegro :

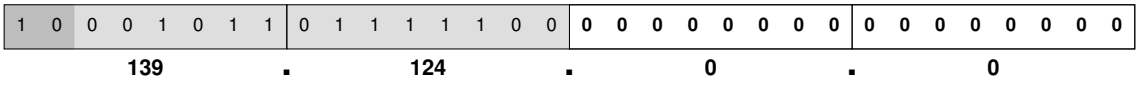


- Des premiers bits du premier octet, on en déduit la classe et donc les bits de l'*id. réseau* et de l'*id. station* :



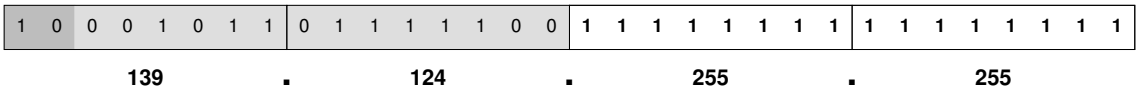
  Ici, les deux premiers bits indiquent qu'il s'agit d'une adresse de classe B. L'*id. réseau* est donc codé sur les 16 premiers bits et l'*id. station* sur les 16 derniers ;

- adresse du réseau d'allegro :



  elle est déduite de l'adresse précédente en mettant à 0 tous les bits de l'*id. station* ;

- adresse de diffusion dans le réseau d'allegro :



  à nouveau déduite de l'adresse de départ (ou de toute autre adresse appartenant à son réseau) en mettant à 1 tous les bits de l'*id. station*.

□

Exercice 2 (Manipulation d'adresse et accessibilité directe)

Sachant que l'hôte de nom SV1 d'un réseau possède l'adresse `175.110.28.82` :

1. quelle est l'adresse IP du réseau de SV1 ?
2. quelle est l'adresse IP de diffusion (dirigée) dans ce réseau ?

> ✍ On rappelle qu'un datagramme envoyé à cette adresse depuis n'importe quel hôte d'Internet est censé être délivré à tous les hôtes du réseau. Il n'y a cependant aucune garantie : le datagramme ne sera délivré à un hôte du réseau que s'il est actif et disponible.
> Aussi, la diffusion peut être interdite/impossible dans le réseau ciblé. Généralement, si elle est possible, elle n'est autorisée qu'en interne (un hôte du réseau en est la source) et les routeurs rejettent (sans envoyer d'erreur) les datagrammes en *broadcast* provenant de l'extérieur.

3. l'hôte possédant l'adresse `175.110.28.50` se trouve-t-il sur le même réseau IP ?
4. l'hôte possédant l'adresse `175.110.23.50` se trouve-t-il sur le même réseau IP ?
5. l'hôte possédant l'adresse `175.104.28.50` se trouve-t-il sur le même réseau IP ?
6. quelle est la plus petite adresse attribuable à un hôte dans le réseau de SV1 ?
7. quelle est la plus grande adresse attribuable à un hôte dans le réseau de SV1 ?

[Corrigé]

1.1.3 Tables de routage historiques

Ainsi qu'il a été dit, le routage historique se fonde sur les adresses de réseaux uniquement. Tout hôte dispose de sa propre table de routage, indiquant comment atteindre les réseaux connus.

Les tables de routage historiques comportent (au minimum) deux colonnes : **destination (réseau)** et **routeur** (ou *next hop*, *gateway* ou passerelle). Chaque entrée (ligne) de la table est un couple (N, R) où N est une adresse de réseau et R est l'adresse IP du routeur menant vers le réseau N.

> **Pour des raisons qui seront vues plus tard avec la résolution d'adresse, les adresses IP des routeurs qui figurent dans la colonne routeur doivent être directement accessibles pour l'hôte possédant la table.**

Pour router un datagramme destiné à une adresse IP D, un routeur cherche dans sa table une entrée (N, R) où N est le réseau de D. Si elle existe, le prochain relais pour ce datagramme sera R. Si aucune entrée n'est trouvée, l'adresse D est inaccessible et un message d'erreur ICMP est renvoyé à la source du datagramme.

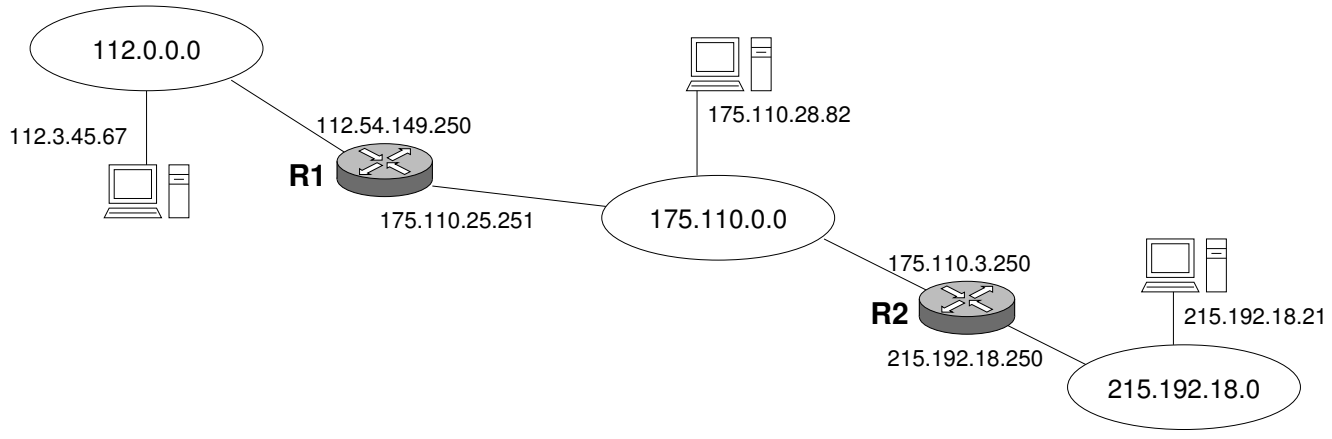
Lors de la configuration IP d'une interface de connexion à un réseau, par exemple une carte Ethernet, l'adresse du réseau est automatiquement ajoutée dans la table comme une destination pour laquelle la colonne routeur contient **`0.0.0.0`** (adresse particulière signifiant "cet ordinateur").

> ✍ Toute entrée dont le routeur est `0.0.0.0` est un réseau directement accessible (remise directe).

> ⓘ Notons que sur certains systèmes, comme Windows, la colonne routeur ne contient pas `0.0.0.0` pour les routes directes mais l'adresse IP de l'hôte (possédant la table) dans le réseau correspondant. C'est un détail d'interprétation par le système d'exploitation (et son module implémentant la couche IP).

Exemple

Soit l'interconnexion de réseaux suivante :



La table de routage de la station `112.3.45.67` devrait être la suivante :

Destination	Routeur
112.0.0.0	0.0.0.0
175.110.0.0	112.54.149.250
215.192.18.0	112.54.149.250

✍ Comme on le voit, l'adresse du routeur R1 utilisée dans la table est celle qu'il possède dans le réseau de la station.

La table du routeur R1 devrait être :

Destination	Routeur
112.0.0.0	0.0.0.0
175.110.0.0	0.0.0.0
215.192.18.0	175.110.3.250

Si la station `112.3.45.67` émet un datagramme à destination de `215.192.18.21`, alors sa table lui dit de le confier à R1 (`112.54.149.250`). Puis, la table de R1 lui dit de le confier à R2. Enfin, la table de R2 lui indique qu'il doit le transmettre à son destinataire (remise directe).

ⓘ Rappelons qu'à chaque passage dans un réseau, le datagramme doit être encapsulé dans une trame (ou autre) de ce réseau.

□

Enfin, il est à noter qu'une table de routage (historique ou pas) ne décrit pas le chemin qui sera suivi par un datagramme. Elle n'en indique qu'une partie : l'adresse du routeur devant servir de relais. La table de ce routeur indiquera une autre partie de ce chemin : l'adresse du routeur devant servir de relais suivant, etc.

✍ Ainsi, le chemin (la route entière suivie) permettant d'atteindre une destination est une information répartie entre les différentes tables des routeurs de l'Internet.

Exercice 3 (Table de routage (historique) de R2)

Écrire la table de routage du routeur R2.

