République Algérienne Démocratique et Populaire

Ministère de l'enseignement superieur et de la recherche scientifique

Université MOULOUD MAMMERI DE TIZI OUZOU

Faculté de Génie Electrique et d'Informatique

Département d'Electronique.

Mémoire de fin d'études

En vue de l'obtention du Diolôme de MASTER II en Electronique.

Spécialité : Réseaux Télécommunication.

THEME :

Etude et configuration des routeurs en utilisant la méthode OSPF

Promoteur :

Mr: **LAHDIR Mourad**

Proposé par :

Mr **LAHDIRI Toufik**

Présenté par :

Mlle. BERSI Tassadit

Promotion 2013/2014.

Remerciements

Nous tenons à remercier en premier lieu « Dieu » le tout puissant, qui nous a donné le courage et la volonté pour bien mener ce modeste travail.

Nous tenons tous d'abord a remercie notre promoteur ***Mr. LAHDIR. M*** *pour sa disponibilité et son aide tout au long de ce travail.*

On est profondément reconnaissant à ***Mr LAHDIRI. T*** *pour sa disponibilité et son aide.*

Que tous ceux qui ont contribué de prés ou de loin à la réalisation de ce modeste travail trouvent ici l'expression de notre sincère gratitude.

Dédicaces

Je dédie ce modeste travail à mes chers parents.
Mes frères (H. A. M. Y. S. K. A.) Mes sœurs (F. N.).
A mon mari Rabah.
A toute ma famille et ma belle-famille.
A tous mes amis sans exception.
A tous ceux que j'aime et qui m'aiment.

Bersi Tassadit

Liste des abriviations

ARP : Adress Resolution Protocol.

DNS: Domain Name System.

DTP: Data Transfer Process.

FTP: File Transfer Protocol.

HTTP: Hyper Text Transfer.

ICMP: Internet Control Message Protocol.

IHL: Internet Header Lengh.

IGMP: Internet Groupe Management Protocol.

IP : Internet Protocol .

ISO : Internationale Standar Oganisation.

MBZ : Must Be Zero.

NVT :Network Virtual Terminal.

OSI: Open Systeme Interconnexion.

PDU : Protocol Data Unit.

PPP :Point To Point Protocol.

RARP: Reverse Address Resolution Protocol.

RFC: Requet For Comments.

SDU: Service Data Unit .

SMTP: Simple Mail Transfer Protocol.

TCP: Transmission Control Protocol.

TFTP: Trivial File Transfer Protocol

Liste des figures

Sommaire

Chapitre III : Les protocoles de Routage.

Introduction Générale

Introduction :

Les technologies de la télécommunication constituent aujourd'hui le principal vecteur de changement au monde car elles contribuent à créer un univers dans lequel les frontières, les distances et les limites physiques perdent de leur importance. Internet, le plus grand réseau mondial, est en extension continue. La stabilité de ce dernier est due à l'implémentation de technologies de routage avancées au sein de son architecture.

GNS3 est un émulateur qui permet de simuler ces réseaux informatiques d'une manière très proche de la façon dont les réseaux réels procèdent. Nous l'utiliserons au cours de notre étude pour mettre en pratique tout ce que nous aurons apprit. Nous l'avons choisit car relativement simple à utiliser, avec une interface précise et sans avoir besoin de matériel réseau tels que des routeurs ou des commutateurs.

Dans le premier Chapitre de notre projet, nous avons procédé à une étude préliminaire sur les réseaux en générale, les différents concepts des modèles en couche OSI et le TCP/IP, pour assimiler le concept d'interaction entre les différents périphériques d'un réseau informatique.

Le deuxième chapitre, nous l'avons consacré à l'étude des différents protocoles de transport et d'application en IP, ainsi que les différents protocoles de routage et commutation.

Le troisième chapitre est consacré à la présentation du logiciel GNS3 avec lequel nous avons crée un réseau virtuel reliant plusieurs PC à des routeurs, nous les avons configuré avec le protocole OSPF et avons fait passé des testes avec différentes invites de commandes afin de vérifier le bon fonctionnement de ce réseau.

Nous avons terminé ce projet par une conclusion.

Chapitre I :

Généralités sur les Réseaux Informatique.

Introduction :

Pour crée un réseau, il faut utiliser un grand nombre de composants matériels et logiciels souvent conçu par des fabricants différents. Pour que le tous ces appareils soient capables de communiquer entre eux. Pour ce chapitre sera consacré à la définition de ces différents organismes de normalisation à savoir le modèle OSI et le TCP/IP.

I . Réseau informatique :

Un réseau est un ensemble d'objets interconnectés les uns avec les autres. Il permet de faire circuler des éléments entre chacun de ces objets selon des règles bien définies. Dans le cas ou les objets sont des ordinateurs on parle d'un réseau informatique.Les réseaux informatiques qui permettaient à leur origine de relier des terminaux passifs à de gros ordinateurs centraux autorisent à l'heure actuelle l'interconnexion de tous types, d'ordinateurs que ce soit de gros serveurs, des stations de travail, des ordinateurs personnels ou de simples terminaux graphiques. Les services qu'ils offrent font partie de la vie courante des entreprises et administrations (banques, gestion, commerce, bases de données, recherche,...) et des particuliers (messagerie, loisirs, services d'informations par minitel et Internet ...).

I.1Intérêts d'un réseau :

Un ordinateur est une machine permettant de manipuler des données. L'homme, un être de communication, a vite compris l'intérêt qu'il pouvait y avoir à relier ces ordinateurs entre eux afin de pouvoir échanger des informations. Voici un certain nombre de raisons pour lesquelles un réseau est utile, un réseau permet:

- ✓ Le partage de fichiers, d'applications et de ressources. La communication entre personnes (grâce au courrier électronique, la discussion endirect, ...).
- ✓ La communication entre processus (entre des machines industrielles).
- ✓ La garantie de l'unicité de l'information (bases de données).
- ✓ Le transfert de la parole, de la vidéo et des données (réseaux à intégration de services ou multimédia).

I .2Les architectures de réseaux :

I .2.1Le modèle de référence OSI d'ISO :

Au début des années 70, chaque constructeur a développé sa propre solution réseau autour d'architecture et de protocole privés et il s'est vite avéré qu'il serait impossible d'interconnecter ces différents réseaux si une norme internationale n'était pas établie.

Cette norme établie par l'internationale standard organisation (ISO) est la norme open system interconnexion (OSI, interconnexion de systèmes ouverts).

Un système ouvert est un ordinateur, un terminal, un réseau, n'importe quel équipement respectant cette norme et donc apte à échanger de l'information avec d'autres équipement hétérogènes et issus de constructeurs différents.

Le premier objectif de la norme OSI a été de définir un modèle de toute architecture de réseau base sur découpage en sept couches chacun de ces couches correspondant à une fonctionnalité particulière d'un réseau.

Les couches 1, 2,3 et 4 sont dites basses et les couches 5,6 et 7 sont dites hautes.

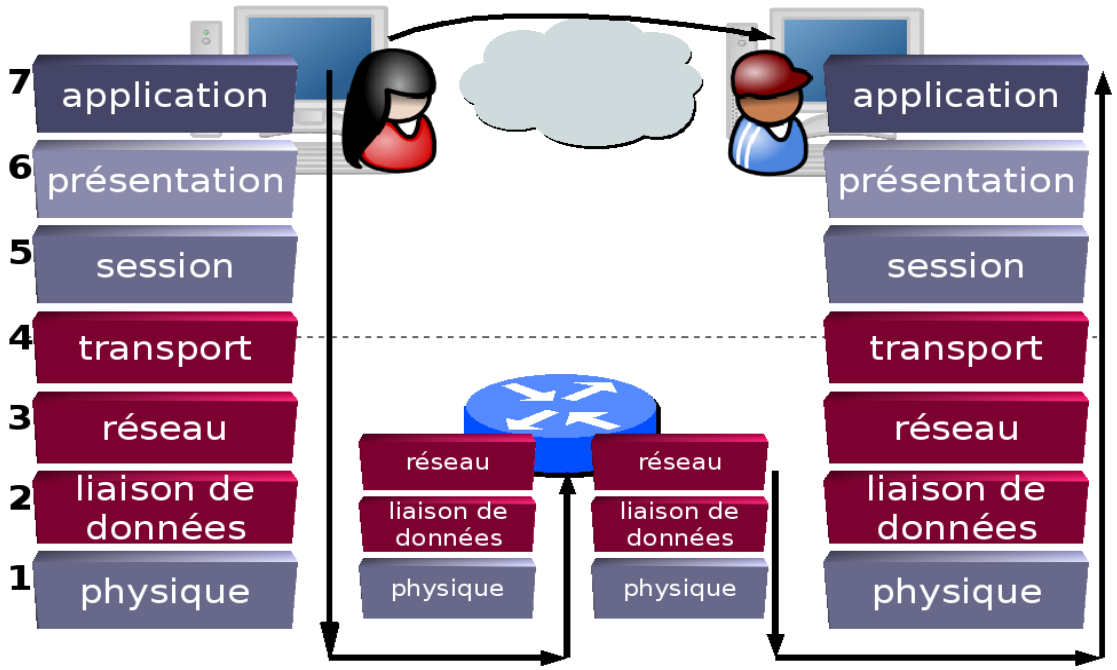


Figure I.2.1: Le model OSI

I.2.2 Les différentes couches du modèle :

- **La couche physique :**

La couche physique est la plus basse de l'organisation hiérarchique de modèle OSI, elle est dite de niveau 1.Elle s'occupe de transmission des bits de façon brute sur un canal de communication. Cette couche doit garantir la parfaite transmission des donnée (un bit 1 envoyé doit bien être reçu comme bit valant 1) .Cette couche doit normaliser les caractéristique électrique (un bit 1 doit être représenté par une tension de 5V),les caractéristique mécaniques (forme des connecteurs, de la topologie…), le type de signaux émis (modulation, puissance, portée…), la durée de bit et la possibilité de transmission bidirectionnelle. Les problèmes de conception sont principalement des problèmes de télécommunication et concernant les interfaces mécanique et électriques,

la synchronisation la nature des caractéristiques des supports (câbles, fibre optique…), le sens de transmission.

- **La couche liaison :**

  Le rôle principal de la couche liaison de donnée est de faire en sorte qu'un moyen de communication brut apparaisse à la couche réseau comme étant une liaison exempte de la transmission. Pour cela elle fraction les données d'entre de l'émetteur en trame et envoie ces trame en séquence. La couche liaison de données s'occupe de l'adressage physique de la topologie du réseau, l'accès au réseau, la notification des erreurs, la livraison ordonné des trames et le contrôle de flux.

  Cette couche détecte et corrige, quand cela est possible, les erreurs de la couche physique et signale à la couche réseau les erreurs irrécupérables.

- **La couche réseau :**

Cette couche s'intéresse à l'interconnexion de plusieurs réseaux physique, ainsi les problèmes à résoudre sont l'acheminement d'un paquet d'un point du réseau à un autre ce qu'on appelle le routage , sachant que l'arrivée et le départ ne sont pas sur le même support physique, l'interconnexion de support physique et de réseau hétérogènes ainsi que le contrôle et la régulation du trafic sur le réseau, A ce niveau apparaissent des protocoles de communication réseau tels que le protocole IP utilisé pat Internet.

- **La couche transport :**

  La fonction de base de la couche transport est d'accepter des données de couche supérieure, de les diviser en unité plus petites si c'est nécessaire, de les transmettre à la couche réseau, et d'assure qu'elles arrivent correctement à l'autre bout.

  La couche transport détermine aussi le type de service à fournir à la couche session, et finalement aux utilisateur de réseau, le type de connexion de transport qui connaît le plus grand succès est le canal point à point exempts d'erreurs qui remet les messages ou les octets dans l'ordre dans lequel ils ont été envoyés.

- **La couche session :**

  La couche session permet aux utilisateurs de différentes machines d'établir des sessions, une session offre divers service parmi lesquels, la gestion du dialogue, gère l'échange des données et la synchronisation. Elle réalise le lien entre les adresses logiques et les adresses physiques des tâches réparties.

La couche session assure un transfert efficace des données, classe de service ainsi que la signalisation des écarts de cette couche, de la couche présentation et la couche application.

- **La couche présentation :**

A la différence des couches les plus basses qui sont principalement concernées par le déplacement des bits, la couche présentation s'intéresse à la syntaxe et la sémantique des informations transmise. C'est elle traite l'information de manière à la rendre compatible entre tâches communicantes.

Cette couche peut convertir les données, les reformater, les crypter et compresser.

- **La couche application :**

La couche application est la couche OSI la plus proche de l'utilisateur. Elle fournit des services réseau aux applications de l'utilisateur, exemple de ce type d'application : réseau pour échanger des informations données, comme elle utilise des protocoles pour le transfert de fichiers, les échanges d'e-mail, la navigation de pages en pages Internet.

I.2L'avenir d'OSI :

Au niveau de son utilisation et implémentation, et ce malgré une mise à jour du modèle en 1994, OSI a clairement perdu la guerre face à TCP/IP. Seuls quelques grands constructeurs dominant conservent le modèle mais il est amené à disparaître d'autant plus vite qu'Internet (et donc TCP/IP) explose. Le modèle OSI restera cependant encore longtemps dans les mémoires pour plusieurs raisons. C'est d'abord l'un des premiers grands efforts en matière de normalisation du monde des réseaux. Les constructeurs ont maintenant tendance à faire avec TCP/IP, mais aussi le WAP, l'UMTS etc. ce qu'il devait faire avec OSI, à savoir proposer des normalisations dès le départ. OSI marquera aussi les mémoires pour une autre raison : même si c'est TCP/IP qui est concrètement utilisé, les gens ont tendance et utilisent OSI comme le modèle réseau de référence actuel. En fait, TCP/IP et OSI ont des structures très proches, et c'est surtout l'effort de normalisation d'OSI qui a imposé cette "confusion" générale entre les 2 modèles. On a communément tendance à considérer TCP/IP comme l'implémentation réelle de OSI.

I.3 Le modèle TCP/IP :

Le modèle TCP/IP peut en effet être décrit comme une architecture réseau à 4 couches :

- **La couche hôte réseau :**

  Cette couche regroupe les couches physiques et liaison de données du modèle OSI. En fait, cette couche n'a pas vraiment été spécifiée ; la seule contrainte de cette couche, c'est de permettre un hôte d'envoyer des paquets IP sur le réseau. L'implémentation de cette couche est laissée libre. De manière plus concrète, cette implémentation est typique de la technologie utilisée sur le réseau local. Par exemple, beaucoup de réseaux locaux utilisent Ethernet ; Ethernet est une implémentation de la couche hôte réseau.

- **La couche internet :**

  Cette couche réalise l'interconnexion des réseaux (hétérogènes) distants sans connexion. Son rôle est de permettre l'injection de paquets dans n'importe quel réseau et l'acheminement des ces paquets indépendamment les uns des autres jusqu'à destination. Comme aucune connexion n'est établie au préalable, les paquets peuvent arriver dans le désordre ; le contrôle de l'ordre de remise est éventuellement la tâche des couches supérieures. Du fait du rôle imminent de cette couche dans l'acheminement des paquets, le point critique de cette couche est le routage. C'est en ce sens que l'on peut se permettre de comparer cette couche avec la couche réseau du modèle OSI. La couche internet possède une implémentation officielle : le protocole IP(Internet Protocol). Remarquons que le nom de la couche ("internet") est écrit avec un i minuscule, pour la simple et bonne raison que le mot internet est pris ici au sens large (littéralement, "interconnexion de réseaux"), même si l'Internet (avec un grand I) utilise cette couche.

- **La couche transport :**

Son rôle est le même que celui de la couche transport du modèle OSI : Permettre à des entités de soutenir une communication. Cette couche possède deux implémentations :

- L**e** protocole **TCP (Transmission Control Protocol)** : c'est un protocole fiable, orienté connexion, qui permet l'acheminement sans erreur de paquets issus d'une machine d'un internet à une autre machine du même internet. Son rôle est de fragmenter les messages en paquets à transmettre de manière à pouvoir le faire passer sur la couche internet (donc au protocole IP). À l'inverse, sur la machine destination, T.C.P. replace dans l'ordre les paquets transmis sur la couche internet pour

reconstruire le message initial. T.C.P. s'occupe également du contrôle de flux de la connexion.

- Le protocole **UDP (User Datagram Protocol)** : UDP est en revanche un protocole plus simple que T.C.P. : il est non fiable et sans connexion. Son utilisation présuppose que l'on n'a pas besoin ni du contrôle de flux, ni de la conservation de l'ordre de remise des paquets. Par exemple, on l'utilise lorsque la couche application se charge de la remise en ordre des messages. Une autre utilisation d'UDP : la transmission de la voix ou de données particulières dont la latence et la taille est faible. C'est-à-dire lorsqu'il est nécessaire d'être rapide dans l'envoi des paquets (un autre exemple est la diffusion vidéo).

- **La couche application :**

Contrairement au modèle OSI, c'est la couche immédiatement supérieure à la couche transport, tout simplement parce que les couches présentation et session sont apparues inutiles. On s'est en effet aperçu avec l'usage que les logiciels réseau n'utilisent que très rarement ces 2 couches, et finalement, le modèle OSI dépouillé de ces 2 couches ressemble fortement au modèle TCP/IP. Cette couche contient tous les protocoles de haut niveau, comme par exemple Telnet, TFTP (trivial File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), HTTP (HyperText Transfer Protocol). Le point important pour cette couche est le choix du protocole de transport à utiliser. Par exemple, TFTP (surtout utilisé sur réseaux locaux) utilisera UDP, car on part du principe que les liaisons physiques sont suffisamment fiables et les temps de transmission suffisamment courts pour qu'il n'y ait pas d'inversion de paquets à l'arrivée. Ce choix rend TFTP plus rapide que le protocole FTP qui utilise TCP. A l'inverse, SMTP utilise TCP, car pour la remise du courrier électronique, on veut que tous les messages parviennent intégralement et sans erreurs.

I.4 Les réseaux IP :

Les réseaux IP (Internet) devient nom seulement un moyen de communication mais aussi un moyen de commerce globale de développement et distribution.

TCP/IP est très connu dans le domaine des réseaux, il correspond à toute une architecture. Il ne correspond pas à un seul protocole mais bien a un ensemble de petits protocoles spécialisée appelés sous protocoles (TCP, IP, UDP, ARP ICMP……).

La plus part- des administrateurs réseaux désignent ce groupe par TCP/IP.

TCP (Transmission Control Protocol) qui est un protocole de niveau message.

IP (Internet Protocol) qui est un protocole de niveau paquet.

1.4.1 Fonctionnement des réseaux IP :

La plupart des réseaux sont des entités indépendantes, mises en place pour rendre service à une population restreinte. Les utilisateurs choisissent les réseaux adaptés à leurs besoins spécifiques, car il est impossible de trouver une technologie satisfaisant tous les types de besoin.

Dans cet environnement de base, les utilisateurs qui ne sont pas connectés au même réseau ne peuvent pas communiquer. L'Internet est le résultat de l'interconnexion de ces différents réseaux physiques par des routeurs.

Pour obtenir l'inter-fonctionnement des différents réseaux, la présence du protocole IP est obligatoire dans les nœuds qui vont faire le routage entre les réseaux. Globalement, l'Internet est donc un réseau à transfert de paquets. Ces paquets traversent un (ou plusieurs) sous-réseaux pour atteindre leur destination, sauf si l'émetteur se trouve dans le même sous-réseau que le récepteur.

I.4.2 L'adressage IP et la structure d'adresses IP :

Sur Internet, les ordinateurs communiquent entre eux grâce au protocole TCP/IP qui utilise des numéros de 32 bits que l'on écrit sous forme de 4 numéros allant de 0 à 255(4fois 8 bits),on les note donc sous forme xxx.xxx.xxx.xxx. Où chaque xxx représente un entier de 0 à 255. Ces numéros servent aux ordinateurs du réseau pour se connaître, ainsi il ne doit pas exister deux ordinateurs sur le réseau ayant la même adresse IP.

Comme nous l'avons vu , une adresse IP est une adresse 32 bits notée sous forme de 4 nombres entiers séparés par des points .on distingue en fais deux parties dans l'adresse IP :

- Une partie des nombres à gauche désigne le réseau (on l'appelle Net-ID).
- Les nombres de droite désignent les ordinateurs de ce réseau (on l'appelle Host-ID).

I.4.3Les classes d'adresses :

On distingue 5 classes d'adresse :

Classe A :

Dans une adresses IP de classe A, le premier octet représente le réseau. le bit de poids fort (le premier bit, celui de gauche) est à zéro, qui signifie qu'il y a 2^7 (00000000 à 01111111) se nombre 127 est réservé pour désigner votre machine, les réseaux disponibles en classe A sont Donc les réseaux les réseaux allant de **1.0.0.0** à **126.0.0.0** (lorsque les derniers octets sont Des zéros cela indique qu'il s'agit d'un réseau et non d'un ordinateur !)

Les trois octets de droite représentent les ordinateurs du réseau, le réseau peut donc contenir :

2^{24}-2 = 16777214 ordinateurs.

Classe B :

Dans une adresses IP de classe B, les deux premiers octets représentent le réseau. les deux premiers bits sont 1 et 0, ce qui signifie qu'il y a 2^{14} (10 000000 00000000 à 10 111111 11111111) possibilités de réseaux, c'est-à-dire 16384. Les réseaux disponibles en classe B sont donc les réseaux allant de **128.0.0.0** à **191.255.0.0.**

Les deux octets de droite représentent les ordinateurs du réseau, le réseau peut donc contenir :

2^{16} -2^{1} = 65534 ordinateurs.

Classe C :

Dans une adresse IP de classe C, les trois premiers octets représentent le réseau. les trois premiers bits sont 1 et 0, ce qui signifie qu'il y a 2^{21} possibilités de réseaux, c'est-à-dire 2097152. Les réseaux disponibles en classe C sont donc les réseaux allant de **192.0.0.0** à **223.255.255.0.**

L'octet de droite représente les ordinateurs du réseau, le réseau peut donc contenir :

2^{8} -2^{1} = 254 ordinateurs.

Classe D :

Les adresses de classe D sont réservées pour les adresses IP de multidiffusion. Les quatre bits de poids fort d'une adresse de classe D sont toujours à la valeur binaire 1 1 1 0. Les bits restants sont pour l'adresse que les hôtes intéressés reconnaissent.

Classe E :

La classe E est une adresse expérimentale qui est réservée pour une utilisation future. Les bits de poids fort d'une adresse de classe E sont fixés à 1111.

