MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU.

FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE

DEPARTEMENT D'ELECTRONIQUE

Mémoire de fin d'études

Présenté en vue de l'Obtention

du Diplôme de MASTER II en Electronique.

Spécialité : Réseaux Télécommunication.

Thème :

Configuration des routeurs et commutateurs avec la syntaxe du constructeur de Cisco système niveau du MSC Mobilis de Tizi-Ouzou.

Proposé et dirigé par :

Mr.LAHDIR Mourad

Encadré par :

Mr.LAHDIRI Toufik

Présenté par :

Mlle.AZDAOU Zineb

Mlle MOUZARINE Ouahiba

Année universitaire 2010/2011

Remerciements

Nous tenons tous d'abord a remercie notre promoteur MR.LAHDIR .M

Sans le quel ce travail n'aurait pas vu le jour. On le remercie de nous avoir soutenue tous au long de ce parcoure.

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

D'autre acteurs sont intervenus ponctuellement pondant la mise en œuvre de ce travail, leurs aide et soutient ont contribué significativement à sa réussite. On souhaite remercier plus spatialement :

Personnel du BSC2 TIZI OUZOU

Et nous remercions tous le corps enseignant qui nous ont suivi toute au long de notre formation.

Dédicaces

Je dédie ce modeste travail à ceux qui sont la source de mon inspiration et de mon courage. Ceux dont je garde le souvenir éternel de sacrifices et de tendresse à :

- *Mes très chers parents que DIEU les protège.*
- *Mes frères et ma petite sœur <Lyna> .*
- *A mon ami Ferhat et sa famille.*
- *Ma chère binôme et sa famille.*
- *tous mes amis.*

AZDAOU ZINEB

Dédicaces

Je dédie ce modeste travail à ceux qui sont la source de mon inspiration et de mon courage. A tous ceux qui croient en moi et qui m'inspire amour et tendresse à :

- *Mes très chers parents que DIEU les protège.*
- *Mes frères.*
- *Ma chère binôme*
- *Ma formidable famille.*
- *Tous mes amis.*

Ouahiba Mouzarine

Sommaire :

Introduction

Les technologies de la télécommunication constituent aujourd'hui le principal vecteur de changement au monde car elle contribue 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 l'implémentation de technologie de routage avancé au sein de son architecture.

Cisco est le plus célèbre des constructeurs de matériel dédié au routage. Il met a la disposition du marché mondial des routeurs et commutateurs intégrant des fonctions avancées leur permettant d'être implémentée au cœur de tous les réseaux : les plus complexes comme les plus simples.

Les entreprises Algériennes ont adopté cette technologie. La problématique est que malgré son implémentation au cœur de leur réseaux, il y'a un sérieux manque d'administrateur réseau qualifiés capable de configurer les routeurs et commutateurs Cisco.

Notre projet se consacrera donc à l'étude des technologies Cisco plus précisément les routeurs de la gamme 2800 adopté par Mobilis pour son architecture réseaux. Pour cela nous avons partagé notre travail en trois parties :

Dans le premier Chapitre, 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.

Dans le deuxième chapitre, nous avons présentés l'architecture du réseau reliant le MSC Mobilis de Tizi-Ouzou aux MSC/BSC Bejaia, Moustapha, Bouira et Alger direction générale, ainsi que la configuration des routeurs et Switchs au niveau de celui-ci.

Le troisième chapitre est consacré à la présentation du système d'exploitation freeBSD, qui nous à permit de réaliser un routeur avec une vielle machine et de le configurer avec la même syntaxe utilisée par Cisco système.

Et nous terminant ce projet par une conclusion.

I. Préambule:

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

I.1. Modèle OSI :

Le modèle OSI (Open Système Interconection) définit de quelle manière les ordinateurs et les périphériques réseau doivent procéder pour communiquer.

- Il spécifie le comportement d'un système dit ouvert.
- Les règles de communication constituent les protocoles normalisés.

I.1.1 Description du modèle :

- Chaque couche est responsable de l'un des aspects de la communication.
- Une couche de niveau N communique avec les couches N+1 et N-1 par le biais d'une interface.
- La couche inferieur transporte les données vers la couche supérieure sans en connaitre la signification.
- Les couches N de deux systèmes communiquent à l'aide de protocoles communs.

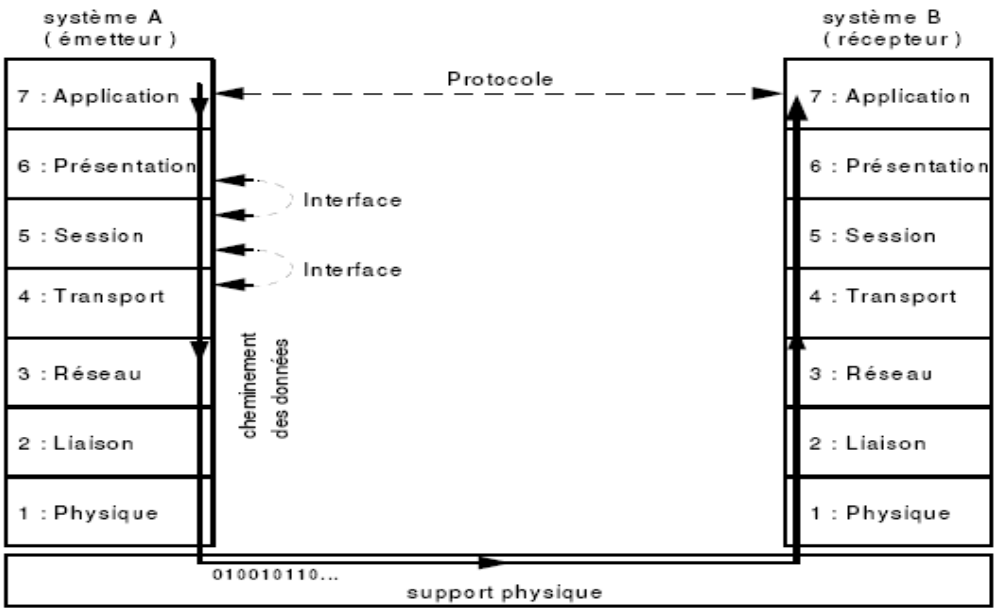


Figure I.1 : modèle en 7 couches de l'OSI.

- **Couche physique (1) :**
  - Elle se charge de l'adaptation du signal aux différents supports de transmission.
  - Elle gère le type de transmission (synchrone ou asynchrone).
  - Elle s'occupe de la modulation et démodulation.
  - L'unité d'échange est le bit.
- **Couche liaison de données (2) :**
  - Elle définit les règles d'émission et de réception des données à travers la connexion physique de deux systèmes.
  - Détection et correction des erreurs s'occupe de la réémission s'il ya eu lieu et détermine la méthode d'accès au support.
  - L'unité d'échange est la trame.
- **Couche réseau (3) :**
  - Assure l'acheminement des données en assurant le routage (chois du trajet) des paquets de données.
  - Routé les données vers un autre nœud en cas de surcharge.
  - La couche réseau assure la translation d'adresse logique en adresse physique.
  - L'unité d'échange est le paquet.
- **Couche transport (4):**
  - Elle assure les services pas pris en compte par les autres couches (contrôle d'erreurs, routage).
  - Elle permet le multiplexage de plusieurs flux de données sur le même support.
  - En tant qu'émetteur, elle segmente les messages en paquets numérotés.
  - En tant que récepteur, elle reconstitue les messages en plaçant les paquets dans l'ordre.
- **Couche session (5) :**
  - Elle permet l'ouverture ou la rupture d'une session de travail entre deux systèmes distants et assure la synchronisation du dialogue.
  - Elle définit le mode de transmission (half-duplex, full-duplex).
  - Elle définit la liaison entre deux programmes d'application et gère le dialogue.

- **La couche présentation(6) :**
  - Elle permet de transcrire les données dans un format compréhensible par les deux systèmes (formatage des données, codage, compression, cryptage, décryptage). La mise en forme des données pour quelle soient accessible aux utilisateurs.
- **La couche d'application(7) :**

Elle fournit des services utilisables sur le réseau par les applications installées.

Les principaux services sont :

- Transfert de fichier FTP.
- Messagerie ou courrier électronique (POP, SMTP).
- Lecture de page internet (http).
- Accès a distance (Telnet).

I.2 Modèle TCP /IP :

Le TCP s'inscrit dans ce modèle mais n'utilise pas systématiquement l'ensemble des septe couches.

TCP/IP désigne communément une architecture réseau, mais cet acronyme désigne en fait deux Protocoles étroitement liés : un protocole de transport, TCP (Transmission Control Protocol) qu'on utilise "par-dessus" un protocole réseau, IP (Internet Protocol).

Ce qu'on entend par "modèle TCP/IP", c'est en fait une architecture réseau en 4 couches dans laquelle les protocoles TCP et IP jouent un rôle prédominant, car ils en constituent l'implémentation la plus courante.la figure suivante illustre cette architecture.

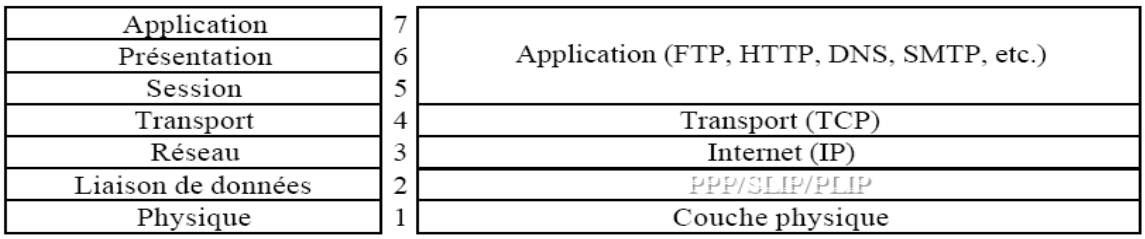


Figure I.2 : modèle de référence OSI et le modèle TCI /IP.

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

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

I.3 Protocoles de communication :

Un protocole de communication est un ensemble de règles et de procédures permettant de définir un type de communication particulier. Les protocoles sont hiérarchisés en couches, pour décomposer et ordonner les différentes tâches. On retrouve :

I.3.1 Les protocoles orienté connexion :

Il s'agit de protocoles opérant un contrôle de transmission de données. Les deux machines émettrice et réceptrice établissent une session avant l'envoie des données, à la réception la machine envoie un accusé de réception. On retrouve pour cette famille de protocole le protocole TCP.

I.3.2 Les protocoles non orienté connexion :

Il s'agit d'un mode de communication dans lequel 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 (data grammes).UDP est un protocole non orienté connexion.

I.4 Encapsulation :

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

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

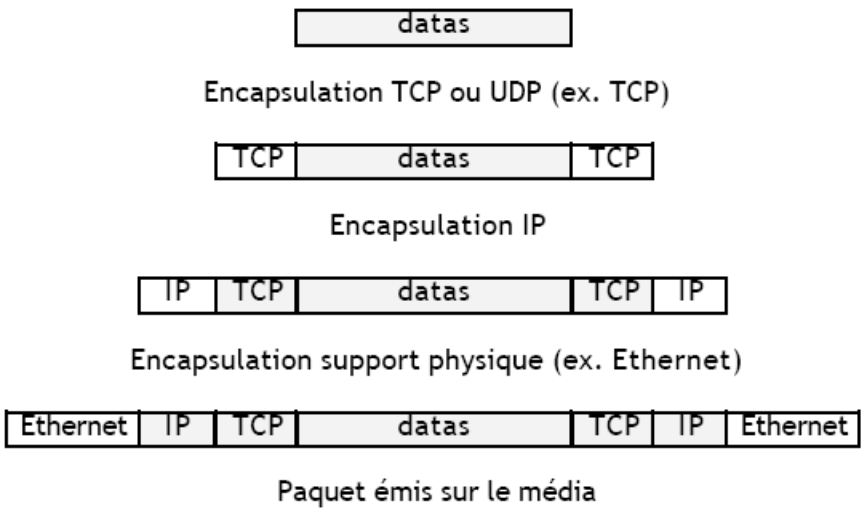


Figure I.3: encapsulation TCP/IP.

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

I.5 Adressage IP :

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

L'adresse IP est codée sur 32 bits, soit 4 octets. **Chaque adresse IP est unique** dans le réseau Internet, ce qui est une première condition nécessaire pour le bon adressage des machines connectées au réseau. Le réseau est en fait composé d'une multitude de sous-réseaux (blocs d'adresses IP contiguës).

I.5.1 Les adresses IP :

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

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

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

Figure I.4 : classes d'adressage IP.

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

Classe	*Plage*	
A	0.0.0.0	127.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Figure I.5 : les plages d'adresses.

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

Il existe 3 classes d'adresses IP :

- Classe A : 10.0.0.0 à 10.255.255.255
- Classe B : 172.16.0.0 à 172.31.255.255
- Classe C : 192.168.0.0 à 192.168.255.0

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

I.6 Masque réseau :

Le masque contient des **1** aux emplacements des bits à conserver (partie réseau), et des **0** pour ceux à rendre égaux à zéro (partie hot). Une fois ce masque créé, il suffit de faire un **ET** entre la valeur à masquer et le masque afin de garder intacte la partie souhaitée et annuler le reste.

I.7 Address Resolution Protocol ou ARP:

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

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

⇒ Pour cela, si 10.23.23.2 cherche à contacter 10.23.23.254, la station va, avant tout dialogue, diffuser (*broadcaster*) à l'ensemble des stations du réseau une requête ARP.

⇒ Chaque station va alors recevoir cette requête ARP, composée du message suivant :

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

⇒ Toutes les stations reliées à ce segment analysent alors cette demande, mais seule la station 10.23.23.254 va répondre à cette demande, en renvoyant le message suivant : Station 10.23.23.254 a pour adresse matérielle yy:yy:yy:yy:yy:yy.

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

Discussion :

Ce chapitre avait pour objectif, d'introduire quelques notions sur les réseaux et la manière dont se fait la communication. Informations nécessaires avant de passer à la deuxième étape de notre travail.

Chapitre II :

La configuration des routeurs et commutateurs Cisco.

II.1 Préambule :

Dans ce chapitre nous étudierons l'organisme ou nous avons effectué note stage, son architecture réseau, les différents MSC / BSC aux quels le MSC de Tizi-Ouzou est relier. Apres quoi nous procéderons à la configuration des routeurs et commutateurs appartenant à ce réseau.

Partie (A) :

II.A.2 Présentation de l'organisme d'accueil :

Mobilis, filiale d'Algérie Télécom, c'est le premier opérateur de téléphone en Algérie, c'est une société par action (spa) au capitale de 100 ,000 ,00DA il propose à ses propre client une large gamme de produit et des services innovants de hante qualité.

- **Situation géographique :**

Ce Centre de Commutations Mobile (MSC), (Mobilis) se situe à la nouvelle ville, exactement au sud de la wilaya de Tizi-Ouzou.

- **Rôle du MSC :**

Le MSC (mobile-services switching center), est le centre de commutation des mobiles, il gère l'établissement des communications entre un mobile et un autre MSC et permet la transmission des messages courts ainsi que l'exécution du hand over.

Il dialogue avec le VLR pour gérer la mobilité des usagers et sert de passerelle active lors d'appel d'abonné fixe ou d'abonné d'autres opérateurs vers un mobile.

Il compte dans son architecture :

- **Une salle d'équipements :** qui compte tous l'équipement du système GSM (commutation « MSC/BSC »)
- **Une salle de contrôle :** les poste de travail des technicien et ou la maintenance des différents équipements du centre sont installés.
- **La salle du chef de service :** elle arbitre le bureau du chef de service.

La figure qui suit nous illustre le tous.

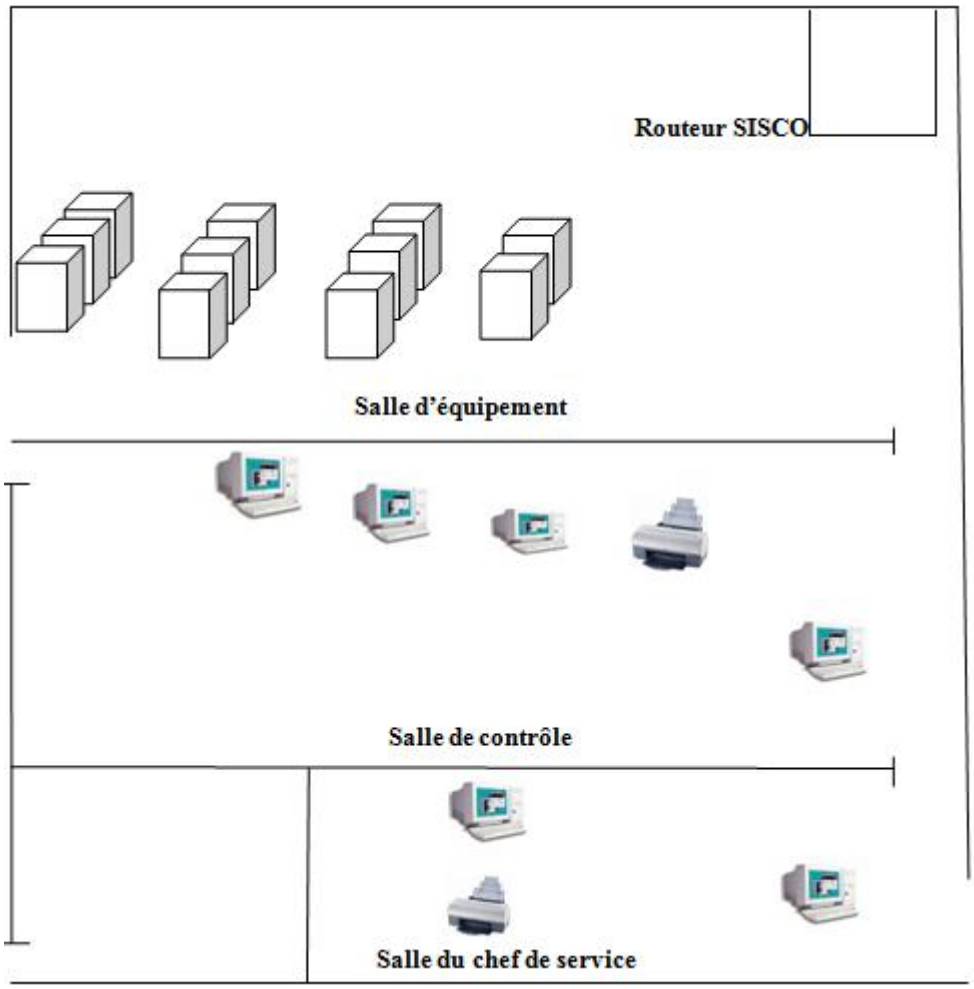


Figure II.A.1 : architecture du centre MSC/BSC de Tizi –Ouzou.

II.A.3 Matérielle utilisé :

II.A.3.1 Routeur :

Le routeur est un équipement d'interconnexion de niveau 3(model OSI) qui achemine (qui route) les données vers un destinataire connu par son adresse de niveau 3(adresse IP).Les routeurs permettent d'interconnecter des réseaux locaux de même topologie ou non.

Le MSC utilise un routeur Cisco de gamme 2800, la gamme est principalement choisie selon la performance d'IOS (Internetworking Operating Système) et selon le nombre de type de ports dont on a besoin.

- **Architecture interne du routeur :**

Tous Les routeurs Cisco ont une architecture interne qui peut être représenté par :

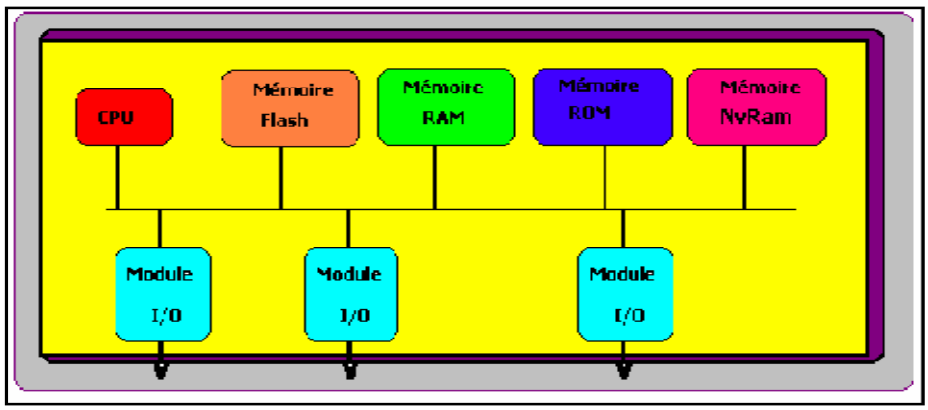


Figure II.A.2 : architecture interne d'un routeur Cisco.

Ils contiennent tous :

- Une mémoire ***NvRam*** pour Ram non Volatile et sur laquelle l'administrateur va Stocker la configuration qu'il aura mise dans le routeur. Elle contient également la configuration de l'IOS.
- Une carte mère qui est en général intégrée au châssis. Par contre les interfaces

Sont des cartes additionnelles. Il existe des interfaces Ethernet, Token-Ring, série V35-V24, RNIS (ISDN), ATM, etc.…

- Une ***CPU*** qui est un microprocesseur avec un *BIOS* spécial nommé " *I.O.S.* " pour *Internetwork Operating System.*
- Une mémoire ***RAM*** principale contenant le logiciel IOS, dans la quelle tout sera exécuté un peu à la manière d'un simple ordinateur. C'est là où l'IOS est exécuté ainsi que le bootstrap…etc. On y retrouve également toutes les tables créées pendant l'utilisation du routeur (tables de routages, ARP, etc.), mais aussi tous les buffers utilisés par les cartes d'entrée sorties. sa capacité est de 256 MO installée.
- Une mémoire ***FLASH*** : de 64 M, également une mémoire non volatile sur laquelle on stocke la version courante de l'IOS du routeur.
- Une mémoire ***ROM*** non volatile et qui, quant à elle, contient les instructions de démarrage (bootstrap) et est utilisée pour des opérations de maintenance difficiles.

- **Interfaces:**

Les interfaces permettent au routeur de se connecter avec l'extérieur. Il possède trois types d'interfaces: LAN, WAN et Console/AUX. Les interfaces LAN sont en général des ports Ethernet ou Token Ring standard qui sont au nombre 16 ports Ethernet sur notre routeur. Les interfaces WAN incluent des ports série, RNIS et une unité de transmission de données (CSU) intégrée. Comme les interfaces LAN, notre routeur dispose de 4 interfaces série parfaitement adapté aux liaisons WAN requissent.

- **Les ports Console/AUX** :

Sont des ports série principalement utilisés pour la configuration initiale du routeur. La console s'utilise en particulier dans les circonstances suivantes :

- Ø Configuration initiale du périphérique réseau ;
- Ø Procédures de reprise après sinistre et dépannage lorsque l'accès distant est impossible ;
- Ø Procédures de récupération des mots de passe.

  On en retrouve deux un aux et un port console. La figure qui suit illustre les différents ports pour voire leur aspect physique.

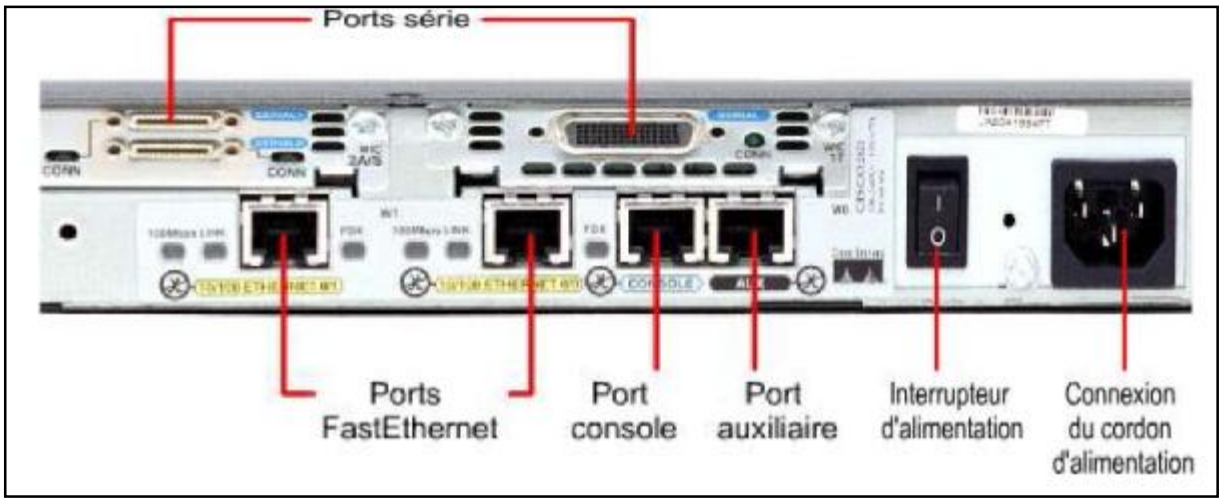


Figure II.A.3 : les différentes interfaces du routeur.

- **Software :**
  - **IOS (Internetwork Operating System):**

À l'instar d'un ordinateur personnel, un routeur ou un commutateur ne peut pas fonctionner sans système d'exploitation. Sans système d'exploitation, le matériel est inopérant.

Cisco IOS est le logiciel système des périphériques Cisco. Il fournit aux périphériques les services réseau suivants :

- Fonctions de routage et de commutation de base.
- Accès fiable et sécurisé aux ressources en réseau.
- Évolutivité du réseau.

Pour notre routeur il a été intégré un IOS de version 12.3(T). Pour accéder aux services fournis par IOS, nous utilisons généralement une interface de ligne de commande (ILC). Les fonctions accessibles à travers ILC varient selon la version de Cisco IOS et le type du périphérique.

- **Interpréteur de commande (ILC) :**

L'interpréteur de commande, comme son nom l'indique, est responsable de l'interprétation des commandes que nous tapons. La commande interprétée, si elle est correcte, réalise l'opération demandée.