[Corrigé]

1.2 Masques de sous-réseau, agrégation et routage actuel

De nos jours, l'utilisation des sous-réseaux IP (subnetting) et de l'adressage hors classe (CIDR), rendent la notion de classe quelque peu obsolète [1] et modifient la manière dont le test de l'accessibilité directe et le routage sont opérés :

- le subnetting est la technique permettant à plusieurs réseaux physiques —appelés sous-réseaux— de partager une même adresse de réseau. Dans ce cas, la partie (bits) d'une adresse qui est prise en compte pour déterminer l'appartenance à un réseau physique est **plus grande** que la partie *id. réseau* mentionnée par la classe ;
- l'adressage hors classe est la technique permettant de regrouper (agréger) plusieurs adresses de réseaux pour former une seule adresse de (sur-)réseau. Dans ce cas, la partie (bits) d'une adresse qui est prise en compte pour déterminer l'appartenance à un réseau physique est **plus petite** que la partie *id. réseau* mentionnée par la classe.

1. même s'il y est encore fait référence selon le contexte.

ⓘ Ces deux techniques seront développées dans le prochain TP.

Ces techniques font qu'un réseau physique peut avoir une adresse IP dont le découpage *id. réseau – id. station* ne correspond pas à celui d'une classe. Or, il reste primordial pour un hôte de déterminer si une destination est directement accessible (remise directe) ou s'il doit passer par un routeur pour l'atteindre (remise indirecte).

Pour cette raison, en configurant un hôte, on lui affecte une adresse IP à laquelle on associe un **masque de sous-réseau**. Le masque est un entier sur 32 bits indiquant où se situent les parties *id. sous-réseau* et *id. station* dans l'adresse IP qu'on affecte à l'hôte :

- ses bits à 1 indiquent où se trouvent les bits de la partie [2] *id. réseau* ;
- ses bits à 0 indiquent où se trouvent les bits de la partie *id. station*.

✍ Comme une adresse IP, un masque de sous-réseau s'écrit en notation décimale pointée.

1.2.1 Masque de sous-réseau et adresse de réseau

Soit un hôte configuré avec une adresse S et un masque M. En **appliquant** le masque M à S, on détermine l'adresse N du réseau auquel l'hôte appartient. Appliquer le masque signifie opérer un **ET bit-à-bit** entre S et M pour obtenir N. Autrement dit,

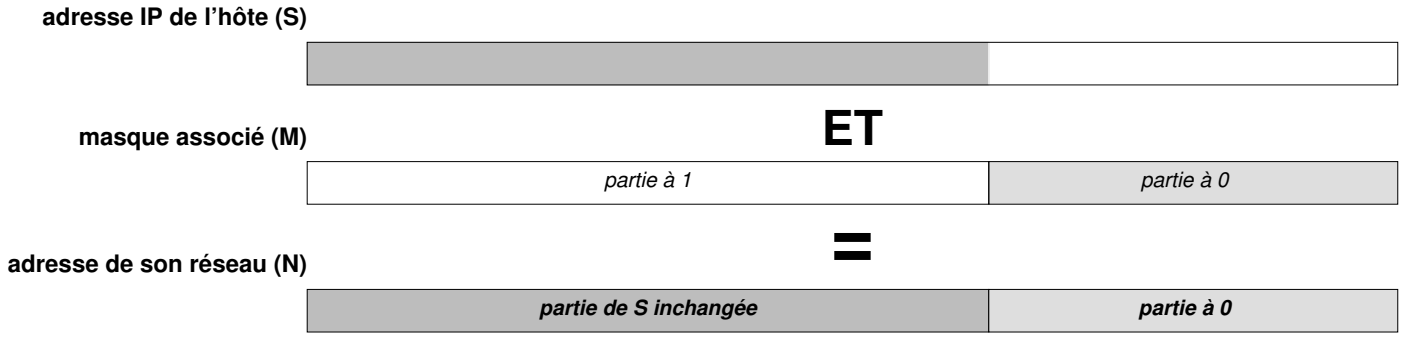
$$N = S \ \& \ M$$

où **&** est l'opérateur du "ET bit-à-bit" en C/C++.

ⓘ Table du ET :

&	0	1
0	0	0
1	0	1

Cette opération laisse inchangée la partie représentant l'*id. réseau* dans S (les bits à 1 du masque) et met à 0 sa partie codant l'*id. station* (les bits à 0 du masque). L'adresse IP obtenue, N, est celle du réseau :



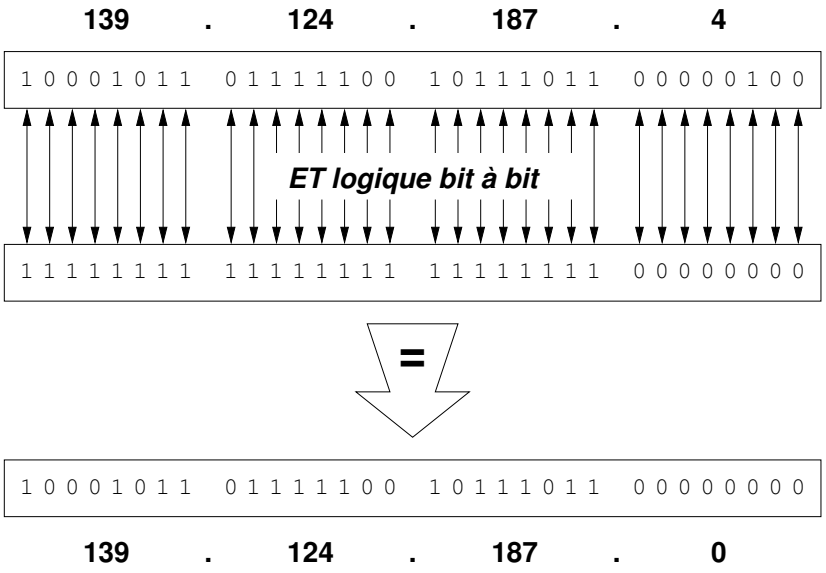
2. dans le cas du subnetting, cela inclut une partie *id. sous-réseau*

Exemple

Soit un hôte dont l'administrateur réseau configure une interface pour le raccorder à un réseau IP ainsi :

- adresse IP : `139.124.187.4`
- masque de sous réseau : `255.255.255.0`

En appliquant le masque à l'adresse, on en déduit que cet hôte appartient au réseau `139.124.187.0` :



□

Exercice 4 (Masques des classes A, B et C)

Supposons que les adresses suivent le format des classes A, B et C. Déterminer, pour chaque classe, le masque à appliquer à une adresse de cette classe pour retrouver l'adresse de son réseau.

> ✍ Ces masques sont les masques par défaut utilisés pour les adresses de ces classes lorsqu'on ne précise pas de masque particulier.

[Corrigé]

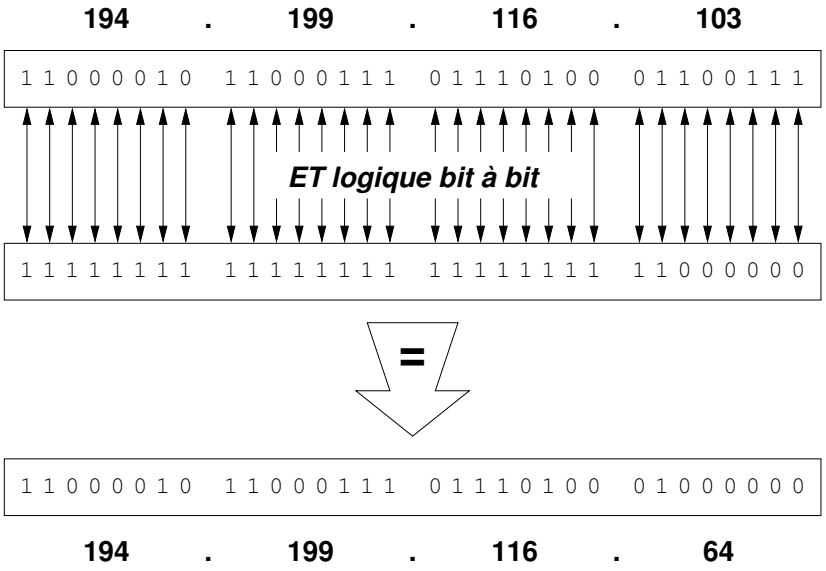
Le masque de l'exemple précédent est assez simple car on voit que l'*id. réseau* est codé sur 3 octets. Ce masque correspondait à celui implicitement utilisé pour la classe C. Toutefois, le nombre de bits à 1 d'un masque peut être quelconque, ce qui se traduit par un masque moins lisible.