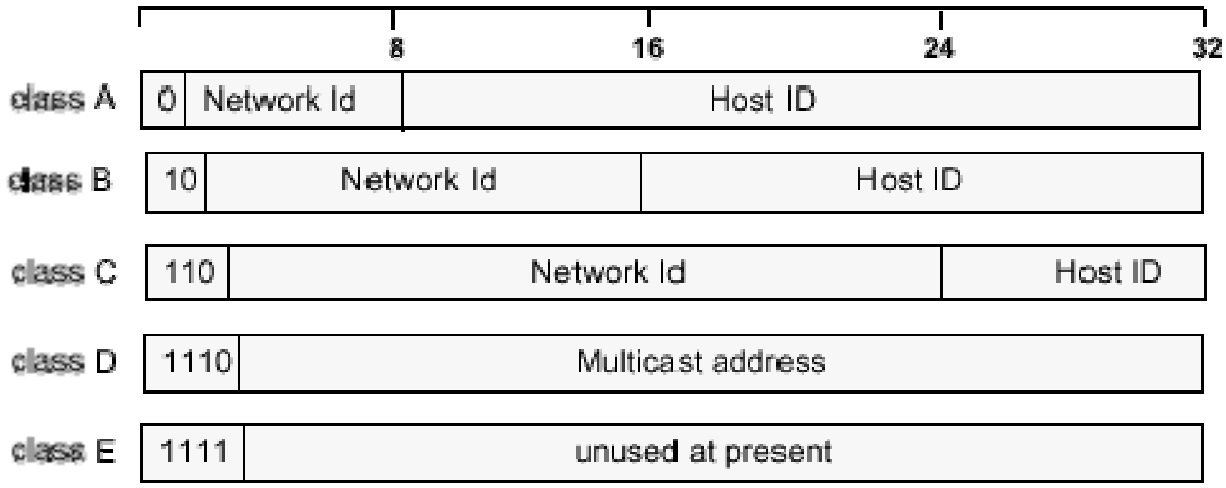


Figure I.2 : Différentes classes d'adresse.

I.4.4Le routage IP :

Un routeur peut être connecté directement à deux ou plusieurs réseaux et les hôtes sont généralement connectes a un seul réseau. Il existe plusieurs types de routages :

- **Routage directe :**

c'est le cas si les deux machines qui veulent communiquer sont rattachées au même réseau et ont donc le même numéro du réseau IP. Il peut s'agir de deux hôtes ou d'un routeur et un hôte. Il suffit pour effectuer le transport du paquet IP, de déterminer l'adresse physique du destinataire et d'encapsuler le datagramme dans une trame avant de l'envoyer sur le réseau.

- **Routage indirecte :**

Dans ce cas le routage est plus complexe car il faut déterminer le routeur au quel les datagrammes doivent être envoies ces deniers peuvent aussi être transmis de routeur en routeur jusqu'à ce qu'ils atteignent l'hôte destinataire. La fonction de routage fonde principalement sur la table de routage.

- **Table de routage :**

Le routage est effectue à partir du numéro du réseau de l'adresse IP du destinataire la table contient pour chaque numéro du réseau à atteindre, l'adresse, l'adresse IP du routeur au quel envoyer le datagramme. Elle peut également comprendre une adresse de routeur par défaut et indication de routage directe. La difficulté du routage provient de l'initialisation et de mise à jour de la table de routage.

- **Le Subnetting :**

C'est une technique d'adressage et de routage normalisée, qui permet de gérer plusieurs réseaux physique à partir d'une seule adresse IP d'Internet. Le principe du Subnetting consiste à diviser la partie numéro d'hôte d'une adresse IP en numéro de sous réseau et numéro d'hôte. En dehors du site les adresses sont interprétées sans qu'il soit tenu compte du Subnetting, le découpage n'étant connu est traité que de l'intérieur. Le découpage du numéro d'hôte permet de choisir librement le nombre de machine en fonction du nombre de réseaux sur le site.

I.4.5 Types d'adresses :

IPv6 reconnaît trois types d'adresses : unicast, multicast et anycast.

Le type **unicast**, est le plus simple. Une adresse de ce type désigne une interface unique. Un paquet envoyé à une telle adresse sera donc remis à l'interface ainsi identifiée.

Une adresse de type **multicast** désigne un groupe d'interfaces qui en général appartiennent à des équipements différents pouvant être situés n'importe où dans l'Internet.

Lorsqu'un paquet a pour destination une adresse de type multicast, il est acheminé par le réseau à toutes les interfaces membres de ce groupe.

Il faut noter qu'il n'y a plus d'adresses de type broadcast comme sous IPv4; elles sont remplacées par des adresses de type multicast.

Le dernier type, **anycast,** est nouveau en IPv6. Les adresses anycast ont deux points communs avec les adresses unicast : elles sont allouées dans le même espace d'adressage et ont les mêmes formats. Comme dans le cas de multicast, une adresse de type anycast désigne un groupe d'interfaces, la différence étant que lorsqu'un paquet a pour destination une telle adresse, il est routé à un seul des éléments du groupe et non pas à tous.

I.5 Masque de réseau :

On fabrique un masque contenant des 1 aux emplacements des bits que l'on désire conserver, et des 0 pour ceux que l'on veut rendre égaux à zéro. Une fois ce masque crée, il suffit de faire un ET entre la valeur que l'on désire masquer et le masque afin de garder intacte la partie que l'on désire et annuler le reste.

Ainsi, un masque réseau (en anglais netmask) se présente sous la forme de 4 octets séparéspar des points (comme une adresse IP), il comprend (dans sa notation binaire) des zéros aux niveaux des bits de l'adresse IP que l'on veut annuler (et des 1 au niveau de ceux que l'on désire conserver).

Le réseau associé à l'adresse 34.56.123.12 est 34.0.0.0 (classe A). Il suffit donc pour connaître l'adresse du réseau associé à l'adresse IP 34.56.123.12 d'appliquer un masque dont le premier octet ne comporte que des 1 (ce qui donne 255), puis des 0 sur les octets suivants (ce qui donne 0..).

Le masque est : 11111111.00000000.00000000.00000000 et le masque associé à l'adresse IP 34.208.12 est donc 255.0.0.0. la valeur binaire de 34.208.123.12 est : 00100010.11010000.01111011.00001100

Un ET entre

00100010.11010000.01111011.00001100

ET

11111111.00000000.00000000.00000000

Donne ;

00100010.000000000.00000000.00000000

C'est-à-dire 34.0.0.0,c'est le réseau associé à l'adresse 34.208.123.12.Donc pour une adresse de Classe A, le masque c'est en notation décimale : **255.0.0.0,** pour la classe B :**255.255.0.0 et 255.255.255.0 classe C.**

I.6Le DHCP (Dynamic Host Configuration Protocol) :

DHCP utilise un modèle client/serveur dans lequel le serveur DHCP assure la gestion centralisée des adresses IP utilisées sur le réseau. Les clients qui prennent en charge DHCP peuvent ensuite demander et obtenir la location d'une adresse IP auprès d'un serveur DHCP dans le cadre de leur procédure d'amorçage réseau.

I.7Le protocole NAT (Network Address Translation) :

Il s'agit d'un procédé permettant de transcrire des adresses IP en d'autres, sans références directes avec les adresses MAC, traitées quand à elles par le protocole ARP. NAT utilise l'adresse IP et le numéro de port d'une station et les transformer en une adresse IP et un numéro de port qui n'est pas attribué a une application standard.

I.8 Protocoles de résolution d'adresses :

Les adressent IP sont attribuées indépendamment des adresses matérielles des machines. Pour envoyer un datagramme sur Internet, le logiciel réseau doit convertir l'adresse IP en une adresse physique, utilisée pour transmettre la trame.

C'est le protocole ARP (Address Resolution Protocol) qui effectue cette traduction en s'appuyant sur le réseau physique. ARP permet aux machines de résoudre les adresses sans utiliser une table statique. Une machine utilise ARP pour déterminer l'adresse physique

destinataire en diffusant, sur le sous réseau, une requête ARP qui contient l'adresse IP à traduire. La machine possédant l'adresse IP concernée répond en renvoyant son adresse physique. Le protocole RARP (Reverse ARP) permet à une machine d'utiliser son adresse physique pour déterminer son adresse logique sur Internet. Le mécanisme RARP permet à un calculateur de se faire identifier comme cible en diffusant sur le réseau une requête RARP.

I.7Entête IP :

IP signifie "Internet Protocol", protocole Internet. Il représente le protocole réseau le plus répandu. Il permet de découper l'information à transmettre en paquets, de les adresser, de les transporter indépendamment les uns des autres et de recomposer le message initial à l'arrivée. Ce protocole utilise ainsi une technique dite de commutation de paquets. Il apporte, en comparaison à Ipx/Spx et Netbeui, l'adressage en couche 3 qui permet, par exemple, la fonction principale de routage. Il est souvent associé à un protocole de contrôle de la transmission des données appelé TCP, on parle ainsi du protocole TCP/IP. Cependant, TCP/IP est un ensemble de protocole dont voici les plus connu:

- IP : Internet Protocol - Couche 3 - IP natif.
- ARP : Address Resolution Protocol - Couche 3 - Résolution d'adresse IP en adresse MAC.
- RARP : Reverse Address Resolution Protocol - Couche 3 - Résolution d'adresse MAC en adresse IP.
- ICMP : Internet Control Message Protocol - Couche 3 - Gestion des messages du protocole IP.
- IGMP : Internet Group Management Protocol - Couche 3 - Protocole de gestion de groupe.
- TCP : Transmission Control Protocol - Couche 4 - Transport en mode connecté.
- UDP : User Datagram Protocol - Couche 4 - Transport en mode non connecté.

I.7.1 Structure de l'entête :

Voici la structure de l'entête IP basé sur 20 octets.

I.7.2 Définition des différents champs :

- **Le champ Vers :**

  Le champ version est codé sur 4 bits. Il représente le numéro de version du protocole IP. Il permet aux piles IP réceptionnant la trame de vérifier le format et d'interpréter correctement la suite du paquet. C'est d'ailleurs pour cette raison qu'il est placé au début, une version inconnue par un équipement conduit au rejet direct.

- **IHL :**

  IHL signifie "Internet header lengh". Ce champ est codé sur 4 bits et représente la longueur en mots de 32 bits de l'entête IP. Par défaut, il est égal à 5 (20 octets), cependant, avec les options de l'entête IP, il peut être compris entre 6 et 15. Le fait que le codage soit sur 4 bits, la taille maximum de l'entête IP est donc de 15*32bits = 60 octets

- **Service :**

  Le champ service "Type Of Service" est codé sur 8 bits, il permet la gestion d'une qualité de service traitée directement en couche 3 du modèle OSI. Cependant, la plupart deséquipements de Backbone, ne tiennent pas compte de ce champ et même certain le réinitialise à 0.

- **Priorité :**

  Le champ Priorité "Precedence" est codé sur 3 bits. Il indique la priorité que possède le paquet.

- **Délai :**

  Le champ Délai "Delay" est codé sur 1 bit. Il indique l'importance du délai d'acheminement du paquet.

- **Débit :**

  Le champ Débit "Throughput" est codé sur 1 bit. Il indique l'importance du débit acheminé.

- **Fiabilité :**

  Le champ Fiabilité "Reliability" est codé sur 1 bit. Il indique l'importance de la qualité du paquet.

- **Coût :**

Le champ Coût "Cost" est codé sur 1 bit. Il indique le coût du paquet.

- **MBZ :**

  Le champ MBZ "Must Be Zero" est codé sur 1 bit. Comme son nom l'indique, il doit être mis à 0.

- **Longueur totale :**

  Le champ Longueur totale est codé sur 16 bits et représente la longueur du paquet incluant l'entête IP et les Data associées. La longueur totale est exprimée en octets, ceci permettant de spécifier une taille maximum de 2^{16}= 65535 octets.

- **Identification :**

  Le champ Identification est codé sur 16 bits et constitue l'identification utilisée pour reconstituer les différents fragments. Chaque fragment possède le même numéro d'identification, les entêtes IP des fragments sont identiques à l'exception des champs Longueur totale, Checksum et Position fragment.

- **Flags :**

  Le champ Flags est codé sur 3 bits et indique l'état de la fragmentation.

- **Reserved :**

  Le premier bit est réservé et positionné à 0.

- **DF :**

  Appelé DF "Don't Fragment", le second bit permet d'indiqué si la fragmentation est autorisée. Si un Datagramme devant être fragmenté possède le flag DF à 1, alors, il sera alors détruit.

- **MF :**

  Appelé MF "More Fragments", le troisième bit indique s'il est à 1 que le fragment n'est pas le dernier.

- **Position fragment :**

  Le champ Position fragment est codé sur 13 bits et indique la position du fragment par rapport à la première trame. Le premier fragment possède donc le champ Position fragment à 0.

- **TTL :**

  Le champ TTL (Time To Live) est codé sur 8 bits et indique la durée de vie maximale du paquet. Il représente la durée de vie en seconde du paquet. Si le TTL arrive à 0, alors l'équipement qui possède le paquet, le détruira. Attention, à chaque passage d'un routeur le paquet se verra décrémenté dans une seconde. De plus, si le paquet reste en file d'attente d'un routeur plus d'une seconde, alors la décrémentation sera plus élevée. Elle sera égale au nombre de seconde passé dans cette même file d'attente. Par défaut, si les temps de réponse sont corrects, alors on peut, entre guillemet, en conclure que le Time To Live représente le nombre de saut maximum du niveau. Le but du champ TTL est d'éviter de faire circuler des trames en boucle infinie.

- **Protocole :**

  Le champ Protocole est codé sur 8 bits et représente le type de Data qui se trouve derrière l'entête IP.

- **Checksum :**

  Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 3. Pour pouvoir calculer le Checksum, il faut positionner le champ du checksum a 0 et ne considérer que l'entête IP. Donc par exemple, si deux trames ont la même entête IP (y compris le champ length) et deux entêtes ICMP et Data différentes (mais de même longueur), le checksum IP sera alors le même.

- **Adresse IP source :**

  Le champ IP source est codé sur 32 bits et représente l'adresse IP source ou de réponse. Il est codé sur 4 octets qui forme l'adresse A.B.C.D.

- **Adresse IP destination :**

  Le champ IP destination est codé sur 32 bits et représente l'adresse IP destination. Il est codé sur 4 octets qui forme l'adresse A.B.C.D.

- **Options :**

  Le champ Options est codé entre 0 et 40 octets. Il n'est pas obligatoire, mais permet le "Tuning de l'entête IP". Afin de bien gérer les Options, cela doit commencer par un octet de renseignement.

- **Bourrage :**

  Le champ Bourrage est de taille variable comprise entre 0 et 7 bits. Il permet de combler le champ option afin d'obtenir un entête IP multiple de 32 bits. La valeur des bits de bourrage est 0.

1.8Conclusion :

L'ingénierie des réseaux IP reste un domaine complexe, encore peu maîtrisé. Le but de la version IPv6 est de proposer un protocole beaucoup plus maîtrisable, grâce à des nouveaux champs permettant d'introduire une adresse mieux construite, une zone d'identification des flux et des options nombreuses, en particulier dans le domaine de la sécurité.

Chapitre II :

Etude des protocoles de transport et d'applications en IP.

Introduction :

Le but des réseaux est de faire communiquer plusieurs ordinateurs ensemble. Si les hommes communiquent entre eux grâce aux différentes langues, les ordinateurs utilisent différents protocoles. Les communications sont souvent internationales, et comme pour les hommes, il n'existe pas de protocole universel. Certains sont plus utilisés que d'autres, il en existe cependant un très grand nombre, chacun cherchant à imposer sa propre norme.

Comment expliquer clairement ce qu'est un protocole? Supposons que quelqu'un veuille envoyer une lettre à quelqu'un d'autre. On va placer cette lettre dans une enveloppe et on y notera l'adresse. Pour l'acheminement du courrier, le contenu de la lettre n'est d'aucune utilité. Les différents services de la poste regardent les différents champs de l'adresse et dirigent l'enveloppe, donc son contenu dans la bonne direction. Il en est de même quand un ordinateur veut envoyer des données à un autre ordinateur. Les données sont enfermées (on dit encapsulées) dans une enveloppe qui contient les informations permettant l'acheminement des données. Un protocole, c'est la façon dont l'adresse est écrite sur l'enveloppe, le fait de mettre d'abord le nom, puis la rue et enfin la ville. Un autre protocole, c'est aussi le fait de mettre le lieu et la date en haut à droite et la signature en bas.

Enfin, un protocole est une description formelle de règles et de conventions à suivre dans un échange d'informations, que ce soit pour acheminer les données jusqu'au destinataire ou pour que le destinataire comprenne comment il doit utiliser les données qu'il a reçues.

II. Notion de protocole :

La notion de protocole se retrouve à tous les niveaux du modèle en couche. Pour une couche donnée, on appelle protocole, l'ensemble des règles et des formats (sémantiques et syntaxiques) prédéfinis déterminant les caractéristiques de communication des processus de la couche. La mise en œuvre d'un protocole est effectuée à partir d'un PDU (Protocole Data Unit).Un sous-système dans une couche est un élément n'ayant des interactions qu'avec les éléments des niveaux immédiatement supérieur et inférieur. Il est composé d'une ou plusieurs entités. On peut donc considérer qu'une entité est un processus qui met en œuvre un protocole particulier. On dit que le programme correspondant implémente le protocole.

Les services d'une couche sont les fonctionnalités offertes par cette couche à la couche supérieure. On y accède par des primitives spécifiques du service. Le point d'accès à des services (SAP) est le point où les services sont fournis par une entité de la couche aux sous systèmes des couches supérieures ou inférieures. Les blocs de données utilisateurs (SDU :

Service Data Unit) sont transmis par l'intermédiaire des SAP. Une couche offre en général plusieurs points d'accès.

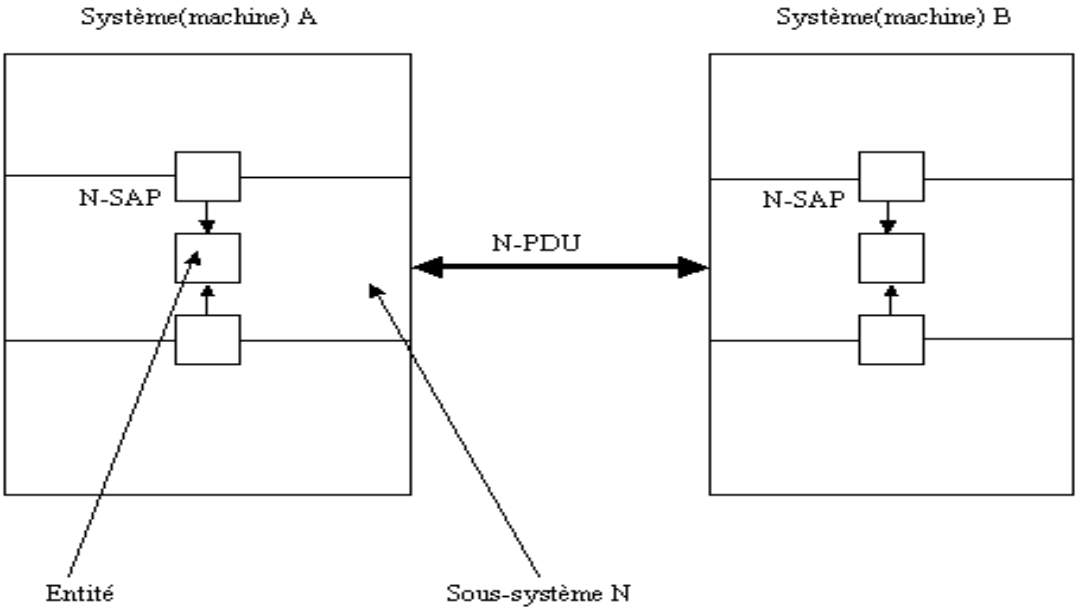


Fig. II.1 Les blocs fonctionnels d'une couche.

- **Exemple** : Pour fixer les idées, on peut tout à fait assimiler le fonctionnement à un service postal. Lorsque quelqu'un écrit à un chercheur dans un laboratoire, il envoie une ou plusieurs enveloppes (ce sont les PDU). S'il s'agit de plusieurs enveloppes à réordonner, il va les numéroter et rajouter donc une information inutile pour la poste mais utile pour le chercheur. Le message ainsi constitué est un SDU. Lorsque le facteur livre le courrier quelle que soit la personne du laboratoire, il la met dans la boite à lettre (N-1SAP) globale. Puis une personne (entité) va les distribuer dans les casiers (N-SAP). Pour mettre en œuvre correctement un protocole, il faut formaliser les actions et les objets utilisés ainsi que les interactions entre prestataires de services et utilisateurs. Une entité protocolaire (sous-système ou couche) est sollicitée par des évènements qui sont :

➢ Externes : primitives d'interactions entre couches,

➢ Internes, résultat de processus ou traitement.

La figure (Fig.II.b) donne une idée schématique des évènements liés à un protocole. On décrit souvent un protocole par un automate d'état fini. Chaque transition est de la forme :

(État départ, évènement, état arrivé, Action vers couche supérieure, action vers couche inférieure, action interne), Les trois derniers champs peuvent être vides.

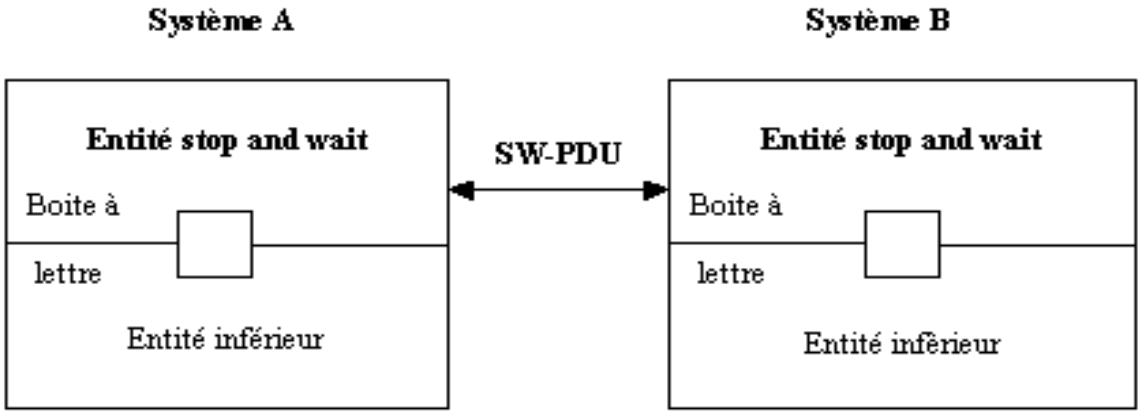


Fig.II.2 Architecture Send and Wait.

- **Exemple de protocole « SEND AND WAIT »**

Le protocole « envoi et attente » est un des plus simple. Chaque entité peut émettre et recevoir des PDU par l'intermédiaire de sa boîte à lettres locale à son sous-système. Après l'envoi d'un PDU, l'entité se met en attente de réception d'un accusé de réception. La figure 3 montre l'architecture du système.

L'automate correspondant est décrit figure (FigII.c).