II.A.3.1.1 Routage :

Permet de construire des réseaux complexes et permet ainsi d'étendre la portée des réseaux sur de longues distances. La transmission des datagrammes IP vers leurs destinataires est appelée **routage**.

Les routeurs sont des périphériques de niveau 3. C'est donc la couche 3 du modèle OSI (Réseau) qui est chargée des fonctions de routage et la détermination du chemin à prendre. Comme l'illustre la figure qui suit :

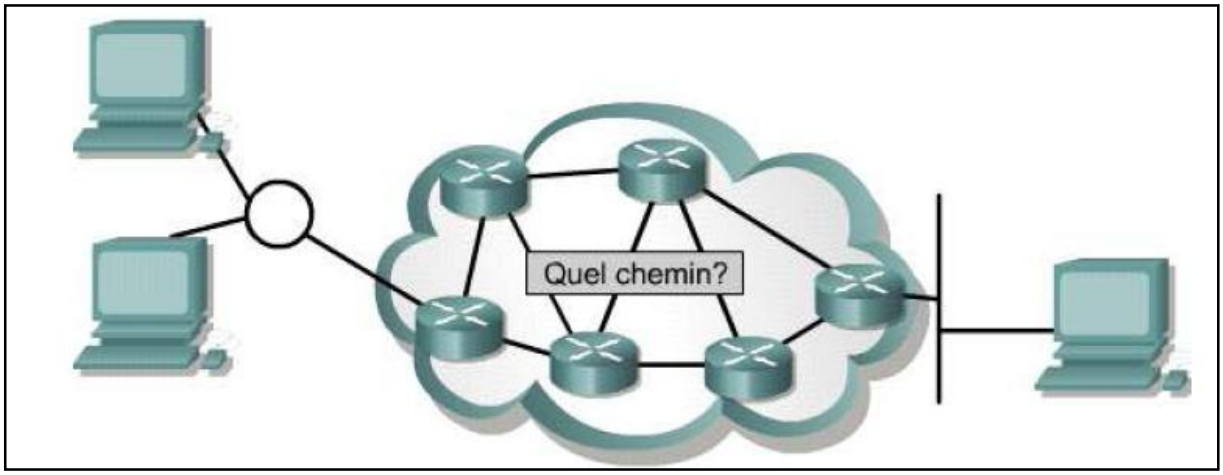


Figure II.A.4 : intérêt du routage.

Un routeur dispose de plusieurs ports (deux au minimum) et donc de plusieurs adresses IP (situées dans des réseaux IP différents). Lorsque les datagrammes IP arrivent sur le port du routeur, le logiciel de routage examine les en-têtes pour déterminer de quelle manière il doit les transmettre. L'information déterminante est alors l'adresse IP de destination du datagramme. Le logiciel de routage consulte alors la **Table de Routage** du routeur pour savoir comment atteindre le destinataire, puis transmet le datagramme sur le port indiqué. La table de routage contient une liste de réseaux et d'hôtes de destination ainsi que la manière de les atteindre.

Les informations de la table de routage peuvent être renseignées de deux façons :

- **Méthode statique** : L'administrateur rentre manuellement les informations de routage

- **Méthode dynamique** : Le routeur assimile dynamiquement les informations grâce aux protocoles de routage.

- **Les protocoles de routage :**

Définissent la manière dont les routeurs s'échangent les informations afin de déterminer la meilleure route vers une destination. Ils échangent leurs messages à l'aide de datagrammes IP.

Certains s'exécutent directement au dessus d'IP (**OSPF : Open Shortest Path First**), d'autres au dessus d'UDP (**RIP : Routing Information Protocol**) ou de TCP (**BGP : Border Gateway Protocol**). Comme ils se comportent comme des clients IP, TCP ou UDP on peut les assimiler à des protocoles applicatifs

Les protocoles de routage ne sont pas utilisés pour effectuer le routage lui-même. Ils sont utilisés par les routeurs pour échanger entre eux leurs informations et constituer de manière dynamique leurs tables de routage.

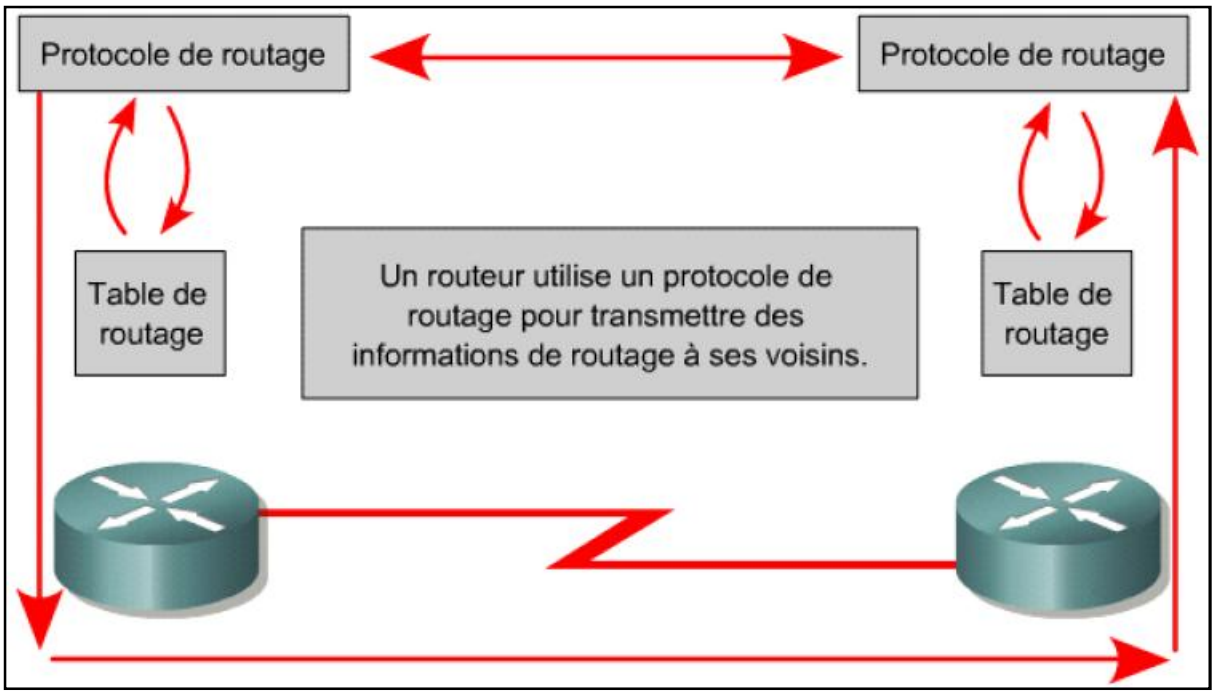


Figure II.5 : fonctionnement du routage dynamique.

La plupart des algorithmes de routage peuvent êtres rangé dans l'une des catégories suivantes :

- Vecteur de distance, dont on retrouve le protocole **RIP** pour Routing Information Protocol et le protocole **BGP** pour Border Gateway Protocol
- Etat de liens, comme par exemple le protocole **OSPF** (Open Shortest Path First).

– **Protocole de routage à vecteur de distance :**

- **RIP (Routing Information Protocol) :**

Est un protocole à vecteur de distance. Il tient à jour dans la table de routage une liste de toutes les routes connues et diffuse périodiquement le contenu de cette table chaque 30 secondes. Chaque entrée de la table est composée de la destination, du routeur de prochain pas et de **la distance (ou métrique)**.

La distance est la mesure du coût associé à la route et se mesure en pas. Le franchissement d'un routeur compte pour un pas. RIP affecte un coût de 1 pour un réseau connecté directement, un coût de 2 à un réseau atteint par le biais d'un routeur, et ainsi de suite.

RIP utilise UDP comme protocole de transport. Les routeurs RIP envoient et reçoivent des mises à jour et des requêtes d'informations de routage sur le port UDP 520

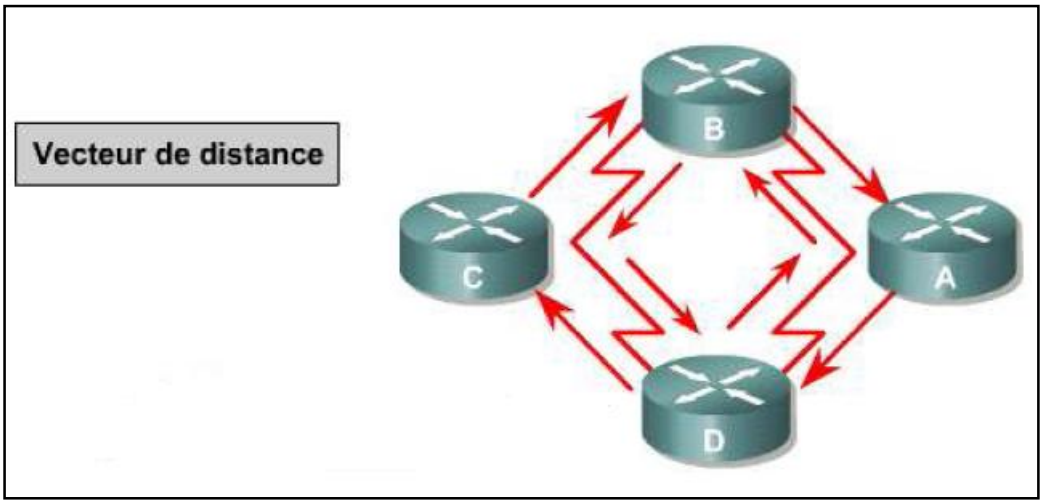


Figure II.A.6 : RIP et le calcule des nombre de sauts.

- **IGRP(Interior Gateway Routing Protocol)**

Est un protocole propriétaire de Cisco, il est doté entre autre des caractéristique suivantes :

- Il s'agit d'un protocole de routage à vecteur de distance.
- La bande passante, le délai, la charge et la fiabilité sont utilisés pour créer une métrique composite.
- Par défaut la mise à jour de la table de routage s'effectue toutes les 90secondes.

- **BGP (Border Gateway Protocol) :**

Est un protocole de routage extérieur, à vecteur de distance.il est utiliser entre les FAI et les FAI et les clients et pour acheminer le trafique internet entres des systèmes autonomes.

- **Protocol de routage à état de liens :**

- **OSPF (Open Shortest Path First) :**

Est un protocole d'état de liaison. Ces protocoles sont mieux adaptés aux réseaux de grande taille car ils utilisent des messages plus courts et moins nombreux pour communiquer et mettre à jour les tables de routage. Chaque routeur dispose de la carte complète du réseau et échangent régulièrement des informations concernant l'état de leur liaison. L'état de la liaison peut être soit UP (en activité), soit DOWN (hors service). A partir des messages reçus, chaque routeur recalculera lui-même sa table de routage. C'est un protocole de routage interne.

OSPF communique directement par le biais de datagrammes IP et n'utilise pas de protocole de transport comme UDP ou TCP.

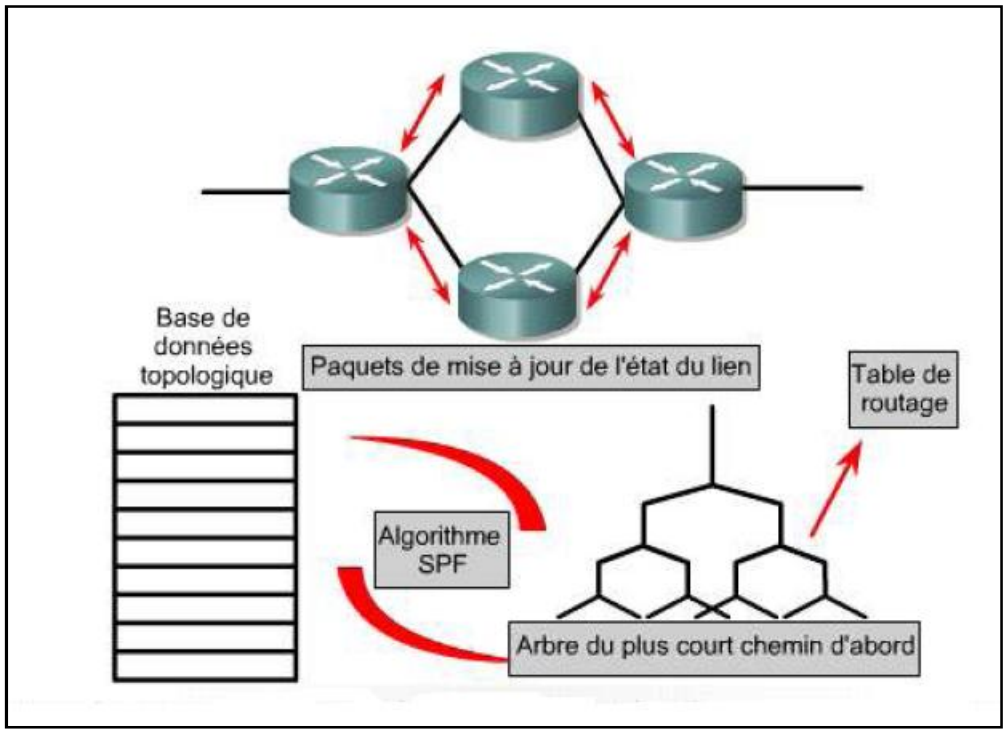


Figure II.A.7 : fonctionnement du protocole OSPF.

II.A.3.2 Commutateur :

Est un équipement d'interconnexion de niveau 2 (du model OSI), il permet de créé un réseau informatique local de type Ethernet. Ce dispositif est dite intelligent par opposition au hub car, alors que ce dernier diffuse les données sur toutes les machines du réseau le Switch à lui, permet de diriger les données uniquement vers la machine de destinataire. Il utilise des adresses physiques ou MAC.

Le commutateur qu'on veut implémenter au niveau du MSC de Tizi-Ouzou est un Catalyst 2950 de Cisco à 24 ports, grâce au quel nous pouvons construit le réseau local du MSC. On a donc besoin de 4 port Ethernet pour les PC un autre pour l'imprimante, et on laisse le reste pour une éventuelle extension.

II.A.3.3 Transceiver :

Dans notre réseau, le Transceiver est utilisé entre la fibre optique et le câble RJ 45 qui lui à son tour va vers un Switch ou un routeur, on parle de Transceiver optique.

Partie (B):

II.B.1 Application :

La figure II.B.1 représente les différents MSC/BSC reliés au MSC de Tizi-Ouzou et les diverses liaisons qui assurent le maintien de leur connexions.

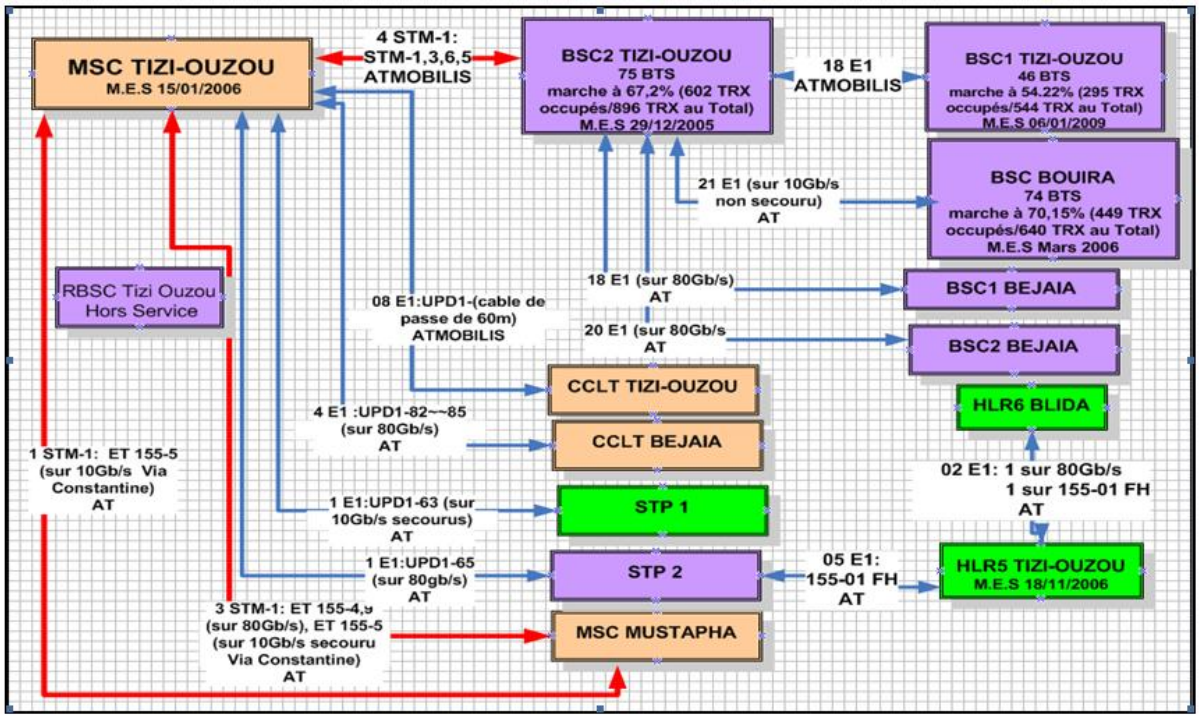


Figure II.B.1: Schéma de transmission MSC de Tizi-Ouzou.

Alors le MSC Mobilis de Tizi-Ouzou est reliée par l'intermédiaire d'un routeur Cisco à la station Mobilis d'Alger au MSC de Moustapha, les deux BSC de Bejaïa et le BSC Brouira, grâce à cette liaison les ingénieurs de ce site peuvent avoir accès aux différentes MSC ou BSC cités avant; avec des utilitaires comme Win fioul qui leur permet la définition l'état des différents BTS , avec cet utilitaire on peut définir de quelle alarme il s'agit pour alerter les technicien de BSC tizi 1ou 2 pou qu'ils puissent intervenir.il permet aussi la mise en service d'une nouvelle antennes , vérifier les liaisons MIC au moment d'un dérangement signalé au niveau d'une BTS et l'élimination d'un dérangement .

Notre tache à nous est de configurer le routeur Cisco 2811 pour que les liaisons vers les réseaux étendu aux quels le MSC est relié soient opérationnelles. Sachant que pour pouvoir configurer le routeur et le commutateur nous avons choisit un logiciel de simulation en temps réel également utilisé par les administrateurs réseau de chez Cisco qui est Packet tracer 5.0.

Grace à ce logiciel on a pu recréer le réseau du MSC LAN et WAN de Mobilis, configurer les différents fonctions requissent pour que les liaisons soient fonctionnelles. La figure qui suit illustre l'environnement packet tracer et ses différentes fonctions.

II.B.2 Présentation du packet tracer :

Packet tracer constitue une solution parfaite pour la construction des réseaux ainsi que leurs teste avant de les mettre en œuvre sur le terrain, il est simple à manipuler et sont avantage le plus important est qu'il fonctionne en temps réel et il nous offre des blocs de tous les routeurs, commutateurs, ponts … que Cisco a mit à disposition de ses clients.

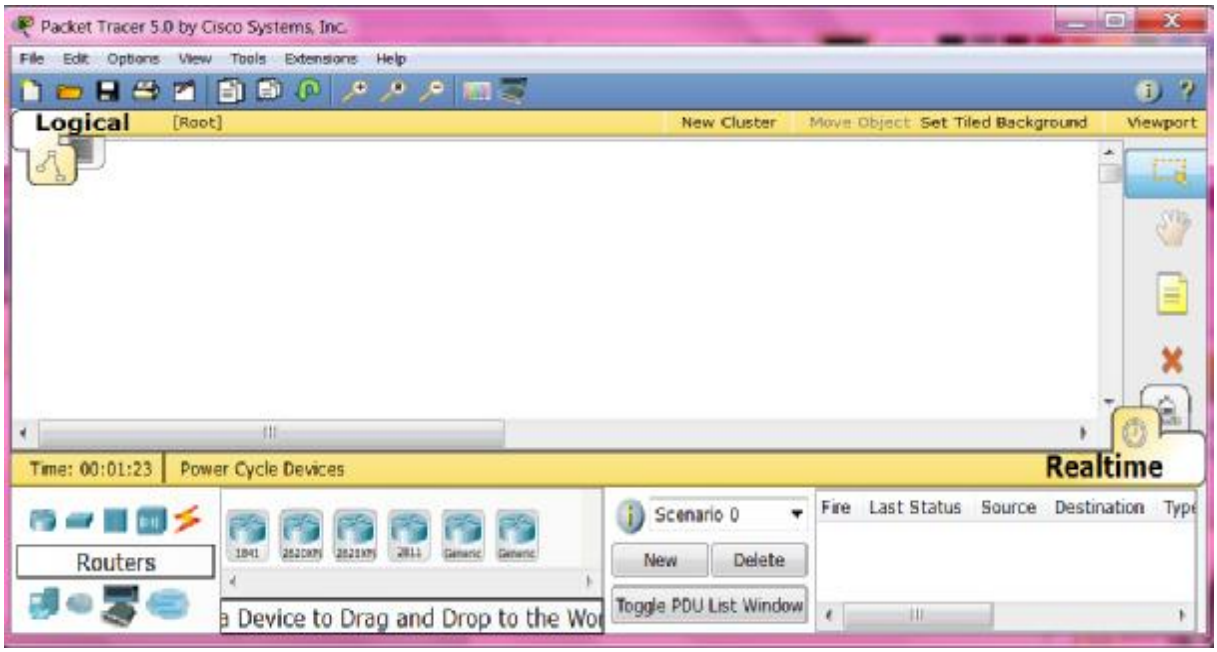


Figure II.B.2 : Environnement de packet tracer 5.0.

Environnement convivial, il mit à notre disposition tous les moyens de connexion fibre optique, câble directe, croiser, console, DTE, DCE…etc

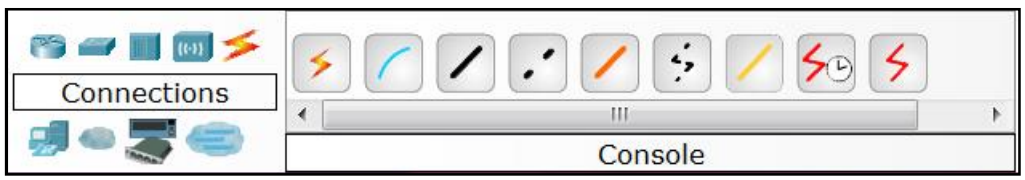


Figure II.B.3 : différents câblages disponible sur packet tracer.

Il nous permet de construire des exemples de simulation identiques à la réalité, par exemple dans le cas réel de configuration de routeur on connecte le port com1 du pc au port console du routeur on fait démarrer une session HyperTerminal (sous Windows-xp) on règle les paramètres nécessaire et on configure notre routeur. Avec packet tracer c'est la même chose.

- On clique sur **End devices** on fait glisser un pc (PC-PT)**,** puis sur **routers** on glisse un routeur on choisit un câble console pour interconnexion des deux éléments (port RS232 pour le pc au port console du routeur), on aura le schéma de la figure suivante :

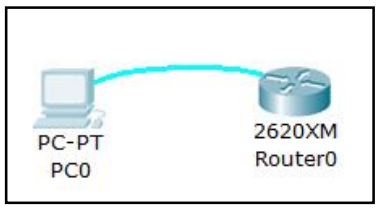


Figure II.B.4 : configuration d'un routeur Cisco avec packet tracer.

- Pour la configuration, un click sur le pc, une autre image se présente on choisit l'onglet **desktop** puis la fenêtre **terminal**, on règle les paramètres pour accélérer la procédure comme suit :

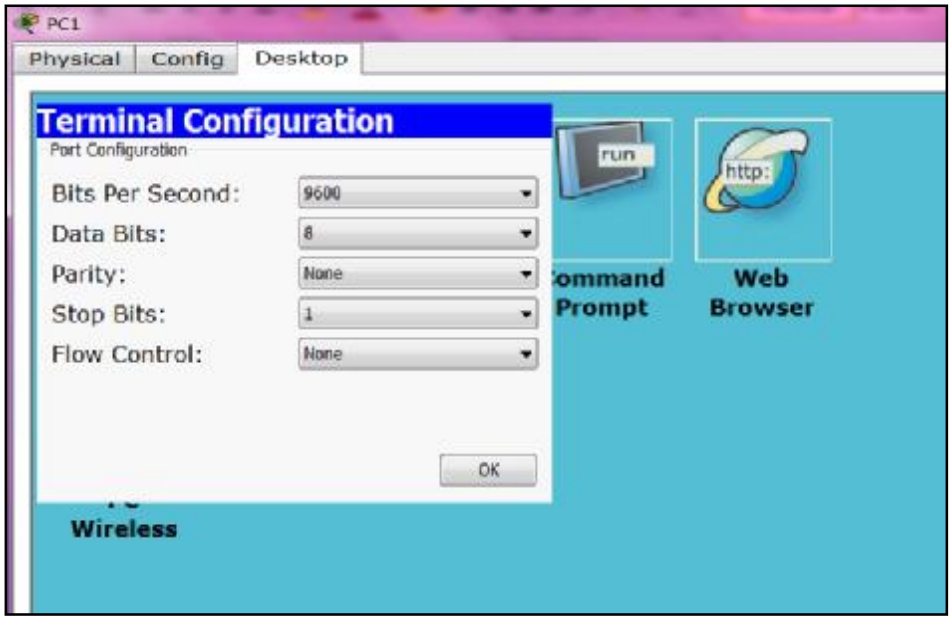


Figure II.B.5 : configuration des paramètres initiaux.

- On click sur ok et on a accès à l'interpréteur de commande CLI du routeur pour enfin commencer la configuration.la figure qui suit illustre l'invite terminal grâce au quel on configure les routeurs ou commutateurs.