Exemple

Soit un hôte dont l'administrateur réseau configure une interface pour le raccorder à un réseau IP ainsi :

- adresse IP : `194.199.116.103`
- masque de sous réseau : `255.255.255.192`

En appliquant le masque à l'adresse, on en déduit que cet hôte appartient au réseau `194.199.116.64` :



□

1.2.2 Notation CIDR pour les masques

Ce que nous devons retenir pour le moment du CIDR (adressage hors classe) est sa notation pour les masques. En CIDR, un masque s'exprime avec */n*, qui indique que l'*id. réseau* est codé sur les *n* premiers bits d'une adresse. C'est une notation pour représenter le masque où les *n* premiers bits sont à 1 et les suivants à 0.

Exercice 5 (Masques et notation CIDR)

Quels sont les masques (en notation décimale pointée) qui correspondent aux notations CIDR suivantes :

1. `/8`
2. `/11`
3. `/21`

[Corrigé]

1.2.3 Adresse de réseau avec masque et remise directe

Un réseau IP est de nos jours identifié par son adresse *N* et son masque *M*. Nous avons vu dans les sections précédentes qu'en appliquant *M* à toute adresse *S* de *N*, on obtient *N*, c'est à dire $N = S \ \& \ M$.

Un hôte configuré avec une adresse IP *S* et son masque associé *M* calcule *N* et est prêt à s'échanger des datagrammes IP avec n'importe quel hôte de *N* car la remise est directe.

S'il doit envoyer un datagramme à une destination *D*, il vérifie si *D* est une adresse de *N* en testant si

$$N == D \ \& \ M$$

Si l'égalité est vraie, alors *D* est une adresse de *N* car sa partie *id. réseau* retenue par *M* est la même que dans *N*. Or, c'est précisément ce qui caractérise les adresses de *N*.
Si l'égalité est fausse, au moins un bit de cette partie dans *D* diffère avec celle de *N*, ce qui suffit pour savoir que ce n'est pas une adresse de *N*, sans pour autant renseigner sur l'adresse réelle du réseau de *D*.

Exercice 6 (Masque, réseau et remise directe)

Soit SV1 un hôte possédant l'adresse IP `175.110.28.82`. Dans chacun des cas suivants :

1. son masque associé est `255.255.255.0`
2. son masque associé est `255.255.240.0`
3. son masque CIDR associé est `/11`

Déterminer :

a) l'adresse IP du réseau de SV1
b) l'adresse IP de diffusion dans son réseau
c) si l'hôte possédant l'adresse `175.110.28.50` se trouve sur le même réseau IP
d) si l'hôte possédant l'adresse `175.110.23.50` se trouve sur le même réseau IP
e) si l'hôte possédant l'adresse `175.104.28.50` se trouve sur le même réseau IP
f) si l'hôte possédant l'adresse `174.110.28.50` se trouve sur le même réseau IP
g) la plus petite adresse attribuable à un hôte dans le réseau de SV1
h) la plus grande adresse attribuable à un hôte dans le réseau de SV1.

[Corrigé]

Un hôte, notamment un routeur, raccordé à plusieurs réseaux par le biais d'interfaces configurées au niveau IP, chacune avec son adresse S_i et son masque M_i, est prêt à communiquer avec les hôtes des réseaux N_i (où $N_i = S_i \ \& \ M_i$).

Lorsqu'il doit envoyer un datagramme à une destination D, il cherche une interface pour laquelle D appartient au réseau N_i (en vérifiant si $N_i == D \ \& \ M_i$). S'il en trouve une, il l'utilise pour envoyer directement le datagramme à D.

Généralement, seul un routeur est ainsi connecté à plusieurs réseaux. Il est alors prêt à router les datagrammes d'un réseau à l'autre.

Exercice 7 (Remise directe et masques de sous-réseaux)

Soit un routeur connecté à quatre réseaux par l'intermédiaire de 4 interfaces :

- la première a pour adresse `139.124.5.250` et pour masque `255.255.255.0`
- la seconde a pour adresse `194.199.10.171` et pour masque `255.255.255.224`
- la troisième d'adresse `194.199.10.82` et pour masque `255.255.255.224`
- la quatrième d'adresse `138.10.5.50` et pour masque (non conseillé) `255.255.0.255`

En déduire les adresses des réseaux auxquels se raccordent ces interfaces, puis déterminer, parmi les destinations suivantes, celles qui lui sont directement accessibles, et par quelle interface :

1. `139.124.20.210`
2. `139.124.5.133`
3. `194.199.10.2`
4. `194.199.10.90`
5. `194.199.10.103`
6. `138.10.5.51`
7. `138.10.6.50`

[Corrigé]

1.2.4 Tables de routage actuelles

Pour tenir compte de ce nouveau partitionnement *id. réseau – id. station*, les tables de routages doivent être modifiées et intégrent désormais une colonne **masque**. Un entrée est alors un triplet (N, M, R) où M est le masque associé à la destination N.

Pour router un datagramme destiné à une adresse IP D, un routeur cherche dans sa table une entrée (N, M, R) telle que $N == D \ \& \ M$. Si elle existe, le prochain relais pour ce datagramme sera R. Si aucune entrée n'est trouvée, la destination D est inaccessible. Si plusieurs entrées correspondent, le routeur choisit celle dont le masque comporte le plus de bits à 1 (car on aura pris en compte une partie plus grande dans l'adresse D).

En configurant (une interface d') un hôte avec une adresse IP S et son masque M, l'entrée représentant la route directe vers son réseau N ($= S \ \& \ M$) est automatiquement ajoutée dans la table :

Destination	Masque	Routeur
...	...	...
N	M	**0.0.0.0**

Exemple

Reprenons l'exemple des tables de routage historiques (section 1.1.3), en considérant que les masques à utiliser sont les masques par défaut. La table de routage de R1 devient :

Destination	Masque	Routeur
112.0.0.0	255.0.0.0	0.0.0.0
175.110.0.0	255.255.0.0	0.0.0.0
215.192.18.0	255.255.255.0	175.110.3.250

□

Exercice 8 (Table de routage actualisée de R2)

Écrire la table de routage du routeur R2.

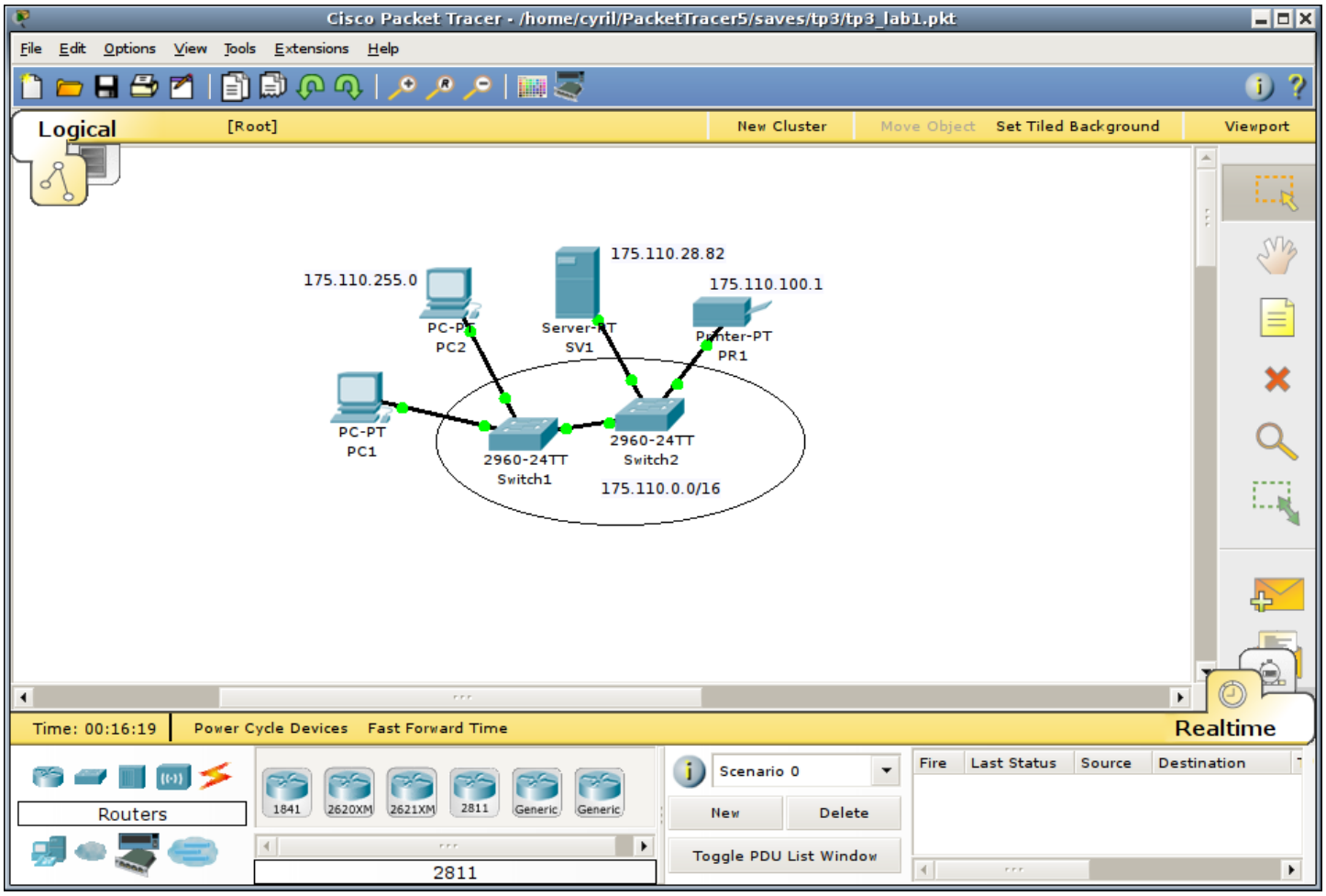
[Corrigé]