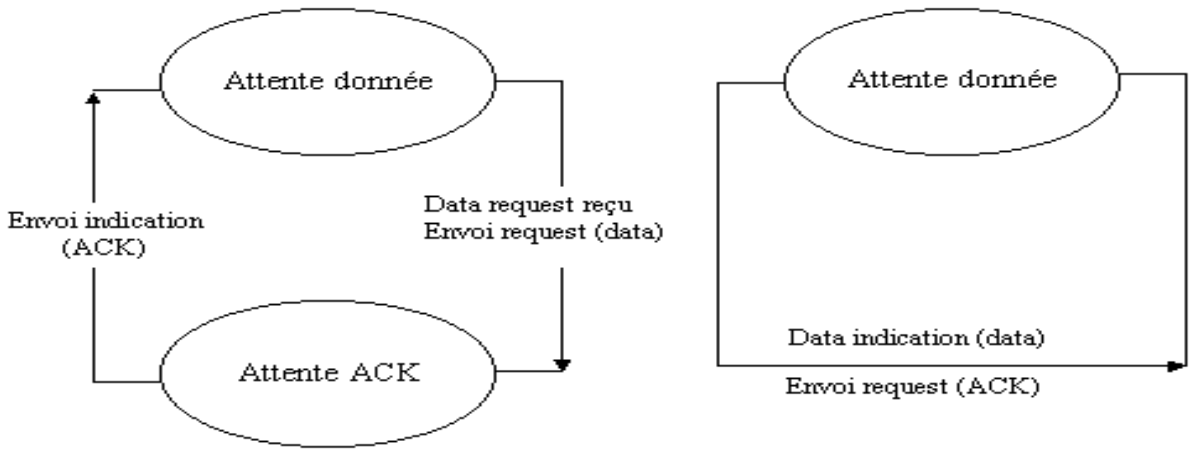


Fig.II. 3 Automate send and wait.

II.1 Les protocoles de la couche Transport :

II.1.1 Protocole TCP :

Introduction à TCP :

Le protocole TCP est défini dans le but de fournir un service de transfert de données de haute fiabilité entre deux ordinateurs "maîtres" raccordés sur un réseau de type "Paquets commutés", et sur tout système résultant de l'interconnexion de ce type de réseaux.

II.1.1.1 Motivation :

TCP est un protocole sécurisé orienté connexion conçu pour s'implanter dans un ensemble de protocoles multicouches, supportant le fonctionnement de réseau hétérogènes. TCP s'intègre dans une architecture multicouche des protocoles, juste au-dessus du protocole Internet IP. Ce dernier permet à TCP l'envoi et la réception de segments de longueur variable, encapsulés dans un paquet Internet appelé aussi "datagramme".

Fig.II.3 Implantation du protocole TCP dans le modèle TCP/IP.

II.1.1.2 Spécifications fonctionnelles de TCP :

a) Format des segments TCP :

Les paquets TCP sont envoyés sous forme de datagrammes Internet. L'en-tête IP transmet un certain nombre de paramètres, tels que les adresses Internet source et destinataires.

0 16 32

Port source								Port destination
Numéro de séquence								
Accusé de réception								
Data Offset	Réservé	U	A	P	R	S	F	Fenêtre
Checksum								Pointeur données urgentes
Option								Bourrage
Data								

Fig.II.4 En-tête TCP.

b) Définition des différents champs:

-Port source : (16 bits) Le numéro de port de la source.

-Port Destinataire : (16 bits) Le numéro de port du destinataire.

-Numéro de séquence : (32 bits) Le numéro du premier octet de données par rapport au début de la transmission (sauf si SYN est marqué). Si SYN est marqué, le numéro de séquence est le numéro de séquence initial (ISN) et le premier octet à pour numéro ISN+1.

-Accusé de réception : (32 bits) Si ACK est marqué ce champ contient le numéro de séquence du prochain octet que le récepteur s'attend à recevoir.

-Data Offset : (4 bits) La taille de l'en-tête TCP en nombre de mots de 32 bits. Il indique là ou commence les données.

-Réservé : (6 bits) Réservés pour usage futur. Doivent nécessairement être à 0.

-Bits de contrôle : (6 bits) (de gauche à droite): URG: Pointeur de données urgentes significatif, ACK: Accusé de réception significatif, PSH: Fonction Push indique à l'hôte en réception de «pousser» toutes les informations en mémoire tampon vers l'application en couche supérieure. L'émetteur notifie le récepteur qu'il a

transmis toutes ses données «pour l'instant». , RST: indique un arrêt ou un refus

de connexion. SYN: Synchronisation des numéros de séquence .FIN: Fin de transmission.

-Fenêtre : (16 bits) Le nombre d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir.

-Checksum : (16 bits) Le Checksum est constitué en calculant le complément à 1 sur 16 bits de la somme des compléments à 1 des octets de l'en-tête et des données pris deux par deux mots de 16 bits.

-Pointeur de données urgentes : (16 bits) Communique la position d'une donnée urgente en donnant son décalage par rapport au numéro de séquence.

-Options :(variable) Les champs d'option peuvent occuper un espace de taille variable à la fin de l'en-tête TCP.

-Bourrage (padding): (variable) Les octets de bourrage terminent l'en-tête TCP: de sorte que le nombre d'octet de celle-ci soit toujours multiple de 4 (32 bits).

c) Modèle de fonctionnement :

Les processus transmettent les données en faisant appel à TCP et en passant des tampons de données comme arguments. TCP met en forme les données de ces tampons, les segmente afin de les transférer au protocole Internet qui a son tour les acheminera vers le TCP distant. Celui-ci reçoit les segments, les copies dans un tampon temporaire, et en avise l'émetteur. Le protocole TCP inclut les informations nécessaires à la "reconstruction" en bon ordre des données originales. Le modèle d'une communication Internet fait qu'il existe pour chaque TCP actif un module de protocole Internet chargé de l'acheminement de données. Ce module Internet "encapsule" à son tour les paquets TCP sous la forme de paquets Internet, transmis à un module Internet distant via des "routeurs.

d) Fiabilité de communication :

Un flux de donnée s'appuyant sur une connexion TCP doit être pouvoir considéré comme "fiable". La fiabilité de cette transmission s'appuie sur l'utilisation de numéros de séquence et sur un mécanisme d'accusés de réception.

II.1.2 Le protocole UDP :

Introduction à UDP :

Le protocole UDP est, comme TCP, un protocole de transport des données. Cependant, contrairement à TCP, on qualifie l'UDP de transmission "en mode non connecté et

non fiable" ou encore de protocole "non orienté connexion". Ceci signifie simplement que la machine émettrice envoie des données sans prévenir la machine réceptrice, et la machine réceptrice reçoit les données sans envoyer d'avis de réception à la première. Les données sont ainsi envoyées sous forme de blocs (datagrammes). Il n'y a pas de contrôle d'erreur. C'est un mécanisme simple d'échange de données entre applications.

Le protocole UDP est utilisé en place de TCP pour un transport rapide et léger des données.

II.1.2.1 Structure de l'en-tête UDP :

Le paquet UDP est conçu pour être encapsulé dans un datagramme IP et permettre un échange de données entre deux applications, sans échange préliminaire. Ainsi, si les données à transmettre n'obligent pas IP à fragmenter un paquet UDP génère un datagramme IP et c'est tout.

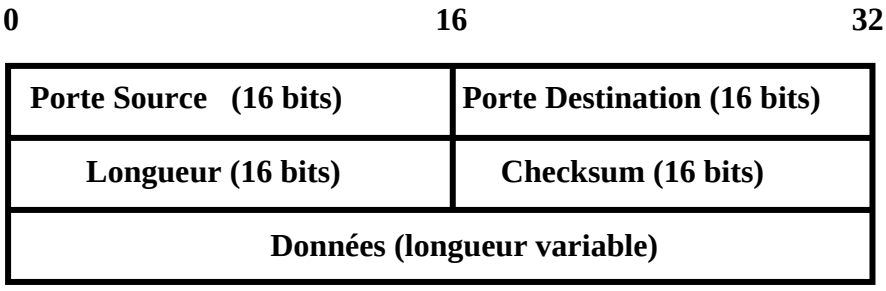


Fig.II. 5En-tête UDP

- **Définition des différents champs :**

  Port source : (16bits) Le champ Port source est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine source.

  Port destination : (16bits) champ Port destination est codé sur 16 bits et il correspond au port relatif à l'application en cours sur la machine de destination.

  Longueur : (16bits) Le champ Longueur est codé sur 16 bits et il représente la taille de l'entête et des données. Sont unité est l'octet et sa valeur maximale est 64 Koctets (2^{16}).

  Checksum : (16bits) Le champ Checksum est codé sur 16 bits et représente la validité du paquet de la couche 4 UDP.

II.1.2.2 Applications du Protocole :

Ce protocole sera utilisé principalement pour les communications avec les serveurs de noms de domaines, et dans les transactions utilisant le protocole Trivial File Transfer. Les protocoles de la couche Application

II.2 Les protocoles de la couche Application:

II.2.1 Le protocole FTP :

Introduction :

Le protocole FTP (File Transfer Protocol) est, comme son nom l'indique, un protocole de transfert de fichier. La mise en place du protocole FTP date de 1971, date à laquelle un mécanisme de transfert de fichiers (décrit dans le RFC 141) entre les machines du MIT (Massachussetts Institute of Technology) avait été mis au point. De nombreux RFC ont ensuite apporté des améliorations au protocole de base, mais les plus grandes innovations datent de juillet 1973.

II.2.1.1 Le rôle du protocole FTP :

Le protocole FTP a pour objectifs de :

- Permettre un partage de fichiers entre machine distante
- Permettre une indépendance aux systèmes de fichiers des machines clientes et serveur
- Permettre de transférer des données de manière efficace

II.2.1.2 Fonctionnement :

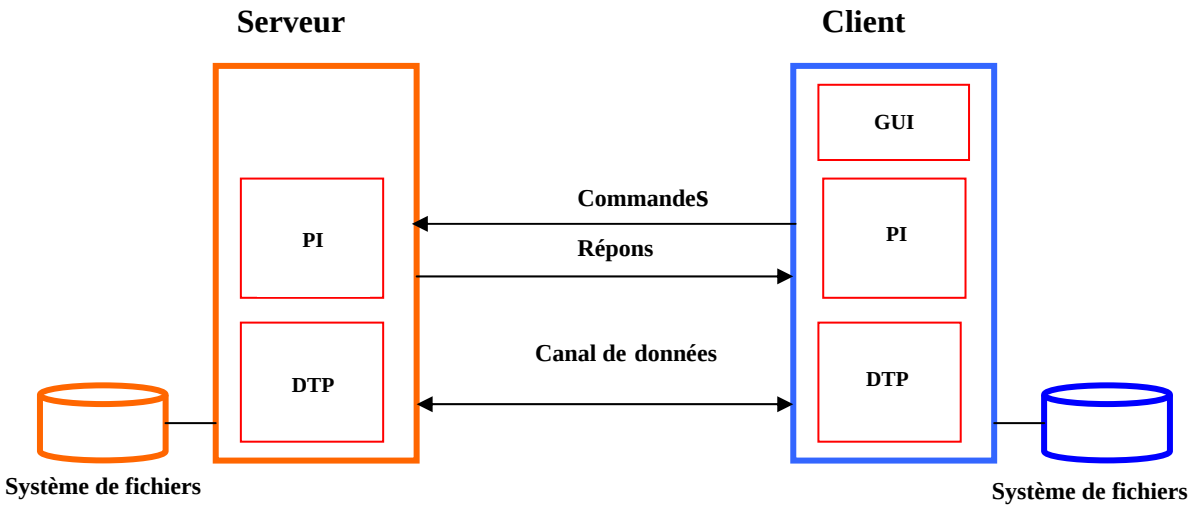


Fig.II.6 Présente le principe d'une connexion FTP entre un client et un serveur.

Le protocole FTP s'inscrit dans un modèle client-serveur, c'est-à-dire qu'une machine envoie des ordres (le client) et que l'autre attend des requêtes pour effectuer des actions (le serveur). Lors d'une connexion FTP, deux canaux de transmission sont ouverts :

Un canal pour les commandes (canal de contrôle) et un canal pour les données. Ainsi, le client comme le serveur possède deux processus permettant de gérer ces deux types d'information :

Lors d'une connexion FTP, deux canaux de transmission sont ouverts :

- **Le DTP (Data Transfer Process):**
  Processus chargé d'établir la connexion et de gérer le canal de données. Le DTP côté serveur est appelé Serveur-DTP, le DTP côté client est appelé Client-DTP Le schéma suivant représente le transfert de données entre deux serveurs.

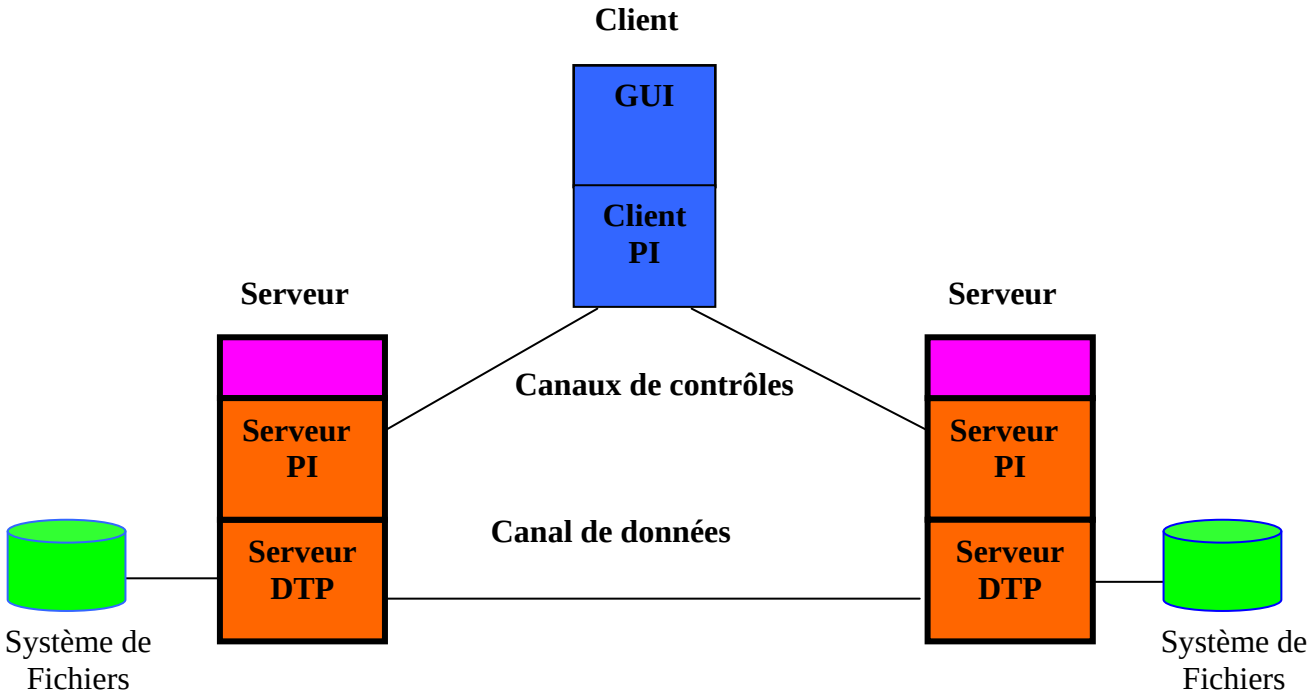


Fig.II.7Transfère des données entre deux serveurs FTP en passant par un client

Dans cette configuration, le protocole impose que les canaux de contrôle restent ouverts pendant tout le transfert de données. Ainsi un serveur peut arrêter une transmission si le canal de contrôle est coupé lors de la transmission.

II.2.2 Le protocole http :

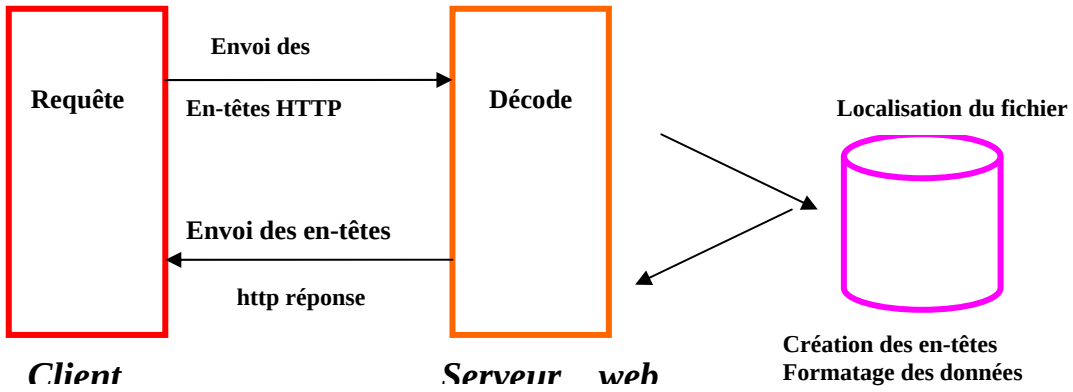
Introduction au protocole http :

Le protocole HTTP (Hyper Text Transfer Protocol) est le protocole le plus utilisé sur Internet depuis 1990. La version 0.9 était uniquement destinée à transférer des données sur Internet (en particulier des pages Web en HTML). La version 1.0 du protocole (la plus utilisée) permet désormais de transférer des messages avec des en-têtes décrivant le contenu du message en utilisant un codage de type MIME**.**

Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisé grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web (appelé d'ailleurs http).

-Communication entre navigateur et serveur :

La communication entre le navigateur et le serveur se fait en deux temps :



(Navigateur)

Fig.II.8 Communication entre le navigateur et le serveur.

Le navigateur effectue une requête http, Le serveur traite la requête puis envoie une réponse HTTP

En réalité la communication s'effectue en plus de temps si on considère le traitement de la requête par le serveur**.**

- **Requête http :**

  Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur.

  Elle comprend :

- **Une ligne de requête** :

C'est une ligne précisant le type de document demandé, la méthode qui doit être appliqué, et la version du protocole utilisée. La ligne comprend trois éléments devant être séparé par un espace :

* La méthode * L'URL

La version du protocole utilisé par le client (généralement HTTP/1.0)

- **Les champs d'en-tête de la requête** :

Il s'agit d'un ensemble de lignes facultatives permettant de donner des

Informations supplémentaires sur la requête et/ou le client (Navigateur,

système d'exploitation,...). Chacune de ces lignes est composé d'un nom qualifiant

le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête.

- **Le corps de la requête**:

C'est un ensemble de ligne optionnel devant être séparé des lignes précédentes par une ligne vide et permettant par exemple un envoi de données par une commande POST lors de l'envoi de données au serveur par un formulaire

Un requête HTTP a donc la syntaxe suivante (<crlf> signifie retour chariot ou saut de ligne) : Voici donc un exemple de requête HTTP :

GET http://www.commentcamarche.net HTTP/1.0

- **Réponses HTTP:**

Une réponse HTTP est un ensemble de lignes envoyé au navigateur par le serveur. Elle comprend :

- **Une ligne de statut** :

C'est une ligne précisant la version du protocole utilisé et l'état du traitement de la requête à l'aide d'un code et d'un texte explicatif. La ligne comprend trois éléments devant être séparé par un espace :

-La version du protocole utilisé ;

-Le code de statut ;

-La signification du code.

- **Les champs d'en-tête de la requête** :

  Il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la réponse et/ou le serveur. Chacune de ces lignes est composé d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête

- **Le corps de la réponse** :

  Il contient le document demandé : Une réponse HTTP a donc la syntaxe suivante (<crlf> signifie retour chariot ou saut de ligne) :

II.2.3 Le protocole Telnet :

Introduction au protocole Telnet :

Le protocole Telnet est un protocole standard d'Internet permettant l'interfaçage de terminaux et d'applications à travers Internet. Ce protocole fournit les règles de base pour permettre de relier un client (système composé d'un affichage et d'un clavier) à un interpréteur de commande (côté serveur).

Le protocole Telnet s'appuie sur une connexion TCP pour envoyer des données au format ASCII codées sur 8 bits entre lesquelles s'intercalent des séquences de contrôle Telnet. Il fournit ainsi un système orienté communication, bidirectionnel (half-duplex), codé sur 8 bits facile à mettre en œuvre.

Le protocole Telnet repose sur trois concepts fondamentaux :

- Le paradigme du terminal réseau virtuel (NVT, Network Virtual Terminal) ;
- Le principe d'options négociées ;
- Les règles de négociation.

Ce protocole est un protocole de base, sur lequel s'appuient certains autres protocoles de la suite TCP/IP (FTP, SMTP, POP3, ...).

Lorsque le protocole Telnet est utilisé pour connecter un hôte distant à la machine sur lequel il est implémenté en tant que serveur, ce protocole est assigné au port 23. Hormis les options et les règles de négociation associées, les spécifications du protocole Telnet sont basiques. La transmission de données à travers Telnet consiste uniquement à transmettre les octets dans le flux TCP (le protocole Telnet précise tout de même que les données doivent par défaut, c'est-à-dire si aucune option ne précise le contraire, être groupées dans un tampon avant d'être envoyées. Plus exactement cela signifie que par défaut les données sont envoyées ligne par ligne.

II.2.3.1La notion de terminal virtuel :

Aux débuts d'Internet, le réseau (ARPANET) était composé de machines dont les configurations étaient très peu homogènes (claviers, jeux de caractères, résolutions, longueur des lignes d'affichage). D'autre part, les sessions des terminaux possédaient également leur propre façon de contrôler les flux de données en entrée/sortie.

Ainsi, au lieu de créer des adaptateurs pour chaque type de terminal afin qu'il puisse y avoir une interopérabilité de ces systèmes, il a été décidé de mettre au point une interface standard, appelée NVT (Network Virtual Terminal, traduisez Terminal réseau virtuel), fournissant une base de communication standard, composée de :

- Caractères ASCII 7 bits auxquels s'ajoutent le code ASCII étendu
- Trois caractères de contrôle.
- Cinq caractères de contrôle optionnels.
- Un jeu de signaux de contrôle basique.

Le protocole Telnet consiste ainsi à créer une abstraction du terminal, permettant à n'importe quel hôte (client ou serveur) de communiquer avec un autre hôte sans connaître ses caractéristiques.

II.2.3.2 Le principe d'options négociées :

Les spécifications du protocole Telnet permettent de prendre en compte le fait que certains terminaux puissent proposer des services additionnels, non définis dans les spécifications de base (mais conformes aux spécifications), afin de pouvoir utiliser des fonctions avancées. Ainsi, ces fonctionnalités se traduisent en terme d'option. Le protocole Telnet propose donc un système de négociations d'options permettant l'utilisation de fonctions avancées sous forme d'options de part et d'autre en initiant des requêtes pour en demander l'autorisation au système distant.

Les options de Telnet affectent séparément chaque direction du canal de données. Ainsi, chaque extrémité est à même de négocier les options, c'est-à-dire de définir les options qu'elle :

-* veut utiliser (DO).