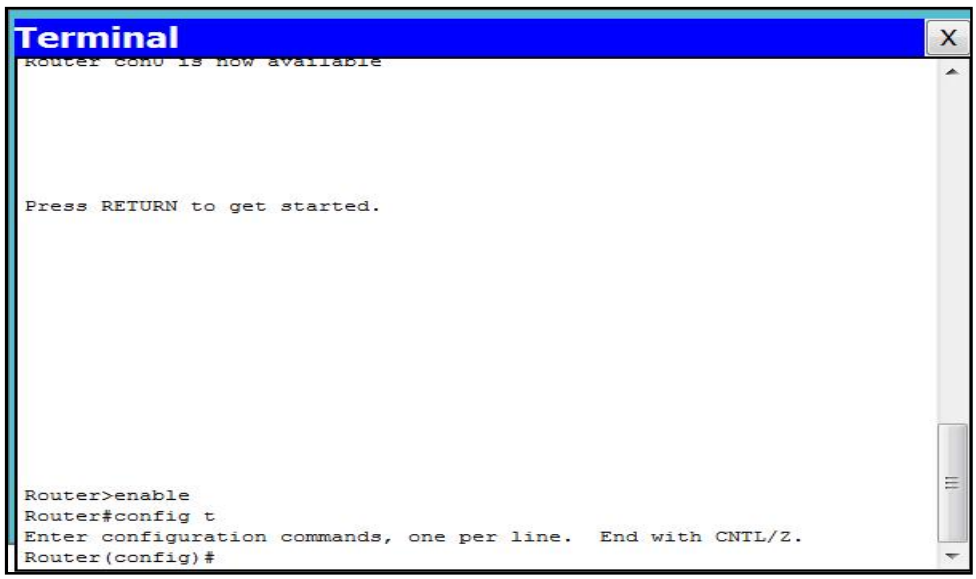


Figure II.B.6 : invite pour configurer les périphériques.

II.B.3 Le réseau de Mobilis sur packet tracer

La figure qui suit montre le travail qu'on a réalisé avec packet tracer pour représenter les liaisons réseaux étendu du MSC de Tizi-Ouzou avec les BSC de Bejaia celle de Bouira de Moustapha et la direction générale d'Alger.

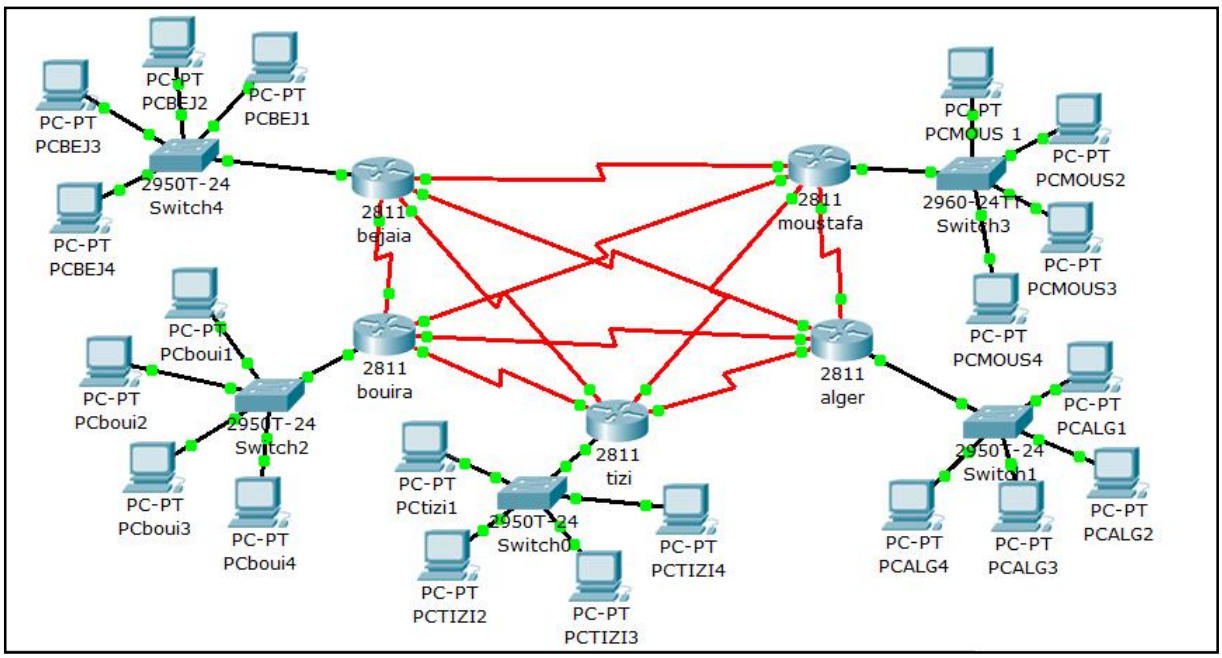


Figure II.B.7: représentation des réseaux étendus reliés au MSC de Tizi-Ouzou.

- **Topologie du reseau étendu :**

Lorsqu'une entreprise s'agrandit pour inclure des succursales, des services de commerce électronique ou des activités globales, un réseau local (LAN) peut s'avérer insuffisant pour satisfaire ses besoins commerciaux. L'accès de réseau étendu (WAN) est devenu aujourd'hui essentiel dans la plupart des grandes entreprises.

Les principaux caractéristiques des réseaux étendus sont les suivantes :

- ils connectent généralement des périphériques séparés par une zone géographique plus étendue que ne peut couvrir un réseau local ;
- ils utilisent les services d'opérateurs, tels que des compagnies de téléphone ou de câble, des systèmes satellite et des fournisseurs de réseau.

- ils utilisent divers types de connexions série pour permettre l'accès à la bande passante sur de vastes zones géographiques.

Dans le ca du réseau de Mobilis, les différentes stations sont reliées entre elles par le biais de connections série dédiées pour cette tache.

Pour ce réseau, on a opté pour une topologie maillé ou chaque routeur est connecté par tous ses ports aux autres routeurs, assurant ainsi une connexion permanente entre les différents cites.

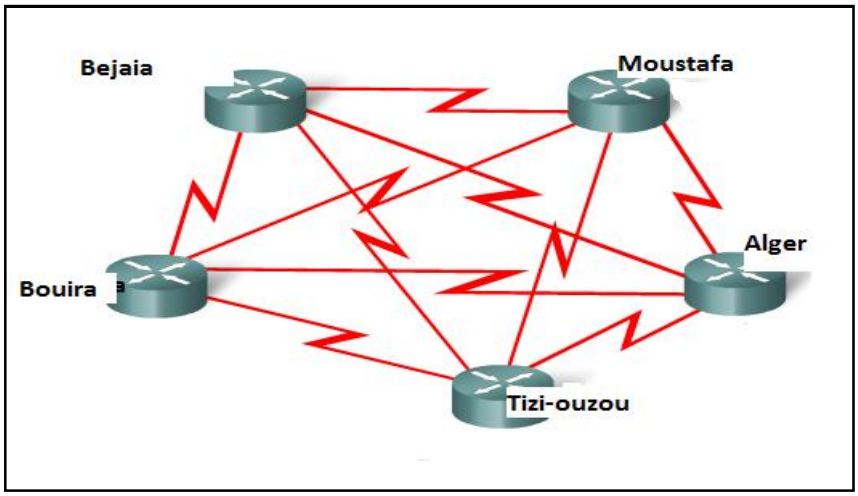


Figure II.B.8 : la topologie du réseau adoptée.

– **Type d'interconnexion des sites :**

Comme le réseau de Mobilis est un réseau privé, des liaisons de commutations dédiées sont requis.les lignes point à point sont utilisées afin d'assurer des connexions permanentes entre la station de Mobilis de Tizi-Ouzou et les différents réseaux étendus aux quels elle est reliée. Les lignes point à point sont généralement louées à un operateur (Algérie télécom) et prennent le nom de ligne louées.

Les lignes louées peuvent présenter des capacités variées, et leur prix dépend généralement de bande passante requise et de la distance entre deux points.

Le port série d'un routeur est requis, une unité CSU/DSU et un circuit provenant du fournisseur de services sont généralement requis pour chaque connexion sur ligne loué.

Si on se refere au schéma de transmission du MSC, les lignes loués sont déployées pour interconecter le MSC vers les deux site de BSC tizi-ouzou(BSC 1 et BSC 2),on compte 18 E1

entre les deux BSC réellement ils utilisent 17 .et 21 E1 de 80G pour etre relier au BSC de Bouira contre 18 E1 pour BSC de Bejaia 1 et 20 pour BSC de Bejaia 2 de bande de 80G chacun .

On retrouve aussi des liaisons STM-1 vers Moustapha en tou 1liaison STM passant par constantinne de 10G et 3 autres liaisons de soucours de 80G.la liaison vers Moustapha et Alger la direction central est assuré par de la fibre optique de capacité 2G instalé par Algerié Télécom.

II.B.3.1 Adressage des machines :

Tous les pc des différentes stations sont sous Windows xp. Nous avons donné à chaque machine une adresse IP selon le plant d'adressage qui va suivre :

Pour sécuriser notre réseau on a choisi un plan d'adressage avec des adresse privé qui son pas routable sur internat pour garantir une meilleure sécurité, ils sont donc résumés dans tableau suivant :

ROUTEURS	INTERFACES	ADRESSE IP	MASQUE	PASSERELLE PAR DEFAUT
R Tizi-Ouzou	Fa0/0	172.16.1.1	255.255.255.0	
	S0/0/0	172.16.2.1	255.255.255.0	
	S0/0/1	172.16.12.1	255.255.255 .0	
	S0/1/0	172.16.15.1	255.255.255.0	
	S0/1/1	172.16.13.1	255.255.255.0	
COMMUTATEUR	VLAN 1	172.16.1.10	255.255.255.0	172.16.1.1
PC TIZI1		172.16.1.251	255.255.255.0	172.16.1.1
PC TIZI2		172.16.1.252	255.255.255.0	172.16.1.1
PC TIZI3		172.16.1.253	255.255.255.0	172.16.1.1
PC TIZI4		172.16.1.254	255.255.255.0	172.16.1.1
R ALGER	FA0/0	172.16.3.1	255.255.255.0	
	S0/0/0	172.16.2.2	255.255.255.0	
	S0/0/1	172.16.11.1	255.255.255.0	
	S0/1/0	172.16.8.2	255.255.255.0	
	S0/1/1	172.16.7.2	255.255.255.0	
COMMUTATEUR	VLAN 1	172.16.3.10	255.255.255.0	172.16.3.1
PC ALG 1		172.16.3.251	255.255.255.0	172.16.3.1
PC ALG 2		172.16.3.252	255.255.255.0	172.16.3.1
PC ALG 3		172.16.3.253	255.255.255.0	172.16.3.1
PC ALG 4		172.16.3.254	255.255.255.0	172.16.3.1
R BOUIRA	FA0/0	172.16.5.1	255.255.255.0	
	S0/0/0	172.16.15.2	255.255.255.0	
	S0/0/1	172.16.7.1	255.255.255.0	
	S0/1/0	172.16.9.1	255.255.255.0	
	S0/1/1	172.16.10.1	255.255.255.0	
COMMUTATEUR	VLAN 1	172.16.5.10	255.255.255.0	172.16.5.1
PC BOUI 1		172.16.5.251	255.255.255.0	172.16.5.1
PC BOUI 2		172.16.3.252	255.255.255.0	172.16.5.1

PC BOUI 3		172.16.3.253	255.255.255.0	172.16.5.1
PC BOUI 4		172.16.3.254	255.255.255.0	172.16.5.1
R MOUSTAPHA	FA0/0	172.16.6.1	255.255.255.0	
	S0/0/0	172.16.12.2	255.255.255.0	
	S0/0/1	172.16.11.2	255.255.255.0	
	S0/1/0	172.16.10.2	255.255.255.0	
	S0/1/1	172.16.14.2	255.255.255.0	
COMMUTATEUR	VLAN 1	172.16.6.10	255.255.255.0	172.16.6.1
PC MOUS 1		172.16.6.1	255.255.255.0	172.16.6.1
PC MOUS 2		172.16.6.1	255.255.255.0	172.16.6.1
PC MOUS 3		172.16.6.1	255.255.255.0	172.16.6.1
PC MOUS 4		172.16.6.1	255.255.255.0	172.16.6.1
R BEJAIA	FA 0/0	172.16.4.1	255.255.255.0	
	S0/0/0	172.16.13.2	255.255.255.0	
	S0/0/1	172.16.8.1	255.255.255.0	
	S0/1/0	172.16.9.2	255.255.255.0	
	S0/1/1	172.16.14.1	255.255.255.0	
COMMUTATEUR	VLAN 1	172.16.4.10	255.255.255.0	172.16.4.1
PC BEJ 1		172.16.4.251	255.255.255.0	172.16.4.1
PC BEJ 2		172.16.4.252	255.255.255.0	172.16.4.1
PC BEJ 3		172.16.4.253	255.255.255.0	172.16.4.1
PC BEJ 4		172.16.4.254	255.255.255.0	172.16.4.1

II.B.3.2 Configuration IP des stations :

Nous avons configuré les adresses IP de chaque PC de tous les sites. L'adresse de la passerelle est celle par la quelle toutes les trames devront passer pour atteindre une autre adresse d'un autre réseau local. On mit donc comme adresse de passerelle celle de l'interface fastEternet du routeur qui est dans le même réseau local.

Pour ceci nous suivons les étapes suivantes:

Ø A fin d'attribuer les adresses aux PC on click sur le pc dans le menu on choisit l'onglet **desktop** on click sur **IP configuration** on aura un menu ou il est possible de faire entrer les adresses IP pour chaque PC le masque ainsi que l'adresse de passerelle par défaut, le tous est illustré par la figure qui suit :

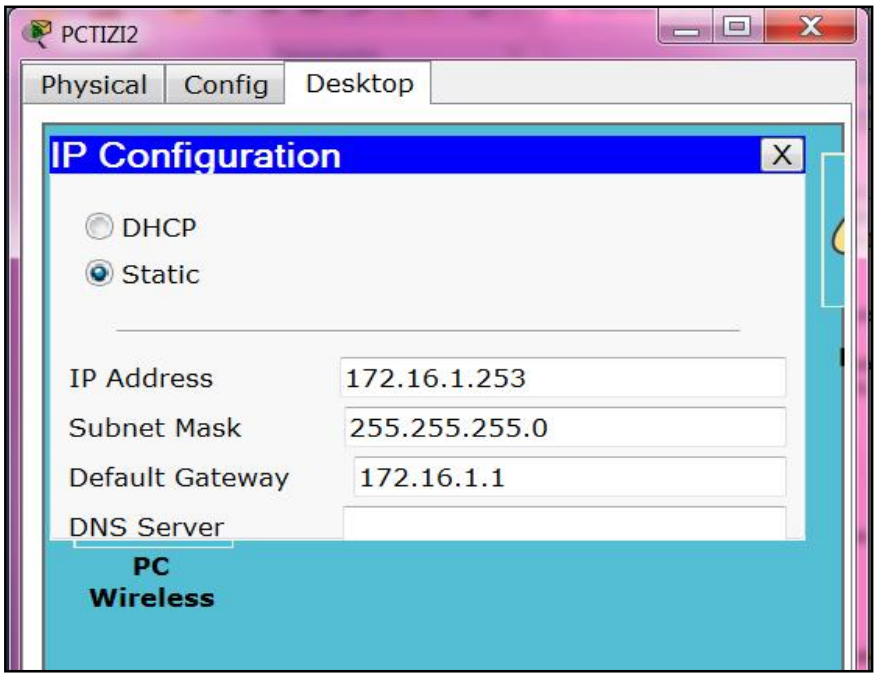


Figure II.B.9: Procédures d'attribution des adresses IP d'un PC.

Notons cependant qu'il faut répéter la même procédure pour tous les PC des différents sites.

II.B.3.3 Attribution d'adresses aux interfaces du Switch :

Comme première adresse celle du VLAN 1 par ce qu'on a configuré qu'un seul vlan qui est le vlan 1 et on lui a attribué l'adresse comme dans le tableau de la façon suivante :

```
switchtizi#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
switchtizi(config)#interface vlan 1
switchtizi(config-if)#ip address 172.16.1.10 255.255.255.0
switchtizi(config-if)#no shut down
switchtizi(config)#ip default gateway 172.16.1.1
```

Après avoir configure cette interface les voyant orange des connexions Switch - pc devient vert indiquant que la liaison est établi.

Bien sur il faut mètre un câble directe entre les PC et le Switch ainsi qu'entre le Switch et le routeur si non même si la configuration est juste les voyants des connexions resteront rouges indiquant un problème de câblage.

- **Configuration de l'interface fa0/5 reliée au routeur TIZI :**

Par défaut cette interface est configurée en mode Access il faut la reconfigurer en mode trunk pour préparer notre Switch à recevoir des trames venant d'un réseau différent ou d'un vlan différent.

```
switchtizi#config ter
enter configuration commands, one per line.  end with cntl/z.
switchtizi(config)#interface fa0/5
switchtizi(config-if)#switchtchport mode trunk
switchtizi(config-if)#^z
%sys-5-config_i: configured from console by console
switchtizi#copy run star
destination filename [startup-config]?
building configuration...
[ok]
```

II.B.3.4 Configuration des fonctions de sécurité sur le Switch :

- **Sécurisation des ports du Switch :**

Adresse mac statique ou dynamique ?

Par défaut les adresse mac d'une table d'adresses mac d'un Switch sont prises d'une façon dynamique c'est à dire dé qu'un pc est débranché du commutateur au bout d'un certain temps son adresse mac sera éliminé et bien sur entre temps l'adresse du nouveau PC sera prise en compte. Dans cette partie on va configurer les adresse de la table du Switch pour quelles soient statiques ainsi aucun intrus branchant son PC ne pourra pénétrer le système.

```
Switch#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#mac-address-table static 00D0.D3C7.5760  vlan 1  interface fa0/1
Switch(config)#mac-address-table static 0001.42E8.A363 vlan 1  interface  fa0/2
Switch(config)#mac-address-table static 0060.47AD.B2BA vlan 1 interface f0/3
Switch(config)#mac-address-table static 00D0.D3E4.6B89  vlan 1 interface f0/4
Switch(config)#mac-address-table static 0090.2B97.4201 vlan 1 interface f0/5
Switch(config)#exit
```

– **Vérification de la configuration:**

```
switchtizi#show mac-address-table
        Mac Address Table
------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----

   1    0001.42e8.a363    STATIC      Fa0/2
   1    0060.47ad.b28a    STATIC      Fa0/3
   1    0090.2b97.4201    STATIC      Fa0/5
   1    00d0.d3c7.5760    STATIC      Fa0/1
   1    00d0.d3e4.6b89    STATIC      Fa0/4
```

Donc la configuration ne permet qu'aux PC de la station de Tizi-Ouzou d'entrer au système et aucun autre PC étranger ne pourra s'introduire, son accès est un échec.

Et puis pour chaque port on configure le nombre maximum d'adresse mac recevables ainsi que de configurer la désactivation du port en ca d'une éventuel intrusion. La nouvelle configuration et comme suit :

```
switchtizi(config)#interface fa0/1
switchtizi(config-if)#switchport port-security
switchtizi(config-if)#switchport port-security maximum 1
switchtizi(config-if)#switchport port-security mac-address sticky
switchtizi(config-if)#switchport port-security violation shutdown
switchtizi(config-if)#exit
switchtizi(config)#interface fa0/2
switchtizi(config-if)#switchport port-security
switchtizi(config-if)#switchport port-security maximum 1
stickyswitchtizi(config-if)#switchport port-security mac-address sticky
switchtizi(config-if)#switchport port-security violation shutdown
switchtizi(config-if)#exit
switchtizi(config)#interface fa0/3
switchtizi(config-if)#switchport port-security
switchtizi(config-if)#switchport port-security maximum 1
switchtizi(config-if)#switchport port-security mac-address sticky
switchtizi(config-if)#switchport port-security violation shutdown
switchtizi(config-if)#exit
switchtizi(config)#interface fa0/4
switchtizi(config-if)#switchport port-security
switchtizi(config-if)#switchport port-security maximum 1
switchtizi(config-if)#switchport port-security mac-address sticky
switchtizi(config-if)#switchport port-security violation shutdown
switchtizi(config-if)#exit
switchtizi(config)#interface fa0/5
switchtizi(config-if)#switchport port-security
switchtizi(config-if)#switchport port-security maximum 1
switchtizi(config-if)#switchport port-security mac-address sticky
switchtizi(config-if)#switchport port-security violation shutdown
switchtizi(config-if)#exit
switchtizi(config)#exit
```

– **Vérification de la configuration :**

```
interface FastEthernet0/1
 switchport port-security mac-address sticky
!
interface FastEthernet0/2
 switchport port-security mac-address sticky
!
interface FastEthernet0/3
```

```
 switchport port-security mac-address sticky
!
interface FastEthernet0/4
 switchport port-security mac-address sticky
!
interface FastEthernet0/5
 switchport mode trunk
 switchport port-security mac-address sticky
```

– **Désactivation des ports non utilisés :**

Une méthode simple à laquelle de nombreux d'administrateurs ont recours pour mieux protéger leur réseau contre tout accès non autorisé, est de désactiver tous les ports qui ne sont pas exploités sur un commutateur réseau. On utilise la commande **« shutdown »** sur toutes les interfaces non utilisées.

II.B.3.5 Configuration des routeurs :

- **Modes d'accès :** On peut configurer un routeur, en utilisant :

- Soit le mode console,
- Soit une plate forme d'administration (exemple Cisco Works qui utilise des requêtes SNMP).

Pour atteindre le menu de configuration d'un routeur, on utilise :

- Soit le port console du routeur,
- Soit des terminaux virtuels en émulation Telnet.

Les routeurs Cisco fonctionnent dans trois modes différents. Le mode ligne de commande **'exc.'** permet d'exécuter quelques commandes de base, mais sans modifier la configuration du routeur. Le mode administrateur **'exc. privilégié'** permet de modifier certains paramètres du routeur et d'accéder à des commandes complémentaires. Le mode **'global configuration'** permet lui de modifier complètement la configuration du routeur. Lorsque l'on se connecte à un routeur Cisco (via console ou via Telnet) on se retrouve dans le mode ligne de commande.

Le prompt est alors > précédé du nom du routeur :

```
Router >
```

L'utilisateur est alors capable de passer différents types de commandes (Ping, show, etc.) mais il ne peut modifier la configuration du routeur.

Pour passer en mode administrateur, il suffit de rentrer l'instruction **enable**, et de donner s'il existe, le mode de passe pour passer dans ce mode. Le prompt change alors pour devenir **#** :

```
Router> enable
Router #
```

Les deux lignes précédentes nous ont permis de passer en mode administrateur, nous allons maintenant passer en mode configuration du routeur :

```
Router # configure terminal
Router (config)#
```

Pour ressortir de ce mode configuration, il suffit de taper la commande **<ctrl> Z** et l'on revient au mode privilégié. Pour remonter les niveaux d'accès ou pour sortir du mode privilégié, on utilise la commande **disable** ou tout simplement la commande **exit**.

Lorsque l'on est dans le mode configuration, il est possible de passer tous les ordres s'appliquant de manière globale au routeur.

- **Sauvegarde de configuration :**

Lorsque l'on passe en mode configuration globale à l'aide de la commande :

```
Router# configure terminal
```

On est alors dans la **RAM** et toute commande passée à un effet immédiat. La commande **# show configure** permet d'afficher la configuration sauvegardée dans la **NvRam**(et pas forcément celle actuellement en fonction dans la **(**RAM). Pour afficher la configuration qui fonctionne actuellement et qui se trouve dans la **RAM** (et qui peut être différente de celle stockée dans la **NvRam**, il suffit de taper la commande : **# write terminal.** Et finalement, lorsque l'on souhaite par précaution (en cas de coupure de courant) sauvegarder la configuration actuelle de la **RAM** dans la **NvRam**, on utilise la commande :

```
# write memory
```

- **Configuration de base d'un routeur Cisco :**

Il existe plusieurs méthodes pour effectuer la configuration basique d'un routeur Cisco :

- L'utilisation du mode « setup ».
- La configuration manuelle en « ligne de commande CLI »on s'intéresse à ça justement.

La configuration des routeurs est stockée (NVRAM) dans un fichier appelé **« startup-config »** qui est utilisé au démarrage du routeur.il est alors chargé en RAM ou il devient le fichier **« running-config ».**

Chaque commande saisie en mode console modifie directement la configuration stockée en RAM. En cas d'erreur, il est donc possible de revenir en arrière pour restaurer une configuration validée **(copy startup-config running-config**) par **reload startup-config,** toutes les modifications sont alors perdues.

II.B.3.5.1 Configuration des noms :

Le nom d'un routeur n'est pas un élément primordial dans la configuration d'un routeur et dans son fonctionnement. Pour autant il peut permettre d'éviter de faire des erreurs lors de la configuration de plusieurs périphériques réseau au même moment.

Comme première étape doc on configure le nom du routeur qui sera « tizi » pour le routeur de Tizi-Ouzou et « ALGER » pour le routeur d'Alger et « BOUIRA », « BEJAIA » et « MOUSTAPHA » pour les autres routeurs.

Routeur de Tizi-Ouzou :

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname tizi
tizi(config)#exit
```

II.B.3.5.2 Configuration des mots de bannière pour les routeurs :

La bennière d'un routeur est un message adressé à tout utilisateurs se connectant au routeur via console ou Telnet et ssh, le prévenir que l'accès au périphérique est restreint et surveiller.