2 Simulation du routage avec Packet Tracer

Packet Tracer est un simulateur développé par Cisco, un constructeur renommé de matériels réseaux. Il est disponible gratuitement mais uniquement pour les membres de la Cisco Academy. Nous remercions nos collègues du département R&T pour nous avoir permis de l'utiliser.

2.1 Premiers tests de connectivité et configuration d'une station

Nous allons travailler dans un premier temps sur un tout petit réseau. Cliquer sur le lien `tp2_lab1.pkt` de la page Web `http://infodoc.iut.univ-aix.fr/~cpb/index.php?page=reseaux`, pour télécharger le fichier et l'ouvrir dans Packet Tracer (qu'on appellera PT). **Pour le moment, ne toucher à rien !**
L'interface de PT se présente ainsi :



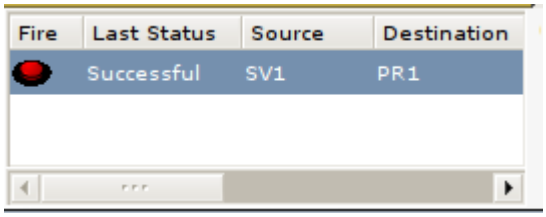
Sur la droite, une barre d'outils contient les boutons suivants, déclenchant quelques actions :

- Bouton pour sélectionner un ou plusieurs objets. Quand il est sélectionné (le défaut) et qu'on laisse un instant la souris devant un matériel, des informations sur sa configuration apparaissent. En cliquant sur le matériel, une fenêtre de configuration de ce matériel apparaît ;
- Bouton pour déplacer un ou plusieurs objets ;
- Permet d'ajouter du texte ;
- Permet de supprimer du texte ou un (groupe d')objet(s) ;


Permet d'inspecter un objet. Si ce bouton est sélectionné, en cliquant sur un objet, on obtient l'état des tables de cet objet (table MAC, table de routage, etc.) ;


Permet de redimensionner un objet (redimensionnable) ;


Permet d'effectuer un **test de connectivité** basique correspondant au PING : envoyer un message (ICMP) depuis un objet vers un autre [3]. Si tout ce passe bien, le message produit l'envoi d'une réponse par son récepteur. Le test est réussi si la réponse parvient correctement à la source du message. Ce test permet de vérifier que la configuration est suffisamment correcte pour que la communication entre ces objets puisse se faire.
En cliquant sur ce bouton, on est invité à cliquer sur l'objet source du message, puis sur l'objet destination. Lorsqu'on le fait, le message est intégré au **scénario** courant et se retrouve dans la fenêtre en bas à droite de l'interface :



Si la source et la destination ont pu correctement s'échanger leur message, *Successful* est écrit dans la colonne *Last Status*, sinon *Failed* apparaît. Dans ce cas, il faut revoir la configuration des matériels impliqués dans cette communication infructueuse.

> Il arrive parfois que le test échoue alors que la configuration est correcte. Il n'y a pas forcément d'explication à ce phénomène ☹. Pour le contourner, cliquer plusieurs fois sur le bouton *Fast Forward Time* en bas de l'interface :
>
> Fast Forward Time
>
> On peut reprendre l'envoi d'un message figurant dans un scénario en cliquant sur le bouton (rouge) de la colonne *Fire*.
> Si le problème persite, on peut inspecter ce qui se passe en passant en mode *Simulation* (voir plus loin).
> En dernier ressort, il y a le bon vieux réflexe de l'informaticien : sauvegarder, quitter puis relancer...


Outil de création de message personnalisé qui permet aussi d'envoyer un message mais dont les caractéristiques sont entrées manuellement. En particulier, ce bouton permet d'envoyer le PDU d'un nombre conséquent de protocoles, pas seulement ICMP.

Exercice 9 (Test de la connectivité)

Le réseau qui a été chargé est formé de 2 switchs et de 4 hôtes : PC1, PC2, SV1 et PR1. Il a pour adresse `175.110.0.0/16`. **La station** PC1 **n'est pas encore configurée au niveau IP** mais PC2, SV1 et PR1 le sont déjà. Leurs adresses sont indiquées sur l'interface comme le montre la figure 1.

3. PING et ICMP seront détaillés plus tard dans le module.

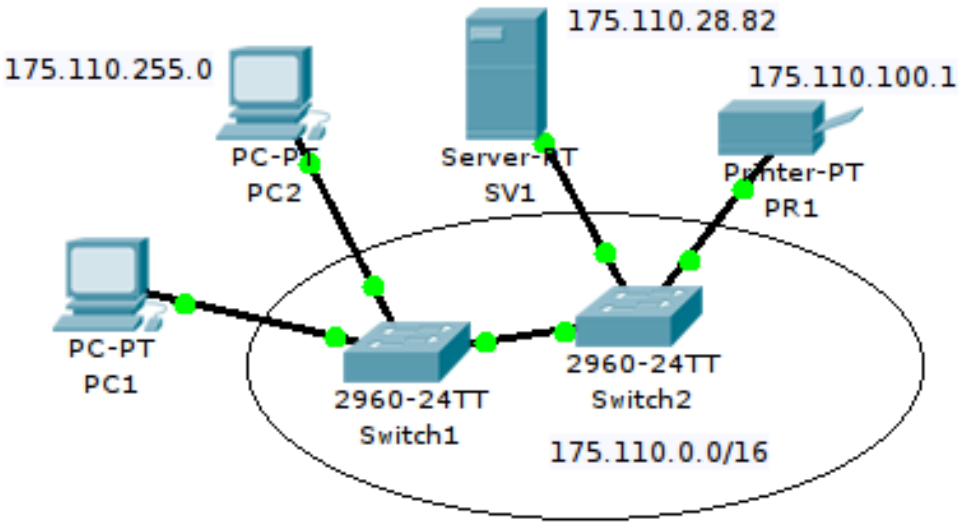
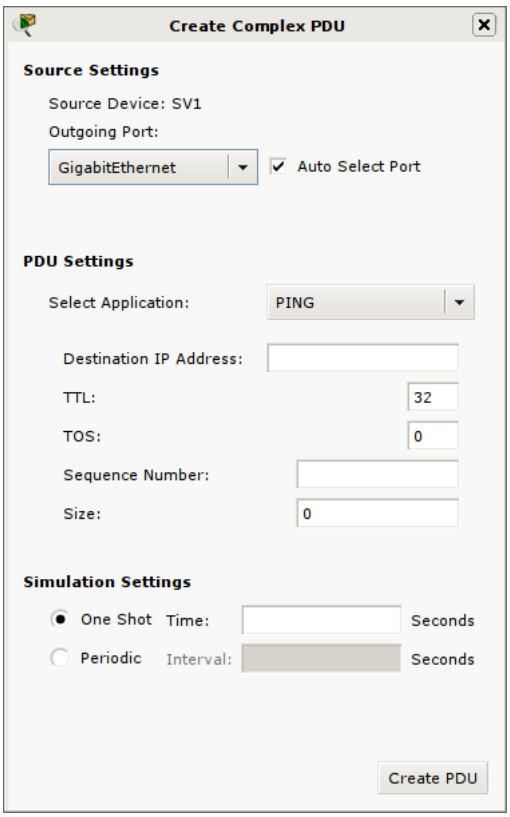


FIGURE 1 – Petit réseau Ethernet du lab1

Sur l'interface, tant que des points orange figurent sur les ports des switchs, aucune trame ne peut être transmise aux hôtes, car cela signifie que les switchs sont en train de communiquer via le *Spanning Tree Protocol* et de découvrir la topologie.