-* refuse d'utiliser (DON'T).

-* veut que l'autre extrémité utilise (WILL).

-* refuse que l'autre extrémité utilise (WON'T).

De cette façon, chacune des parties peut émettre une demande d'utilisation d'une option. L'autre partie doit alors répondre si elle accepte ou non l'utilisation de l'option. Dans le cas où la requête concerne une désactivation d'option, le destinataire de la requête ne doit pas refuser pour être totalement compatible avec le modèle NVT.

II.2.3.3 Les règles de négociation :

Des règles de négociation d'options permettent d'éviter des situations de bouclage (par exemple qu'une des parties envoie des requêtes de négociation d'options à chaque confirmation de l'autre partie).

1- Les requêtes ne doivent être émises que lors d'un changement de mode ;

2- Lorsqu'une des parties reçoit une requête de changement de mode, il ne doit la quitter que s'il ne se trouve pas déjà dans le mode approprié ;

3- Une requête ne doit être insérée dans le flux de données qu'à l'endroit où elle prend effet.

II.2.3.4 La négociation d'options Telnet :

DO WILL L'émetteur commence en utilisant l'option ;

WON'T L'émetteur ne doit pas utiliser l'option ;

WILL DO L'émetteur commence en utilisant l'option, après -avoir envoyé un DO ;

DON'T L'émetteur ne doit pas utiliser l'option ;

DON'T WON'T L'émetteur signale qu'il a désactivé l'option ;

WON'T DON'T L'émetteur signale que l'émetteur doit désactiver l'option.

II.2.4 Protocole RIP2 :

Introduction :

RIP2 est utilisé pour échanger des informations de routage. Il dérive d'un premier protocole développé par Xerox (RIP). Chaque machine qui utilise un protocole RIP2 a un processus qui envoie et reçoit des datagrammes transportés par de l'UDP port numéro 520.

II.2.4.1L'en-tête RIP2 :

La structure des trames RIP2 est décrite ci-dessous :

Commande	Version	Inutilisé
Identifiant de la famille d'adresses		Route tag
Adresse IP		
Masque de sous réseau		
Saut suivant		
Métrique		

Fig.II.9 En-tête RIP2.

II.2.4.2 Définition de différents champs :

Commande : utilisé pour définir le sujet du datagramme.

1 Requête ;

2 Réponse ;

3 Réservé (Utilisé par Sun Microsystems).

Version : numéro de la version RIP.

Identifiant de la famille d'adresses : indique quel type d'adresse est utilisé cela car RIP2 peut transporter d'autres informations de routage.

Route tag : (utilisé par RIP2 ; 0 pour RIP) attribue une route qui doit être préservée par une route. Ce champ permet de séparer les routes RIP internes des routes externes qui ont pu être importée d'un EGP ou d'un autre IGP.

Adresse IP : adresse IP de la destination.

Masque de sous réseau : (utilisé par RIP2; 0 pour RIP) masque du sous réseau destination.

Saut suivant : adresse IP à laquelle les paquets devront être envoyé au prochain saut.

Métrique : représente le coût total de la source à la destination (en nombre de sauts).

II.2.5 Protocoles SLIP :

SLIP signifie Serial Link Internet Protocol, traduisez protocole Internet de liaison en série. SLIP est le résultat de l'intégration des protocoles modems précédent à la suite de protocoles TCP/IP.

Il s'agit d'un protocole de liaison Internet simple n'effectuant ni contrôle d'adresse, ni contrôle d'erreur, c'est la raison pour laquelle il est vite devenu obsolète par rapport à PPP.

Données à transmettre	END

Fig.II.10 Trame SLIP.

II.2.6 Le protocole PPP: Point To Point Protocol :

Introduction :

PPP fut développé pour transférer des données sur des liens synchrones ou asynchrones entre deux points en utilisant HDLC comme base d'encapsulation et un Frame Check Séquence (FCS) HDLC pour la détection des erreurs. Cette liaison permet le full duplex et garantit l'ordre d'arrivée des paquets.

Une fonctionnalité intéressante de ce protocole est le multiplexage simultané de plusieurs protocoles de niveau 3 du modèle OSI.

Ce protocole encapsule des paquets IP, IPX et NetBEUI, dans des trames PPP, puis transmet ces paquets PPP encapsulés à travers la liaison point à point. PPP est donc utilisé entre un client distant et un serveur d'accès distant.

II.2.6.1 Format de la trame PPP :

Fanion	Adresse	Contrôle	Protocole	Données	FCS	Fanion
01111110	11111111	00000011	16 bits		16 bits	01111110

Fig.II.11 Trame PPP.

II.2.6.2 Les déférents champs :

Fanion : séparateur de trame.

Adresse : Le champ adresse correspond à une adresse HDLC, or PPP ne permet pas un adressage individuel des stations donc ce champ doit être à 0xFF (toutes les stations), toute adresse non reconnue fera que la trame sera détruite.

Contrôle : Le champ contrôle doit être à 0x03, ce qui correspond à une trame HDLC non numérotée. Toute autre valeur fera que la trame sera détruite.

Protocole : La valeur contenue dans ce champ doit être impaire, l'octet de poids fort étant pair. Ce champ identifie le protocole encapsulé dans le champ informations de la trame.

Informations : De longueur comprise entre 0 et 1500 octets, ce champ contient le datagramme du protocole supérieur indiqué dans le champ «protocole». Sa longueur est détectée par le drapeau de fin de trame moins 2 octets de contrôle.

FCS (Frame Check Sequence) : Ce champ contient la valeur du checksum de la trame. PPP vérifie le contenu du FCS lorsqu'il reçoit un paquet.

II.2.6.3 Noms de domaines DNS :

Le système de nom de domaine (DNS, *Domain Name System*) a été initialement défini dans les RFC (*Request for Comments*, demandes de commentaires) 1034 et 1035. Ces documents spécifient les éléments communs à toutes les implémentations des logiciels DNS, qui comprennent entre autres :

- Un espace de noms de domaines DNS, qui définit une structure hiérarchique des domaines permettant d'organiser les noms.
- Des enregistrements de ressources, qui mappent les noms de domaines DNS sur un type spécifique d'informations de ressources et sont utilisés lorsque le nom est inscrit ou résolu dans l'espace de noms.
- Des serveurs DNS, qui stockent les requêtes de noms portant sur des enregistrements de ressources et y répondent.

II.2.6.4 Domain Name System (DNS):

Est un système permettant d'établir une correspondance entre une adresse IP et un nom de domaine.

Quand un utilisateur souhaite accéder à un site, comme par exemple www.free.fr, son ordinateur émet une requête spéciale à un serveur DNS, demandant 'Quelle est l'adresse de www.free.fr ?'. Le serveur répond en retournant l'adresse IP du serveur.

Il est également possible de poser la question inverse, à savoir 'Quel est le nom de domaine de telle adresse IP ?'. On parle alors de résolution inverse.

II. 7 Conclusion :

Le logiciel d'un réseau est composé de protocoles, règles qui permettent à des processus de communiquer. Ces protocoles peuvent être sans connexion ou orientés connexion. La plupart des réseaux disposent d'une hiérarchie de protocoles dans laquelle chaque couche fournit des services à la couche supérieure sans avoir à connaître les détails des protocoles des couches inférieures. Les piles de protocoles sont le plus souvent fondées soit sur le modèle OSI soit sur le modèle TCP/IP. Ces modèles ont en commun les couches réseau, transport et application, les autres étant différentes.

Chapitre III :

Les Protocoles de Routage.

Introduction :

Dans ce chapitre, nous allons tenter de créer des réseaux entre des PC par le biais de routeurs, pour cela nous allons utiliser le logiciel GNS3 qui nous permettra d'avoir une topologie des deux petits réseaux virtuels que nous configurerons avec le protocole OSPF et que nous testerons à l'aide de commandes. Nous aurons à nous familiariser avec GNS3. Ce logiciel va nous permettre de construire une topologie simple du réseau. Ce réseau sera composé d'un Routeur, un Switch et deux PC connectés au Switch.

Mais d'abord, nous allons essayer d'avoir une vue globale sur quelques protocoles de routage et de commutation et leur fonctionnement.

III . Routage et Commutation IP:

III.1. Routage IP :

Cette opération consiste à transmettre les données d'un réseau à un autre en prenant le bon chemin. Pour cela, il faut un routeur.

Un routeur est un dispositif qui transmet les paquets de données entre les réseaux informatiques, un routeur peut être connecté à deux ou plusieurs lignes de données provenant de différents réseaux. Quand un paquet de données arrive à l'une des lignes, le routeur lit l'information d'adresse dans le paquet afin de déterminer sa destination finale. Puis, en utilisant les informations dans sa table de routage ou de la politique de routage, il dirige le paquet vers le réseau suivant de son voyage. Ils remplissent les fonctions "de mise en scène de la circulation" sur Internet. Un paquet de données est transmis généralement d'un routeur à un autre à travers les réseaux qui constituent l'inter-réseau jusqu'à ce qu'il atteigne son nœud de destination.

Le routage peut être dynamique ou statique.

1. **Le routage statique :**

   Dans ce type de routage chaque route est saisie manuellement par l'administrateur. Il est utilisé dans les tous petits réseaux. C'est l'un des moyens de routage les plus sûrs. Il est facile à gérer lorsque le nombre de routes reste limité. Lorsqu'une route est en panne, l'intervention de l'administrateur est obligatoire pour saisir une route de secours.

Le routage statique est un concept décrivant un moyen de sélection de chemin de configuration des routeurs dans les réseaux informatiques. C'est le type de routage caractérisé par une absence de communication entre les routeurs concernant la topologie en cours du réseau. Ceci est accompli en ajoutant manuellement des routes à la table de routage.

2. **Le routage dynamique :**

Ici, les routes sont calculées et saisies grâce à un protocole de routage. Il est utilisé dans les plus gros réseaux. Il est plus difficile à mettre en place, mais plus facile à maintenir. Lorsqu'une route est en panne, il recalcule automatiquement un autre chemin.

III.1.1 Protocoles de routage :

a. Protocole RIP :

RIP signifie Routing Information Protocol (protocole d'information de routage). Il s'agit d'un protocole de type Vector Distance (Vecteur Distance), c'est-à-dire que chaque routeur communique aux autres routeurs la distance qui les sépare (le nombre de saut qui les sépare). Ainsi, lorsqu'un routeur reçoit un de ces messages il incrémente cette distance de 1 et communique le message aux routeurs directement accessibles. Les routeurs peuvent donc conserver de cette façon la route optimale d'un message en stockant l'adresse du routeur suivant dans la table de routage de telle façon que le nombre de saut pour atteindre un réseau soit minimal. Toutefois ce protocole ne prend en compte que la distance entre deux machines en termes de saut, mais il ne considère pas l'état de la liaison afin de choisir la meilleure bande passante possible.

Il existe deux versions de RIP, Ripv1 et Ripv2, la deuxième étant une amélioration de la première. RIPv2 apporte des modifications à RIPv1.

RIP v1 est un algorithme de routage, considéré comme un IGP, basé sur des vecteurs distances. Chaque table de routage est complètement diffusée sur le réseau à intervalle de temps pré-déterminé (par défaut 30s). La distance maximale géré par cet algorithme est de 15 hops (16 = infini).

Quand un routeur reçoit des informations relatives à un réseau sur une interface appartenant au même réseau, mais à un sous-réseau différent, le routeur applique le masque de sous-réseau de cette interface.

Mais cette version avait des limites car elle n'envoie pas les masques de sous-réseau dans ses mises-à-jours, les mises-à-jour (updates) sont broadcastés, il n'y a pas d'authentification, elle ne supporte ni le VLSM ni le CIDR (Classless Inter Domain Routing), et utilise Split Horizon et le « holddown » pour détecter les boucles.

Quand à RIP v 2, il supporte le VLSM par la transmission d'un masque de sous-réseau avec les routes. Il permet l'authentification de la source d'une mise à jour de routage qui peut se fonder sur un texte clair ou sur un texte crypté avec l'algorithme MD5 ainsi que la mise à jour de routage par adresse multicast contrairement à RIP v1 qui diffuse ses mises à jour via l'adresse 255.255.255.255. RIP v2 transmet à l'adresse IP de destination multicast 224.0.0.9, réservée pour une utilisation par RIP v2.

b. Le protocole OSPF :

OSPF (*Open Shortest Path First*), un protocole de routage interne, il est plus performant que RIP et commence donc à le remplacer petit à petit. Il s'agit d'un protocole de type *protocole route-link* (que l'on pourrait traduire par Protocole à état des liens), ce qui signifie que, contrairement à RIP, ce protocole n'envoie pas aux routeurs adjacents le nombre de sauts qui les séparent, mais l'état de la liaison qui les sépare. De cette façon, chaque routeur est capable de dresser une carte de l'état du réseau et peut par conséquent choisir à tout moment la route la plus appropriée pour un message donné.

Nous pouvons penser qu'un lien est l'interface d'un routeur. L'état d'un lien est une description de cette interface et de la relation qu''elle entretient avec ses routeurs voisins. Une description de cette interface pourrait comprendre, par exemple, son adresse IP, le masque, le type de réseau connecté, les routeurs connectés, etc. L'ensemble de ces états de liens forme la « link-state database ». La « link-state database » ou « topology table », est identique sur tous les routeurs d'une zone.

De plus, ce protocole évite aux routeurs intermédiaires d'avoir à incrémenter le nombre de sauts, ce qui se traduit par une information beaucoup moins abondante, ce qui permet d'avoir une meilleure bande passante utile qu'avec RIP.

- **Quelques caractéristiques de OSPF:**

-Les routeurs OSPF entretiennent une relation orientée connexion avec les routeurs d'un même segment physique. Dans la terminologie OSPF, on parlera d*''adjacency*, en français, d'adjacence ou de contiguïté.

-Au lieu d'envoyer des mises à jour entières lors d'un changement topologique, OSPF envoie des mises à jour incrémentielles.

-OSPF n'est pas limité par une segmentation dépendante de l'adressage IP ou des sous-réseaux, il utilise la notion d'*area* pour désigner un groupe de routeurs.

-OSPF supporte entièrement les possibilités du VLSM et de la *summarization* manuelle des routes.

-Grâce à la possibilité de donner des rôles particuliers aux routeurs, la communication inter-routeurs est efficace.

-Bien qu'OSPF permette une communication *inter-area*, il reste un protocole de routage intérieur (IGP).

c. Le protocole BGP :

C'est un protocole de routage externe et un protocole de passerelle extérieure normalisée, par opposition à RIP, OSPF et EIGRP qui sont des protocoles de passerelle intérieure dont la version utilisée depuis 1994 est la BGPv4, les précédentes étant considérées obsolètes. BGP est un protocole de routage de type vecteur de chemin, il n'a pas été construit pour effectuer des routages au sein d'un système autonome, mais plutôt entre de tels systèmes car au sein d'un AS, le protocole de routage est qualifié d'« interne » (par exemple, Open shor test path first, abrégé en OSPF). Entre deux systèmes autonomes, le routage est «externe» (par exemple Border Gateway Protocol, abrégé en BGP).

BGP peut fournir des informations plus détaillées concernant chaque route et peut les utiliser pour sélectionner la meilleure voie. BGP appelle ces informations, attributs de chemin. Les possibilités de BGP permettent de mettre en œuvre une nouvelle structure de réseau constituée de systèmes autonomes équivalents pouvant évoluer davantage que l'ancienne structure hiérarchique. Les protocoles externes sont uniquement requis si un AS doit échanger

des informations d'acheminement avec un autre AS. La plupart des ordinateurs appartenant à un AS exécutent un protocole interne tel que RIP. Seules les passerelles assurant la connexion entre l'AS et un autre AS doivent exécuter un protocole de routage externe.

d. IS-IS (*Intermediate system to intermediate system*):

C'est un protocole de routage interne, à état de lien (link state) tout comme OSPF. C'est un protocole de routage dynamique basé sur l'algorithme de routage SPF de Dijkstra et e standard ISO 10589. Ce protocole est utilisé à l'intérieur d'un système autonome (contrairement à BGP qui s'utilise uniquement en routage externe et donc entre deux systèmes autonomes). Les routeurs IS-IS maintiennent une vue topologique commune. La base de données topologique est construite individuellement et ensuite partagée entre tous les routeurs.

III. 2 Commutation IP :

Commutation IP est la technique utilisée pour acheminer les paquets IP par la couche réseau en utilisant les architectures de commutation. Toutes les approches des technologies de commutation IP ont en commun une idée : remplacer au maximum les routeurs par des commutateurs pour profiter de leurs performances (une capacité de traitement de 20 millions de paquets par seconde contre seulement 500 000 paquets pour les routeurs les plus performants), tout en conservant une caractéristique fondamentale des routeurs : le calcul du chemin optimal entre deux ressources.

Un commutateur IP possède des mécanismes pour classer les paquets à transmettre.

- ✓ **Le protocole de Spanning Tree**

  « The Spanning Tree Protocol » (STP) ou algorithme de l'arbre recouvrant, est un protocole de couche 2 (liaison de données), il a été conçu pour les switchs et défini par l'IEEE. Son principe de fonctionnement est d'utiliser un algorithme qui calcule le meilleur chemin sans boucle à travers le réseau. STP détecte et désactive ces boucles et fournit un mécanisme de liens de sauvegarde. Son fonctionnement est basé sur la sélection d'un commutateur Root (principal) et de calculs des chemins les plus courts vers ce commutateur. Pour cela, un réseau interconnecté par des ponts peut être assimilé à un graphe. On peut y appliquer la théorie des graphes si dans ce graphe toutes les boucles sont supprimées, on a un chemin unique entre deux nœuds et on parlera d'arbre. Et si cet arbre passe par tous les nœuds, on parle d'arbre à

recouvrement total. L'algorithme spanning tree permet de superposer un arbre à recouvrement total sur un réseau de grand étendue et fournit des chemins redondants en les laissant à l'état bloqué. À intervalles réguliers, les commutateurs dans le réseau émettent et reçoivent des paquets spanning tree qu'ils emploient pour identifier le chemin. Si un segment de réseau devient inaccessible ou si les coûts spanning tree changent, l'algorithme spanning tree reconfigure la topologie spanning tree et rétablit la liaison en activant le chemin de réserve.

Rapid Spanning Tree (RSTP) est la version améliorée de STP qui fait passer les temps de convergence de 50 secondes à quelques secondes.

III.2.1 Les fonctions des ports :

Le protocole Spanning Tree assigne des rôles à chaque port selon la fonction du

port dans la topologie courante. Ces rôles sont :

- Designated port – port élu pour la connexion d'un segment dans le réseau.
- Root port – port utilisé pour la voie ascendante vers le root bridge.
- Alternate port – un port bloqué fournissant un chemin de réserve vers le port Root dans le spanning tree
- Backup port – un port bloqué dans une configuration loopback

III .2.2 Mode des ports sur les Switches :

-Blocking: Dans cette opération, le port provoque une boucle de commutation si elle est active, aucune donnée de l'utilisateur n'est envoyée ou reçue sur un port en état de blocage, mais il peut passer en mode de transfert si les autres liens en cours d'utilisation échouent. L'algorithme de spanning tree qui prévient l'utilisation des chemins en boucle, détermine si le port peut passer à l'état de transfert. Les données BPDU sont toujours reçues dans l'état de blocage.

-Listening : Le commutateur traite les données BPDU et attend de nouvelles informations qui le ferait revenir à l'état de blocage. Il ne remplit pas la table d'adresses MAC et il ne transmet pas les trames.

-Learning : Le commutateur rejette toutes les trames de données venant du segment attaché, il rejette également les trames de données venant d'un autre port de transfert. Le commutateur construit une table faisant correspondre les adresses MAC aux

numéros des ports**,** reçoit les BPDUs et les transmet à son système. Il envoie les BPDUs reçus de son système et répond à SNMP « ***Simple Network Management Protocol »***.

-Forwarding : Opération normale où le port reçoit et envoie des données. STP surveille toujours les données BPDU entrant qui pourrait indiquer qu'il devrait revenir à l'état de blocage pour empêcher une boucle.

-Disabled : Dans cet état qui n'est pas strictement une partie de STP, un administrateur peut manuellement désactiver un port s'il le souhaite.

-Le protocole ARP :

Le protocole ARP (Address Resolution Protocol) est un standard TCP/IP obligatoire défini dans la RFC 826, « Address Resolution Protocol (ARP) ». Le protocole ARP résout les adresses IP utilisées par le logiciel exploitant TCP/IP pour les adresses de contrôle d'accès au support utilisées par le matériel du réseau local. Le protocole **ARP** a un rôle phare parmi les protocoles de la couche Internet de la suite TCP/IP, car il permet de connaître l'adresse physique d'une carte réseau correspondant à une adresse IP.

Pour faire correspondre les adresses physiques aux adresses logiques, le protocole ARP interroge les machines du réseau pour connaître leur adresse physique, puis crée une table de correspondance entre les adresses logiques et les adresses physiques dans une mémoire cache.

III.3 VLANs :

VLAN, pour Virtual LAN, est un réseau informatique virtuel, logique et indépendant qui regroupe un ensemble de machines de façon logique et non physique.

Les réseaux VLAN permettent une séparation logique du réseau, offrent une sécurité car les informations sont encapsulées dans un niveau supplémentaire et éventuellement analysées. Ils offrent aussi l'avantage de réduire la diffusion du trafic sur le réseau.

Il y a trois types différents de VLAN :

III.3.1 Un VLAN de niveau 1 : (aussi appelés **VLAN par port**, en anglais *Port-Based VLAN*) définit un réseau virtuel en fonction des ports de raccordement sur le commutateur ;

III.3.2 Un VLAN de niveau 2 :(également appelé **VLAN MAC**, *VLAN par adresse IEEE* ou en anglais *MAC Address-Based VLAN*) consiste à définir un réseau virtuel en fonction des

adresses MAC des stations. Ce type de VLAN est beaucoup plus souple que le VLAN par port car le réseau est indépendant de la localisation de la station ;

III.3.3 Un VLAN de niveau 3 : on distingue plusieurs types de VLAN de niveau 3 :

- Le **VLAN par sous-réseau** (en anglais *Network Address-Based VLAN*) associe des sous-réseaux selon l'adresse IP source des datagrammes. Ce type de solution apporte une grande souplesse dans la mesure où la configuration des commutateurs se modifient automatiquement en cas de déplacement d'une station. En contrepartie une légère dégradation de performances peut se faire sentir dans la mesure où les informations contenues dans les paquets doivent être analysées plus finement.
- Le **VLAN par protocole** (en anglais *Protocol-Based VLAN*) permet de créer un réseau virtuel par type de protocole (par exemple TCP/IP, IPX, AppleTalk, etc.), regroupant ainsi toutes les machines utilisant le même protocole au sein d'un même réseau.
- **VLAN taggé (le marquage) :**

  Le marquage de trame LAN est une technologie qui est utilisée pour identifier le VLAN auquel le paquet appartient, c'est à dire reconnaître le VLAN d'origine d'une trame. Ce marquage peut être implicite ou explicite. Quand il est implicite, cela veut dire que le marquage est déduit des informations contenues dans la trame (par exemple l'adresse IEEE, le protocole, le sous-réseau IP), ou par son origine (port). Les données sont déjà là, et rien n'a été ajouté; et le commutateur doit simplement examiner les données dans l'en-tête de trame et implicitement décider à quel VLAN il appartient. Lorsque ce type de marquage est utilisé, aucune donnée supplémentaire n'a besoin être ajoutée à la trame par l'ordinateur émetteur. Les dispositifs fonctionnent comme ils le feraient normalement et ne « savent » pas s'ils sont sur l'un ou l'autre VLAN.