```
tizi#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
tizi(config)#banner motd #acces avec autorisation seulement#
tizi(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
```

La prochaine connexion au routeur nous donnera :

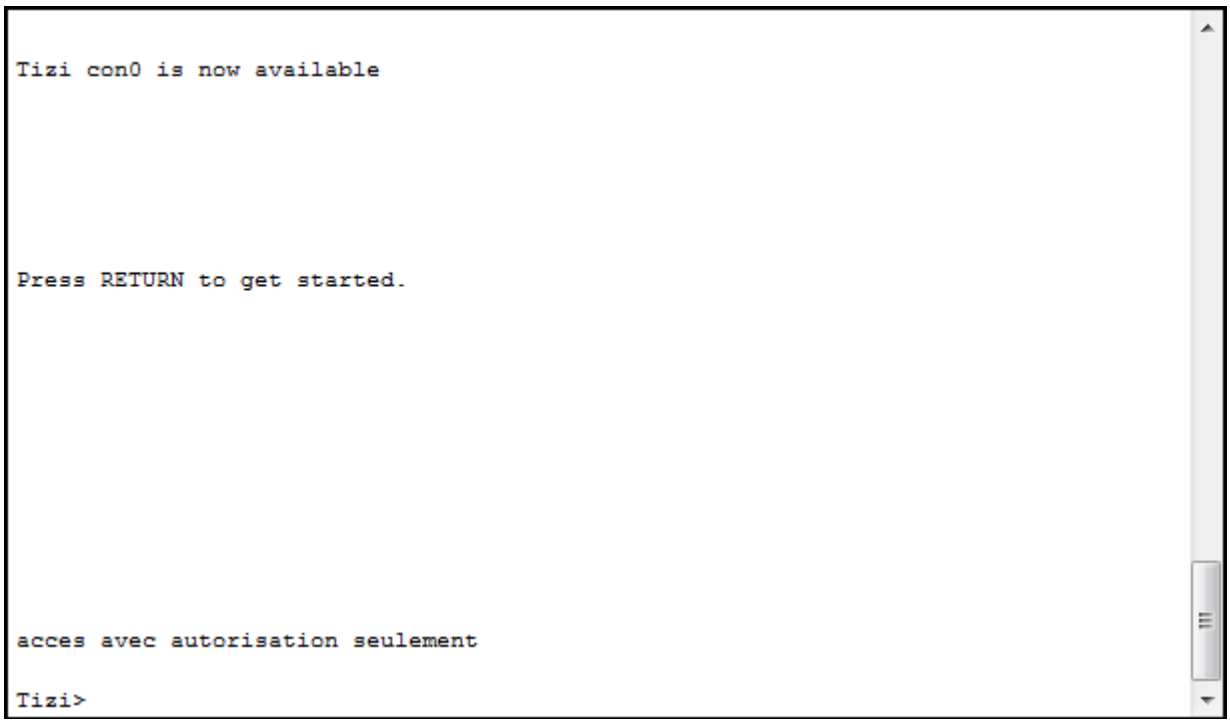


Figure II.B.10 : affichage de la bannière.

II.B.3.5.3 La configuration des mots de passes :

- **Configuration de l'accès à la console :**

Pour sécuriser les routeurs et commutateurs, on doit avant tout les protéger contre tout accès non autorisé.

On pourra effectuer toutes les opérations de configuration directement à partir de la console. Pour accéder à la console, on doit bénéficier d'un accès physique local au périphérique. Si on ne sécurise pas le port de la console comme il se doit, un utilisateur malveillant risque de compromettre la configuration de ces périphériques.

```
tizi#configure terminal
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#line console 0
```

La on passe en mode de configuration de line console 0, l'invite change et la commande qui suit nous aide à introduire le mot de passe :

```
tizi(config-line)#password cisco
```

Une foi que c'est fait il faut s'assurer que l'utilisateur n'accèdera en mode console qu'une foi le mot de passe saisi. On tape donc la commande **« login ».**

```
tizi(config-line)#login
tizi(config-line)#end
```

- **Configuration du mot de passe Telnet ou VTY :**

La configuration se fait comme suit :

```
tizi#configure terminal
enter configuration commands, one per line.  end with cntl/z.
tizi(config-line)#line vty 0 15
tizi(config-line)#password cisco
tizi(config-line)#login
tizi(config-line)#end
```

- **Configuration du mot de passe en mode d'exécution :**

Le mode d'exécution privilégié permet à toutes les personnes qui l'utilisent sur un routeur ou commutateur Cisco de configurer toutes les options disponibles sur ces derniers. On peut également afficher tous les paramètres actuellement configurés sur ces périphériques, y compris certains des mots de passe non chiffrés. C'est pourquoi il est primordial de sécuriser l'accès au mode d'exécution privilégié.

```
tizi#config ter
enter configuration commands, one per line.  end with cntl/z.
tizi (config)#enable secret class
tizi (config)#end
```

- **Verification de la configuration:**

Après avoir exécuté la commande « **show run** » on aura les informations suivantes.

```
line con 0
 password cisco
 login
!
line vty 0 4
 password cisco
 login
line vty 5 15
 password cisco ….[omni d'autres résultats]
```

La on Remarque que les mots de passes sont visible, pour plus de sécurité les mots de passes doivent êtres chiffrés grâce à la syntaxe :

```
tizi (config)#service password-encryption
```

Et on n'oubli pas de sauvegarder la nouvelle configuration.

```
tizi #copy run star
Destination filename [startup-config]?
Building configuration...
[OK]
```

Vérification si la nouvelle instruction a été prise en compte toujours « show run » :

```
!
line con 0
 password 7 08325B470A1016141D
 login
!
line vty 0 4
 password 7 0822455D0A16
 login
line vty 5 15
 password 7 0822455D0A16[omni autres résultats]
```

Les mots de passé sont maintenant chiffrés.

La configuration des mots de passes d'accès Telnet, console et celui du mode privilégier reste les même pour les différents routeurs et commutateurs du réseau. Sachant que les mots de passe sont des mots de passe par défaut et qu'il ne faut jamais configurer des mots de passe par défaut, question de sécurité.

II.B.3.5.4 Configuration des interfaces du routeur :

- **Interface vers le réseau LAN (passerelle) :**

tizi>enable

Le résultat de cette commende change l'invite en :

tizi#

Puis on entre la commende « configure terminal » pour pouvoir configurer le routeur :

tizi#configure terminal

enter configuration commands, one per line. end with cntl/z.

tizi(config)#interface fa0/0

On a spécifié l'interface, il s'agit de l'interface du routeur relier au switch et maintenant on lui donne une adresse et un masque comme suit :

tizi(config-if)#ip address 172.16.1.1 255.255.255.0

A la fin on active l'interface grâce à la commande **no shutdown.**

tizi(config-if)#no shutdown

- **Configuration des interfaces séries:**

Ces interfaces nous permet de connecter le routeur aux différents réseaux étendus, on dispose de 4 connections réseau étendu pour le routeur Cisco 2811.la configuration des interfaces séries ne diffère pas vraiment des interfaces réseau local, la seul différence est l'attribution du signale d'horloge. En absence d'un modem, qui généralement fournie le signale d'horloge nous serons donc contrains de designer le routeur qui va jouer le rôle du DCE pour l'architecture de la figure II .B .14.

- Pour toutes les liaisons séries des routeur de la station MOBILIS de Tizi-Ouzou Alger, Moustapha, Brouira et Bejaïa on désignera le routeur de Tizi-Ouzou comme DCE.
- Les liaisons Alger, Bouira, Bejaia et Moustafa on désigne le routeur d'Alger comme DCE.
- Celle de Moustafa vers Bejaia se sera donc Moustafa comme DCE.
- Il reste Bouira vers Moustafa et Bejaia on a pris Bouira comme DCE.

Les autres terminaux intermédiaires seront pour ces liaisons des DTE.

- La configuration des interfaces séries pour le routeur de Tizi par exemple sera :

```
tizi>enable
tizi#configure terminal
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#interface s0/0/0
tizi(config-if)#ip address 172.16.2.1 255.255.255.0
tizi(config-if)#clock rate 2000000
tizi(config-if)#description liaison vers alger
tizi(config-if)#no shutdown
```

Et à la fin il faut toujours sauvegarder la nouvelle configuration dans la NVRAM (copy running -config startup -config).

Pour les autres interfaces séries qui ne sont pas des DCE la configuration reste la même on élimine uniquement la fréquence d'horloge.

```
TIZI#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        172.16.1.1      YES manual up                    up
FastEthernet0/1        unassigned      YES manual administratively down down
Serial0/0/0            172.16.2.1      YES manual up                    up
Serial0/0/1            172.16.12.1     YES manual up                    up
Serial0/1/0            172.16.15.1     YES manual up                    up
Serial0/1/1            172.16.13.1     YES manual up                    up
FastEthernet1/0        unassigned      YES manual administratively down down
FastEthernet1/1        unassigned      YES manual administratively down down
```

Up veut dire quelles sont activé niveau haut.

Down celles qui son pas actives.

II.B.3.5.5 Configuration de l'encapsulation ppp :

Les données de la couche réseau sont transférées à la couche liaison de données afin d'être livrées sur une liaison physique, généralement point à point sur une connexion de réseau étendu. La couche liaison de données établit une trame autour des données de la couche réseau, de telle sorte que les vérifications et contrôles nécessaires puissent être effectués. Chaque type de connexion de réseau étendu utilise un protocole de couche 2 pour encapsuler un paquet pendant qu'il traverse la liaison longue distance. Pour assurer que le protocole d'encapsulation correct est utilisé, le type d'encapsulation de couche 2 utilisé pour l'interface série de chaque routeur doit être configuré. Le choix du protocole d'encapsulation est fonction de la technologie de réseau étendu et de l'équipement.

Par défaut l'encapsulation utilisée par les routeurs Cisco est le HDLC mais nous on va configurer l'encapsulation PPP, cette dernière permet deux types d'authentification différents : **PAP** (Password Authentication Protocol ou protocole d'authentification du mot de passe) et **CHAP** (Challenge Handshake Authentication Protocol) ou protocole d'authentification à échanges confirmés). PAP utilise un mot de passe sous forme de texte en clair, tandis que CHAP fait appel à une empreinte numérique à sens unique qui offre plus de sécurité que PAP.

```
tizi(config)#interface s0/0/0
tizi(config-if)#encapsulation PPP
tizi (config-if)#exit
tizi (config)#interface s0/0/1
tizi (config-if)#encapsulation PPP
tizi (config-if)#exit
tizi (config)#interface s0/1/1
tizi (config-if)#encapsulation PPP
tizi (config-if)#exit
tizi (config)#interface s0/1/0
```

```
tizi (config-if)#encapsulation PPP
tizi (config-if)#exit
```

– **Configuration du CHAP :**

Maintenant la configuration du l'encapsulation est faite, mais aucun système d'authentification est activé. Comme dit précédemment au niveau du routeur il ya deux types d'authentification soit du PAP soit du CHAP. Pour nous ça sera plutôt le CHAP.

Pour pouvoir configurer l'authentification par le CHAP entre les routeurs et celui du MSC de Tizi, la première étape est de créer des comptes locaux pour tous les routeurs, puis définir quelle est l'interface sur la quelle on configure l'authentification par exemple entre le routeur de Tizi et celui d'Alger la configuration est :

Ø On se palace en premier lieu sur le routeur d'Alger et on créé un compte locale pour le routeur de Tizi, il aura pour nom « tizi » et mot de passe « toto » :

```
Password:
ALGER>enable
Password:
ALGER#configure Terminal
Enter configuration commands, one per line.  End with CNTL/Z.
ALGER(config)#username tizi pass toto
```

Ø L'interface qui relié Tizi-Ouzou à Alger est S0/0/0, donc on entre dans la configuration de l'interface est on active le CHAP :

```
ALGER(config)#int s0/0/0

ALGER(config-if)#ppp authentication chap
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
ALGER(config-if)#ppp chap sent_username ALGER pass toto
ALGER(config-if)#no shutdown
ALGER(config-if)#exit
```

```
ALGER(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
ALGER#copy run star
Destination filename [startup-config]?
```

La commende qui est mit en jaune signifie que le login envoyé vers le routeur de Tizi-Ouzou est ALLGER qui est le nom du routeur lui-même et le mot de passe est « toto ».

- Maintenant on pace au routeur de Tizi et on active aussi l'authentification CHAP :

```
tizi>ena
Password:
tizi#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
tizi(config)#username ALGER pass toto
tizi(config)#int s0/0/0
tizi(config-if)#ppp authentication chap
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
tizi(config-if)#ppp chap sent-username tizi pass toto
tizi(config-if)#no shutdown
tizi(config-if)#exit
tizi(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
tizi#copy run star
Destination filename [startup-config]?
Building configuration...
```

- **Vérification de la nouvelle configuration :**

- **Sur le routeur de tizi :**

```
interface Serial0/0/0
 ip address 172.16.2.1 255.255.255.0
```

```
encapsulation ppp
ppp authentication chap
ppp pap sent-username tizi password 0 toto
clock rate 2000000000 (resultat Omni autres)
```

ü **Sur celui d'Alger:**

```
!
interface Serial0/0/0
 ip address 172.16.2.2 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username ALGER password 0 toto
!(resutat omni autres)
```

- Les mots de passe utilisés pour l'authentification doivent être identiques parce que Cisco utilise le MD5 pour la vérification de mots de passe.
- La configuration reste la même sur les autres routeurs du réseau.

II.B.3.5.6 Configuration du routage :

Une foi qu'on a donné des adresses IP à toutes les interfaces et des vlan de notre réseau vient maintenant le choix du routage. Deux types de routage sont utilisables. Le premier est le routage statique où les tables de routage sont alimentées manuellement sur chaque routeur du réseau. Le second est le routage dynamique où un protocole de routage est utilisé pour distribuer les routes dans l'ensemble du réseau.

Pour le réseau de Mobilis le protocole qui prend en charge le routage est le RIP version 2, mais pour des raisons de sécurité on choisit plutôt un routage statique.

Ø La configuration sur le routeur de Tizi-Ouzou sera comme suit :

```
tizi#configure terminal
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#ip route 172.16.5.0  255.255.255.0  172.16.15.2
```

```
tizi(config)#ip route 172.16.4.0  255.255.255.0  172.16.13.2
tizi(config)#ip route 172.16.6.0  255.255.255.0  172.16.12.2
tizi(config)#ip route 172.16.3.0  255.255.255.0  172.16.2.2
tizi(config-router)#exit
```

À noter que s'est la même méthode pour configurer les routes pour les autres routeurs du réseau.

- **Vérification de la configuration sur le routeur de Tizi-Ouzou :**

```
tizi#show ip route
Gateway of last resort is not set
    172.16.0.0/24 is subnetted, 9 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
S       172.16.3.0 [1/0] via 172.16.2.2
S       172.16.4.0 [1/0] via 172.16.13.2
S       172.16.5.0 [1/0] via 172.16.15.2
S       172.16.6.0 [1/0] via 172.16.2.2
                   [1/0] via 172.16.12.2
C       172.16.12.0 is directly connected, Serial0/0/1
C       172.16.13.0 is directly connected, Serial0/1/1
C       172.16.15.0 is directly connected, Serial0/1/0
```

C pour les réseaux directement connectés.

On remarque que dans la table de routage sont ajoutés des route statique permettant au routeur d'accéder aux différents réseaux.

II.B.3.5.7 Teste de la connectivité :

Ping du réseau local de Tizi-Ouzou vers tous les réseaux aux quels il est relier, le résultat est comme suivante :

```
tizi#ping 172.16.5.0
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.5.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/2 ms
```

```
tizi#ping 172.16.4.0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.4.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/4/12 ms
tizi#ping 172.16.3.0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/10/17 ms
tizi#ping 172.16.6.0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/13/17 ms
```

II.B.3.5.8 Configuration du NAT:

Le NAT :

Le NAT (Network Address Translation) ou translation d'adresses IP permet de faire la relation entre une ou plusieurs adresses privées et une ou plusieurs adresses publiques. Nous distinguons deux types de NAT.

- **Le NAT statique :**

Permet la translation d'une adresse privée en adresse publique.

- **Le NAT dynamique :**

Il s'agit de partager une adresse publique entre plusieurs adresses privées.

Pour un lien vers l'extérieur on a choisit l'interface fastEthernet 0/1 à la quelle on a attribué l'adresse 172.16.25.1 pour un masque 255.255.255.0.

```
tizi>ena
password:
tizi#config ter
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#int fa0/1
tizi(config-if)#ip address 172.16.25.1 255.255.255.0
tizi(config-if)#no shutdown
%link-5-changed: interface fastethernet0/1, changed state to up
%lineproto-5-updown: line protocol on interface fastethernet0/1, changed state to up
tizi(config-if)#exit
```

L'interface est maintenant active, il reste de lui attribuer l'adresse publique vers l'extérieur, la configuration est comme suit :

```
tizi(config)#ip nat inside source static 172.16.25.1 120.0.0.1
tizi(config)#int fa0/1
tizi(config-if)#IP nat inside
tizi(config-if)#exit
tizi(config)#int fa0/1
tizi(config-if)#ip nat outside
tizi(config-if)#exit
tizi(config)#
```

On a choisi une configuration statique pour des raisons de sécurité.

II.B.3.5.9 Filtrage du trafique :

Les administrateurs réseau doivent trouver le moyen d'interdire l'accès au réseau à certains utilisateurs tout en permettant aux utilisateurs internes d'accéder aux services nécessaires. Bien que les outils permettant d'assurer la sécurité, tels que les mots de passe, l'équipement de rappel et les dispositifs de sécurité physiques, se révèlent utiles, dans la plupart des cas, ils n'offrent pas la souplesse que procurent le filtrage de trafic de base et les contrôles spécifiques que leur préfèrent la majorité des administrateurs. Ainsi, il se peut qu'un

administrateur réseau souhaite accorder l'accès à Internet aux utilisateurs, tout en interdisant à des utilisateurs externes l'accès au réseau LAN via Telnet.

Les routeurs assurent les fonctions de base de filtrage du trafic, telles que le blocage du trafic Internet, à l'aide de listes de contrôle d'accès. Une liste de contrôle d'accès est un ensemble séquentiel d'instructions d'autorisation ou de refus qui s'appliquent aux adresses ou aux protocoles de couche supérieure.

Les **ACL** permettent de filtrer des paquets suivant des critères définis par l'utilisateur. Il est ainsi possible de filtrer les paquets entrants ou sortants d'un routeur en fonction de l'adresse IP source ou IP de destination.

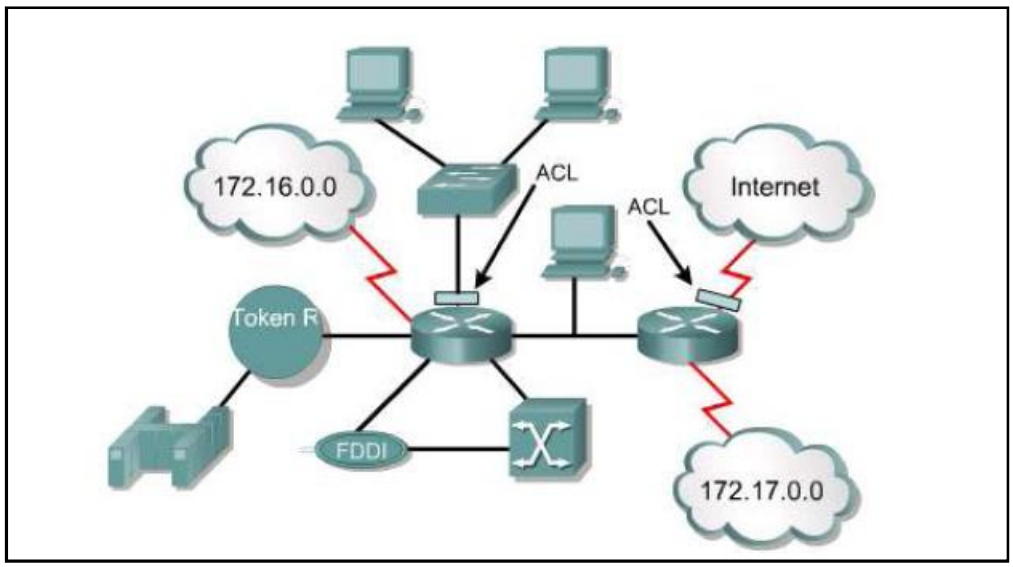


Figure II.B.11: emplacement des ACL.

– **Fonctionnement des ACL :**

Il est possible de résumer le fonctionnement d'une ACL comme suit :

- Le paquet est vérifié par rapport au 1er critère défini.
- S'il vérifie le critère, l'action définie est appliquée.
- Sinon le paquet est comparé successivement par rapport aux ACL suivants.
- S'il ne satisfait aucun critère, l'action **deny** est appliquée.

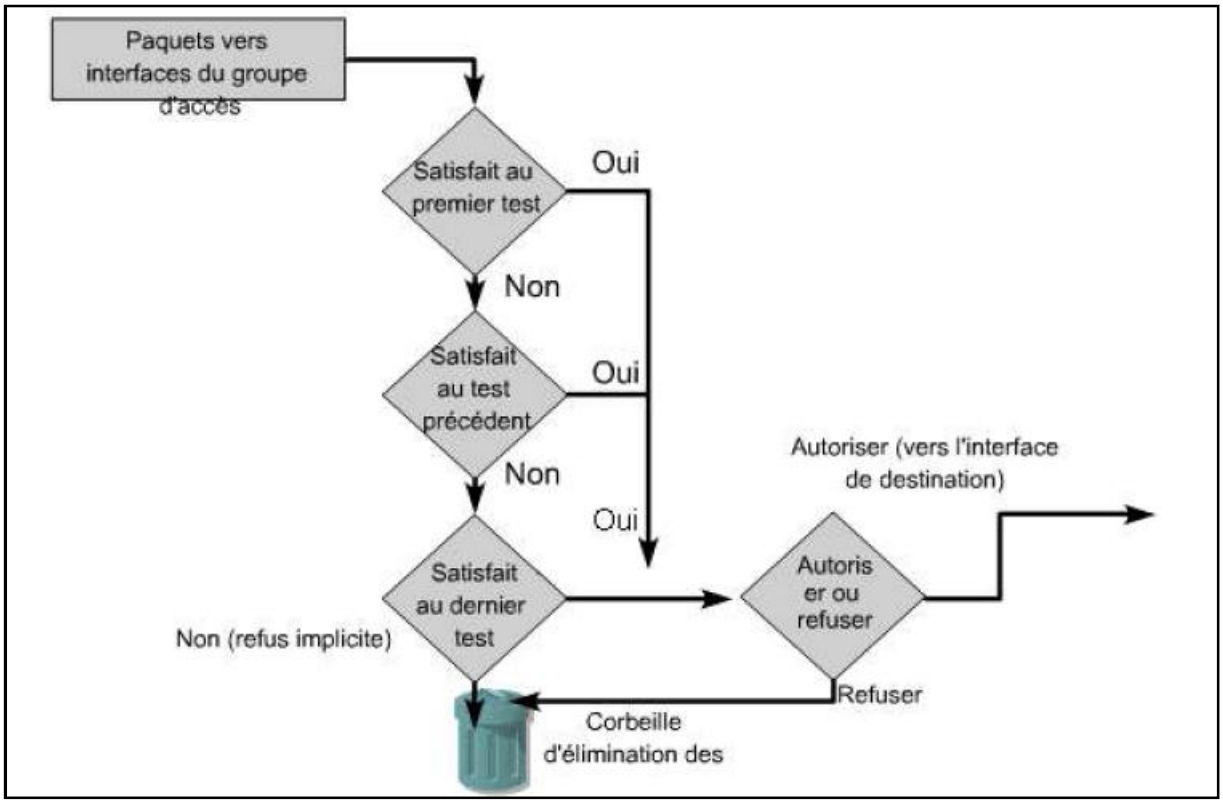


Figure II.B.12 : Fonctionnement des ACL.

Les critères sont définit sur les informations contenues dans les en-têtes IP, TCP ou UDP.

Des masques (wildcard mask) ont été définis pour pouvoir identifier une ou plusieurs adresses IP en une seule définition. Ce masque défini la portion de l'adresse IP qui doit être examinée.

- **Configuration des ACL :**
  - § La protection contre les attaques douteuses. Pour cela on interdit les adresses IP commençant par 127 et qui sont des adresses réservées.

```
tizi(config)#access-list 12 deny  127.0.0.0 0.0.0.0
tizi(config)#int fa0/0
tizi(config-if)#ip access-group 12 out
tizi(config-if)#exit
```

§ Filtrer certains services venant de l'extérieur :

bootp (67 UDP),), tftpd(69,UDP), syslog(514,UDP), sunrpc(111,TCP/UDP), snmp(161,UDP) xdmcp(177,UDP), login(513,TCP), shell(514,TCP), RDP (3389), Netbios (1001,1002), imap (143 TCP), ipx(213,UDP/TCP), wins(1512,UDP/TCP), microsoft (135>139), SNMP,194 …etc.