1. Tenter d'échanger un message entre SV1 et PR1. Cela devrait fonctionner : la colonne *Last Status* devrait afficher ***Successful***.
2. Utiliser l'outil de création de message personnalisé afin d'envoyer depuis SV1 un message (PING) dont la destination est l'adresse de diffusion (dirigée) de ce réseau. Pour cela, une fenêtre *Create Complex PDU* s'ouvre après avoir choisi l'émetteur de ce message :



Comme son nom l'indique, cette fenêtre permet de créer le PDU d'un protocole (ou application) au choix (par défaut, un message PING du protocole ICMP), en fixant manuellement des valeurs pour certains champs qu'il comporte.
Renseigner l'adresse de destination puis donner une valeur numérique à la zone *Sequence Number* (exemple 1234) et fixer à 0 la zone *One Shot Time*. Terminer en cliquant sur *Create PDU*.

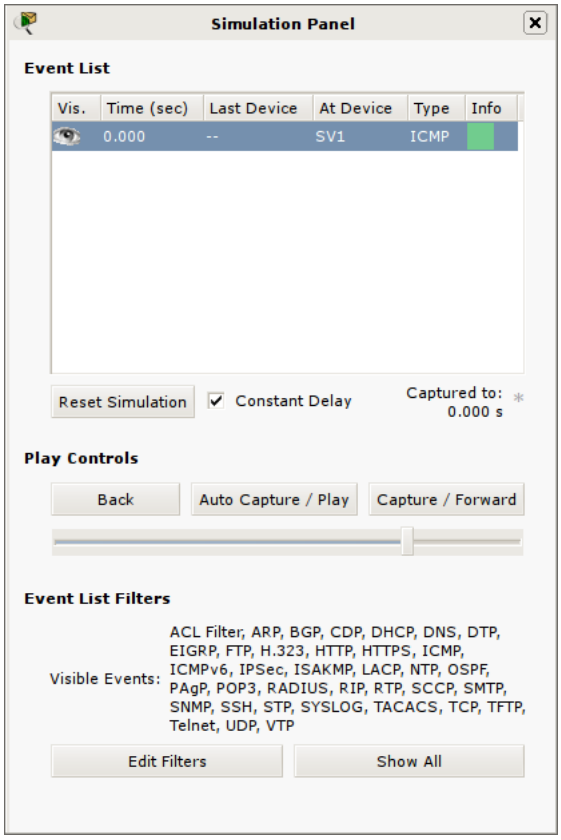
3. En principe, ce message personnalisé devrait apparaître *Successful* sans qu'on ait pu voir ce qui se passait. Ceci par ce que, par défaut, on se trouve dans le mode *Realtime*.
4. Supprimer le premier message **sans supprimer le scénario** en double-cliquant sur *(delete)* dans la colonne à droite dans la zone où il se trouve :

Fire	Last Status	Source	Destination	Type	Color	Time (sec)	Periodic	Num	Edit	Delete
	Successful	SV1	PR1	ICMP		0.000	N	0	(edit)	(delete)
	Successful	SV1	175.110.255.255	ICMP		0.000	N	1	(edit)	(delete)

5. Pour visualiser le trafic, il faut passer en mode *Simulation*. Cliquer sur le bouton derrière l'horloge en bas à droite afin de basculer en mode *Simulation* :

En cliquant à nouveau sur l'horloge, on reviendrait au mode *Realtime*.

6. Dans le mode *Simulation*, un panneau apparaît sur la droite, qui va permettre de tracer les échanges de messages :



> ⓘ Sur l'installation actuelle, ce panneau peut être instable et trembler, en particulier si PT est en plein écran. Dans ce cas, double-cliquer sur la barre du haut du panneau, ce qui le détache de la fenêtre de PT et le stabilise.

Le message que nous voulons tracer est un message ICMP, qui sera traité plus tard en cours. Pour le moment, retenons juste que ce message demande à son récepteur de retourner une réponse (PING suivi d'un PONG).

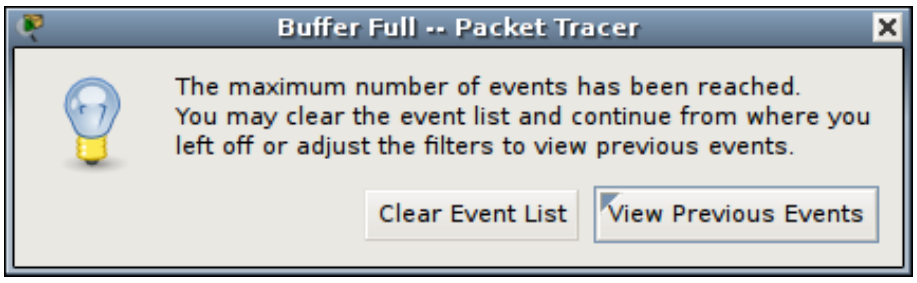
7. Cliquer sur *Auto Capture / Play* afin de dérouler automatiquement la simulation. On peut accélérer la simulation en faisant glisser vers la droite le curseur sous le bouton *Auto Capture / Play*.
   On devrait remarquer que tous les hôtes du réseau répondent à SV1 sauf PC1 qui n'est pas configuré.
   Les messages en erreur apparaissent avec une croix rouge. Les réponses reçues normalement arrivent avec une marque verte.
   Quand SV1 aura reçu toutes les réponses, cliquer à nouveau sur *Auto Capture / Play* pour arrêter la simulation.

> ⓘ Il se peut que d'autres types de message (comme STP, ARP, CDP ou autres) apparaissent sur le panneau et l'interface. Certains, comme ARP, peuvent être consécutifs aux messages qu'on a générés. Nous traiterons ARP plus tard dans le cours. D'autres messages, comme STP ou CDP, sont générés automatiquement pour simuler le fonctionnement des équipements réseaux. Les messages STP sont les BPDU générés par les switchs pour le *Spanning Tree Protocol*.

8. Revenir au mode *Realtime*. Dans la zone où figure le message, double-cliquer sur *(edit)* afin de le modifier. Remplacer la destination par une adresse telle que `139.124.187.4`. Puis, cliquer sur *Apply Changes*. Le message modifié devrait être en état *Failed*.
9. Revenir au mode *Simulation* afin de vérifier ce qu'il s'est passé. Avant de cliquer sur *Auto Capture / Play*, cliquer sur *Edit Filters* et ne laisser sélectionné que ICMP. Enfin, cliquer sur *Auto Capture / Play*.

> ⓘ On peut aussi utiliser le bouton *Capture / Forward* qui déroule la simulation pas à pas. Un clic déroule le traitement d'un événement (réception/envoi d'un message) qui s'arrête au prochain événement (objet réceptionnant/envoyant un message).

Rien ne devrait se passer, si ce n'est qu'un croix rouge devrait apparaître sur le message (sur SV1) ! Cela signifie que le message ne peut être envoyé.

> ⓘ Il se peut qu'une fenêtre d'erreur s'affiche pour informer du dépassement du nombre d'événements. C'est un peu gênant mais pas vraiment grave. Cliquer simplement sur *Clear Event List* :
>
> 
>

10. Dans le panneau de simulation (partie *Event List*), cliquer sur la colone *Info* (sur la ligne du message). Une fenêtre *PDU Information* s'affiche montrant le traitement par les couches réseau de la station SV1. Il devrait être indiqué que la couche 3 n'a pas envoyé le message car elle ne sait pas comment atteindre la destination qui n'est pas directement accessible. En effet, la table de routage de SV1 n'est pas encore configurée (et il n'y a pas de routeur). Fermer cette fenêtre *PDU Information*.
11. Revenir au mode *Realtime* et supprimer le message.

[Corrigé]

Exercice 10 (Configuration de PC1)

Il faut maintenant configurer l'hôte PC1. Nous choissons de lui affecter **la plus petite adresse dans son réseau** :

1. Cliquer sur le bouton de sélection et cliquer sur PC1. Une fenêtre s'ouvre. Dans l'onglet *Config*, cliquer sur l'interface *FastEthernet*. Dans la zone *IP Configuration*, renseigner l'adresse IP et le masque associé (le masque par défaut de la classe devrait être proposé). Fermer la fenêtre de configuration.

   > ⓘ On rappelle qu'en configurant au niveau IP une interface réseau, la route directe vers le réseau est automatiquement ajoutée dans la table.