Lorsque le marquage implicite est utilisé, les données de trame qui sont généralement utilisées pour créer des règles d'association de VLAN sont:

- Protocole "Le protocole de réseau, comme IP ou AppleTalk.
- Adresse source de liaison des données, L'adresse matérielle de la source de l'image.

Nous ferons remarquer que les adresses matérielles, également appelées adresses MAC, sont des adresses uniques gravées dans la carte, à l'usine, lorsque la carte est fabriquée. Elles

fournissent un espace d'adressage plat, mais doivent être uniques et différentes de toute autre telle adresse à travers le monde.

- Couche protocole de plus haut niveau : En plus d'un type de protocole, tels que IP, une adresse identifiante de sous-réseau peut être utilisée pour identifier le VLAN auquel une image est associée.

On trouve ensuite le marquage explicite. Dans ce type de marquage l'information (par exemple un numéro de VLAN) est insérée dans la trame.

Le type de marquage dépend du type de VLAN.

Par exemple :

- Dans le cas d'un VLAN par port, le transfert d'une trame vers un autre commutateur ne conserve pas d'information sur l'appartenance à tel ou tel VLAN. Il est nécessaire de mettre en œuvre un marquage explicite des trames.
- Dans le cas d'un VLAN par adresse IEEE, il est possible d'envisager que la table de correspondance entre les adresses IEEE et les numéros de VLAN soit distribuée sur tous les commutateurs. C'est une solution lourde à laquelle on peut préférer un marquage explicite.
- Les VLAN de niveau 3 utilisent un marquage implicite. Il n'est pas nécessaire de marquer les trames sur les liaisons inter-commutateurs. L'analyse des trames dégradant les performances, il est là aussi préférable de marquer explicitement les trames.

Passons maintenant à la pratique.

III .4 Lancer GNS3 :

En lançant GNS3, la fenêtre suivante apparait.

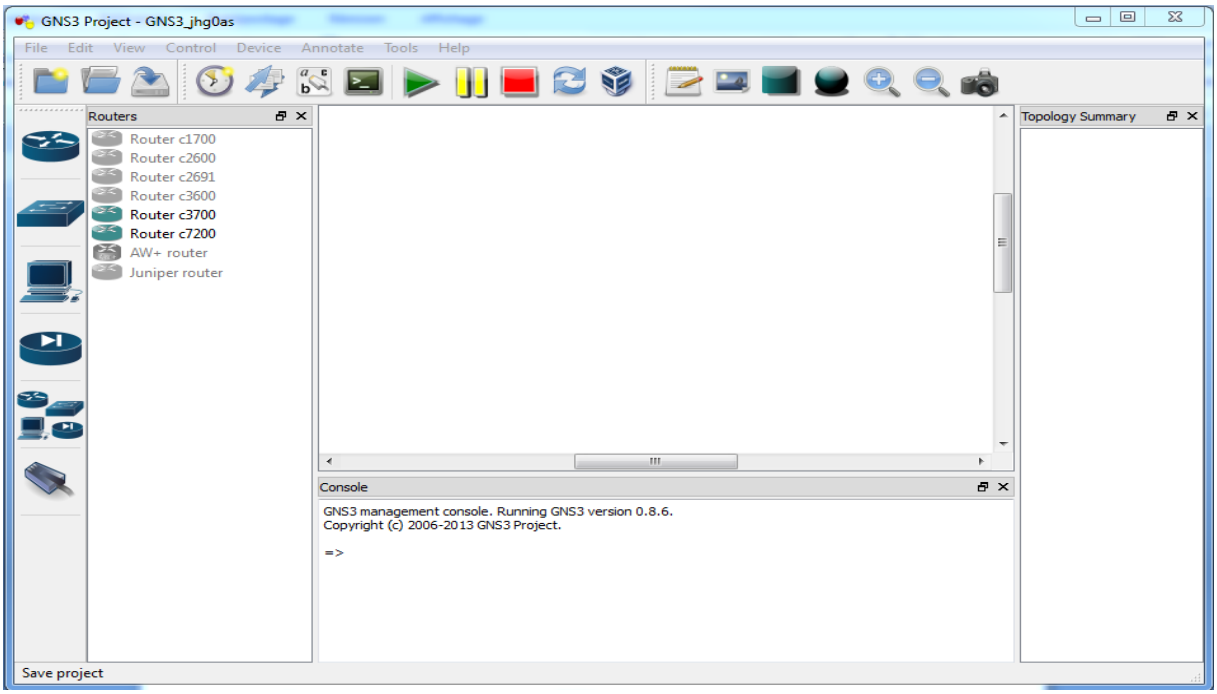


Fig.III.1 : Emulateur GNS3.

Il s'agit de la fenêtre principale pour GNS3, elle est divisée en quatre panneaux par défaut.

Le volet à gauche répertorie les types de nœuds disponibles. On voit affichées les icônes routeur pour les différentes plates-formes prises en charge, un pare-feu Pix (**PIX firewall)**, un commutateur Ethernet, un pont ATM, un commutateur ATM, un commutateur de relais de trame, et le « nuage » (Cloud). D'autres types de nœuds peuvent être ajoutés.

Le volet à droite fournit un résumé de la topologie qui sera mieux compris lorsque nous construirons des topologies plus complexes. En résumé, il présente les nœuds et les liaisons de chaque nœud.

La section du milieu contient deux volets. Le panneau supérieur est notre zone de travail où une topologie peut être graphiquement construite, et le volet inférieur, appelé la console, est utilisé pour Dynagen.

III.4.1Construire une topologie de réseau :

Pour commencer à construire une topologie de réseau, à partir de la liste du volet des nœuds, "Types de nœud", nous allons glisser le routeur c 2691 dans le volet du milieu de la topologie. Nous remarquerons que le routeur est désormais nommé «R0» par défaut.

Ensuite, nous allons glisser un autre routeur C2691 du volet de nœuds au volet de la topologie, ce nœud sera appelé "R1" par défaut.

La topologie devra maintenant être similaire à la figure de dessous.

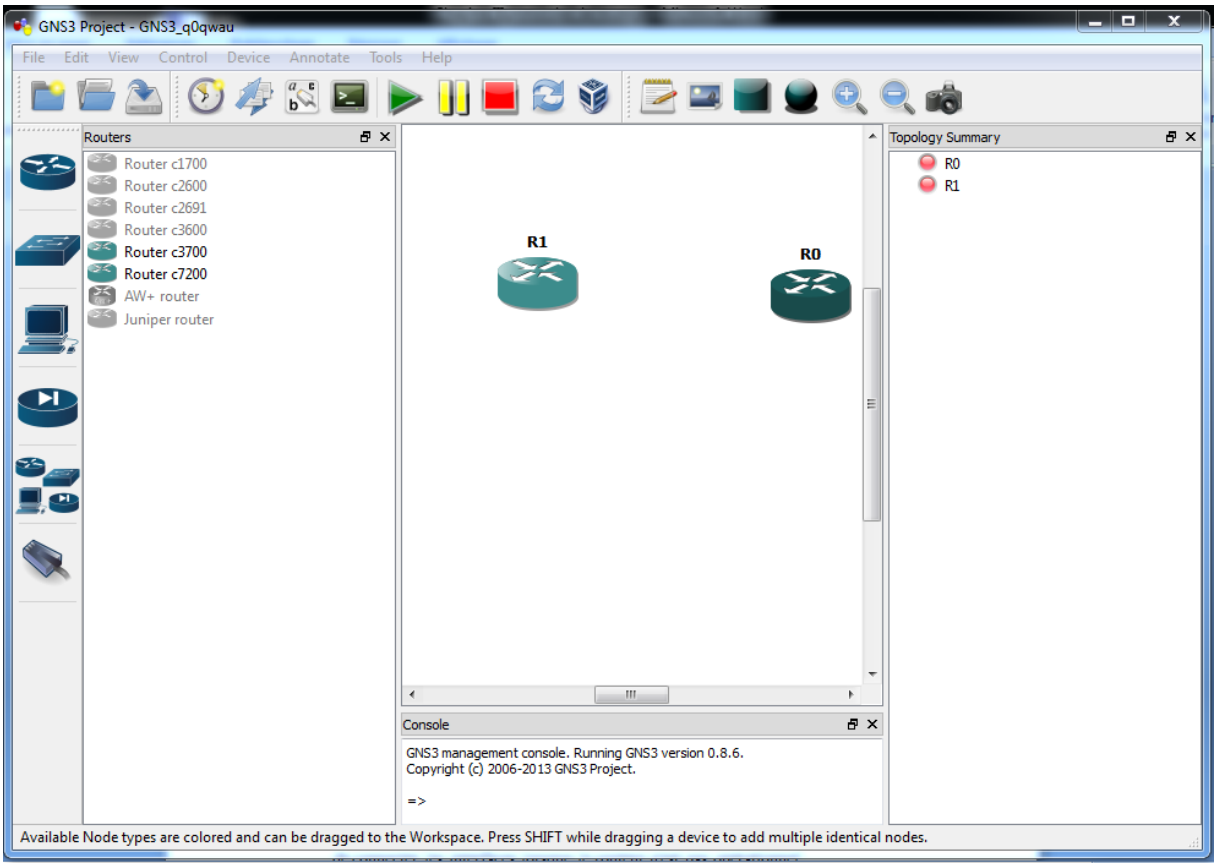


Figure. III.2 : La topologie des nœuds R1 et R0.

Note: les feux rouges dans le volet « Topology Summary » indiquent que les routeurs ne sont pas encore en cours d'exécution.

Dans cette application, nous allons utiliser le nœud "R0" comme interrupteur et "R1" comme routeur de sorte que, pour chaque nœud, il faudra une configuration spécifique.

III.4.2 Configurer les périphériques (Les ports physiques) :

La première étape pour que les Routeurs deviennent opérationnels est de configurer leurs « slots » et leurs ports physiques.

Tout comme avec un routeur physique normal, il est plus sûr d'insérer les modules et de connecter les interfaces lorsque le routeur n'est pas opérationnel.

Dans le volet de la topologie, on place le curseur / souris sur le nœud "R0", on appuie sur le bouton droit de la souris et on sélectionne l'option "configurer", qui apparaît dans le menu. Une fois sélectionné, la fenêtre de configuration des nœuds s'affiche.

On procède par placer les adaptateurs appropriés dans les emplacements disponibles dans le routeur.

On clique sur "R0" dans le volet gauche de la fenêtre. Ensuite dans le volet droit, on sélectionne l'onglet appelé «slots» puis on configure les deux cartes suivantes comme indiqué:

- Slot0: GT96100-FE

-Slot1: NM-16ESW

Le graphique suivant s'affiche.

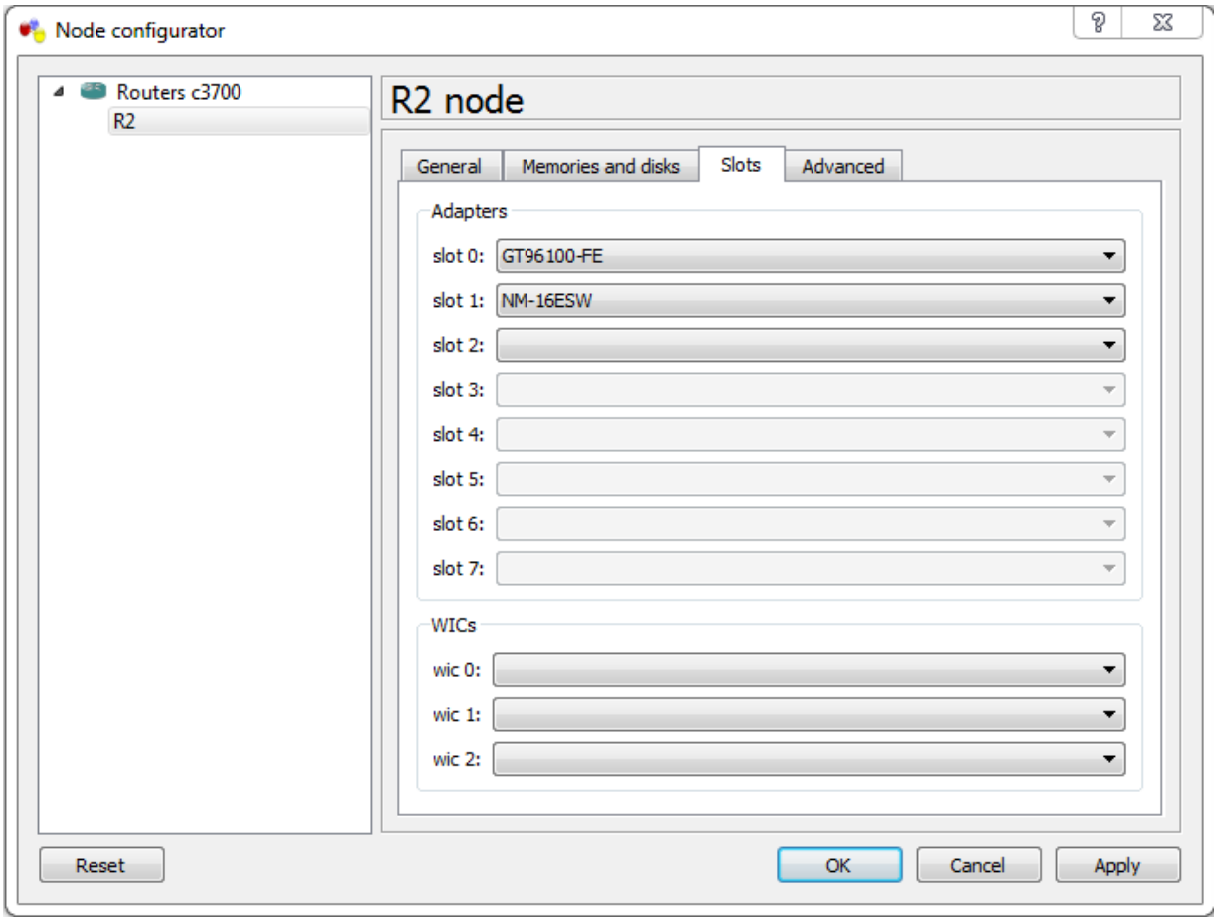


Figure III.3 : GNS3, Fenêtre de configuration des nœuds RO (configuration des slots).

Ensuite, on appui sur "appliquer" puis sur OK.

Cela configure le nœud du **Switch** "R0" avec deux ports Ethernet rapides dans "slot0" et le port 16 du module de commutation Ethernet dans "slot1"- à travers ces ports "switch" nous allons plutard connecter les PC et le routeur "R1".

Car ce nœud fonctionne comme un interrupteur dans notre topologie et pour ne pas nous tromper, nous allons changer maintenant le nom du nœud de "R0" à "Swith1". Pour ce faire, dans le volet de la topologie, on place le curseur / souris sur le nœud "R0", avec le bouton droit de la souris, on sélectionne l'option "changer le nom d'hôte" dans le menu qui apparaît. La fenêtre **(FigureIII.4.),** nous permet alors de changer le nom, on tape "switch1" et on appui sur OK.

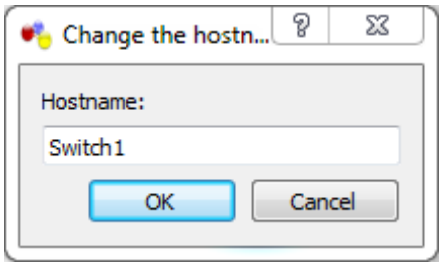


Figure III.4 : Changement du nom d'hôte pour R0.

Ensuite, nous allons configurer le nœud de routeur "R1". Dans le volet de la topologie, on place le curseur / souris sur le nœud" R1 ", on appui sur le bouton droit de la souris et on sélectionne l'option "configurer" dans le menu qui apparaît, la fenêtre de configuration des nœuds **(Figure III.5)** s'affiche.

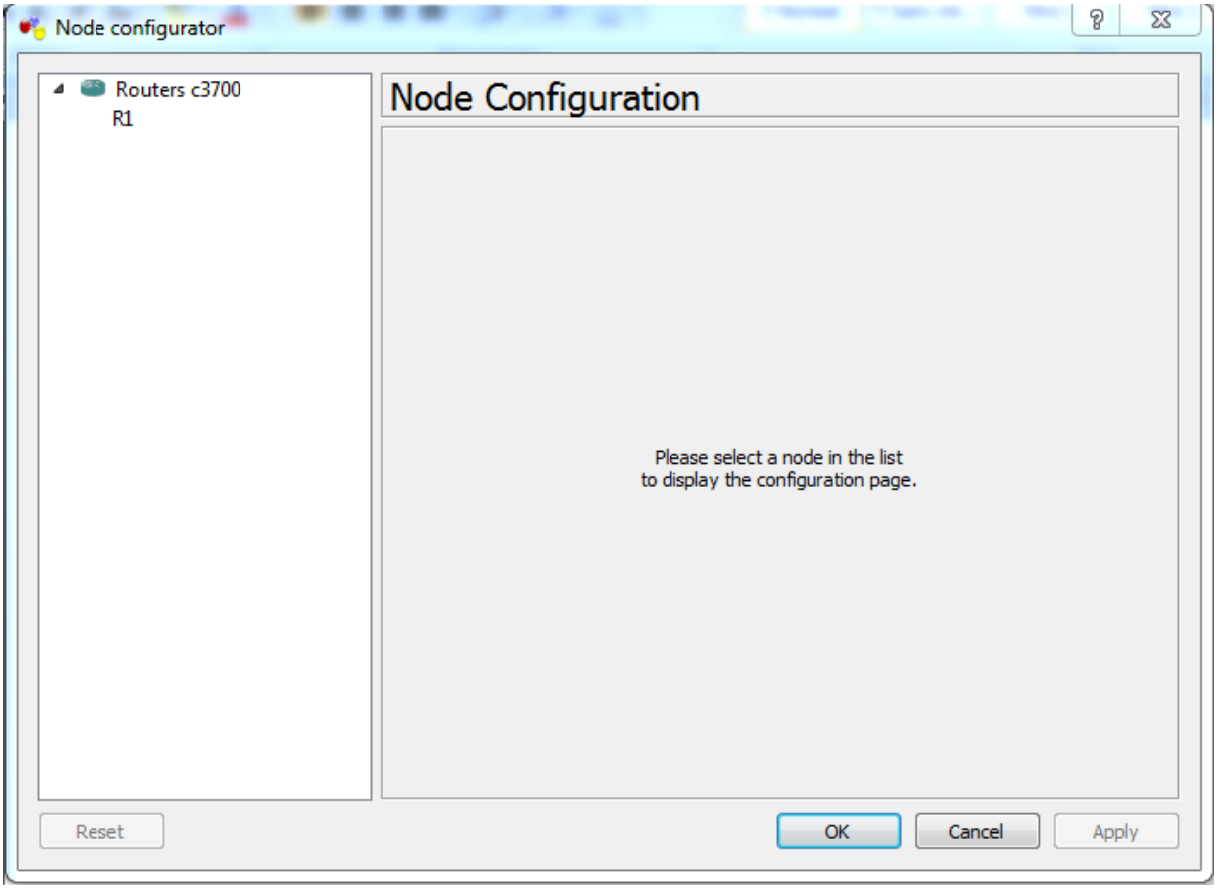


Figure III.5 : GNS3, Fenêtre de configuration du nœud pour R1.

On clique sur "R1" dans le volet de gauche de la fenêtre. Ensuite, dans le volet de droite, on sélectionne l'onglet «slots», puis on configure les adaptateurs suivants comme indiqué:

-Slot0: GT96100-FE

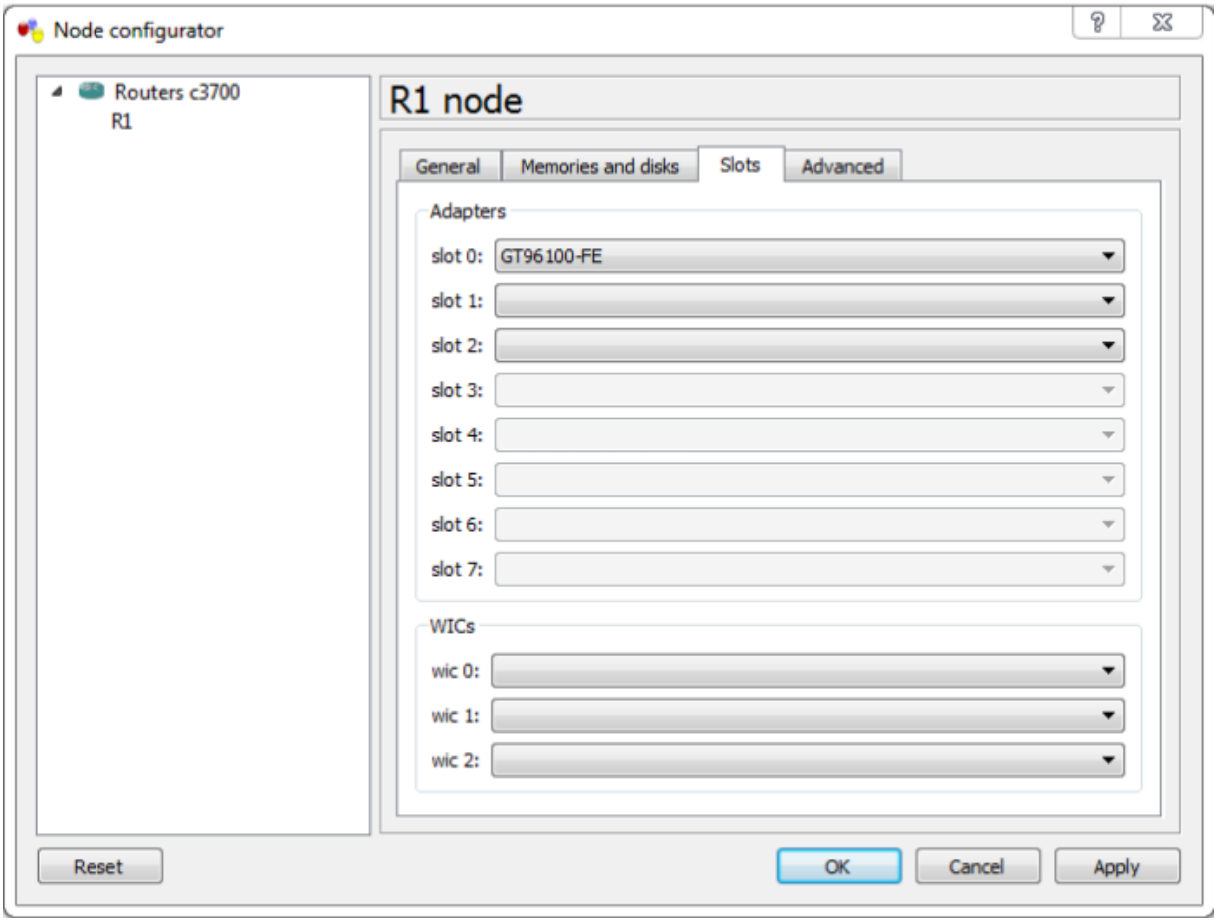


Figure III.6: GNS3, Fenêtre de configuration de nœud pour R1 (configuration du slot)

Cela configure le noeud du « **routeur** R1 » avec deux ports fast Ethernet dans « slot0 ». Dans ce nœud (**routeur),** le module de commutateur Ethernet port 16 n'est pas nécessaire dans « slot1 ». Comme avec "switch1", nous allons changer le nom du nœud de "R1" vers "**Router1**". Pour ce faire, dans le plan de la topologie, on place le curseur / souris sur le nœud "R1", on appuit sur le bouton droit de la souris et on sélectionne l'option "changer le nom d'hôte " dans le menu qui apparaît. La fenêtre **(Figure III.7)** nous permet alors de modifier le nom, on tape « Router1 » et on appuit sur OK.