```
tizi(config)#access-list 101 deny udp any any eq 138
tizi(config)#access-list 101 deny udp any any eq 137
tizi(config)#access-list 101 deny udp any any eq 135
tizi(config)#access-list 101 deny udp any any eq bootps
tizi(config)#access-list 101 deny udp any any eq tftp
tizi(config)#access-list 101 deny tcp any any eq 111
tizi(config)#access-list 101 deny tcp any any eq 135
tizi(config)#access-list 101 deny tcp any any eq 95
tizi(config)#access-list 101 deny tcp any any eq 87
tizi(config)#access-list 101 deny tcp any any eq 137
tizi(config)#access-list 101 deny udp any any eq 514
tizi(config)#access-list 101 deny udp any any eq 67
tizi(config)#access-list 101 deny udp any any eq 161
tizi(config)#int fa0/0
tizi(config-if)#ip access-group 101 out
tizi(config-if)#ext
```

§ Interdire au autres PC du réseau local du MSC l'accès en dors du réseau local :

```
 tizi>ena
password:
password:
tizi#config ter
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#ip access-list standar no-access
tizi(config-std-nacl)#deny host 172.16.1.251
tizi(config-std-nacl)#deny host 172.16.1.252
tizi(config-std-nacl)#deny host 172.16.1.253
tizi(config-std-nacl)#permit any
tizi(config-std-nacl)#exit
tizi(config)#int fa0/0
tizi(config-if)#ip accs
tizi(config-if)#ip access-group no-access in
tizi(config-if)#exit
tizi(config)#exit
```

§ Permettre au pc du chef du centre l'accès libre vers tous les réseaux :

```
tizi(config)#no access-list 100
tizi(config)#access-list 100 permit ip 172.16.1.254  0.0.0.0  172.16.3.0 0.0.0.255
tizi(config)#access-list 100 permit ip 172.16.1.254  0.0.0.0  172.16.4.0 0.0.0.255
tizi(config)#access-list 100 permit ip 172.16.1.254  0.0.0.0  172.16.5.0 0.0.0.255
tizi(config)#access-list 100 permit ip 172.16.1.254  0.0.0.0  172.16.6.0 0.0.0.255
tizi(config)#int fa0/0
tizi(config-if)#ip access-group 100 in
tizi(config-if)#exit
tizi(config)#exit
```

§ Définir les réseaux qui sont autorisées à accéder au routeur du MSC :

```
tizi(config)#no access-list 2
tizi(config)#access list 2 permit 172.16.3.0
tizi(config)#access-list 2 deny any
tizi(config)#int s0/0/0
tizi(config-if)#ip access-group 2 in
tizi(config-if)#exit
tizi(config)#no access-list 3
tizi(config)#access-list 3 permit 172.16.5.0
tizi(config)#access-list 3 deny any
tizi(config)#int s0/1/0
tizi(config-if)#ip access-group 3 in
tizi(config-if)#exit
tizi(config)#no access-list 4
tizi(config)#access-list 4 permit 172.16.6.0
tizi(config)#access-list 4 deny any
tizi(config)#int s0/1/1
tizi(config-if)#ip access-group 4 in
tizi(config-if)#exit
```

```
tizi(config)#no access-list 5
tizi(config)#access-list 5 permit 172.16.4.0
tizi(config)#access-list 5 deny any
tizi(config)#int s0/1/1
tizi(config-if)#ip access-group 5 in
tizi(config-if)#exit
```

§ Restreindre l'accès aux vty:

```
tizi#conf t
enter configuration commands, one per line.  end with cntl/z.
tizi(config)#access-list 10 permit 172.16.4.254 0.0.0.0
tizi(config)#access-list 10 permit 172.16.5.254 0.0.0.0
tizi(config)#access-list 10 permit 172.16.6.254 0.0.0.0
tizi(config)#access-list 10 permit 172.16.3.254 0.0.0.0
tizi(config)#access-list 10 deny any
tizi(config)#line vty 0 15
tizi(config-line)#login
tizi(config-line)#password secret
tizi(config-line)#access-class 10 in
tizi(config-line)#exit
tizi(config)#exit
```

§ Configuration d'une liste de contrôle d'accès étendus nommée pour permettre les requêtes tcp par le port 80 et 443 :

```
tizi(config)#ip access-list extended dd
tizi(config-ext-nacl)#permit tcp 172.16.1.254 0.0.0.0 any eq 80
tizi(config-ext-nacl)#permit tcp 172.16.1.254 0.0.0.0 any eq 443
tizi(config-ext-nacl)#exit
tizi(config)#int fa0/0
tizi(config-if)#ip access-group dd in
tizi(config-if)#exit
tizi(config)#int fa0/0
tizi(config-if)#ip access-group dd in
tizi(config-if)#exit
tizi(config)#exit
```

§ Permettre des requêtes icmp de tous les réseaux vers le routeur de Tizi-Ouzou :

```
tizi#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
tizi(config)#access-
tizi(config)#access-list 103 permit icmp 172.16.3.254 0.0.0.0 172.16.1.0 0.0.0.255
tizi(config)#access-list 104 permit icmp 172.16.4.254 0.0.0.0 172.16.1.0 0.0.0.255
tizi(config)#access-list 105 permit icmp 172.16.5.254 0.0.0.0 172.16.1.0 0.0.0.255
tizi(config)#access-list 106 permit icmp 172.16.6.254 0.0.0.0 172.16.1.0 0.0.0.255
tizi(config)#access-list 108 permit icmp 172.16.1.254 0.0.0.0 172.16.1.0 0.0.0.255
tizi(config)#int S0/0/0
tizi(config-if)#ip access-group 103 in
tizi(config-if)#exit
tizi(config)# int S0/1/1
```

```
tizi(config-if)#ip access-group 104 in
tizi(config-if)#exit
tizi(config)# int S0/1/0
tizi(config-if)#ip access-group 105 in
tizi(config-if)#exit
tizi(config)# int S0/0/1
tizi(config-if)#ip access-group 106 in
tizi(config-if)#exit
tizi(config)#int fa0/0
tizi(config-if)#ip  access-group 108 in
tizi(config-if)#exit
```

§ Permettre des requêtes icmp pour le pc du chef de centre du MSC vers tous les réseaux voisins a travers toutes les interfaces séries :

```
tizi(config)#access-list 115 permit icmp 172.16.1.254 0.0.0.0 172.16.3.0 0.0.0.255
tizi(config)#access-list 115 permit icmp 172.16.1.254 0.0.0.0 172.16.4.0 0.0.0.255
tizi(config)#access-list 115 permit icmp 172.16.1.254 0.0.0.0 172.16.5.0 0.0.0.255
tizi(config)#access-list 115 permit icmp 172.16.1.254 0.0.0.0 172.16.6.0 0.0.0.255
tizi(config)#int s0/0/0
tizi(config-if)#ip access-group 115 out
tizi(config-if)#exit
tizi(config)#int s0/0/1
tizi(config-if)#ip access-group 115 out
tizi(config-if)#exit
tizi(config)#int s0/1/1
tizi(config-if)#ip access-group 115 out
```

```
tizi(config-if)#exit
tizi(config)#int s0/1/0
tizi(config-if)#ip access-group 115 out
tizi(config-if)#exit
tizi(config)#exit
```

§ Eviter des attaques douteuses :

- ne pas laisser entrer les réseaux suivants.
- 0.0.0.0/8 Historical Broadcast.
- 10.0.0.0/8 RFC.
- 169.254.0.0/16 Link.
- Local Networks.
- 192.0.2.0/24 TESTNET.
- 192.168.0.0/16 RFC.
- 240.0.0.0/5 Class E Reserved.
- 248.0.0.0/5 Unallocated.
- 255.255.255.255/32 Broadcast.
- 224.0.0.0/4 Class D Multicast.

```
tizi(config)#access-list  130 deny ip 0.0.0.0 0.255.255.255 any
tizi(config)#access-list  130 deny ip 10.0.0.0 0.255.255.255 any
tizi(config)#access-list  130 deny ip 169.254.0.0 0.0.255.255 any
tizi(config)#access-list  130 deny ip 192.168.0.0 0.0.0.255 any
tizi(config)#access-list  130 deny ip 240.0.0.0 7.255.255.255 any
tizi(config)#access-list  130 deny ip 248.0.0.0 7.255.255.255 any
tizi(config)#access-list  130 deny ip 255.255.255.255 0.0.0.0 any
tizi(config)#access-list 130 permit ip any any
tizi(config)#int s0/0/0
tizi(config-if)#ip access-group 130 in
tizi(config-if)#exit
```

```
tizi(config)#int s0/1/0
tizi(config-if)#ip access-group 130 in
tizi(config-if)#exit
tizi(config)#int s0/1/1
tizi(config-if)#ip access-group 130 in
tizi(config-if)#exit
tizi(config)#int s0/0/1
tizi(config-if)#ip access-group 130 in
tizi(config-if)#exit
tizi(config)#exit
```

Discussion:

Dans ce chapitre, nous avons configuré les commutateurs ainsi que les routeurs de tout le réseau, nous avons configuré les interfaces vers le réseau LAN et WAN. Les protocoles d'encapsulation Pour finir avec la configuration des fonctions de sécurité afin de prémunir les réseaux LAN des éventuels menaces venant de l'extérieur comme de l'intérieur du réseau.

Chapitre III :

Création d'un routeur avec FreeBSD

III.1 préambule:

L'idée de créer un routeur nous est venue lors de nos recherches sur la technologie Cisco. Nous avons appris que parmi les Unix libres il existe un système d'exploitation stable appelé FreeBSD. Ce dernier permet de concevoir un PC routeur se configurant avec la syntaxe de Cisco système grâce au routeur logiciel : le daemon **ZEBRA**.

Une occasion d'apprendre à configurer et à programmer avec Unix tout ceci avec un matériel recyclé permanant et réel ainsi que des applications avec des licences gratuites.

III.2 FreeBSD :

FreeBSD est un système d'exploitation haut de gamme dérivé de BSD ou Berkeley Software Distribution, la version Unix développée par le Computer Systems Research Group de l'Université de Californie de Berkeley entre 1975 et 1993. A la fois stable et performant, il est supporté sur un très grand nombre de plates-formes notamment (PC, Alpha, IA-64, PC-98 et UltraSPARC). Il est entièrement gratuit et fourni avec l'intégralité de son code source. La licence BSD (acronyme de Berkeley Software Distribution), autorise tout à chacun de réutiliser tout ou en partie le logiciel sans aucune restriction. Le développement du système est quasi permanent et est assuré par une équipe de développeurs volontaires et d'utilisateurs à travers le monde.

III.2.1 Avantage de FreeBSD par rapport autre version d'Unix:

- Le but principal de l'équipe de FreeBSD était de créer un système fonctionnant au mieux sur le compatible PC. C'est grâce à cela qu'il a était le premier capable d'utiliser des PC multiprocesseurs et de reconnaitre un grand nombre de carte d'extension sur PC.
- L'équipe du projet FreeBSD est composée de 100 développeurs dirigés par 17 chercheurs. Celle-ci garantit à FreeBSD une convergence, une gestion centralisée, une stabilité qui manquent dans les versions de Linux confrontées à des développements plus anarchiques, des incompatibilités entre distributions etc. puisque cette dernière est gérée autocratiquement par Linus Torvalds. D'ailleurs FreeBSD qui est un descendant d'Unix est compatible avec linux qui un clone d'Unix mais linux n'est pas compatible avec FreeBSD.
- FreeBSD est complètement gratuit. Le code source du système d'exploitation est fourni dans le package de base. Sa licence ne nous oblige pas de joindre le code

source tant que nous mentionnant clairement les emprunts que nous avons réalisé. Le code source Linux est aussi public mais est soumis aux conditions d'utilisation plus restrictives sa Licence oblige à inclure les codes sources de chaque changement effectuer sur le système.

- FreeBSD est un système d'exploitation 32 bits de A à Z. alors que, par exemple, Windows 98 contient encore énormément de portions de code 16 bits dans son noyau même s'il a été conçu pour accueillir des applications 32 bits.
- Il propose un multitâche préemptif avec ajustement dynamique des priorités des processus. Windows 3 est un système multitâche mais de type *coopératif*. Dans un système coopératif toutes les applications se partagent une seule queue (liste d'attente) et c'est l'application qui a la responsabilité de passer la main aux autres tâches. Si une application se perd dans une boucle de longue durée ou contient un bug qui l'empêche de se comporter correctement, l'ensemble des autres tâches n'a plus accès au temps CPU. Par contre dans un système multitâche *préemptif*, c'est le système d'exploitation lui-même qui, détermine quelle application prend le contrôle du processeur.
- FreeBSD est multiutilisateur. C'est probablement ce qui manque le plus à Windows NT qui ne dispose pas de plusieurs consoles simultanées et ne contient pas en standard de serveur Telnet.
- FreeBSD est doté de couches TCP/IP (Transport Control Protocol/Internet Protocole) très élaborées comportant notamment le support SLIP (Serial Line Internet Protocol), PPP (Point to Point Protocol), NFS (Network File System) et NIS (Network information System). FreeBSD propose les applications les plus populaires comme sendmail pour la gestion du serveur SMTP (Simple Mail Transfert Protocol), Apache pour le daemon http (Hypertexte Text Transport Protocol), Bind/named pour le serveur DNS (Domain Name System), squid comme serveur proxy etc. Des services de routage, pare feu et translation d'adresses réseau de très bonne qualité sont également fournis.
- Le gestionnaire de mémoire fournit un mécanisme de protection de la mémoire empêchant une application de manipuler un espace mémoire qui appartient à une autre application. Chaque programme a à sa disposition un espace mémoire découpé en pages de 4 KB. Le noyau de FreeBSD décharge dans le disque dur les pages qui n'ont plus été utilisées depuis longtemps. A chaque fois qu'une application tente d'accéder à une page qui n'est pas physiquement présente en mémoire, le noyau FreeBSD décharge de la mémoire les pages non utilisé et charge en mémoire la page demandée.

- Outils de développement C, C++, Fortran, Perl, fournis en standard.
- 2300 applications ont été portées sur FreeBSD et sont utilisables gracieusement. Il faut y ajouter toutes les applications Unix, BSD et Linux.
- L'installation de FreeBSD est nettement plus simple qu'une installation Linux. Une seule disquette d'installation suffit quelle que soit la configuration disque ou réseau. La création des partitions swap [annexe B] est automatique. Un seul programme : **sysinstall** gère la totalité des composants du système d'exploitation. Voir l'annexe B pour plus d'informations.

III.2.2 La preuve de la performance de FreeBSD :

De nombreux développeurs et managers ont choisi FreeBSD pour sa stabilité et ses performances en tant que serveur Web. Les PSI (Prestataires de Services Internet) l'ont adopté en masse et pour paraphraser IBM "FreeBSD est un meilleur Linux que Linux". FreeBSD n'a pas la prétention de concurrencer Microsoft sur le terrain mais outrepasse Windows NT en termes de stabilité et de performance côté des serveurs.

Pour ne citer qu'eux, deux des sites Internet les plus visités au monde comme Yahoo et cdrom qui n'ont trouvé que FreeBSD pour supporter de tels volumes de données en ligne. Ces derniers on mit à l'épreuve la performance et la stabilité de FreeBSD et comme nous le constatons tous ça marchent ! Ce qui est assez impressionnant est comment un vieux PC tournant sous FreeBSD peu supporter des térabits de téléchargement. Ainsi que des millions de requêtes HTTP par jour. En ayant supporté un très grand rendu 3D c'est sur des machines tournant sous FreeBSD que le film 'MATRIX' a était réaliser.

III.3 Zebra :

Zebra est un daemon prévu pour fonctionner sous Linux (noyau 2.0.37 et suivants) et BSD. qui permet d'intégrer des options de routage très avancées. Comme les protocoles de routage dynamique : RIPv1, RIPv2, OSPF et BGP. Tous les éléments de configuration restent conformes à la syntaxe de l'IOS du constructeur Cisco Systems™. Au fil des ans, cette syntaxe est devenue le standard de fait de la configuration des équipements d'interconnexion.

ZEBRA est un excellent logiciel, pour les raisons suivantes :

• c'est un logiciel libre sous une licence gratuite.

• il propose une interface de configuration interactive accessible via Telnet.

• il fonctionne selon une philosophie et un langage de configuration proche de routeurs répandus dans les entreprises (ce qui permet d'avoir accès à une bonne bibliographie).

• il supporte les principaux protocoles de routage.

• il fonctionne avec IPv6.

III.4 Création d'un routeur sous FreeBSD :

Parmi les options qu'offre ce système hors du commun, celui qui nous intéresse est le routage. La possibilité de transformer un vieux pc en routeur performant.

Bonne surprise : les besoins en performances sont faibles ; faire passer des paquets d'une interface à l'autre ne sollicite pas le disque dur, requiert peu de mémoire et peu de puissance de calcul. Bien évidemment plus le réseau sera étendu, plus les besoins en mémoire et en processeur pour stocker et parcourir les tables de routage seront importants.

Nous constaterons qu'un routeur sous FreeBSD est difficile à mettre en œuvre, mais dispose d'une grande flexibilité et fournira des fonctionnalités similaires à un routeur dédié pour un coût très inférieur.

III.4.1 Installation et configuration du logiciel porté zébra :

-Nous allons tester Zebra sur ce petit réseau :

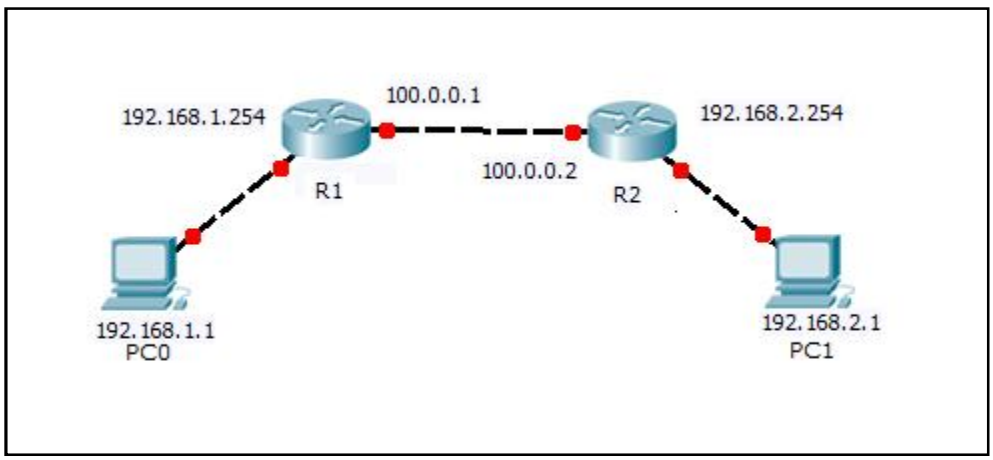


Figure 3.1 : le petit réseau à connecter avec le routeur logiciel zebra.

-Après avoir téléchargé le logiciel du serveur ftp des ports de FreeBSD, on procède à l'installation puis la configuration de celui-ci.

1- pour pouvoir utiliser Zebra il faut éditer sont fichier zebra.conf et le compiler au niveau du noyau de FreeBSD :

`%_cd_/usr/local/etc/zebra/` : accéder aux fichiers de configuration fourni avec le logiciel Zebra.
`%_cp_/usr/local/etc/zebra/zebra.conf.sample_/usr/local/etc/zebra/zebra.conf` : copier ceux-ci dans le dossier enregistré sur le noyau.

→ `cd` : cette commande nous permet d'accéder au dossier de Zebra.

→ `cp` : avec cette commande on copy le contenu d'un fichier dans un autre.

→ '_' : elle représente l'espace entre les commandes.

-afin que le noyau puisse activer le daemon Zebra il faut ajouter les lignes suivantes au fichier rc.conf :

`default_route= "NO"` : pas de passerelle par défaut.
`router_enable= "YES"` : activer le routeur logiciel.
`router="/usr/local/sbin/zebractl"`
**`router_flags="start`»: la commande zebractl start sera utiliser pour active le daemon zebra

- Avant de lancer zebra nous allons éditer son fichier de configuration pour changer ou ajouter le nom d'hôte et le mot de passé:

`% ee_/etc/zebra/zebra.conf` : accéder au fichier zebra.conf pour modifier le mot de passe et le nom d'hôte.

2-Après avoir édité le fichier de configuration, et afin que le logiciel de routage s'exécute en tâche de fond. Nous allons utiliser la commande qu'on avait préparée avant :

 # zebractl start : activer Zebra

ZebraR1#telnet localhost 2601 : lancer la configuration par Telnet

-On va s'identifier avec le mot de passe que nous avons précisé dans le fichier zebra.conf

3-on passe en suite en mode de configuration (mode privilégié appelé mode **enable** dans le logiciel) :

```
R1 > enable : accéder au mode privilégié
R1#
```

-A ce stade on pourrait taper à n'importe quel moment **?** Pour connaitre les commandes disponibles.

III.4.2 Configuration des interfaces :

Ø L'interface em0 du routeur1 :

```
R1#configure terminal
R1 (config) #interface emo
R1 (config-if) #ip address 192.168.1.254 255.255.255.0
R1 (config-if) #no shut down
R1 (config-if)#exit
```

Ø L'interface em1 du routeur1 :

```
R1 (config) #interface em1
R1 (config-if) #ip address 100.0.0.1 255.0.0.0
R1 (config-if) #no shut down
R1 (config-if)#exit
```

Ø L'interface em0 du routeur2 :

```
R2 > enable
R2# configure terminal
R2 (config) #interface em0
R2 (config-if) #ip address 192.168.2.254 255.255.255.0
R2 (config-if) #no shut down
R2 (config-if)#exit
```

Ø L'interface em1 du routeur2 :

```
R2 (config) #interface em1
R2 (config-if) #ip address 100.0.0.2 255.0.0.0
R2 (config-if) #no shut down
R2 (config-if)#exit
```

-sur R1 :

```
R2 (Zebra) # show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
I - ISIS, B - BGP, > - selected route, * - FIB route
C>* 100.0.0.0/8 is directly connected, em1
C>* 127.0.0.0/8 is directly connected, lo0
C>* 192.168.1.0/24 is directly connected, em0
```

-sur R2 :

```
R2 (Zebra) # show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
I - ISIS, B - BGP, > - selected route, * - FIB route
C>* 100.0.0.0/8 is directly connected, em1
C>* 127.0.0.0/8 is directly connected, lo0
C>* 192.168.2.0/24 is directly connected, em0
```

-Afin de parvenir à connecter les deux stations il faudrait ajouter deux routes dites statiques. Nous vérifierons les résultats au fur et à mesure :

-sur R1 :

```
R1#configure terminal
R1 (config) # ip route 192.168.2.0/24 100.0.0.1
R1 (config) #end
R1# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF, I - ISIS,
       B - BGP, > - selected route, * - FIB route
C>* 100.0.0.0/8 is directly connected, em1
C>* 127.0.0.0/8 is directly connected, lo0
S>* 192.168.2.0/24 [1/0] via 100.0.0.1, em1
C>* 192.168.1.0/24 is directly connected, em0
```

-sur R2 :

```
R2#configure terminal
R2 (config) # ip route 192.168.1.0/24 100.0.0.2
R2 (config) #end
R2# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF, I - ISIS,
       B - BGP, > - selected route,
            * - FIB route
C>* 100.0.0.0/8 is directly connected, em1
C>* 127.0.0.0/8 is directly connected, lo0
S>* 192.168.1.0/24 [1/0] via 100.0.0.2, em1
C>* 192.168.2.0/24 is directly connected, em0
```

- Teste de la configuration :

A ce stade nous pouvant vérifier nos résultats par un **ping** soit depuis les routeurs ou les stations:

```
R1 # ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1) from 192.168.1.1 : 56(84) bytes of data.
64 byte from 192.168.2.1 : icmp_seq=0 ttl=64 time=0,219ms
64 byte from 192.168.2.1 : icmp_seq=1 ttl=64 time=2,684ms
64 byte from 192.168.2.1 : icmp_seq=0 ttl=64 time=0,597ms
(Ctrl+C)
--- 192.168.2.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
```

- Le contenu du fichier de configurations de R1 :

-R1 :

```
R1# show running-config
Current configuration:
!
hostname R1
password zebra
!
interface lo0
!
interface em0
!
```

```
interface em1
!
ip route 192.168.2.0/24 100.0.0.2
!
line vty
!
        end
```

Quand à R2 c'est la même chose, seule différence :

```
ip route 192.168.1.0/24 100.0.0.1
```

A partir du moment où nous arrivons à installer, configurer et utiliser Zebra pour de petites applications ; la prise en main devient de plus en plus facile. Puisque tous ce qui à était utiliser comme commande au chapitre précédant peut être utilisé avec Zebra.

Prenant comme exemple le routage RIP, il est pris on charge par la daemon **ripd.** Pour activer se dernier il faut tapez la commande **ripd –d** après l'exécution de Zebra.

Discussion :

Le but de ce troisième chapitre était de créer un environnement de configuration à la fois réel et permanant. Nous avion vérifié que nous pouvant transformer une machine peu performante en routeur intégrant des options de routage avancées.

Notre route était semait d'embuches, en premier lieu nous avons eu du mal à choisir la version de FreeBSD allant avec notre matériel, d'autant plus que le téléchargement dure des heures. Ce premier problème à était vite résolu à la découverte de l'application libre VirtualBox. Elle nous permet d'essayer des versions sans avoir besoin de cd bootable. Bienheureusement car il n'existe pas de fournisseur de cd bootable FreeBSD en Algérie.

En suite, le fait que FreeBSD n'est utilisé que par des spécialistes qui se connaissent bien en système Unix toute la documentation que n'avons trouvée à son sujet était incomplète. Nous avons réussi à assimiler certain concepts par nous même malgré que ça nous ait pris beaucoup de temps ; mais pour les autres concepts tout aussi capitaux nous avons utilisé les foires aux questions et les news pour trouver des réponses aux indénombrables questions qu'on se pauser à chaque étape. L'aide de ces dernier à était bénéfique même si la plupart du temps les solutions données sont pas celles qui peuvent nous aidés.

Le plus dur était la configuration du routeur logiciel Zebra puisque les étapes ne sont pas du tout évident. Les méthodes d'édition des fichiers de configuration, leur compilation, puis leur exécution sont assez difficiles à maitre en œuvre. Ce qui nous a le plus aider à ce niveau c'est le fait que nous somme initier à certains langages de programmation.

Les applications contenues dans notre projet ont été testé et ne présente aucun disfonctionnement. Sachant que le système exploitation FreeBSD assure un très grand nombre d'options réseau avancé comme la création de serveur, pare feu, etc. notre travail présente un recueil de notion de base pour ceux qui s'intéresseront à ce système d'exploitation et aux différents services qu'il offre.

Conclusion

Conclusion :

Ce projet, mené pendant plus de cinq mois, nous a été bénéfique et les apports personnels ont été multiples. En effet, il nous a non seulement permis de mètre en œuvre les connaissances que nous avons acquises au cours de notre cursus scolaire, mais aussi de développer un esprit d'équipe adéquat à l'avancement de notre travail.

Notre thème de départ été uniquement la configuration des routeurs et commutateurs **Cisco** présent sur le site du MSC de Tizi-Ouzou, la problématique est qu'on ne dispose pas de l'accès au matériel il fallait donc trouver une autre solution pour la réalisation et la configuration, packet tracer pouvait réaliser ces fonctions .une deuxième solution se présente à nous, un système d'exploitation hors du commun **freeBSD,** grâce à cette solution on pouvait rendre une vielle machine usé un routeur et la meilleurs est que la configuration reste la même syntaxe que **Cisco** et c'est gratuit.

Donc au premier chapitre nous avons introduit quelques notions fondamentales sur les réseaux et la manière dont se fait la communication entre les machines

Le deuxième chapitre été consacré à la configuration du routeur et commutateur **Cisco**, une configuration bien adapté au réseau reliant le MSC Mobilis de Tizi-Ouzou à celui d'Alger, Bejaia, Bouira et Moustapha.

Le troisième chapitre, une présentation du système d'exploitation **freeBSD**, ses avantage et bien sur une petite configuration pour illustré la configuration avec le deamont **zabra**.

Ainsi il ressort de cette étude une comparaison entre les deux systèmes sensiblement équivalents en thermes de syntaxe de configuration. Cependant, l'avantage majeur des routeurs **Cisco** reste la configuration via une interface graphique, ce qui le rend plus accessible à un publique dont les connaissances en système Unix et l'interface DOS sont limitées. En plus bien sur du nombre de ports que peut offrir un routeur **Cisco** par rapport à une vielle machine.

Bibliographie:

- Sécurité Firewall Linux/Cisco fait par SAMBA Mamadou et DEME Mohamed Université Montpellier II 2008/2009.

- Dreyfus, Emmanuel. Cahiers de l'Admin: BSD 2nde Ed. (en Français), Eyrolles, 2004. ISBN 2-212-11463-X.

- Computer Systems Research Group, UC Berkeley. *4.4BSD System Manager's Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-080-5.

- Hunt, Craig. *TCP/IP Network Administration*, 2nd Ed. O'Reilly & Associates, Inc., 1997. ISBN 1-56592-322-7.

- Absolute BSD: The Ultimate Guide to FreeBSD, publié par No Starch Press, 2002. ISBN: 1886411743.

- The Complete FreeBSD, publié par O'Reilly, 2003. ISBN: 0596005164.

- The FreeBSD Corporate Networker's Guide, publié par Addison-Wesley, 2000. ISBN: 0201704811.

http ://www.netlab.tkk.fi/opetus /s383133 /cisco- how-to.pdf

http//www.laissus.fr/pub/cours/cours.ps

https://repo.zenk-security.com/.../cours%20d.introduction%20tcp-ip.pdf

http//doc.hackbbs.org/réseaux/Cisco/ccna4.pdf

http//www .simon web.be/afiche-commandes-routeurs-Cisco-ccna.html

http//www.kittler.fr/docs/Ciscoconfig.pdf

http//cosy univ-reims.fr/

http//www.Cisco.goffinet.org

http//www. Cisco.com

http://www.reseaumaroc.com/files/Synthese%20sur%20le%20routage%2014.pdf

.http://www.cisco.com/univercd/cc/td/doc/cisintwk/idg4/nd2003.htm

http//www.Commentçamarche.net

http//www.freebsd.org

http://lists.freebsd.org/mailman/listinfo/freebsd-questions

http://lists.freebsd.org/mailman/listinfo/freebsd-test

news:fr.comp.os.bsd

news:comp.unix.bsd.freebsd.announce

news:comp.unix.questions

http://www.fr.freebsd.org/

http://www1.fr.freebsd.org/

http://8help.osu.edu/wks/unix_course/unix.html

http://www.docbook.org

http://www.debian.org

http://www.linux-france.org/

http://www.linuxmag-france.org/

http://www.zebra.org/

Annexe A : fichier de configuration

```
tizi#sh run
Building configuration...