2. Vérifier la connectivité (PING) entre PC1 et SV1 : le message doit être *Successful*.

[Corrigé]

2.2 Première configuration du routage

Exercice 11 (Configuration du routage sur R2)

Télécharger le fichier `tp2_lab2.pkt` et l'ouvrir dans Packet Tracer (PT). C'est une extension du réseau précédent, correspondant au réseau de la figure 2.

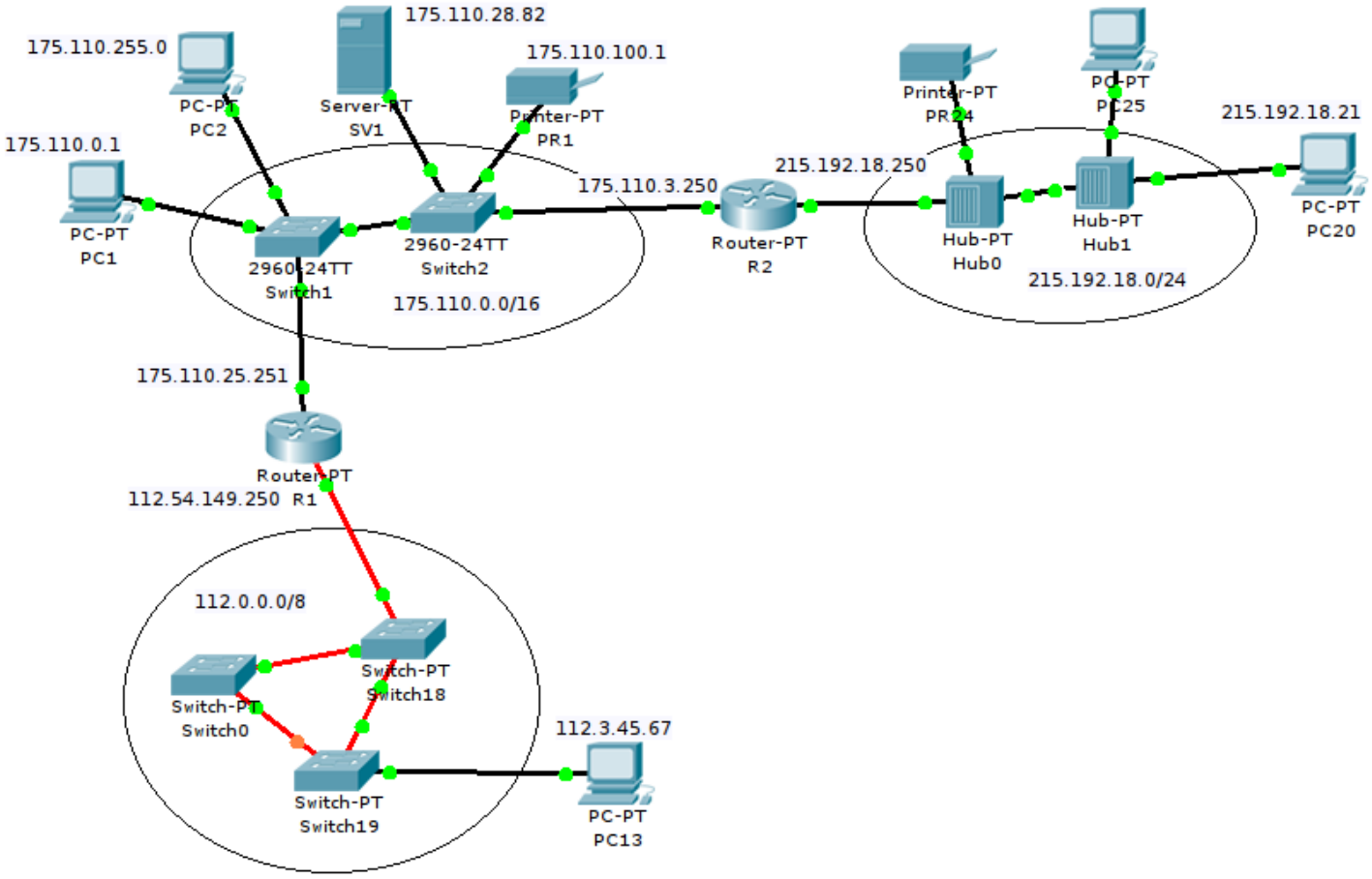


FIGURE 2 – Réseau du lab2, extension du lab1

Dans cet inter-réseau, certains hôtes sont déjà configurés :

- routeur R1 : adresses IP configurées conformément aux labels (textes) sur l'interface, et une route vers le réseau `215.192.18.0/24` via le routeur `175.110.3.250` a été ajoutée dans sa table de routage ;
- à part R2, tous les hôtes du réseau `175.110.0.0/16` ont leur adresse IP configurée, ainsi que PC20 du réseau `215.192.18.0/24` et PC13 du réseau `112.0.0.0/8` ;
- les tables de routage de SV1, PC20 et PC13 sont correctement configurées.

S'assurer d'être en mode *Realtime* et réaliser les opérations suivantes :

1. Envoyer un message depuis PC13 vers SV1. Puisque PC13, R1 et SV1 sont correctement configurés, ce message devrait être *Successful* (insister s'il le faut) ;
2. Envoyer un message depuis PC20 vers SV1. Puisque R2 n'est pas configuré, ce message devrait être *Failed* ;
3. Configurer R2 en lui affectant les adresses figurant sur l'interface (masque par défaut de la classe). Les interfaces d'un routeur apparaissent à l'écran sous des noms réduits. Par exemple, **`fa0/0`** correspond à l'interface **`FastEthernet0/0`** ;

   > ⓘ On peut (dés)activer l'affichage du nom (réduit) des interfaces via le menu *Options* ⟶ *Preferences...*, en (dé)cochant *Always Show Port Labels* dans l'onglet *Interface*.
   > Dans le même genre, l'option *Show Port Labels When Mouse Over* active l'affichage du nom des ports des routeurs/switchs reliés par une liaison sur laquelle on fait passer la souris.

4. Vérifier la connectivité entre R2 et SV1, puis entre R2 et PC20. Les deux messages doivent être *Successful* ;
5. Cliquer sur le bouton d'inspection (la loupe) et visualiser la table de routage (*Routing Table*) de R2. Puisqu'on n'a pas encore ajouté d'entrée manuellement, elle ne contient que les routes directes (type C pour *Connected*) qui ont été automatiquement ajoutées lors de la configuration de ses adresses IP ;
6. Visualiser la table de routage de R1. Elle devrait être :

   Routing Table for R1

Type	Network	Port	Next Hop IP	Metric
C	112.0.0.0/8	GigabitEthernet4/0	---	0/0
C	175.110.0.0/16	FastEthernet0/0	---	0/0
S	215.192.18.0/24	---	175.110.3.250	1/0

   Comme la route montrée en surbrillance ci-dessus, les routes de type S (pour *Static*) sont celles ajoutées manuellement.
7. Modifier la table de routage de R2 afin d'ajouter la route vers le réseau `112.0.0.0`. Pour cela, dans l'onglet *Config* de la configuration du routeur, choisir *Routing* → *Static* et ajouter la route en renseignant les champs *Network*, *Mask* et *Next Hop* (Routeur) puis en cliquant sur *Add* ;
8. Vérifier la connectivité entre PC20 et SV1, ce qui permet de vérifier que R2 assure correctement son rôle de routeur entre ses réseaux directement accessibles ;
9. Vérifier la connectivité entre R2 et PC13, ce qui permet de vérifier que la route vers le réseau `112.0.0.0` est correctement configurée sur R2 ;
10. Vérifier la connectivité entre PC20 et PC13.

[Corrigé]

2.3 Routes par défaut

Bien souvent, les stations connectées à Internet n'ont qu'un seul routeur à disposition. Tous les datagrammes émis par une telle station, à destination de l'extérieur de son réseau passent par le routeur du réseau. Ainsi, toutes les entrées de sa table de routage, sauf celle de la remise directe, indiquent la même adresse de routeur. Ce devrait être le cas de PC13 (`112.3.45.67`). Pour les stations du réseau `112.0.0.0`, R1 joue le rôle de **routeur par défaut**. Peu importe la destination, elles doivent passer par lui. De façon analogue, R2 devrait jouer le rôle de routeur par défaut pour les hôtes du réseau `215.192.18.0`.

C'est pourquoi, IP accepte la spécification d'une **route par défaut**. Elle est employée lorsqu'aucune entrée correspondant à la destination d'un datagramme n'est trouvée dans la table.