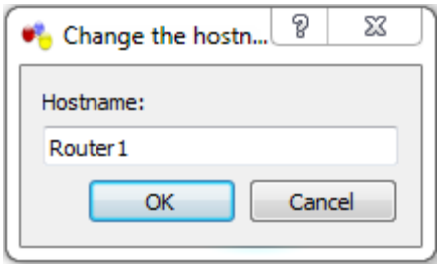


Figure III.7 : Changement du nom d'hôte pour R1.

Maitenant que le routeur et le commutateur ont leur configuration "physique", la tâche suivante consiste à apporter les deux PC et connecter tous les périphériques ensemble pour former la topologie requise.

III.4.7 Développer la topologie avec des PCs :

A partir du volet du nœud GNS3, nous allons faire glisser l'icône de l'ordinateur de la liste "Types de nœuds" vers le volet du milieu de la topologie et la positionner en haut à droite de « **Switch1** ». Nous remarquons que GNS3 nomme l'ordinateur "C0" par défaut. Nous glissons une autre icône de l'ordinateur de la liste "Types de nœuds" vers le volet de la topologie, et nous le positionnions en bas à droite de Switch1, ce nœud sera appelé "C1" par défaut.

La figure **(Figure III.8)** montre comment la topologie devrait apparaître à ce niveau.

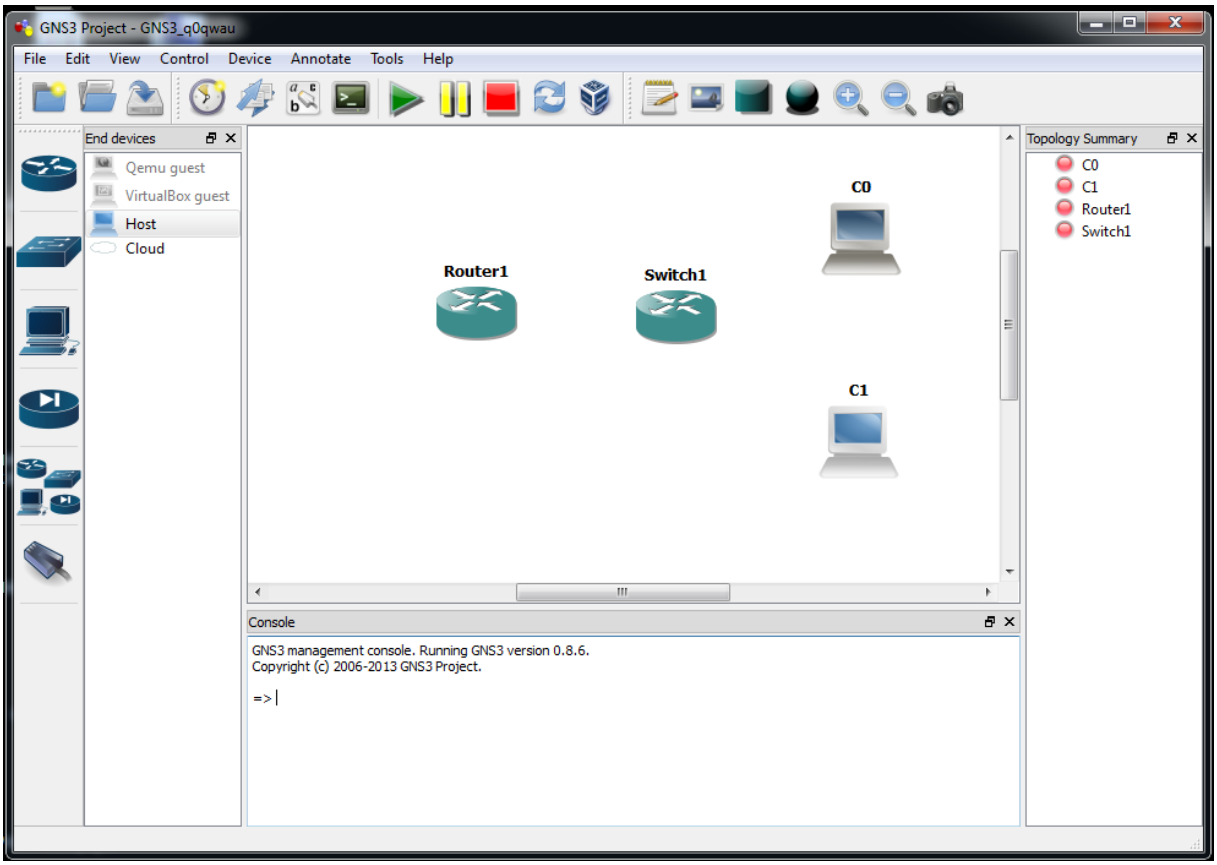


Figure III.8 : La topologie après inclusion des deux PC.

III.4.8 Configurer les PCs:

Comme pour les nœuds Switch1 et Router1, nous devons aussi "physiquement" configurer les PC. C'est un peu plus complexe que pour le Switch et le Routeur car nous devons avoir les ordinateurs configurés pour fonctionner avec un simulateur de PC virtuel appelé "VPCS".

Dans le volet de la topologie, on place le curseur / souris sur le nœud "C0", on clique avec le bouton droit de la souris et on sélectionne la rubrique "configurer" dans le menu qui apparaît, la fenêtre Configuration du nœud s'affiche. On clique sur "C0" dans le volet de gauche, puis on sélectionne l'onglet appelé «NIO UDP" dans le volet du côté droit.

On configure les paramètres comme indiqué ci-dessous:

- Port local: 30000

- L'hôte distant: 127.0.0.1

- Port distant: 20000

On clique sur le bouton Ajouter afin que les réglages apparaissent sous la rubrique «NIOs», ensuite on clique sur Appliquer puis sur OK comme indiqué dans la **Figure III.9**

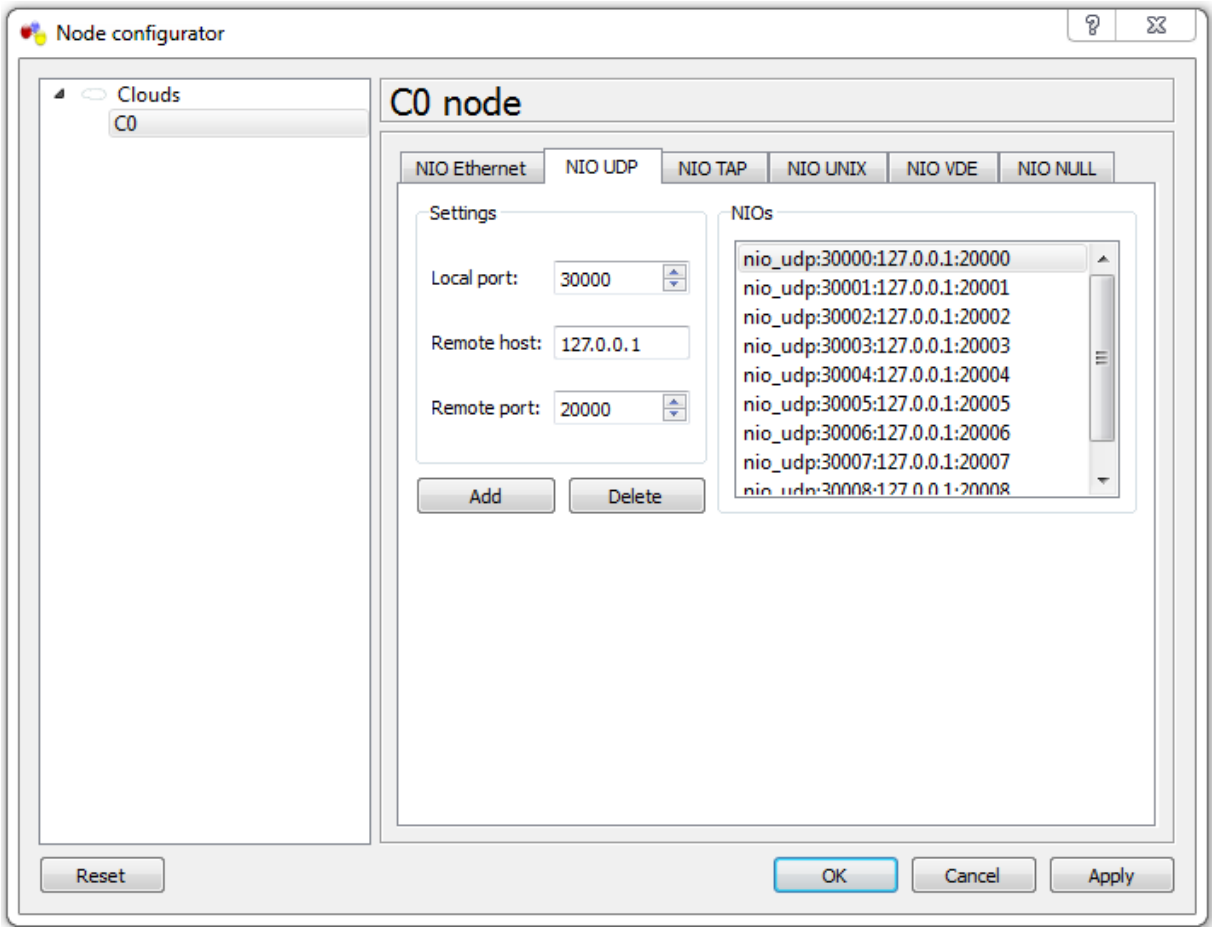


Figure III.9 : GNS3, Paramètres de configuration pour node C0

Maintenant nous allons passer à la configuration du deuxième ordinateur. Dans le volet de la topologie, on place le curseur / souris sur le nœud "C1", on appui sur le bouton droit de la souris et on sélectionne la rubrique "configurer" dans le menu qui apparaît, la fenêtre Node Configuration s'affiche. On clique sur "C1" dans le volet gauche, puis on sélectionne l'onglet intitulé «NIO UDP" dans le volet droit.

On configure les paramètres comme indiqué ci-dessous:

. port local: 30001

. L'hôte distant: 127.0.0.1

. Port distant: 20001

Ces paramètres correspondent au deuxième PC virtuel que nous allons configurer plus tard (via un programme distinct).

On clique sur le bouton Ajouter afin que les réglages apparaissent sous la rubrique «NIOs», ensuite on clique sur Appliquer puis sur OK comme indiqué dans la **Figure III.4.9**.

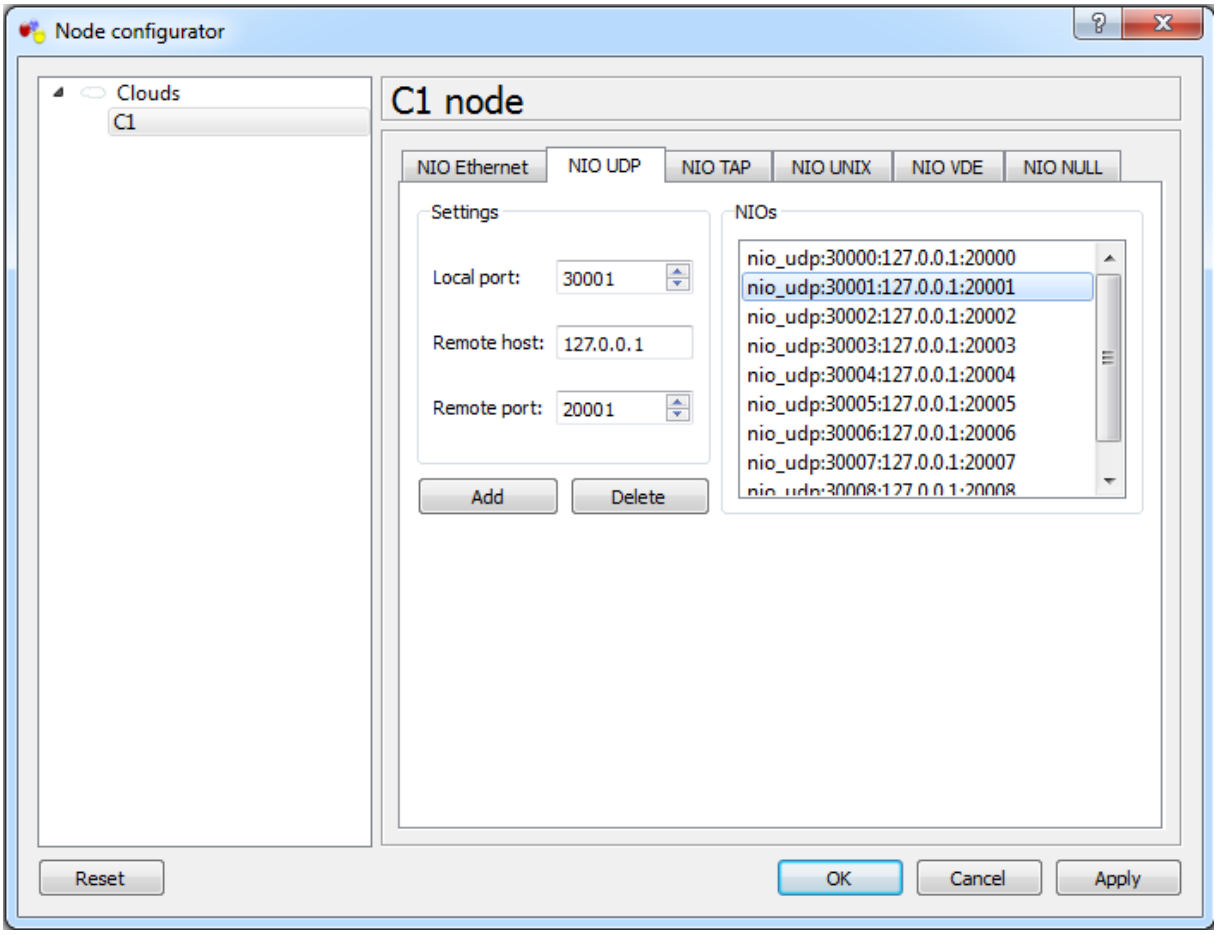


Figure III.10 : GNS3 : Paramètres de configuration pour node C1.

Maintenant, on change le nom à la fois pour C0 et C1 vers « PC1 » pour C0 et « 10C2 » pour C1.

Pour ce faire, (dans le volet de la topologie) on place le curseur / souris sur node "C0", on appuit sur le bouton droit de la souris et on sélectionne l'option "changer le nom d'hôte" dans le menu qui apparaît. La fenêtre nous permet alors de modifier le nom, on tape "PC1" et on appuit sur OK.

Dans le volet de la topologie, on place le curseur / souris sur le noeud "C1", avec le bouton droit de la souris on sélectionne l'option «Change the hostname » dans le menu qui apparaît. La fenêtre nous permet alors de modifier le nom, on tape «PC2» et on clique sur OK.

III.4.10 Connecter la topologie physique :

Maintenant, nous allons connecter les appareils entre eux à partir d'un petit réseau qui correspond à la topologie que nous voulons.

Dans le menu de symbole GNS3 on sélectionne l'outil de connexion (ressemble à une souris mais représente en réalité la fin d'un câble). Le bouton est encerclé dans la figure ci-dessous **(Figure III.11).**

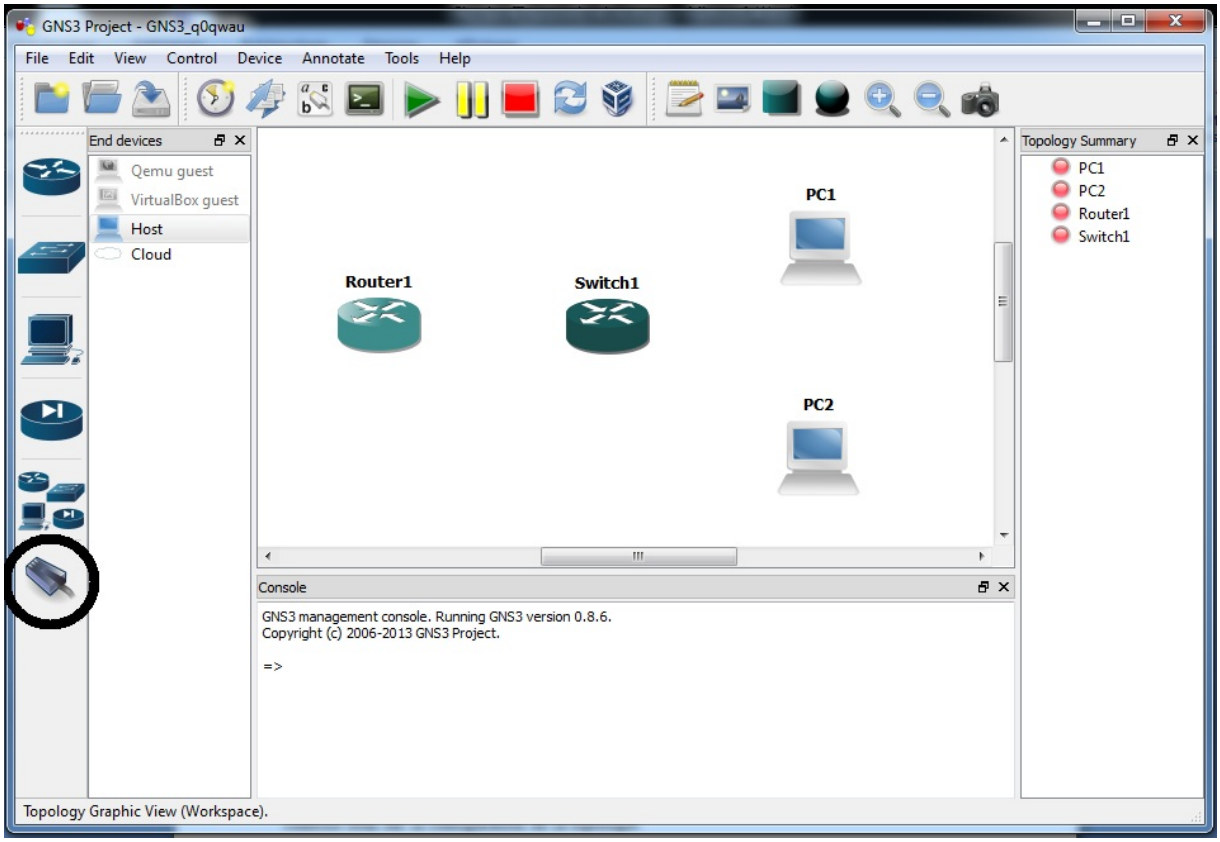


Figure III.11:GNS3 : Outil de connexion.

Dans la boîte de menu qui apparaît, on sélectionne l'option de connexion manuelle. Remarque: il est préférable de choisir la connexion manuelle, de sorte que nous avons un contrôle total sur la configuration de la topologie.

Ensuite, on sélectionne le premier périphérique à connecter, on clique sur **Router1** et on sélectionne l'interface **f0/0** de la liste des interfaces, on fait glisser la ligne qui apparaît vers **Switch1** puis on sélectionne l'interface **f1/15** à partir de la liste d'interface de **Switch1**, la topologie va montrer maintenant une ligne entre les deux nœuds pour indiquer la connexion.

Ceci est illustré sur la **Figure III.12**.

Note: Nous avons sélectionné slot1 sur le nœud du Switch, car c'est l'emplacement qui contient notre module de commutation (NM-16ESW). Nous devrions également relier les deux PCs à ce module de commutation.

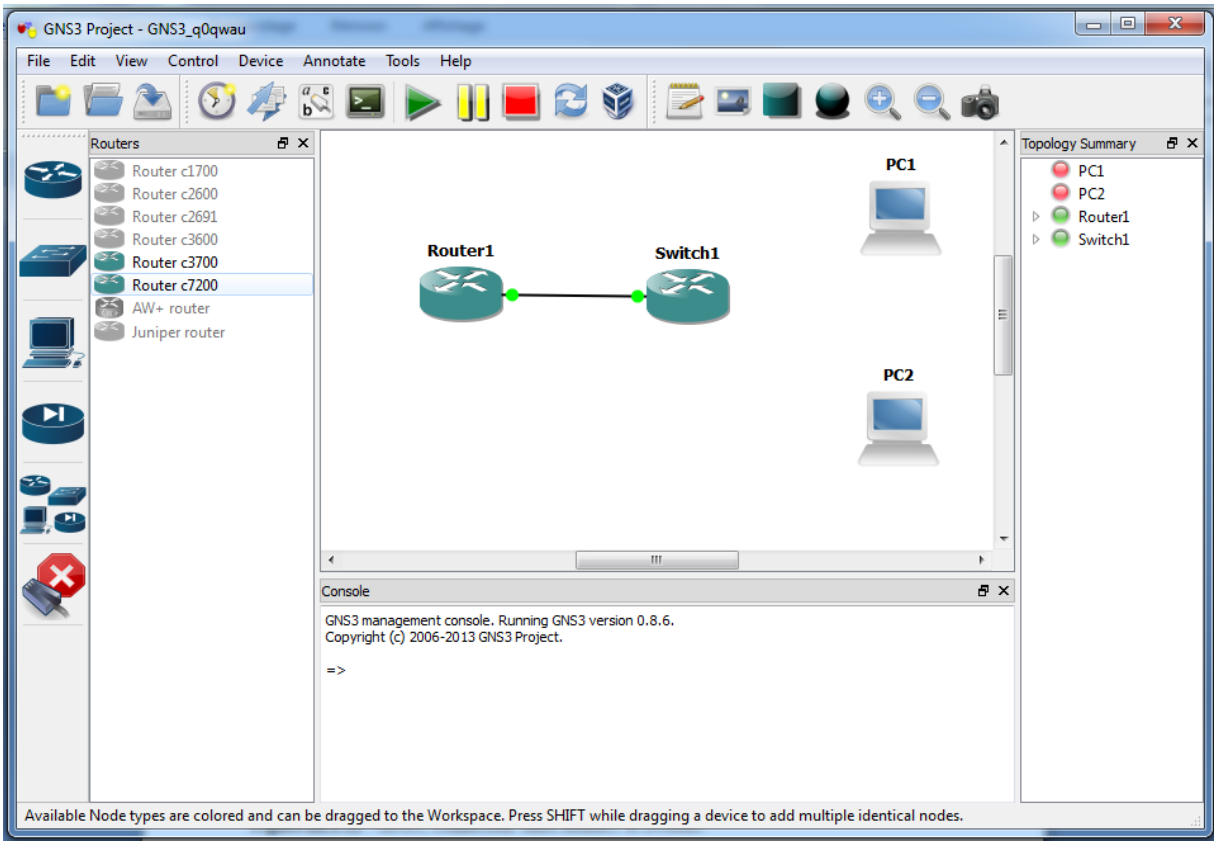


Figure III.12 : GNS3, Connection entre Router1 et Switch1.