Current configuration : 5427 bytes
!
version 12.4
service password-encryption
!
hostname tizi
!
!
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1
!
!
!
!
username ALGER password 7 0835435A06
username BOUIRA password 7 0835435A06
username MOUSTAFA password 7 0835435A06
username tizi password 7 0835435A06
!
ip ssh version 1
!
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 ip access-group 120 in
 ip access-group 108 out
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 duplex auto
```

```
 speed auto
 shutdown
!
interface Serial0/0/0
 ip address 172.16.2.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/0/1
 description LIAISON VERS MOUSTAFA
 ip address 172.16.12.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/1/0
 description LIAISON VERS BOUIRA
 ip address 172.16.15.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/1/1
 description LIAISON VERS BEJAIA
```

```
 ip address 172.16.13.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface FastEthernet1/0
 switchport mode access

tizi#sh run
Building configuration...

Current configuration : 5427 bytes
!
version 12.4
service password-encryption
!
hostname tizi
!
!
enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1
!
!
!
!
username ALGER password 7 0835435A06
username BOUIRA password 7 0835435A06
username MOUSTAFA password 7 0835435A06
username tizi password 7 0835435A06
!
ip ssh version 1
!
```

```
!
interface FastEthernet0/0
 ip address 172.16.1.1 255.255.255.0
 ip access-group 120 in
 ip access-group 108 out
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
!
interface Serial0/0/0
 ip address 172.16.2.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/0/1
 description LIAISON VERS MOUSTAFA
 ip address 172.16.12.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/1/0
```

```
 description LIAISON VERS BOUIRA
 ip address 172.16.15.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface Serial0/1/1
 description LIAISON VERS BEJAIA
 ip address 172.16.13.1 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 ppp pap sent-username TIZI password 0 TOTO
 ip access-group 130 in
 ip access-group 115 out
 clock rate 2000000
!
interface FastEthernet1/0
 switchport mode access
 shutdown
!
interface FastEthernet1/1
 switchport mode access
 shutdown
!
interface FastEthernet1/2
 switchport mode access
 shutdown
!
interface FastEthernet1/3
 switchport mode access
 shutdown
```

```
!
interface FastEthernet1/4
 switchport mode access
 shutdown
!
interface FastEthernet1/5
 switchport mode access
 shutdown
!
interface FastEthernet1/6
 switchport mode access
 shutdown
!
interface FastEthernet1/7
 switchport mode access
 shutdown
!
interface FastEthernet1/8
 switchport mode access
 shutdown
!
interface FastEthernet1/9
 switchport mode access
 shutdown
!
interface FastEthernet1/10
 switchport mode access
 shutdown
!
interface FastEthernet1/11
 switchport mode access
 shutdown
!
interface FastEthernet1/12
```

```
 switchport mode access
 shutdown
!
interface FastEthernet1/13
 switchport mode access
 shutdown
!
interface FastEthernet1/14
 switchport mode access
 shutdown
!
interface FastEthernet1/15
 switchport mode access
 shutdown
!
interface Vlan1
 no ip address
 shutdown
!
ip classless
ip route 172.16.5.0 255.255.255.0 172.16.15.2
ip route 172.16.4.0 255.255.255.0 172.16.13.2
ip route 172.16.6.0 255.255.255.0 172.16.2.2
ip route 172.16.6.0 255.255.255.0 172.16.12.2
ip route 172.16.3.0 255.255.255.0 172.16.2.2
!
!
ip access-list standard no-access
 deny host 172.16.1.251
 deny host 172.16.1.252
 deny host 172.16.1.253
 permit any
access-list 100 permit ip host 172.16.1.254 172.16.3.0 0.0.0.255
access-list 100 permit ip host 172.16.1.254 172.16.4.0 0.0.0.255
```

```
access-list 100 permit ip host 172.16.1.254 172.16.5.0 0.0.0.255
access-list 100 permit ip host 172.16.1.254 172.16.6.0 0.0.0.255
access-list 2 permit host 172.16.3.0
access-list 2 deny any
access-list 3 permit host 172.16.5.0
access-list 3 deny any
access-list 4 permit host 172.16.6.0
access-list 4 deny any
access-list 5 permit host 172.16.4.0
access-list 5 deny any
access-list 10 permit host 172.16.4.254
access-list 10 permit host 172.16.5.254
access-list 10 permit host 172.16.6.254
access-list 10 permit host 172.16.3.254
access-list 10 deny any
ip access-list extended dd
 permit tcp host 172.16.1.254 any eq www
 permit tcp host 172.16.1.254 any eq 443
access-list 101 deny udp any any eq 138
access-list 101 deny udp any any eq 137
access-list 101 deny udp any any eq 135
access-list 101 deny udp any any eq bootps
access-list 101 deny udp any any eq tftp
access-list 101 deny tcp any any eq 111
access-list 101 deny tcp any any eq 135
access-list 101 deny tcp any any eq 95
access-list 101 deny tcp any any eq 87
access-list 101 deny tcp any any eq 137
access-list 101 deny udp any any eq 514
access-list 101 deny udp any any eq snmp
access-list 12 deny host 127.0.0.0
access-list 103 permit icmp host 172.16.3.254 172.16.1.0 0.0.0.255
access-list 104 permit icmp host 172.16.4.254 172.16.1.0 0.0.0.255
access-list 104 permit icmp host 172.16.5.254 172.16.1.0 0.0.0.255
```