En utilisant les masques, il est très aisé d'indiquer une route par défaut : puisque la destination n'a pas d'importance, aucun bit n'a besoin d'être comparé : il faut tous les ignorer.

✍ La route par défaut en passant par un routeur d'adresse *R* s'exprime ainsi simplement dans la table de routage par l'entrée :

(**0.0.0.0**, **0.0.0.0**, *R*)

La table de PC13 (`112.3.45.67`), qui utilise R1 comme routeur par défaut est alors :

Destination	Masque	Routeur
112.0.0.0	255.0.0.0	0.0.0.0
0.0.0.0	0.0.0.0	112.54.149.250

On remarque que plusieurs routes peuvent correspondre pour une destination : **c'est celle dont le masque comporte le plus de bits à 1 qui sera retenue**. Par exemple, on préférera la destination `112.0.0.0` à la route par défaut pour l'adresse `112.1.2.3`.

✍ Sur ce simulateur, les stations ne peuvent utiliser qu'un seul routeur qui est donc le routeur par défaut (appelé *gateway* ou passerelle). Les stations du réseau `175.110.0.0/16` doivent s'en remettre à un unique routeur. Celui choisi est R1. L'inconvénient est que les messages envoyés aux hôtes du réseau `215.192.18.0/24` doivent forcément passer par R1 alors que le chemin le plus court est de s'adresser à R2.
En principe, dans ce cas R1 doit s'en rendre compte et demander aux stations qui le sollicitent à la place de R2 d'utiliser R2 la prochaine fois (redirection de route). Les stations doivent alors modifier leur table. Ainsi, même en utilisant un routeur par défaut, les stations peuvent "apprendre" des routes plus judicieuses. Malheureusement, ce mécanisme est absent du simulateur.

Les avantages de la route par défaut sont multiples :

- le nombre d'entrées des tables peut être considérablement réduit : de plusieurs millions à une seule ;
- les évolutions de l'Internet, extérieures à l'environnement d'une station utilisant une route par défaut, nécessitent rarement la mise à jour de sa table de routage, voire jamais si elle ne dispose que d'un routeur. Si les routes par défaut n'existaient pas, tout ajout ou suppression de réseau dans l'Internet doit donner lieu à une mise à jour de la table de tous les hôtes d'Internet.

Mais il y aussi quelques inconvénients :

- des datagrammes dont la destination ne correspond à aucun réseau peuvent se trouver à boucler entre des routeurs utilisant des routes par défaut, jusqu'à ce que leur TTL [4] arrive à 0, ce qui est très inefficace ;

4. Durée de vie d'un datagramme IP.

- on ne peut plus interdire les communications avec des réseaux sans utiliser de firewall (logiciel actif sur un routeur, filtrant les datagrammes passant par lui) ou d'autres facilités. En effet, sans route par défaut, tous les réseaux qui ne sont pas mentionnés dans la table de routage ne peuvent être atteints. La communication avec les stations de ces réseaux est donc impossible. Cela peut s'avérer pratique pour sécuriser des ordinateurs. En présence de route par défaut, ce n'est pas possible de cette façon.

> ⓘ Notons tout de même qu'il est aisé sur un Unix d'exprimer des destinations interdites dans les tables de routages en utilisant l'option **reject** de la commande **route**. Mais les administrateurs réseau préféreront (à juste titre) utiliser un firewall plutôt que cette possibilité...

Exercice 12 (Routes par défaut sur R2 et PC25)

1. Sur R2, remplacer la route vers le réseau `112.0.0.0` par une route par défaut passant par R1 ;
2. Configurer PC25 en lui affectant **la plus petite adresse disponible** dans son réseau et indiquer le routeur R2 comme routeur par défaut. Cette dernière configuration se fait dans l'onglet *Config*, rubrique *Settings* dans la zone *Gateway* ;
3. Vérifier la connectivité entre PC25 et PC20, pour vérifier la bonne configuration IP de PC25 ;
4. Vérifier la connectivité entre R2 et PC13, pour vérifier la bonne modification de la route de R2 ;
5. Vérifier la connectivité entre PC25 et SV1, pour vérifier la route par défaut de PC25 ;
6. Vérifier la connectivité entre PC25 et PC13 ;
7. Supprimer tous les messages du scénario courant sauf celui concernant l'échange PC25 — SV1
8. Passer en mode *Simulation* puis cliquer sur *Auto Capture / Play*. Observer ce qu'il se passe : la réponse envoyée par SV1 passe par R1 pour revenir vers R2. Clairement, ce n'est pas efficace (mais on ne peut rien y faire sur ce simulateur).

[Corrigé]

2.4 Encapsulation des datagrammes dans des trames

Qu'un datagramme soit transmis à un routeur ou à la destination finale, il est toujours encapsulé dans une trame (ou un paquet) du réseau (physique) sur lequel s'appuie IP. Cette trame est envoyée à l'adresse physique du routeur ou de la destination finale, correspondant à l'adresse de son interface sur le réseau utilisé. Celle-ci est déterminée en effectuant une **résolution d'adresse**, à partir de l'adresse IP du destinataire de la trame. La résolution d'adresse sera traitée prochainement en cours.

Exercice 13 (Adresses physiques et adresses IP)

Les trois réseaux présents sont des réseaux Ethernet. On peut connaître les adresses MAC (appelées aussi adresses physiques) des interfaces (cartes Ethernet) :

- soit en laissant la souris devant un objet : ce renseignement finit par apparaître ;
- soit en cliquant sur un objet et dans l'onglet *Config*, et en cliquant sur une interface. Dans ce cas, on peut même modifier son adresse MAC.

> Les adresses MAC sont présentées différemment d'un système à un autre. Voici 3 formats d'affichage différents pour la même adresse :
>
> - Unix : `08:00:57:f5:8d:01`
> - Windows : `08-00-57-f5-8d-01`
> - Cisco : `0800.57F5.8D01`

Notamment, l'adresse MAC de l'interface de PC13 est `0002.1672.1783`.

1. Retrouver les adresses MAC des interfaces utilisées de R1, R2 et de PC20, ainsi que les adresses IP associées à ces interfaces ;
2. Trois trames (hors celles transportant des éventuels datagrammes ARP) sont nécessaires pour acheminer un datagramme émis par PC13 à destination de PC20. Pour les deux premières trames uniquement, remplir le tableau suivant pour indiquer les adresses MAC figurant dans les trames, ainsi que les adresses figurant dans les datagrammes IP que ces trames encapsulent :

Numéro	Adresses de la trame		Adresses du datagramme	
Trame	MAC Source	MAC Destination	IP Source	IP Destination
1				
2				

Vous pouvez utiliser le mode *Simulation* pour vérifier votre réponse, en inspectant le traitement par les couches 2 et 3 des messages transmis.

[Corrigé]

2.5 Configurations sur un réseau plus conséquent

Télécharger le fichier `tp2_lab3.pkt`, où le réseau précédent a évolué comme le montre la figure 3. On peut remarquer que de nouveaux réseaux, hôtes et routeurs ont été ajoutés. Certaines configurations sont déjà (partiellement) faites et devront être complétées dans le prochain exercice :

- dans le réseau `215.192.18.0/24` : les hôtes PC20 et PC25 ont leur configuration IP et utilisent R2 comme routeur par défaut ;
- R2 est configuré et utilise R1 comme routeur par défaut ;
- dans le réseau `175.110.0.0/16` : tous les hôtes ont leur configuration IP et utilisent R1 comme routeur par défaut ;
- dans le réseau `198.199.0.0/24` : PC30 a sa configuration IP mais pas de routeur par défaut ;
- R1 n'est pas configuré dans le réseau `198.199.0.0/24`. Son interface dans ce réseau est même désactivée et il faudra l'activer. À part les routes directes, il ne connaît que la route vers `215.192.18.0/24` ;
- dans le réseau `112.0.0.0/8` : PC13 a sa configuration IP et utilise R1 comme routeur par défaut ;
- dans le réseau `205.254.1.0/24` : PC40 a sa configuration IP mais pas de routeur par défaut ;
- dans le réseau `205.254.133.0/24` : PC50 a sa configuration IP et utilise R3 comme routeur par défaut ;
- R3 a toutes ses interfaces désactivées et aucune configuration ;
- le réseau `117.0.0.0/8` est un WAN de type *Frame Relay* sur lequel seuls 3 routeurs sont connectés (R4-Paris, R0-Toulouse et R5-Marseille) ;