Ensuite, nous allons connecter Switch1 aux deux PCs. Comme précédemment, on clique d'abord sur Switch1, on sélectionne l'interface **f1/0** de la liste des interfaces et on le connecte à l'interface « **udp nio-: 30000: 127.0.0.1: 20000** » dans PC1. L'interface « udp nio-: 30000: 127.0.0.1: 20000 » fait référence à l'instance virtuelle de PC que nous avons configuré précédemment.

Ensuite, nous allons faire la même chose pour l'interface **f1/1** de **Switch1**, sauf quand il s'agit de connecter cette interface à l'interface "udp nio-: 30001: 127.0.0.1:20001" sur PC2.

La topologie devrait apparaître comme indiqué ci-dessous **(Figure III.13).**

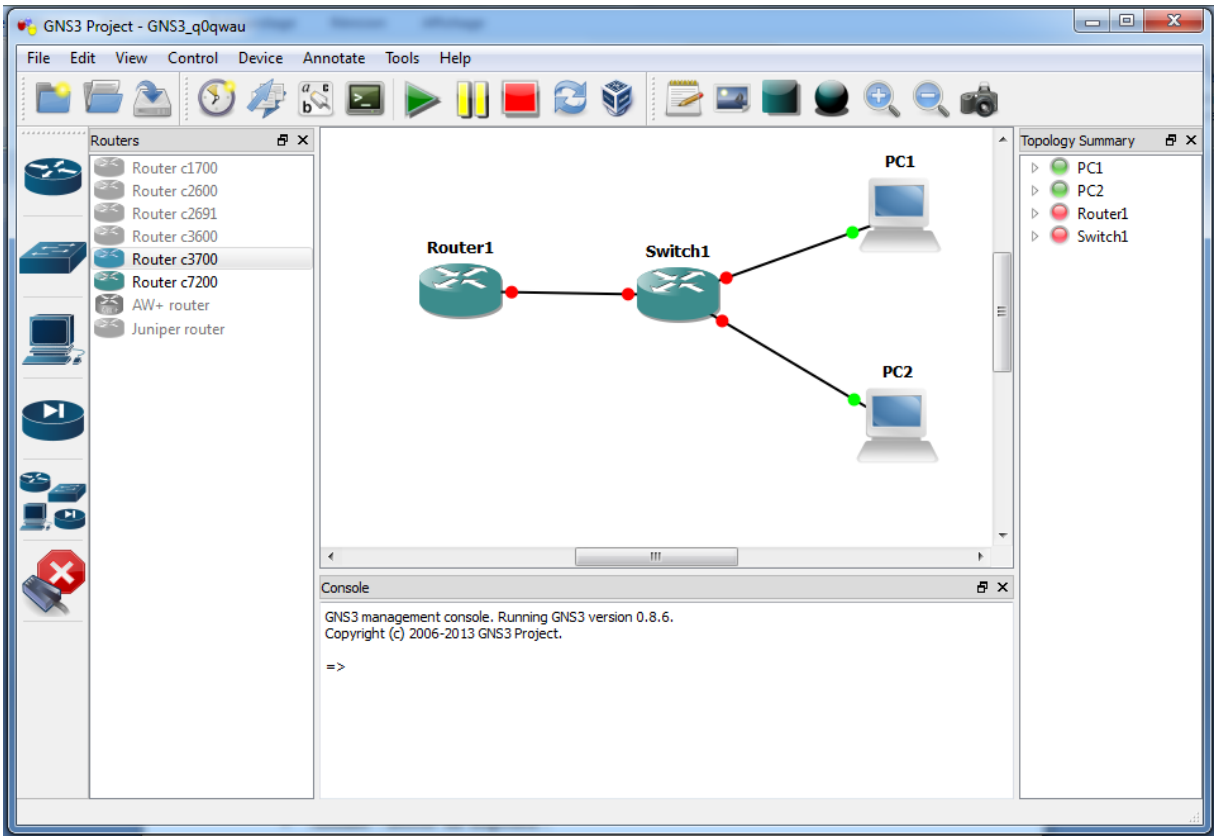


Figure III.13 : Topologie avec Router1, Switch1 et les PC connectés.

III.4.11 Allumer / Booter un dispositif :

Jusqu'à ce stade, nous avons uniquement connecté des dispositifs qui ne sont pas en cours d'exécution, nous allons donc maintenant démarrer le routeur et le switch!

Pour démarrer les périphériques on appui sur la flèche verte dans la barre d'outils GNS3 (au-dessus du volet de la topologie), ce bouton est entouré dans la figure ci-dessous **(Figure .III.14).**

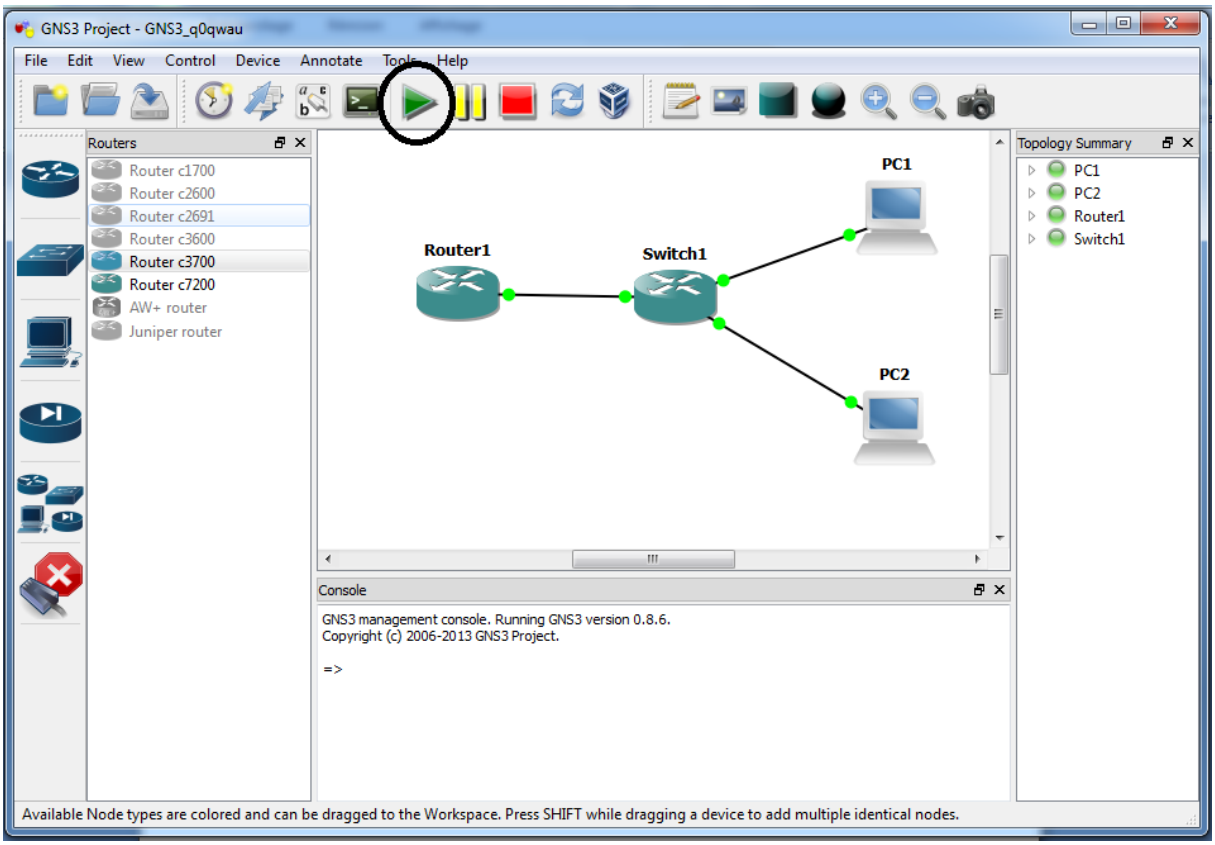


Figure III.14: GNS3, Boutton de démarrage.

Au moment où les dispositifs démarrent nous remarquons deux choses:

- Dans la fenêtre principale de la topologie, les points rouges figurant sur les connexions pour chaque nœud doivent maintenant devenir des points verts (comme le lien devient actif).
- Dans le volet nommé Topology Summary du Switch et du Routeur doivent maintenant apparaître des lumières vertes (une fois que le dispositif est entièrement démarré) avant leurs noms (pour indiquer que chaque dispositif est en cours d'exécution).

III.4.12 Lancement de la console WINDOWS

Nous allons maintenant lancer la console Windows. On clique sur le bouton "telnet to routeur" dans la barre d'outils GNS3 (situé à gauche de la flèche verte). Cela permet de lancer les fenêtres de la console pour Router 1 et Switch1.

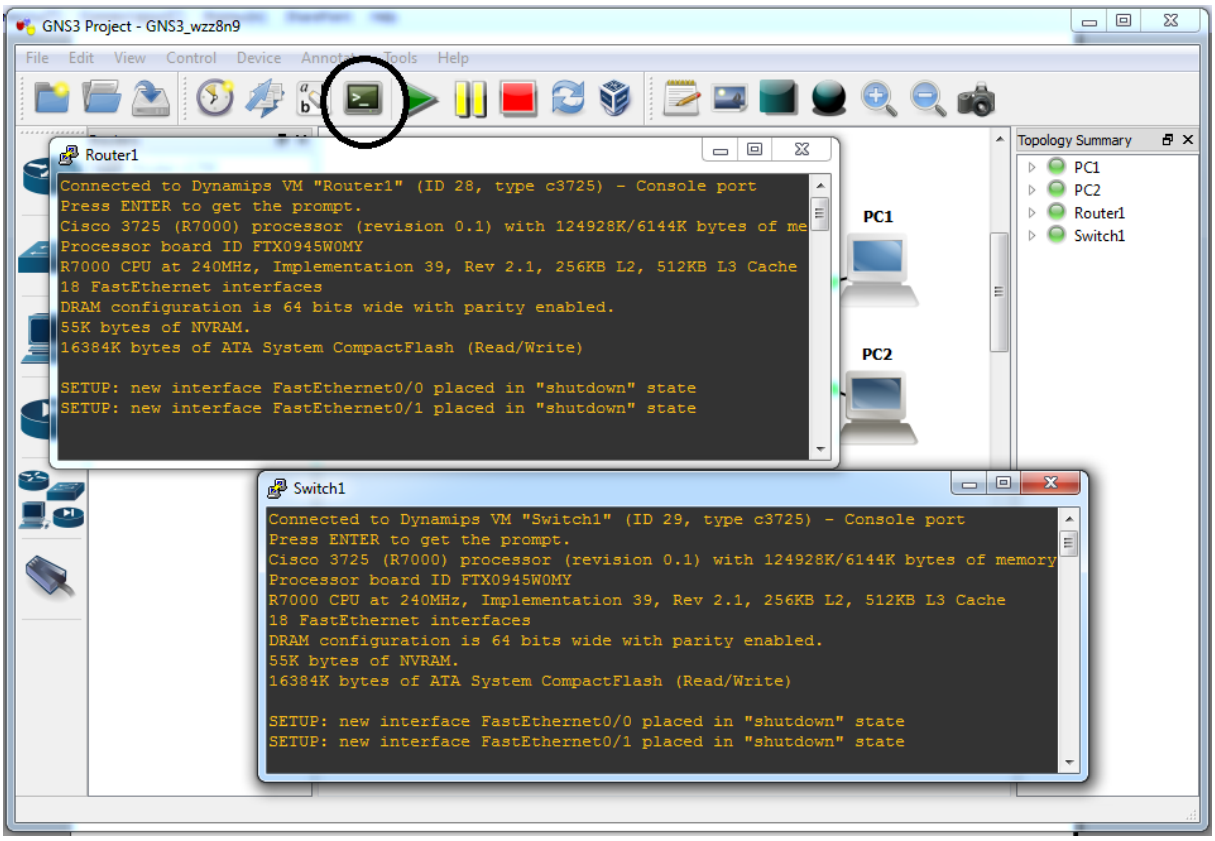


Figure III.15 Fenêtres Telnet pour Switch1 et Router1.

Nous pouvons remarquer que Router 1 et Switch1 s'amorcent en arrière-plan, comme c'est indiqué par des changements à l'information figurant dans les fenêtres de console.

Router1 et Switch1 nous aideront à répondre à une question de configuration:

"Voulez-vous entrer dans la boîte de dialogue de configuration initiale? [Oui / non]".

On répond non **"n"** à la question et puis on appui sur retour.

On appuit sur Entrée à nouveau lorsque le routeur nous invite à:

"appuyez sur Entrée pour commencer!" <entrer>

Remarque: Les anciens feux rouges dans le volet sommaire de la topologie (le volet le plus à droite) devrait maintenant être verts, indiquant que Router 1, Switch1 et les deux PC sont maintenant en cours d'exécution.

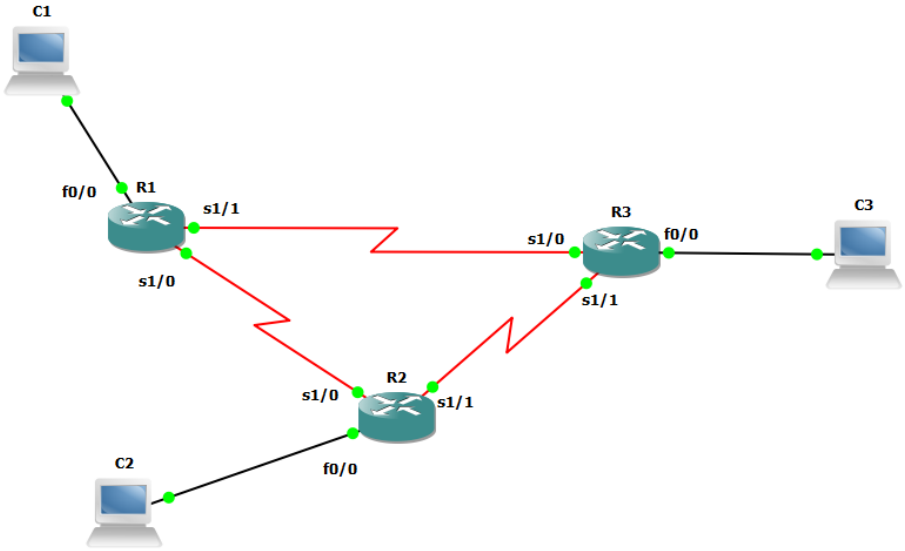


Figure III.16 L : topologie de réseau.

III.5 Nous commençons par configurer les routeurs :

Router1

Router> enable

Router #configure terminal

Router(confing)#hosname Router1

Router1(config)# int f0/0

Router1(config-if)# ip address 192.168.1.1 255.255.255.128

Router1(config-if)# no shutdown

Router1(config-if)# exit

Router1(config)# int s1/0

Router1(config-if)# ip address 172.16.1.1 255.255.255.252

Router1(config-if)# clock rate 56000

Router1(config-if)# no shutdown

Router1(config-if)# exit

Router1(config)# int s1/1

Router1(config-if)# ip address 172.16.1.9 255.255.255.252

Router1(config-if)# clock rate 56000

Router1(config-if)# no shutdown

Router1(config-if)#end

Router2

Router> enable

Router #configure terminal

Router(confing)#hosname Router2

Router2(config)# int f0/0

Router2(config-if)# ip address 192.168.2.1 255.255.255.128

Router2(config-if)# no shutdown

Router2(config-if)# exit

Router2(config)# int s1/0

Router2(config-if)# ip address 172.16.1.2 255.255.255.252

Router2(config-if)# clock rate 56000

Router2(config-if)# no shutdown

Router2(config-if)# exit

Router2(config)# int s1/1

Router2(config-if)# ip address 172.16.1.5 255.255.255.252

Router2(config-if)# clock rate 56000

Router2(config-if)# no shutdown

Router2(config-if)#end

Router3

Router> enable

Router #configure terminal

Router(confing)#hosname Router3

Router3(config)# int f0/0

Router3(config-if)# ip address 192.168.3.1 255.255.255.128

Router3(config-if)# no shutdown

Router3(config-if)# exit

Router3(config)# int s1/0

Router3(config-if)# ip address 172.16.1.10 255.255.255.252

Router3(config-if)# clock rate 56000

Router3(config-if)# no shutdown

Router3(config-if)# exit

Router3(config)# int s1/1

Router3(config-if)# ip address 172.16.1.6 255.255.255.252

Router3(config-if)# clock rate 56000

Router3(config-if)# no shutdown

Router3(config-if)#end

III.6 Configuration des PC :

PC1

VPCS9>1

VPCS1> IP 192.168.1.2 255.255.255.128

PC2

VPCS1>2

VPCS 2> IP 192.168.2.2 255.255.255.128

PC3

VPCS2>3

VPCS3> IP 192.168.3.2 255.255.255.128

Maintenant nous passons à la configuration OSPF de ces routeurs:

III.7 Configuration OSPF :

Nous configurons OSPF sur tous les routeurs. La commande « router ospf 100 » informe le routeur qu'il doit activer OSPF et les commandes « réseau » informent le routeur sur quelle interface activer OSPF (et donc quel réseau diffuser).

Router1

```
Router1#configure terminal
Router1(config)#router ospf 100
Router1(config-router)#network 192.168.1.0  0.0.0.255 area 0
Router1(config-router)#network 172.16.1.1  0.0.0.0 area 0
Router1(config-router)#network 172.16.1.9  0.0.0.0 area 0
Router1(config-router)#exit
Router1(config)#int s1/0
Router1(config-if)#no shutdown
Router1(config-if)# int s1/1
Router1(config-if)#no shutdown
Router1(config-if)#end
```

Router2

```
Router2#configure terminal
Router2(config)#router ospf 100
Router2(config-router)#network 192.168.2.0  0.0.0.255 area 0
Router2(config-router)#network 172.16.1.2  0.0.0.0 area 0
Router2(config-router)#network 172.16.1.5  0.0.0.0 area 0
Router2(config-router)#exit
Router2(config)#int s1/0
Router2(config-if)#no shutdown
Router2(config-if)# int s1/1
Router2(config-if)#no shutdown
Router2(config-if)#end
```

Router3

Router3#configure terminal

Router3(config)#router ospf 100

Router3(config-router)#network 192.168.3.0 0.0.0.255 area 0

Router3(config-router)#network 172.16.1.6 0.0.0.0 area 0

Router3(config-router)#network 172.16.1.10 0.0.0.0 area 0

Router3(config-router)#exit

Router3(config)#int s1/0

Router3(config-if)#no shutdown

Router3(config-if)# int s1/1

Router3(config-if)#no shutdown

Router3(config-if)#end

Enfin, nous allons effectuer des tests à l'aide de commandes pour vérifier l'activité du protocole (OSPF) sur ces routeurs.

On vérifie que les routeurs fonctionnent sous OSPF en utilisant la commande

« **show ip protocols »**

Router1 :

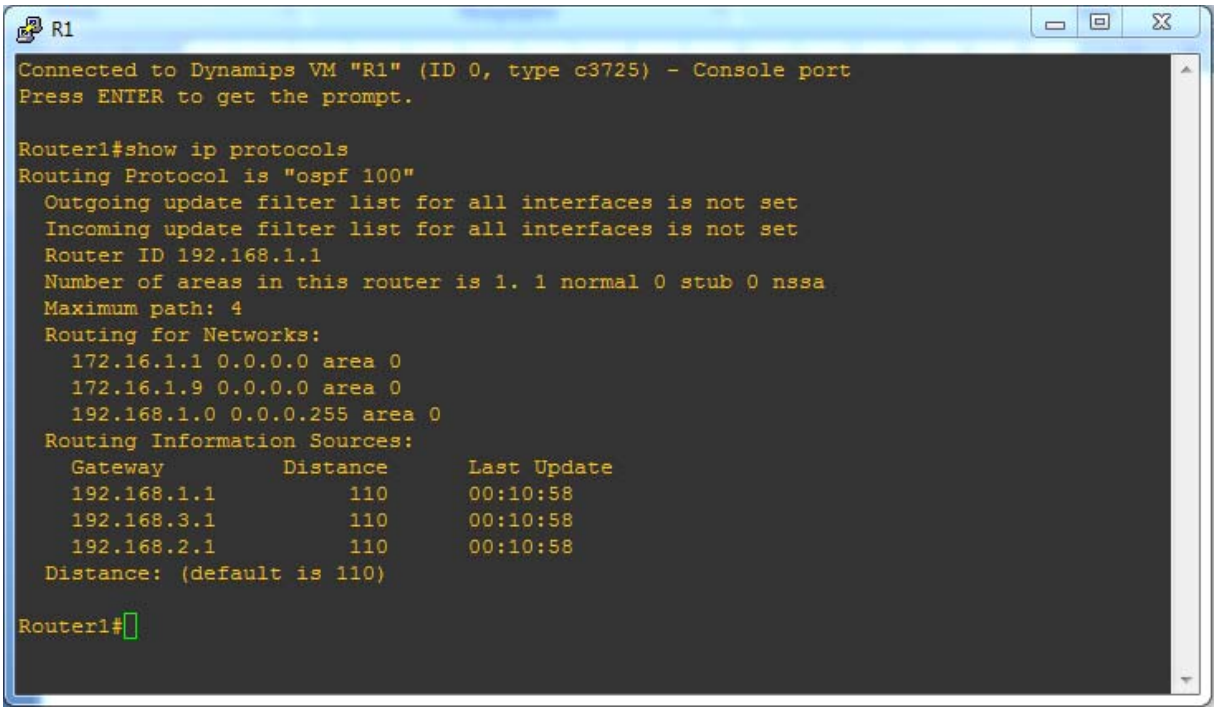


Figure III.17 Router1 :show ip protocols (OSPF active).