```
access-list 105 permit icmp host 172.16.5.254 172.16.1.0 0.0.0.255
access-list 115 permit icmp host 172.16.1.254 172.16.3.0 0.0.0.255
access-list 115 permit icmp host 172.16.1.254 172.16.4.0 0.0.0.255
access-list 115 permit icmp host 172.16.1.254 172.16.5.0 0.0.0.255
access-list 115 permit icmp host 172.16.1.254 172.16.6.0 0.0.0.255
access-list 120 permit icmp 172.16.1.0 0.0.0.255 any
access-list 130 deny ip 0.0.0.0 0.255.255.255 any
access-list 130 deny ip 10.0.0.0 0.255.255.255 any
access-list 130 deny ip 169.254.0.0 0.0.255.255 any
access-list 130 deny ip 192.168.0.0 0.0.0.255 any
access-list 130 deny ip 240.0.0.0 7.255.255.255 any
access-list 130 deny ip 248.0.0.0 7.255.255.255 any
access-list 130 deny ip host 255.255.255.255 any
access-list 130 permit ip any any
!
!
!
banner motd ^C acces avec autorisation seulement ^C
line con 0
 password 7 0822455D0A16
 login
line vty 0 4
 access-class 10 in
 password 7 0812696D3B3C31
 login
line vty 5 15
 access-class 10 in
 password 7 0812696D3B3C31
 login
!
!
end
```

ANNEXE B: Virtual box :

Comme le choix de la version à utiliser de FreeBSD dépend du matériel utiliser et comme très sincèrement on ne peut pas éviter la confusion car il y a trop de version et on ne sait pas laquelle choisir on a trouvé une solution : simuler l'installation, la configuration de la machine en tant que routeur. Cela nous fera gagner un temps précieux et nous évitera beaucoup d'ennui et nous permettra d'apprendre à manipuler sous ce nouveau système d'exploitation sans crainte d'endommager le matériel ni de perdre des informations et ainsi tout refaire, etc. est ce possible ? C'est ce que nous allons voir avec l'application VirtualBox.

Ø Qu'est ce que VirtualBox ?

VirtualBox est une application de virtualisation multiplateforme. C'est-à-dire qu'on peut l'installer sur un ordinateur Intel ou des ordinateurs à base d'AMD, qu'ils soient sous Windows, Mac, Linux ou les systèmes d'exploitation Solaris. Simple et très performante, elle nous permet de simuler plusieurs systèmes d'exploitation (à l'intérieur de plusieurs machines virtuelles) en même temps. Ainsi, par exemple, on peut exécuter Windows et Linux sur un Mac, lancez Windows Server 2008 sur un serveur Linux, Linux sur un PC Windows, et ainsi de suite, tous les côtés des applications existantes. On peut installer et exécuter autant de machines virtuelles qu'on le souhaite. Les seules limites pratiques sont l'espace disque et mémoire.

Ø Quel est l'intérêt de la virtualisation ?

La virtualisation consiste à simuler la présence de support physique réel : processeurs, carte réseaux, disque dur, ports, etc.; dans un environnement virtuel. Elle utilise l'espace libre pour créer le support virtuel sur le quel exécuter les systèmes d'exploitations. Parmi les avantages qu'offre cette option nous citerons :

- **Exécution de plusieurs systèmes d'exploitation simultanément :**

  VirtualBox permet d'exécuter plus d'un système d'exploitation à la fois. De cette façon, on peut exécuter un logiciel écrit pour un système d'exploitation sur un autre (par exemple exécuter des logiciels Windows ou Linux sur un Mac) sans avoir à redémarrer l'ordinateur pour l'utiliser. Si on peut configurer ce genre de matériel

"virtuel", on peut installer un système d'exploitation anciens tels que DOS, même si ce système d'exploitation n'est plus pris en charge par le matériel de l'ordinateur réel.

- **Installations facile de logiciels :**

Les éditeurs de logiciels peuvent utiliser des machines virtuelles pour envoyer des configurations de logiciel entières. Par exemple, l'installation complète d'un serveur de messagerie sur une machine réelle peut être une tâche fastidieuse. Avec VirtualBox, une telle configuration complexe peut être emballée dans une machine virtuelle qu'on pourrait importer ou exporter.

- **Essais et reprise après sinistre :**

Une fois installé, une machine virtuelle et ses disques durs virtuels peuvent être considérés comme un "conteneur" qui peut être arbitrairement congelés, réveillé, copié, sauvegardé, et transporté entre les hôtes.

En plus de cela, avec l'utilisation d'une caractéristique de VirtualBox appelé **«instantanés»,** on peut enregistrer un état particulier d'une machine virtuelle et revenir à cet état, si nécessaire. De cette façon, on peut apprendre librement avec un environnement informatique. Si quelque chose se passe mal (par exemple après l'installation d'un mauvais logiciel ou l'infection de l'hôte avec un virus), on peut facilement revenir à un **instantané** (on utilise ce terme pout faire référence à la touche instantané de la barre d'outil de VirtualBox) précédent et d'éviter ainsi de refaire des reconfigurations entières.

- **consolidation de l'infrastructure :**

La virtualisation permet de réduire considérablement les coûts de matériel et de l'électricité. La plupart du temps, les ordinateurs actuels utilisent seulement une fraction de leur potentielle en matière de puissance. Un grand nombre de ressources matérielles ainsi que l'électricité est ainsi gaspillée. Donc, au lieu de faire marcher beaucoup d'ordinateurs physiques qui ne sont que partiellement utilisés, on peut créer plusieurs machines virtuelles sur un hôte puissant et équilibrer les charges entre elles. Grace à ça, on profitera d'une économie d'énergie et de l'optimisation quant à l'utilisation du matériel.

- **Contrôle des machines virtuelles à distance** :

VirtualBox a une conception modulaire très bien définis avec des interfaces de programmation interne et une séparation nette entre le client et le code serveur. Il est donc facile de le contrôler à partir de plusieurs interfaces à la fois: par exemple, on peut démarrer une machine virtuelle en cliquant simplement sur un bouton dans l'interface utilisateur graphique VirtualBox, puis de contrôler la machine soit par la ligne de commande ou même à distance.

Ø Quel est le matériel pris en charge par VirtualBox ?

Prise en charge matérielle des Grands, entre autres, VirtualBox supporte:

- VirtualBox peut présenter jusqu'à 32 processeurs virtuels pour chaque machine virtuelle, quel que soit le nombre de cœurs de processeurs physiquement présent sur l'hôte. Ainsi qu'elle peu réservé jusqu'à 8 cartes réseaux par machine virtuel.
- Elle met en œuvre un contrôleur USB virtuel et permet de connecter des périphériques USB arbitraire aux machines virtuelles sans avoir à installer les pilotes de périphériques spécifiques sur l'hôte.
- VirtualBox virtualise une vaste gamme de périphériques virtuels, parmi eux de nombreux appareils qui sont généralement fournies par les plates-formes de virtualisation. Cela comprend les IDE, SCSI et les contrôleurs de disque dur SATA, plusieurs cartes réseau virtuelles et des cartes son, série virtuels et les ports parallèles et une entrée / sortie Advanced Programmable Interrupt Controller (I / O APIC), qui se trouve dans de nombreux systèmes PC moderne.
- Les machines virtuelles VirtualBox support beaucoup plus de résolution d'écran qu'un écran physique, leur permettant d'être réparties sur un grand nombre d'écrans connectés au système hôte.

Ø Quel sont les systèmes hôtes sur les quels on peut installer VirtualBox ?

Il existe des versions de VirtualBox pour **Windows** (Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008 et Windows 7), **Mac OS X** (10.5 Leopard, 10.6 Snow Leopard), **Linux** (Ubuntu, Debian GNU / Linux Oracle Enterprise Linux, Linux Redhat Enterprise Gentoo Linux…) et **Solaris hôtes**. Bien évidemment on désigne par hôte l'ordinateur sur le quelles les machines virtuelle sont installées.

Le système d'exploitation qu'on a utilisé est **WINDOWS7** donc on avait besoin de la version dédié à cet OS qui est ***VirtualBox-4.0.8-71778-Win***.

Ø Quel sont les systèmes invités qu'on peut simuler avec VirtualBox ?

VirtualBox est conçu pour fournir un environnement de virtualisation génériques pour les systèmes, il peut exécuter plusieurs systèmes d'exploitation de toute nature, même ceux qui ne figurent pas ici. Toutefois, on a prouvé l'optimisation de VirtualBox pour les systèmes invités suivants:

- **Windows NT 4.0**
- **Windows 2000 / XP / Server 2003 / Vista / Server 2008 / Windows 7 Windows 2000 / XP / Server 2003 / Vista / Server 2008 / Windows 7 :** Toutes les versions, éditions et service packs sont entièrement pris en charge.
- **DOS / Windows 3.x / 95 / 98 / ME DOS / Windows 3.x / 95 / 98 / ME :**Des essais limités ont été effectués.
- **Linux 2.4 :** Prise en charge limitée.
- **Linux 2.6** : Toutes les versions / éditions sont entièrement pris en charge (32 bits et 64 bits).
- **Solaris 10, OpenSolaris :** Entièrement pris en charge (32 bits et 64 bits).
- **FreeBSD et OpenBSD** : virtualisation matérielle nécessite d'être activé. (Pour notre plus grand bonheur il est dans la liste).
- **Mac OS X Server** : VirtualBox 3.2 a ajouté un support expérimental pour les clients Mac OS X Server, mais cela est plein de restrictions.

Ø Installation de VirtualBox :

Apres avoir eu une brève idée sur l'application VirtualBox nous allons procéder à l'installation de la version dédié à WINDOWS7 (figure B.1). Rien de plus simple :