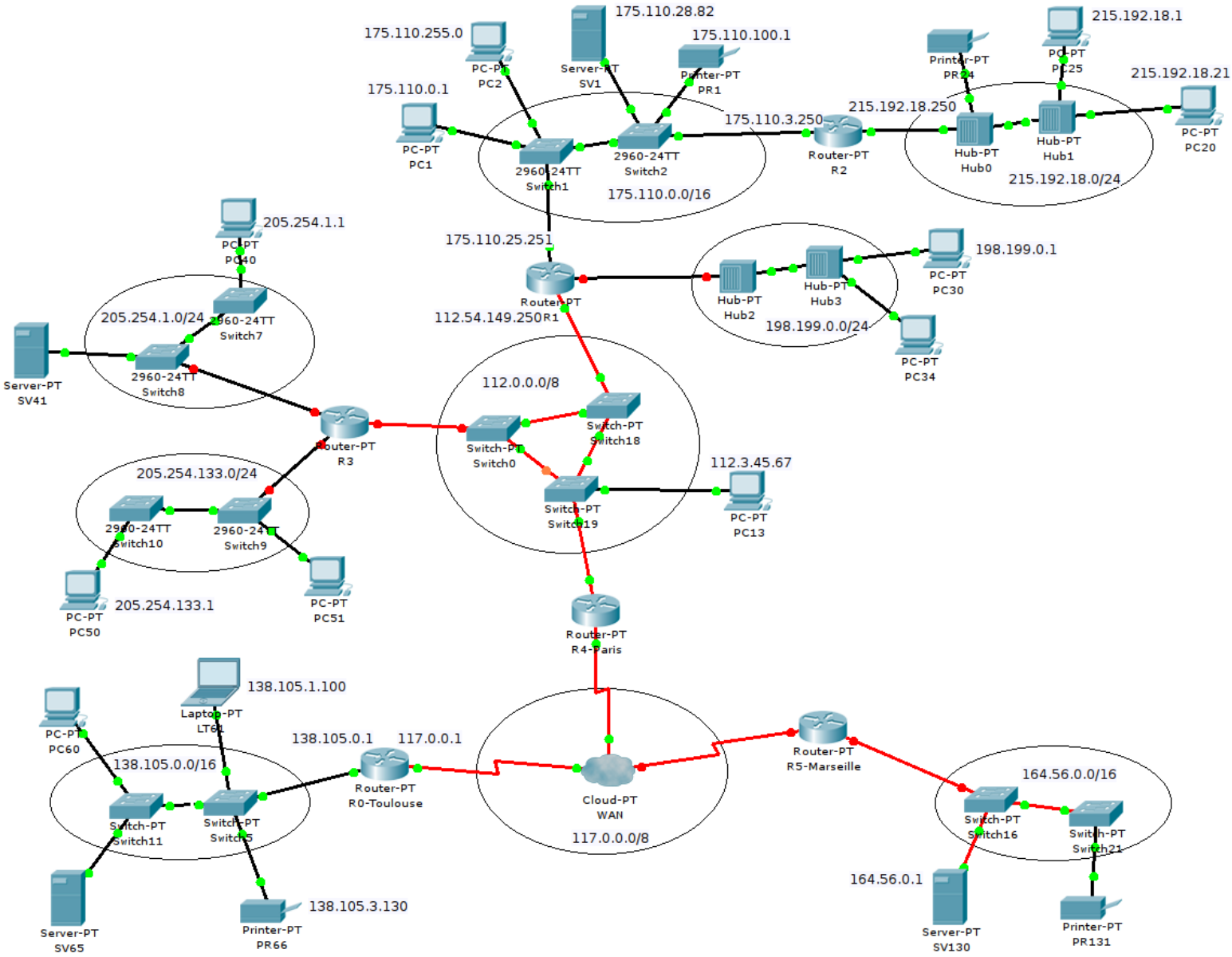


FIGURE 3 – Topologie du lab3, extension du réseau précédent

- dans le réseau `138.105.0.0/16` : LT61 et PR66 sont configurés et utilisent R0-Toulouse comme routeur par défaut ;
- R0-Toulouse a ses interfaces configurées, possède une route vers `164.56.0.0/16` via R5-Marseille, et utilise R4-Paris comme routeur par défaut ;
- dans le réseau `164.56.0.0/16` : SV130 est configuré au niveau IP mais n'a pas de routeur par défaut ;
- R5-Marseille n'est pas configuré et a ses interfaces désactivées.

> ✍ Pour activer une interface d'un routeur, il faut cliquer dessus et dans l'onglet *Config*, choisir l'interface puis cocher la case *Port Status*. Pour savoir quelle interface activer, se référer aux labels affichés. S'ils ne le sont pas, aller dans le menu *Options* ⟶ *Preferences...*, et cocher *Always Show Port Labels* dans l'onglet *Interface*.

Exercice 14 (Extension du réseau)

Configurer correctement ce réseau pour tenir compte de son extension en suivant les instructions suivantes **sans modifier les adresses qui ont déjà été configurées** :

1. Sur les routeurs R1, R3, R4-Paris et R5-Marseille, activer les interfaces qui doivent l'être et, pour celles sans adresse IP, leur affecter la plus **grande adresse disponible dans leur réseau**. Si plusieurs routeurs sont connectés à un réseau, attribuer la plus grande adresse au routeur portant le plus grand numéro. Indiquer ces adresses sur le simulateur en utilisant le bouton d'ajout de texte ;

   > **Les préconfigurations actuelles des matériels ne seront correctes que si vous respectez scrupuleusement les indications pour l'affectation des adresses des routeurs. Le réseau ne fonctionnera pas correctement sinon.**

2. Vérifier la connectivité des routeurs modifiés avec les hôtes déjà configurés des réseaux auxquels ils sont directement reliés ;
3. Attribuer aux hôtes PC34, SV41, PC51 et PR131 la plus petite adresse IP **disponible** dans leur réseau. Indiquer ces adresses sur PT.
4. Vérifier la connectivité de ces hôtes avec les hôtes déjà configurés de leur réseau ;
5. Sur papier dans un premier temps, écrire les tables de routage, sans les routes directes (qui sont/seront ajoutées automatiquement), que devraient avoir tous les routeurs. Utiliser des routes par défaut autant que possible. S'assurer de la cohérence des routes car toutes les destinations doivent être accessibles à partir de n'importe quel routeur/station. Les routeurs ne doivent pas se renvoyer un datagramme à cause de tables mal élaborées.
   De plus, les datagrammes dont l'adresse destination n'appartient pas à une adresse de réseau existante doivent être rapidement détruits. Une solution pour cela est d'utiliser un réseau fédérateur (*backbone*, ou *épine dorsale*), central, contenant au moins un routeur ayant connaissance de l'ensemble du réseau et n'utilisant pas de route par défaut. Les autres routeurs peuvent alors s'appuyer sur ce routeur.
   Le réseau désigné comme réseau fédérateur est le réseau *GigabitEthernet* `112.0.0.0/8`. Configurer R4-Paris pour connaître toutes les routes, bien qu'il serait plus efficace que tous les routeurs du *backbone* en aient aussi connaissance.

   > Un *backbone* est simplement un réseau qui sert d'artère de communication pour connecter des réseaux. Celui de l'exercice a la bizzarerie de comporter une station (`112.3.45.67`), ce qui n'est pas le cas en pratique pour ce type de réseau.

6. Une fois les tables établies, modifier en conséquence dans PT les tables des routeurs qui doivent être revues ;
7. Vérifier la connectivité des routeurs entre eux, d'abord avec les routeurs les plus proches, puis avec les plus éloignés ;
8. Vérifier la connectivité des routeurs avec PC13, SV1 et PC20 ;
9. Configurer le routeur par défaut (*Gateway*) adéquat sur les hôtes PC34, SV41, PC51, PR131 ;
10. Vérifier la connectivité de ces hôtes avec PC13, SV1 et PC20 ;
11. Vérifier la connectivité depuis PC20 vers LT60 puis vers PR131 ;
12. Supprimer le scénario courant puis créer un message ICMP personnalisé (*Sequence Number*=4567, *One Shot Time*=0) depuis PC1 vers l'adresse `139.124.187.4`. Ce message doit être *Failed* ;
13. Passer en mode *Simulation* et vérifier que le message s'arrête [5] à R4-Paris (s'il est le seul à connaître toutes les routes) et ne tourne pas indéfiniment, sinon revoir la configuration des tables.

[Corrigé]

5. Plus exactement, R4-Paris renvoie un message d'erreur à PC1 car il ne connaît pas la destination.