Router2:

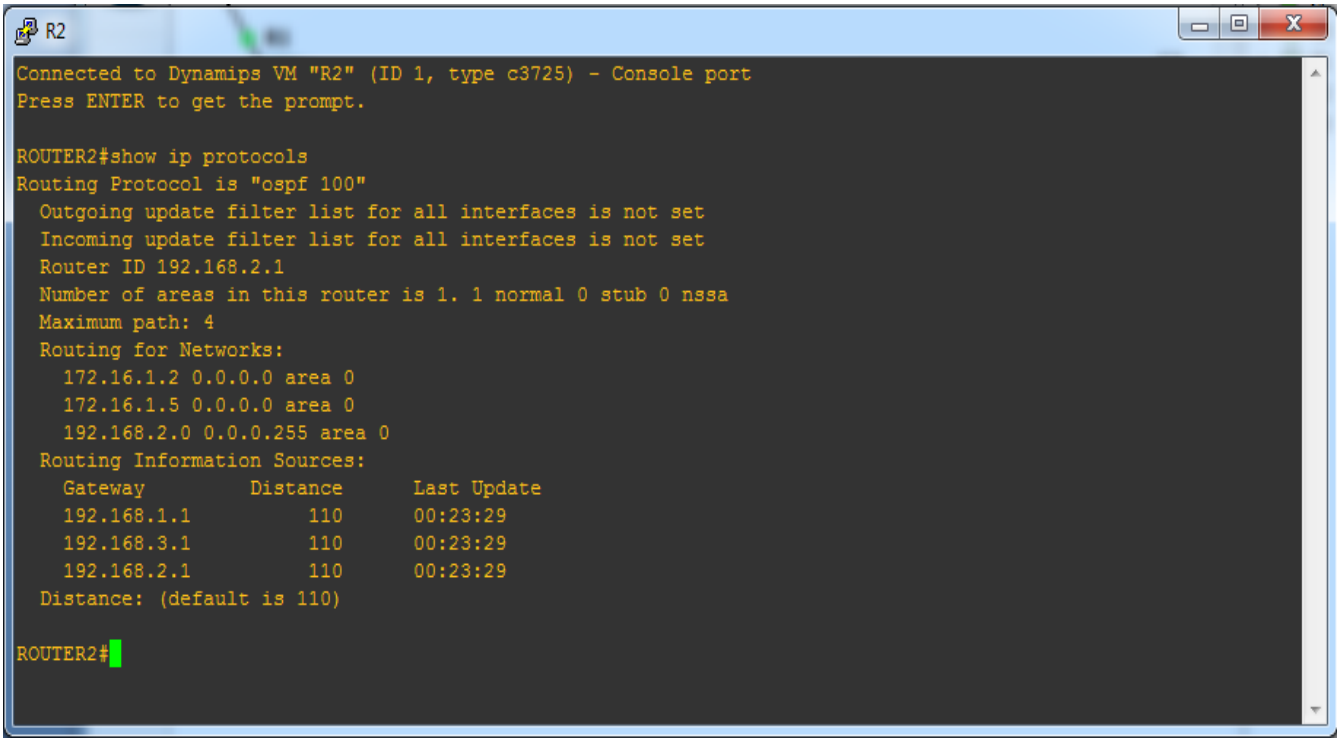


Figure III.18 Router2 :show ip protocols (OSPF active).

Router3:

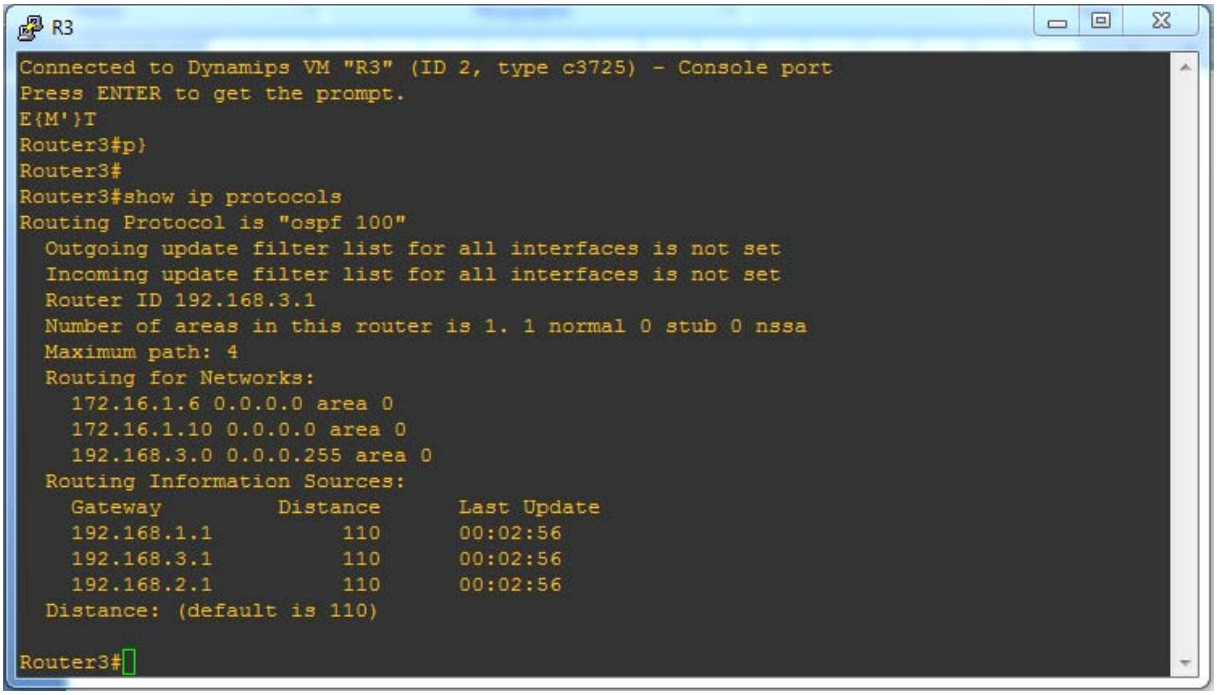


Figure III.19 Router3 :show ip protocols (OSPF active).

On affiche la table de routage IP et on vérifie les routes IP pour chaque routeur.

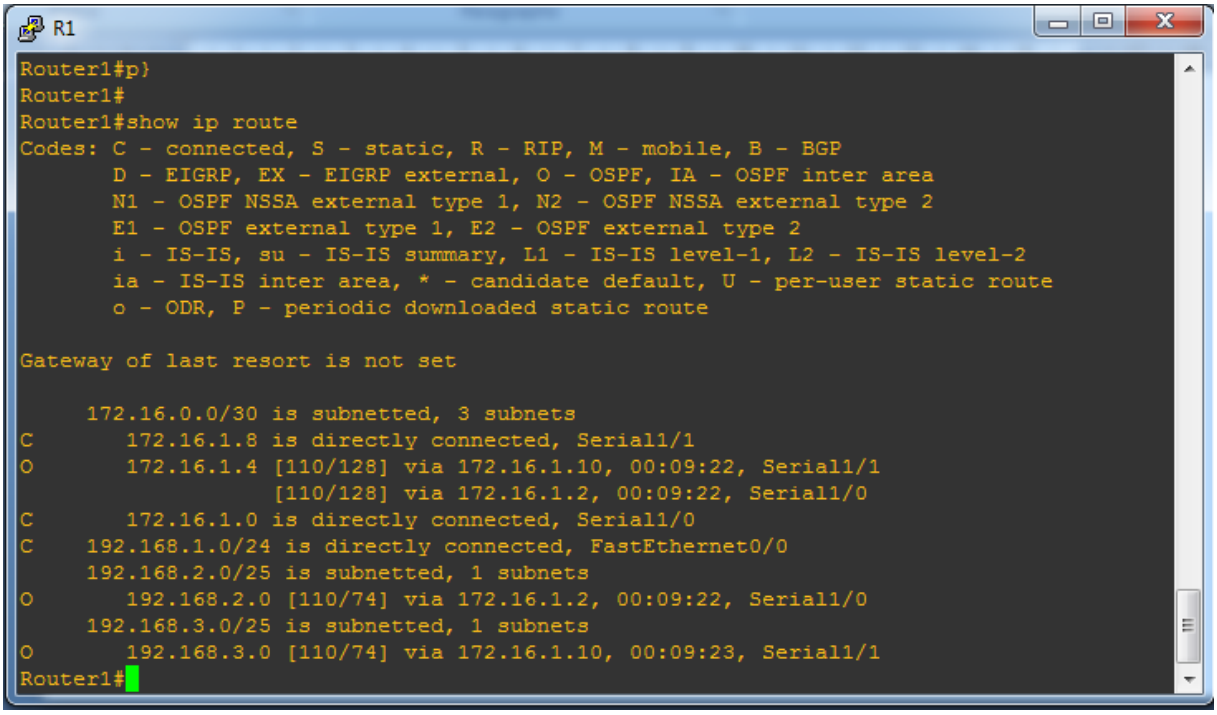


Figure III.20 Router1 : show ip route.

Affichage la table de routage IP **(show ip route)** et affichage des paramètres d'interface **(show ip ospf brief)** sur le même routeur (Router1).

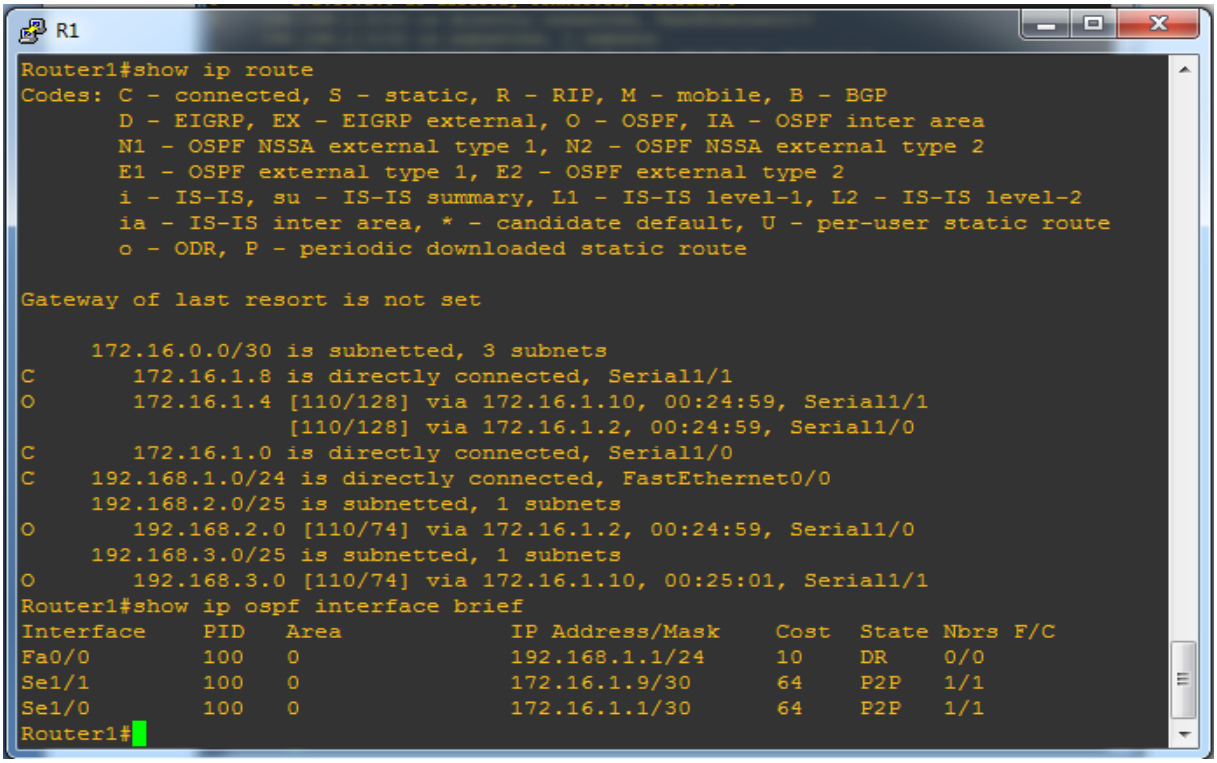


Figure III.21: Router1: show ip ospf interface brief.

Selon les normes, chaque router au sein de la même zone de protocole OSPF doit posséder une base de données OSPF identique.

Donc, pour chaque routeur, on affiche la base de données OSPF via la commande **« show ip ospf database »** et on compare les contenus de ces bases de données.

Router1#show ip ospf database

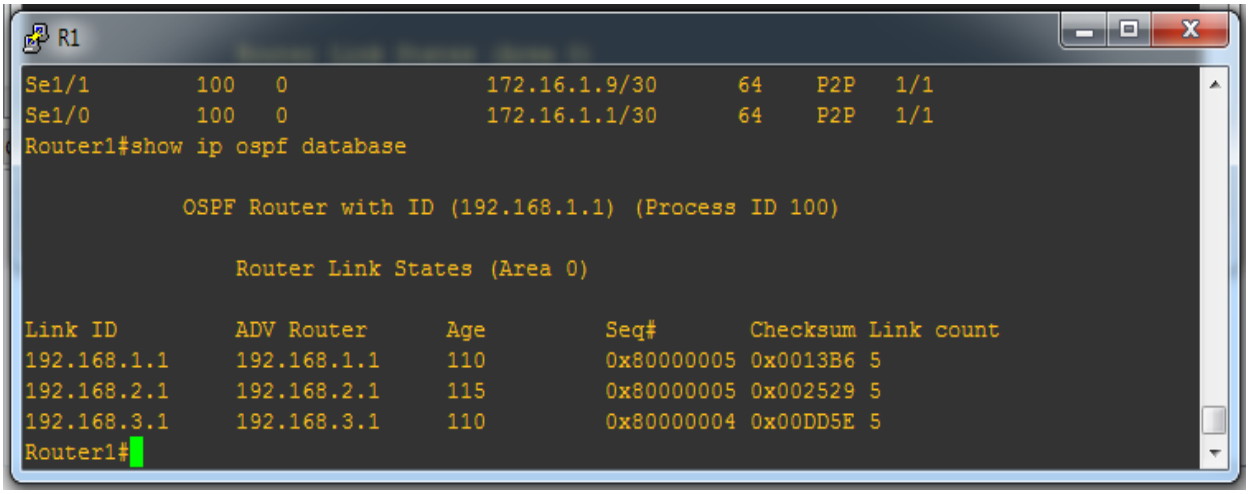


Figure III.21: Router1; show ip ospf database.

Router2#show ip ospf database

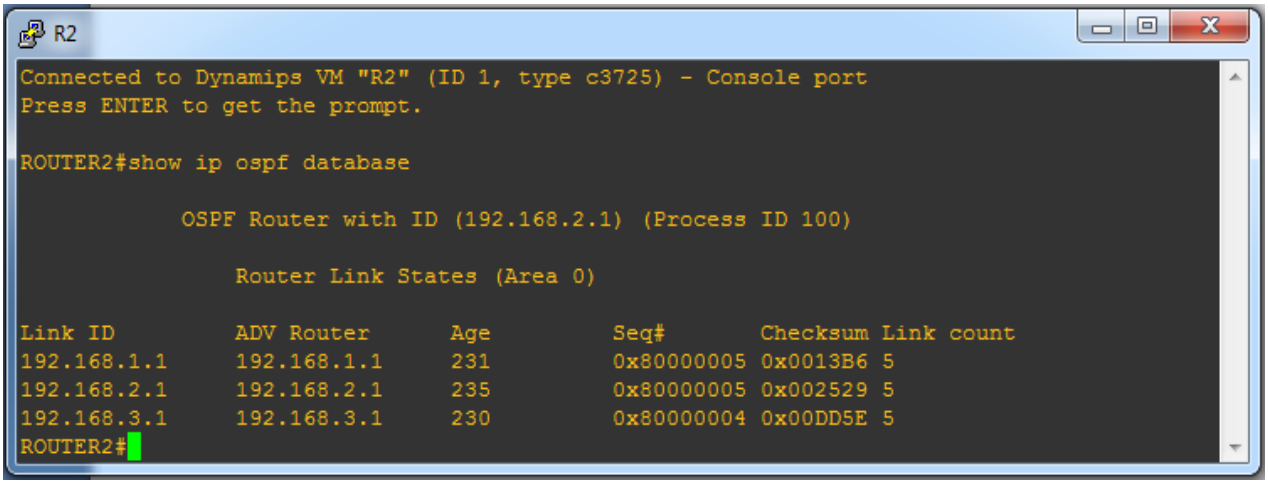


Figure III.22: Router2: show ip ospf database.

Router3#show ip ospf database

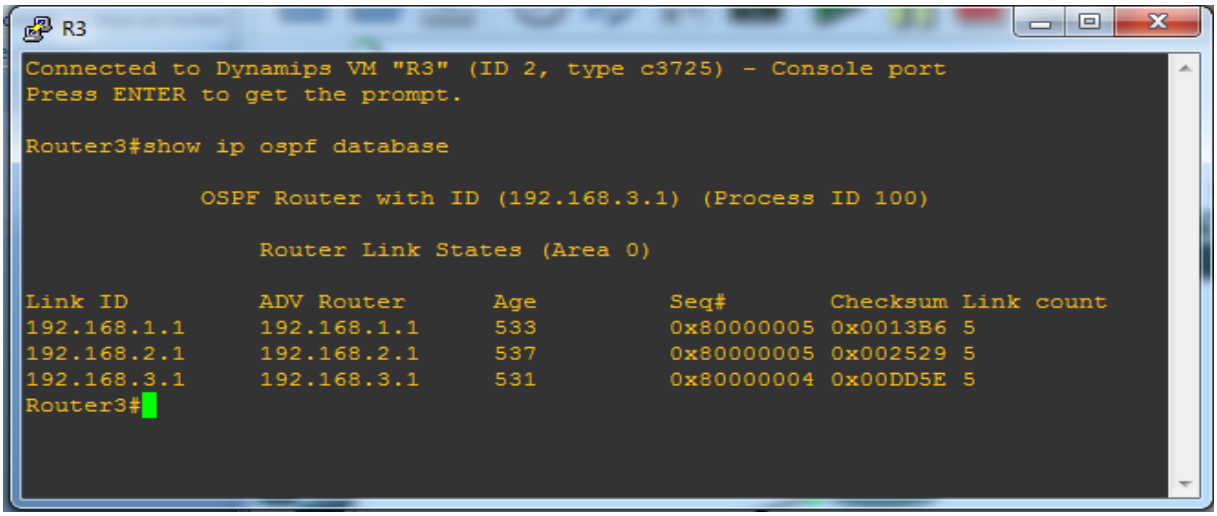


Figure III.23: Router3: show ip ospf database.

Comme nous le constatons à travers ces trois dernières figures, la table des données des trois routeurs est identique ainsi que le requièrent les normes standards des protocoles à état de lien.

III.8 Conclusion:

Dans ce chapitre nous avons présenté les résultats des tests appliqués sur l'impact des protocoles de routage sur les réseaux, ce chapitre illustre les différentes étapes de notre travail pratique. Il est aussi le principal but de notre étude.

Conclusion générale

Conclusion :

Ce projet, mené sur une période de cinq mois, nous a été bénéfique et les apports personnels ont été divers. Il nous a permit d'acquérir des connaissances dans un premier temps, de les approfondir ensuite et enfin de les mettre en pratique.

L'objectif n'étant pas précis d'abord, un travail de recherche sur internet ainsi que dans les bibliothèques et auprès d'entreprises comme Mobilis a été nécessaire pour esquisser un plan de travail et tracer un but à notre étude.

Nous avons essayé de faire connaître quelques protocoles et les différences qu'il y a entre eux, et de montrer leur rôle dans la gestion et le fonctionnement des réseaux virtuels, nous avons ensuite essayé de mettre en pratique nos connaissances en nous centrant sur un protocole en particulier, à savoir OSPF. Pour cela nous avons utilisé le logiciel qui nous a parut le plus adéquat à ce genre d'expérimentations, GNS3.

Nous espérons avoir atteint notre objectif tout en sachant qu'il pourrait être plus élaboré vu les progrès permanents dans les domaines de la télécommunication.

Bibliographie :

- ✓ Mémoire THEME: interconnexion du réseau tcp/ip a base des routeurs cisco Promotion 2006/2007 encadré par (lahdir M).
- ✓ Mémoire de fin d'études, département de l'électronique, Etude et simulation d'un réseau de téléphonie sur IP (TOIP) juin 2008.
- ✓

Les sites internet

http://idum.fr/spip.php?article213

http://www.commentcamarche.net/contents/534-routage-ip

http://www.networklab.fr/routage-inter-vlan-et-switch-l3/

http://www.digital-connexion.info/post/2010/03/10/is-is-intermediate-system-to-intermediate-system

http://fr.wikipedia.org/wiki/IS-IS

http://wapiti.telecom-lille1.eu/commun/ens/peda/options/st/rio/pub/exposes/exposesrio1998/commutation_ip/index.htm

http://fr.wikipedia.org/wiki/Commutateur_r%C3%A9seau

http://fr.wikipedia.org/wiki/Spanning_Tree_Protocol

http://cisco.goffinet.org/s3/spanning_tree#.VBl1T5R5PKO

http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2950/software/release/12-1_6_ea2c/configuration/guide/scg/swgports.html

http://www.commentcamarche.net/contents/516-le-protocole-arp

http://msdn.microsoft.com/fr-fr/library/cc758357(v=ws.10).aspx

http://www.commentcamarche.net/contents/516-le-protocole-arp

http://fr.wikipedia.org/wiki/Address_Resolution_Protocol

http://fr.wikipedia.org/wiki/R%C3%A9seau_local_virtuel

Résumé :

Ce projet, mené sur une période de cinq mois, nous a été bénéfique et les apports personnels ont été divers. Il nous a permit d'acquérir des connaissances dans un premier temps, de les approfondir ensuite et enfin de les mettre en pratique.

L'objectif n'étant pas précis d'abord, un travail de recherche sur internet ainsi que dans les bibliothèques et auprès d'entreprises comme Mobilis a été nécessaire pour esquisser un plan de travail et tracer un but à notre étude.

Nous avons essayé de faire connaître quelques protocoles et les différences qu'il y a entre eux, et de montrer leur rôle dans la gestion et le fonctionnement des réseaux virtuels, nous avons ensuite essayé de mettre en pratique nos connaissances en nous centrant sur un protocole en particulier, à savoir OSPF. Pour cela nous avons utilisé le logiciel qui nous a parut le plus adéquat à ce genre d'expérimentations, GNS3.

Nous espérons avoir atteint notre objectif tout en sachant qu'il pourrait être plus élaboré vu les progrès permanents dans les domaines de la télécommunication.

Mots clé.

- Protocoles OSPF, BGP, telnet et RIP.
- Modèle OSI, TCP/IP.
- ARP, HTTP, UDP.
- Le Routage IP.
- Simulateur GNS3