-télécharger le fichier .exe sur le site de VirtualBox:

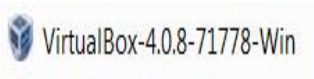


Figure B.1 : la version de VirtualBox dédié à Windows 7

-exécuter le fichier et suivre les étapes comme pour un logiciel standard.

Ø Création d'une machine virtuelle FreeBSD sur VirtualBox:

Apres l'installation on ouvre l'icône sur le bureau qui correspond à l'application (figure B.2) :

Figure B.2 :l'icône de VirtualBox

Grace à la première boite de dialogue (figureB.3) on peut débuter la création d'une MV (machine virtuelle) :

Figure B.3 : la page d'accueille de VirtualBox

ü pour créer une première MV on clic sur l'icône **créer** puis une assistante de création (figure B.4) nous permet de créer notre machine virtuelle aisément:

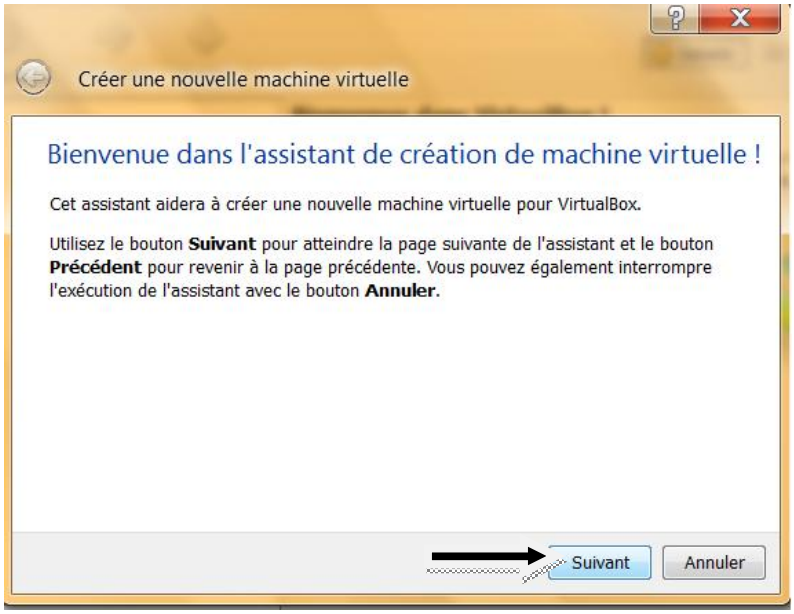


Figure B.4 : l'assistante de création.

ü après avoir appuyé sur **suivant,** la boite de dialogue (figureB.5) nous demande de choisir :

r un nom pour la machine virtuelle : **FreeBSD**

r le système d'exploitation qu'on va installer sur la machine virtuelle et sa version.

Dans notre cas et juste après avoir choisi le nom de la MV le reste se fait automatiquement :

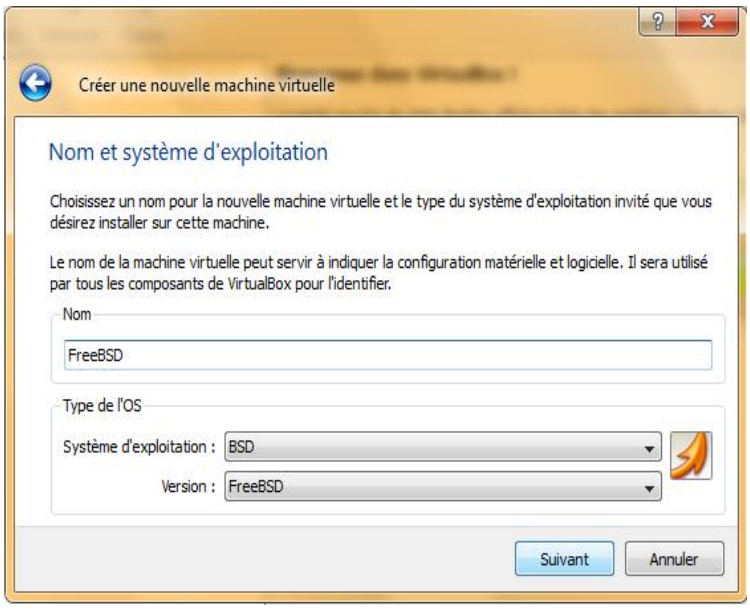


Figure B.5 : ajout des caractéristiques de l'OS.

ü On clic sur **suivant.** Une boite de dialogue (figure B.6) apparait pour demander la taille de la RAM alloué à la machine virtuelle. On remarque qu'un minimum de 128 MO est nécessaire :

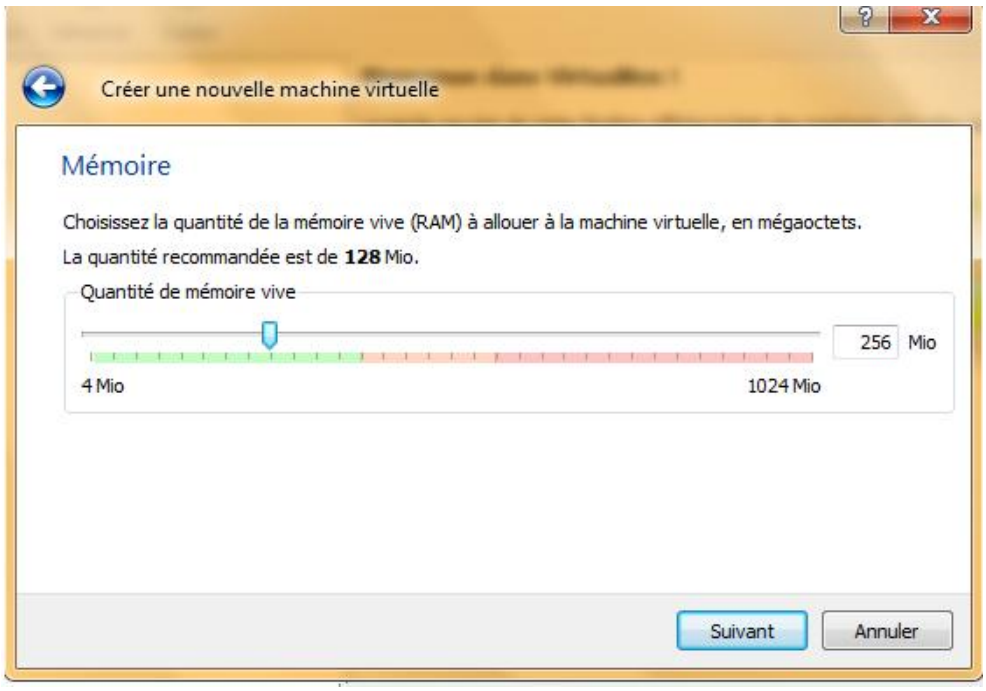


Figure B.6 : précision de la taille de la RAM.

ü Il est temps maintenant de créer un disque dur virtuel (figure B.7) utile à l'installation du système :

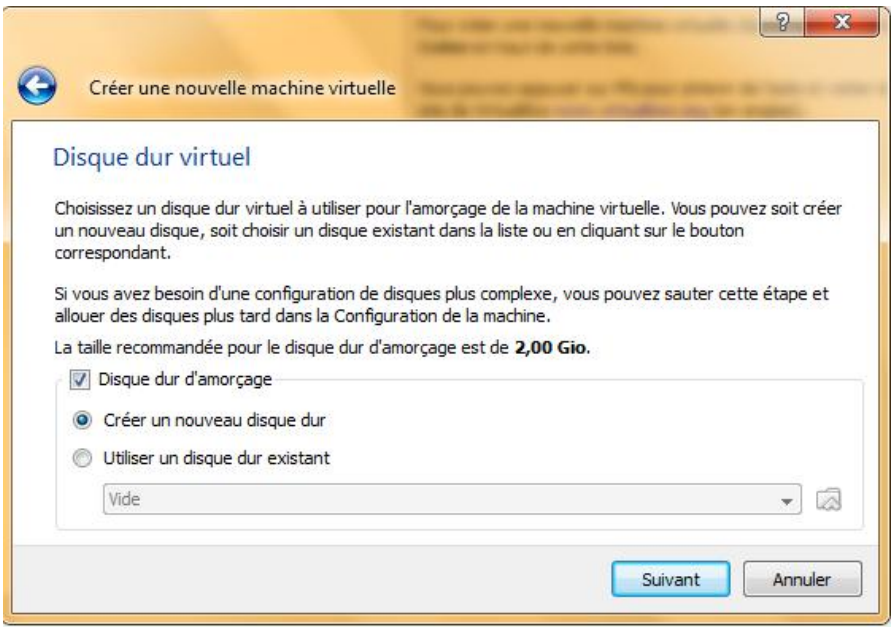


Figure B.7 : création d'un disque dur virtuel

On peut choisir à tout moment d'annuler la création en appuyant sur **annuler,** de retourner à l'étape précédant avec **précédant** (fig. B.8)**.** Etant sur de la configuration on appui sur suivant :

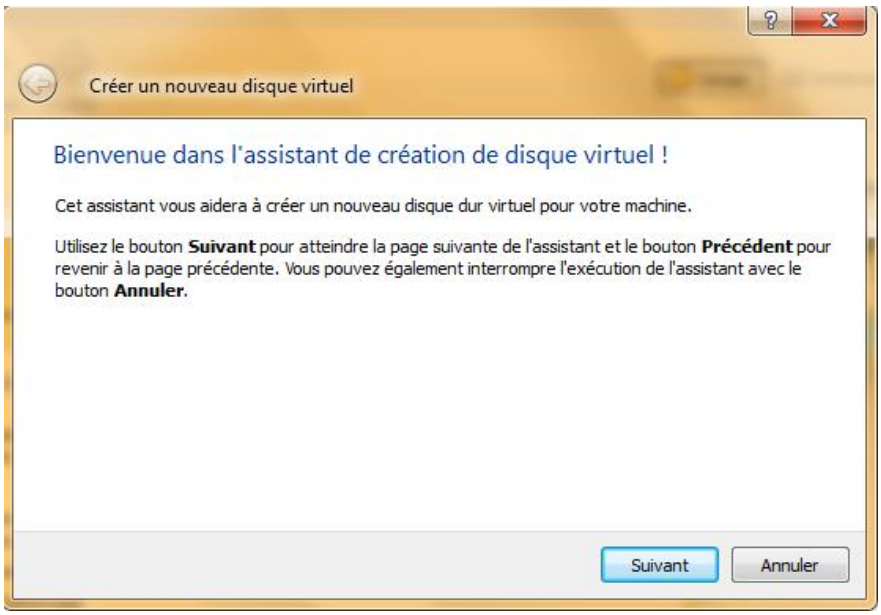


Figure B.8 : confirmation de la création du disque.

On nous propose à cette étape de créer un disque virtuel de taille variable ou fixe (figure B.9). On préfère de loin un disque de taille variable pour que le système lui-même se charge de la taille nécessaire au différent type de fichier. Ainsi il n'occupera pas un espace disque sans l'utiliser :

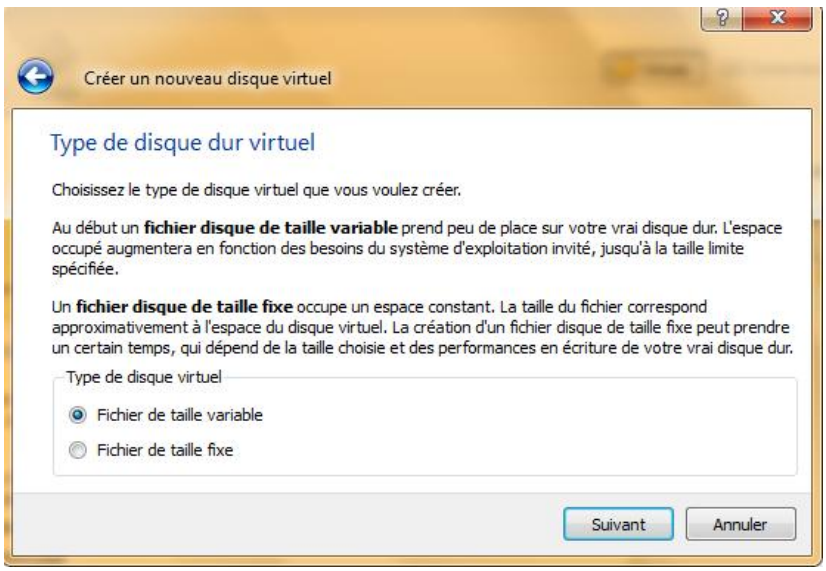


Figure B.9 : choix du type de disque virtuel.

On choisira néanmoins la taille maximale du disque dur virtuel (figure B.10). Il ne faux pas qu'elle dépasse la taille de l'espace libre disponible dans la partition ou le système sera installer. Ça n'est par peur de perdre des données. C'est juste que la machine s'arrêtera si elle ne trouve pas l'espace qui lui était allouée :

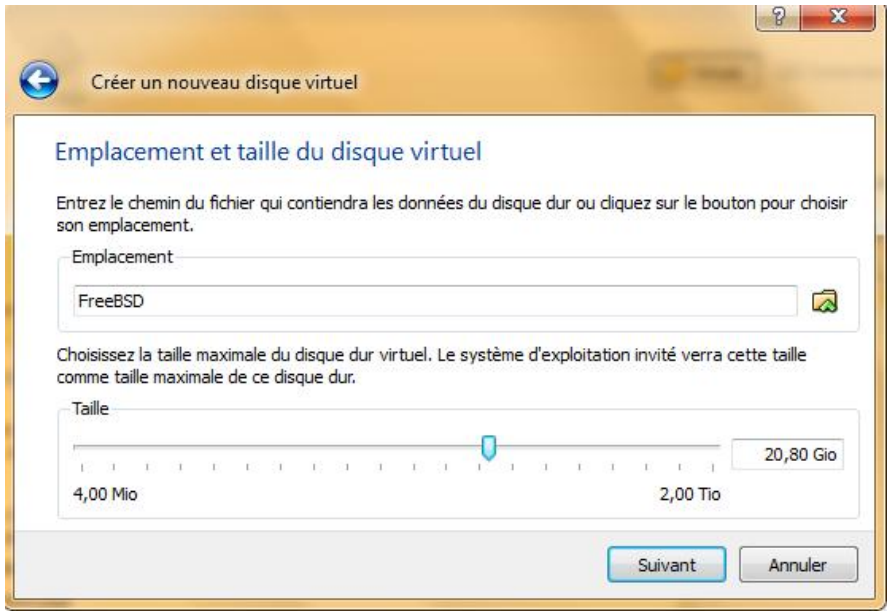


Figure B.10 : le choix de la taille max du disque dur virtuel.

Un récapitulatif pour vérifier qu'il n'y a pas d'erreur et qu'on a rien oublié concernant le disque virtuel (figure B.11):

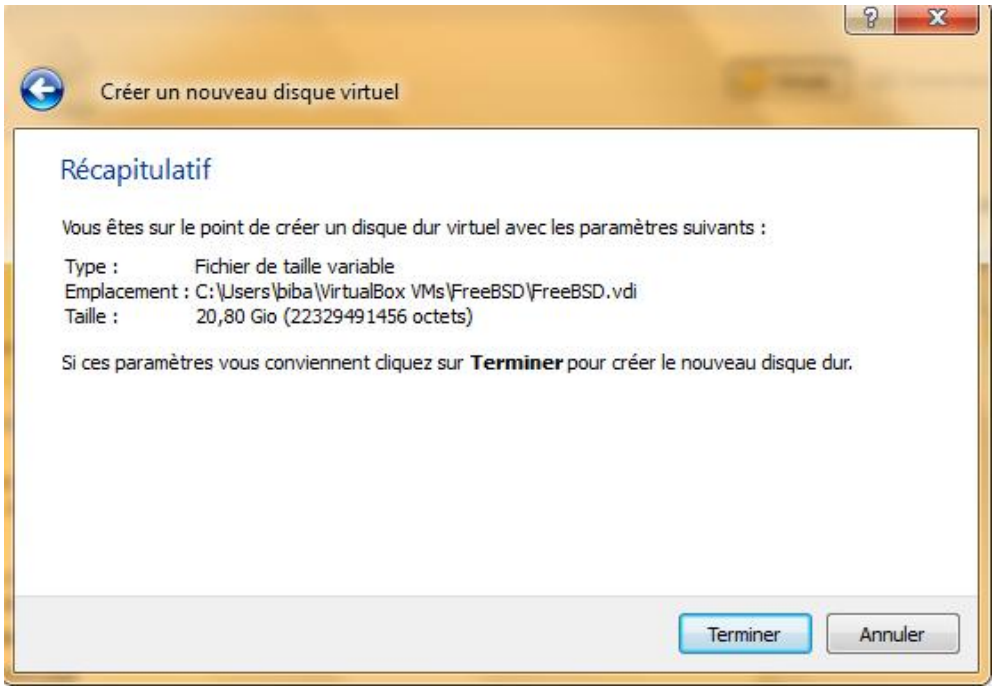


Figure B.11 : récapitulatif sur les propriétés du disque dur virtuel.

Un autre récapitulatif au sujet de la machine virtuelle (figure B.12). On nous informe que nous pouvant changer les paramètres de la MV avec la fenêtre **configuration**:

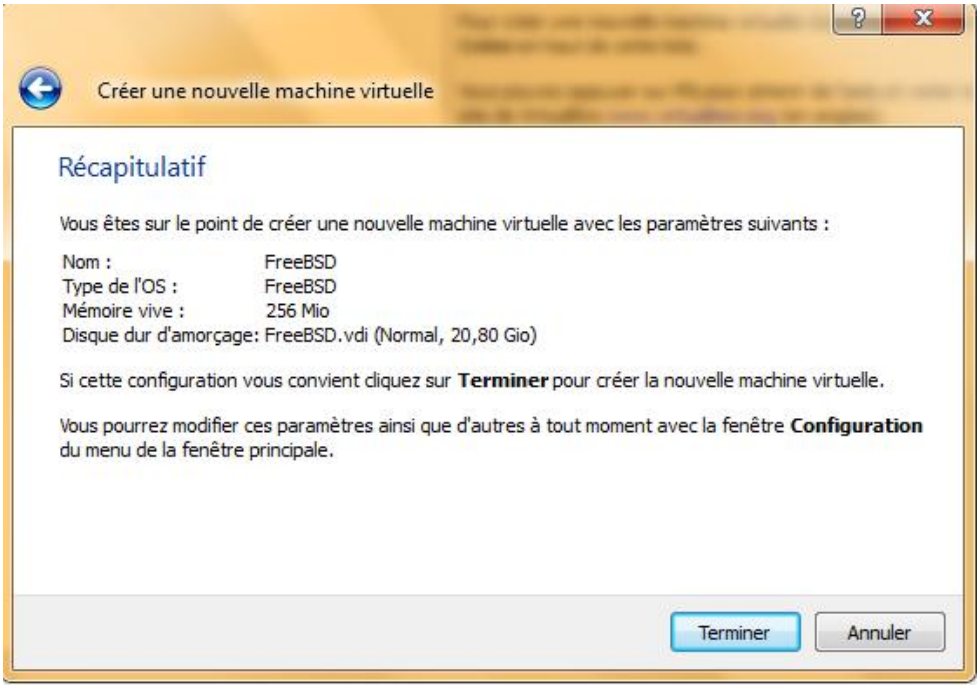


Figure B.12 : récapitulatif sur les propriétés de la machine virtuelle.

La machine virtuelle est maintenant prête à l'emploi. Il suffira d'appuyer sur l'icone démarrer (figure B.13) :

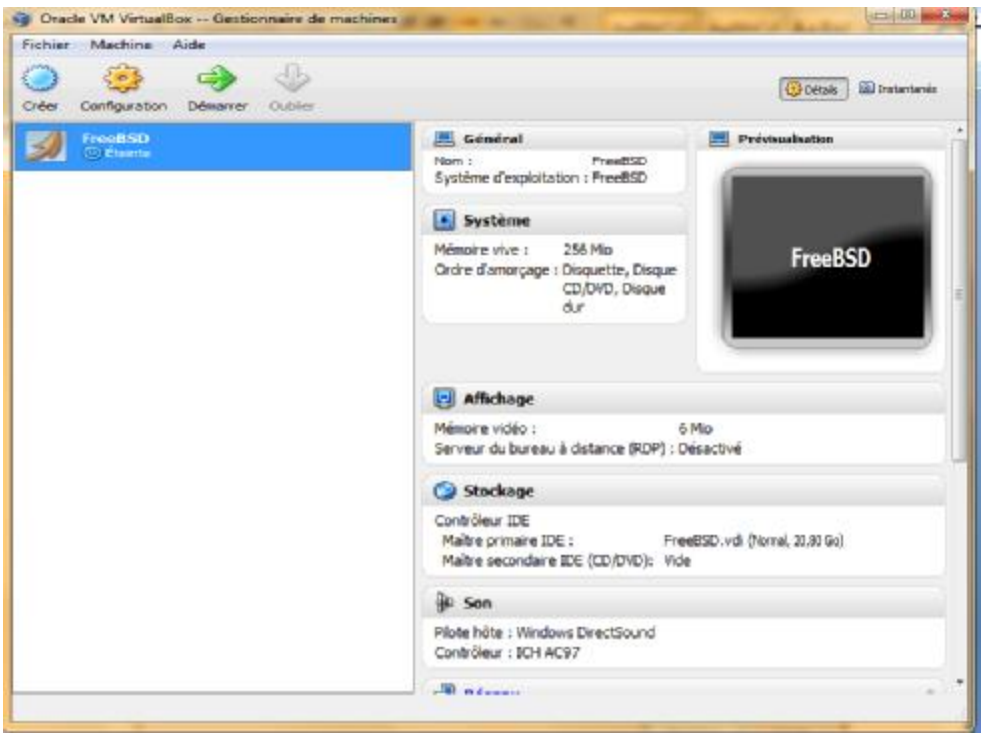


Figure B.13 : la machine virtuelle FreeBSD est prête à l'emploi.

L'assistante d'installation nous parle un peu de la façon avec laquelle le clavier et la souris seront partagé entre les deux systèmes (figure B.14). C'est simple, à chaque fois que la machine sera activée elle capturera le clavier. Pour libérer le clavier quand elle est en marche il suffit d'appuyer sur la touche **Ctrl** droite du clavier.

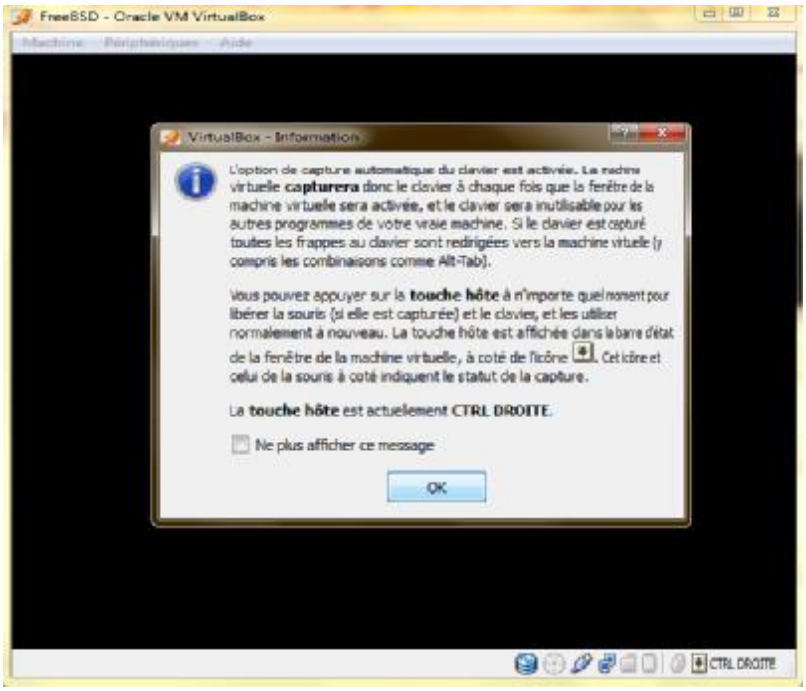


Figure B.14: le partage de la souris et du clavier entre la machines hôte et invité.

L'assistante de lancement nous aidera à installer le système d'exploitation (figure B.15) :

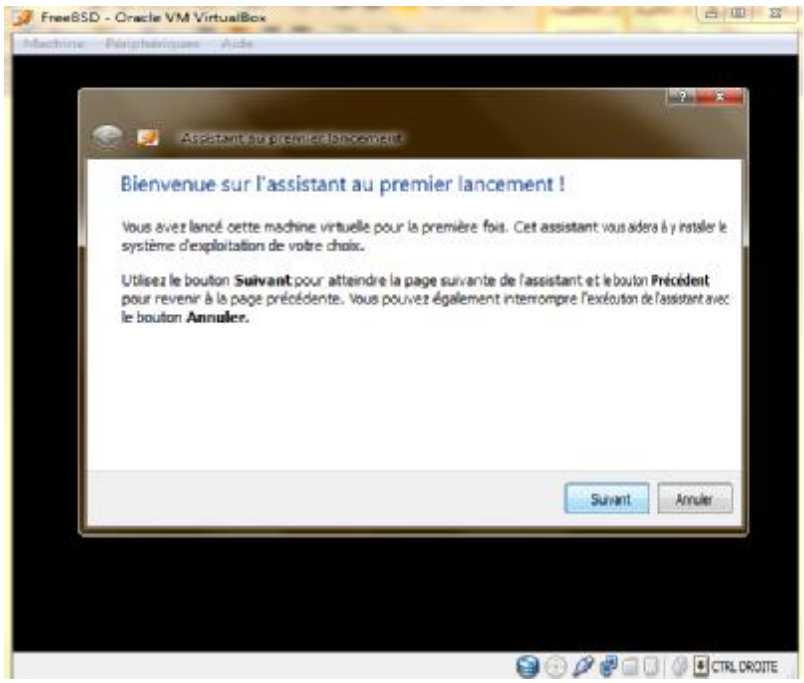


Figure B.15 : l'assistante au premier lancement.

On choisi le lieu ou se trouve le système d'exploitation donc L'image ISO (figure B.16) :

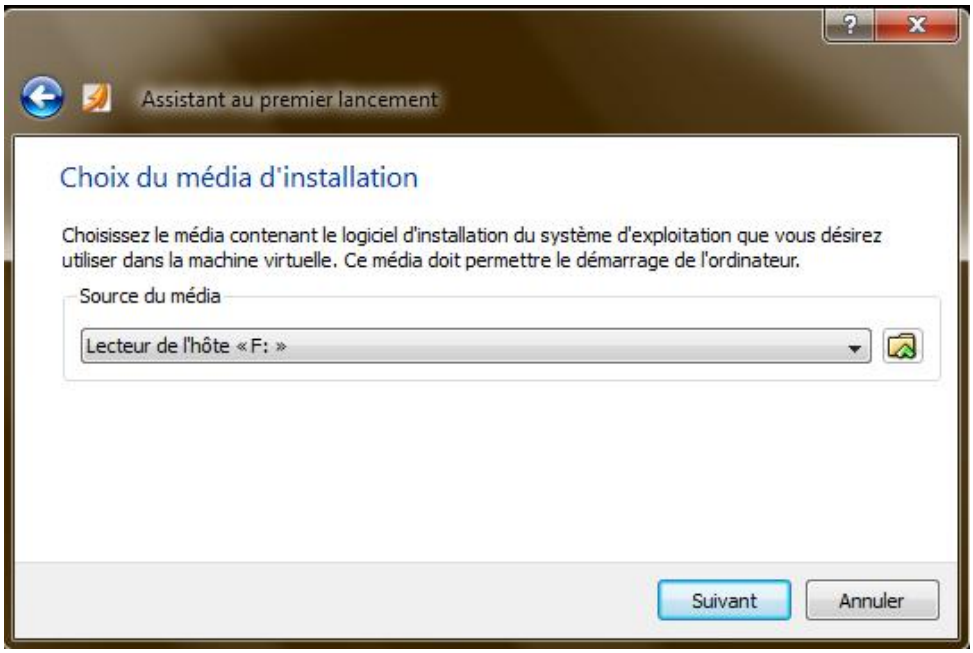


Figure B.16 :

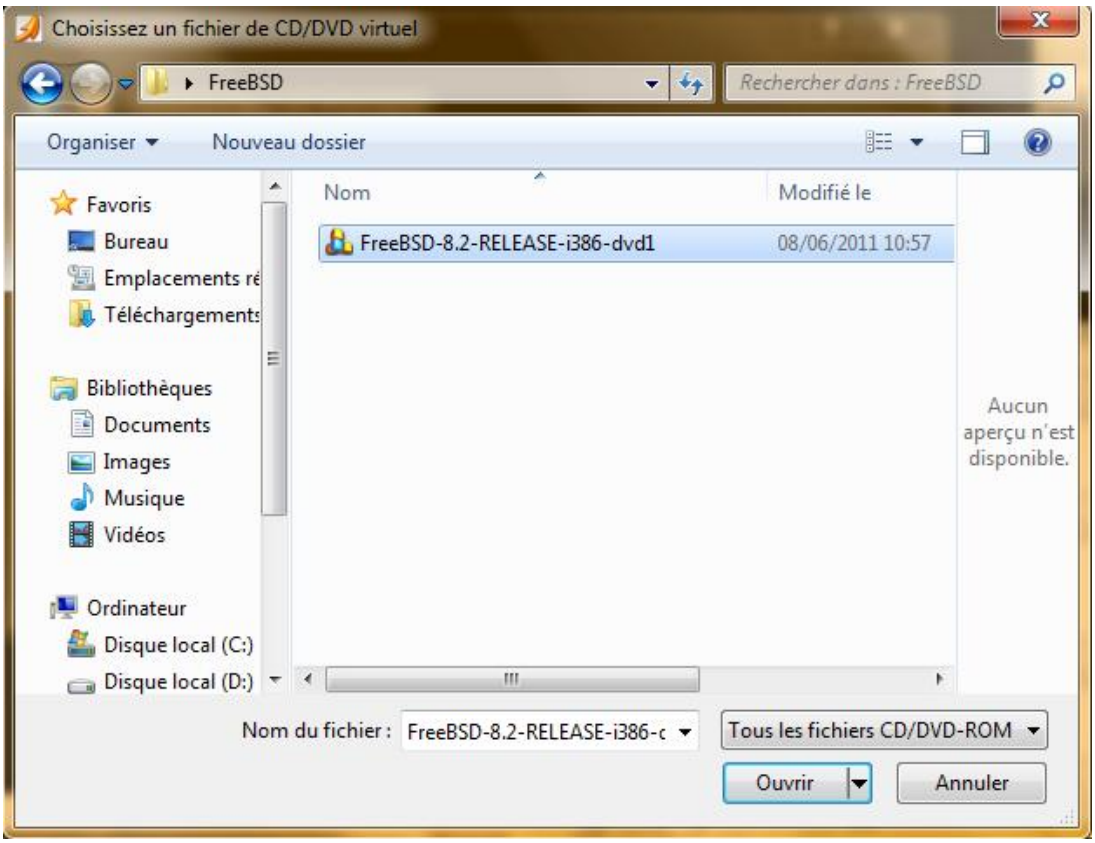


Figure B.16 et **Figure B.17** : choix du media d'installation.

Confirmation du média d'installation (figure B.18):

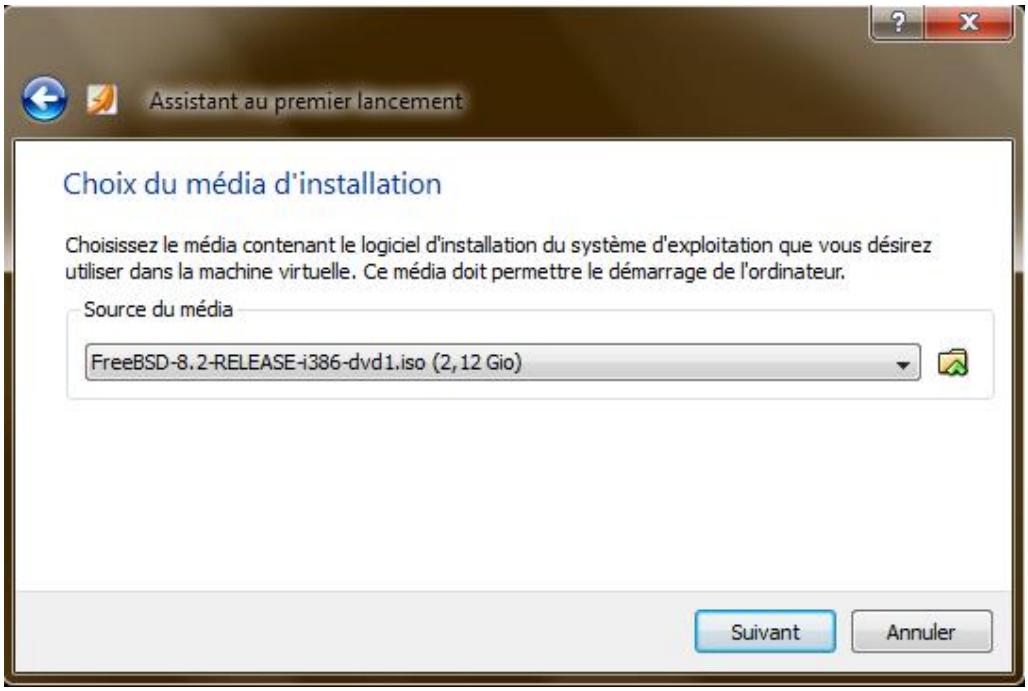


Figure B.18 : chargement de la source du media.

Dernier récapitulatif pour la route (figure B.19) :

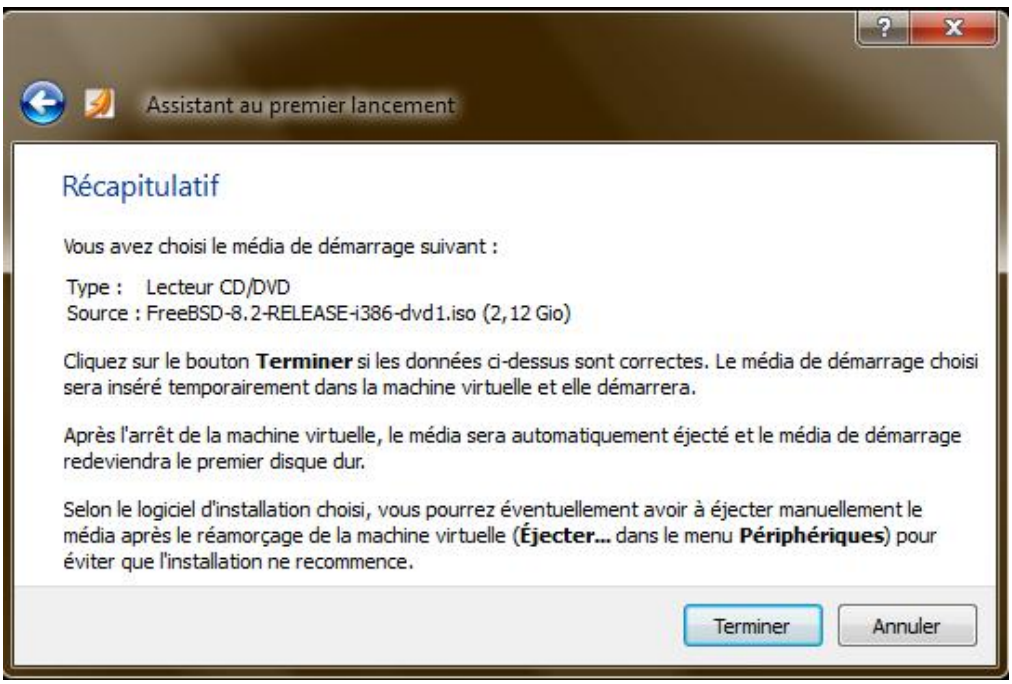


Figure B.19 : récapitulatif.

La machine virtuelle va lancer le système d'exploitation (figure B.20) :

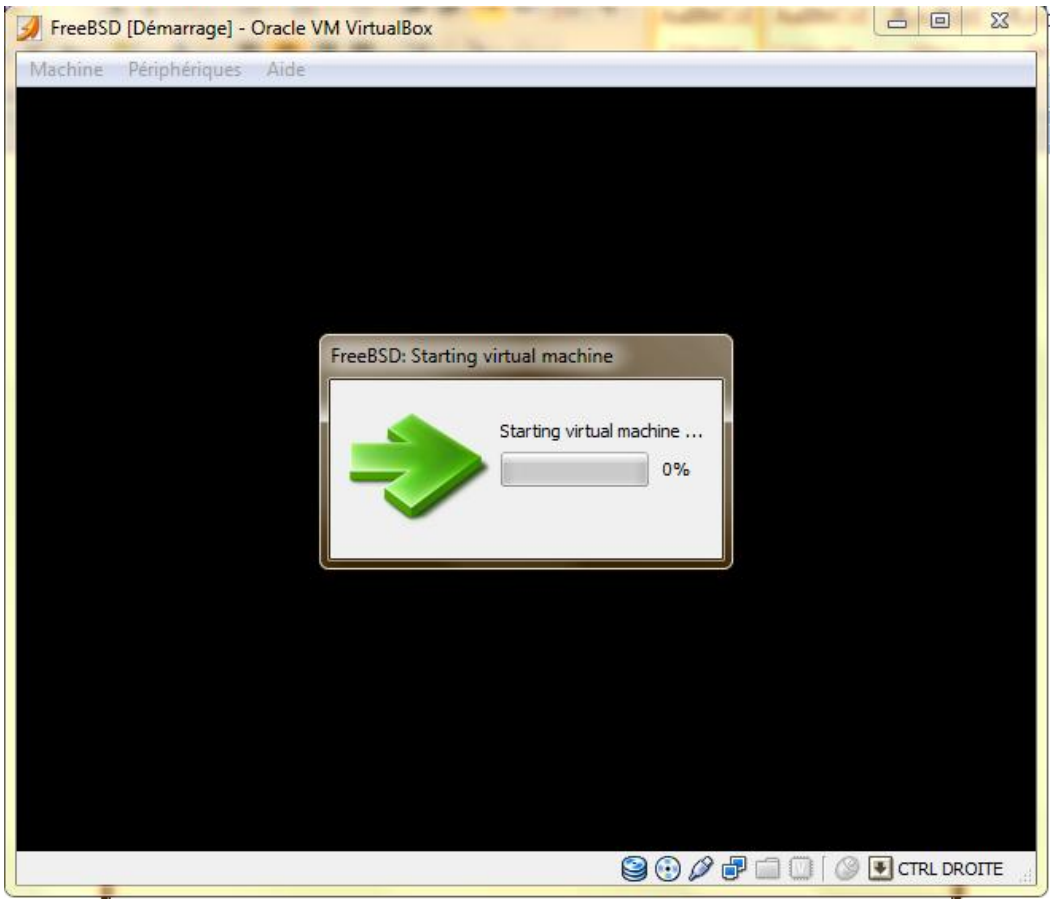


Figure B.20 : premier démarrage de la machine virtuelle.

La figure ci-dessous est la première boite de dialogue se référant au démarrage de FreeBSD nous demande qu'elle mode de boot nous voulant effectuer (figure B.21).

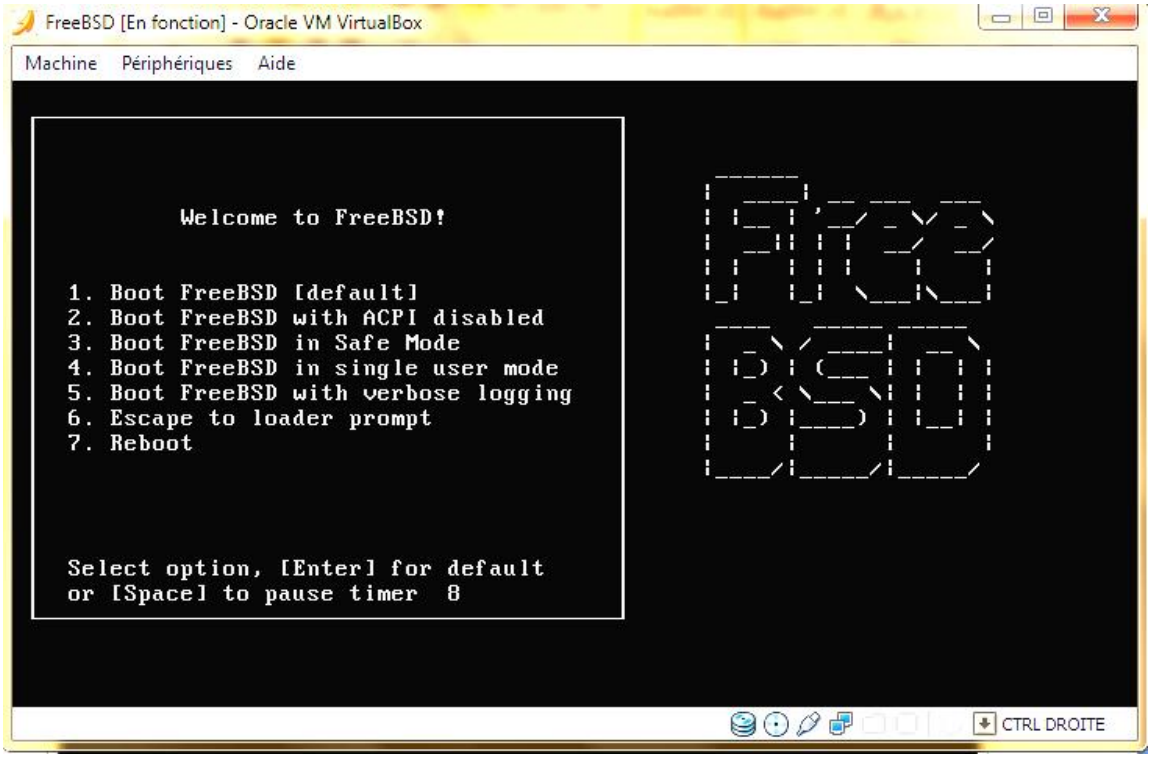


Figure B.21 : premier démarrage de FreeBSD.

Nous avons ainsi réussi à créer notre première machine virtuelle. En se qui concerne l'installation du système d'exploitation elle est expliquée en détaille dans l'annexe **C.**

- **Installation de FreeBSD :**

Les procédures d'installation varient un peu d'une version à l'autre, beaucoup selon le système installé, énormément en fonction de l'architecture.

Comme le PC sur lequel on travail à un processeur **Intel 32 bits**, la version adéquate de FreeBSD est **i386.**

On expliquera comment installer FreeBSD version **8.2-i386 (la dernière version)** de deux façons différentes:

-en utilisant VirtualBox pour l'installer sur une machine virtuelle créée sur le pc.

-en l'installant directement sur l'ordinateur avec un cd bootable.

- **Téléchargement du système d'exploitation :**

Avant de procéder à l'installation la première chose à faire est de télécharger la version De FreeBSD sur le serveur ftp de celui-ci. Car il n'existe pas en Algérie un fournisseur de CD BOOTABLE FreeBSD, ce qui fait qu'on est obligé de le créer nous même si nous envisageant l'installation sur un support physique réel.

Sur le site officielle de FreeBSD, lorsqu'on appui sur l'image ISO à téléchargée une page web (figure C.1) s'affiche pour choisir quelle type de compilation on veut télécharger. Nous avions choisi de la compilation **FreeBSD-8.2.RELAESE-i386-dvd.iso.xz** et la MD5

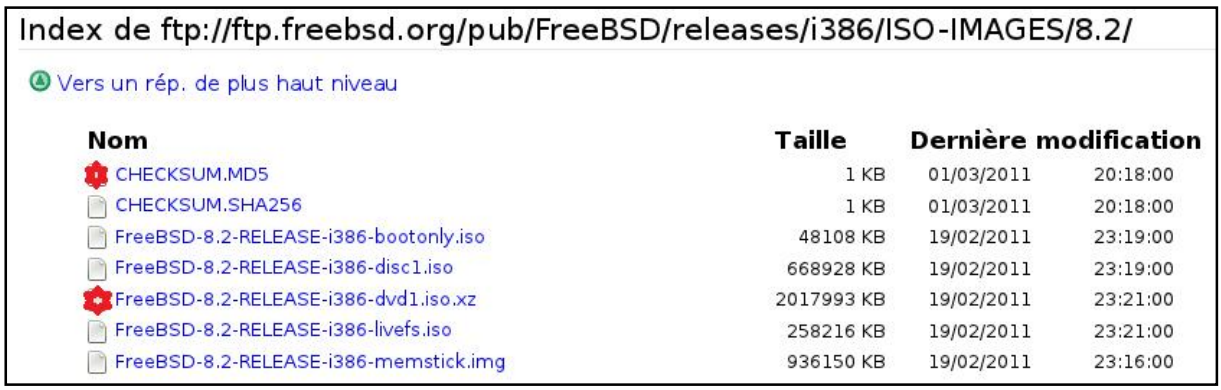


Figure C.1 : le site FTP de téléchargement de **FreeBSD i386** version **8.2**

La somme MD5 nous permet de savoir si le fichier que nous avons téléchargé est arrivé en bon état sur l'ordinateur. Chaque fois qu'on télécharge une image ISO (quelle qu'elle soit) on peu vérifier si elle est corrompu avec le logiciel md5summer. On calcule la

somme MD5 du fichier téléchargé (après l'avoir décompresser) avec le logiciel puis on compare le résultat avec la somme téléchargé et ils doivent correspondre.

On remarque que le fichier télécharger à une extension .xz et un gestionnaire d'archives standard comme **WINRAR** n'arrivera pas à le décompresser. Le meilleur moyen est de faire appel au web pour chercher l'application qui est capable de la faire, puis la télécharger et décompresser le fichier. Le résultat sera une image ISO dont nous nous servirons pour l'installation.

1-installer FreeBSD sur VirtualBox :

On a juste besoin de l'image ISO du système d'exploitation à travers laquelle celui-ci va booté 'démarrer' pour plus d'information sur la procédure de démarrage sur VirtualBox voir **l'annexe B**.

2-installer FreeBSD sur un support physique réel :

Pour ce faire il faut créer un CD bootable avec l'image ISO du système d'exploitation. En gravant cette même image sur un dvd avec l'utilitaire approprié. Lorsque ces étapes sont terminées la configuration est la même pour les deux cas puisqu'on procède à l'installation. La première boite de dialogue (figure C.2) nous demande de choisir quel type de démarrage nous voulons effectuer :

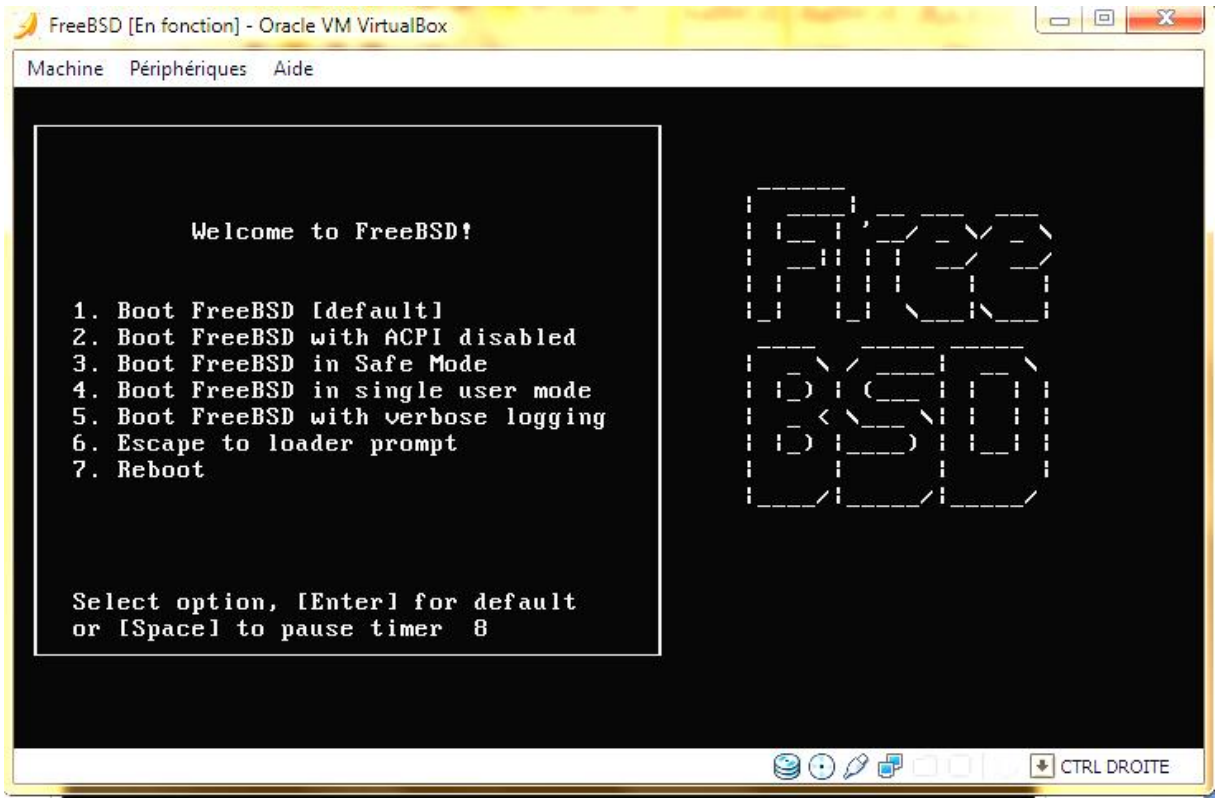


Figure C.2 : choix du démarrage de **FreeBSD.**

- on choisi le démarrage par défaut pour une première installation et on appui sur **entrer**.

Le programme d'installation de FreeBSD est **sysinstall** celui-ci se charge des qu'on choisi le mode de démarrage.

- En premier lieu on choisi le pays (figure 2.3). On a choisi la France pour des raisons de langue.

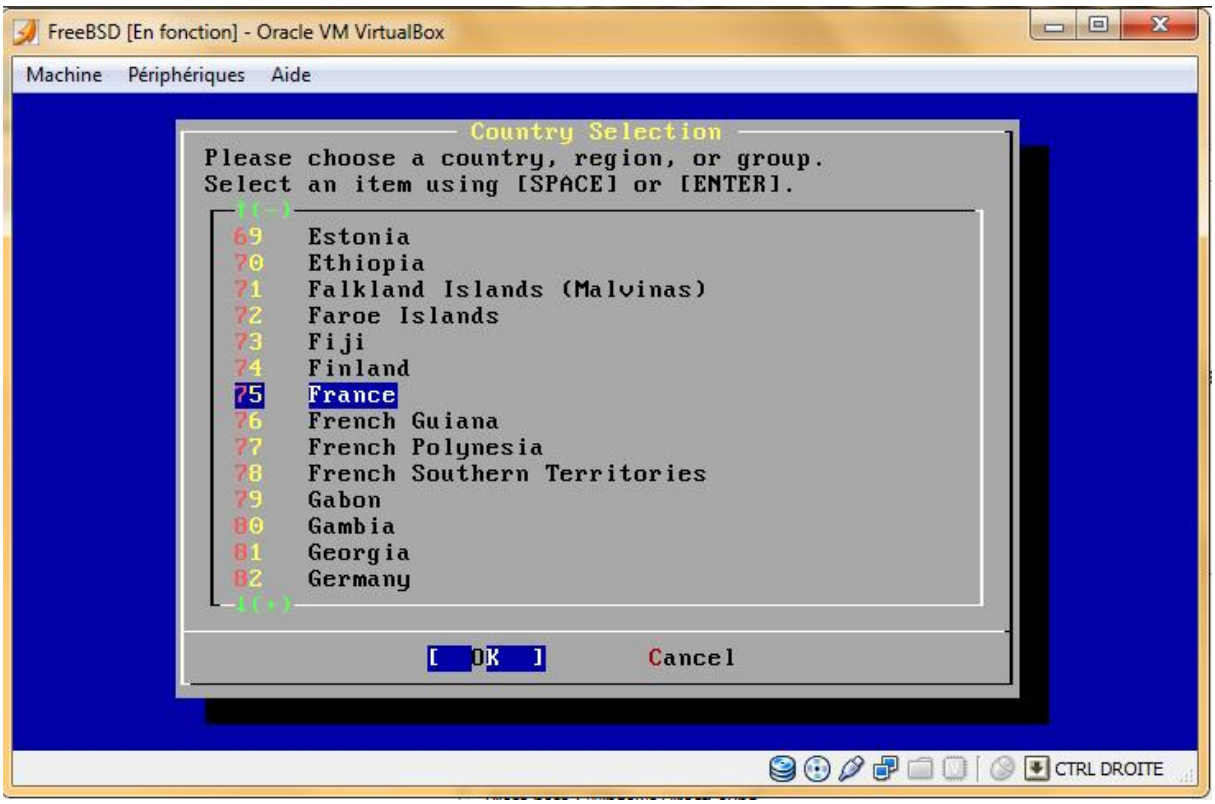


Figure C.3 : choix du pays.

Ø puis on choisi le type de clavier (figure C.4) :

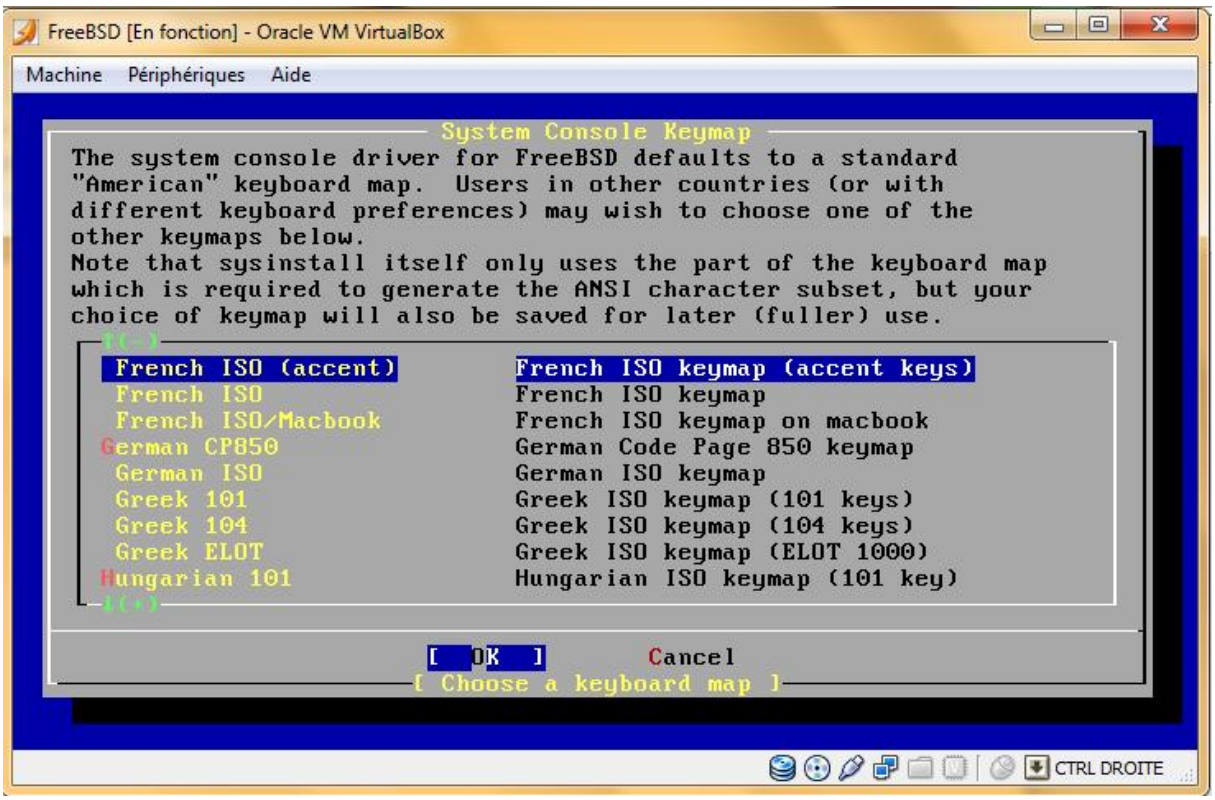


Figure C.4 : choix du type de clavier.

Ø pour une première installation on choisi l'installation standard (figure C.5):

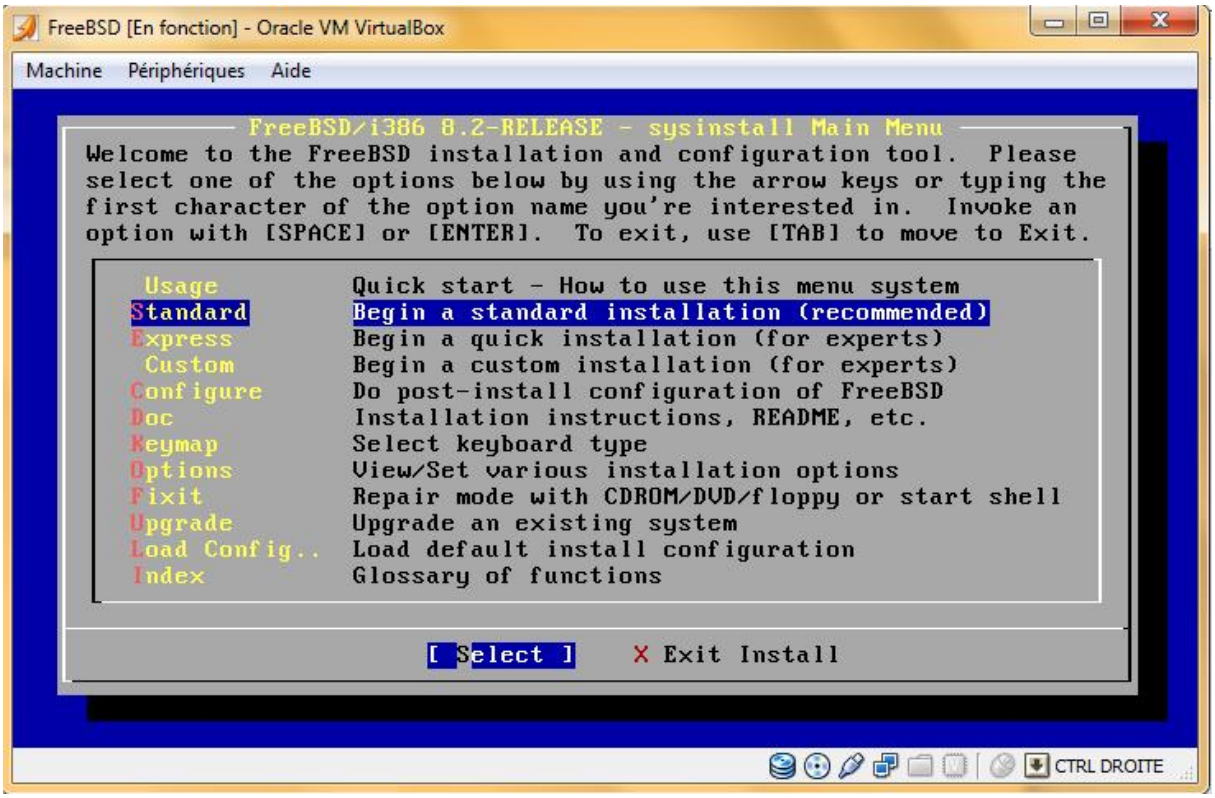


Figure C.5 : choix d'une installation **standard**.

Ø C'est alors que vient la partie du partitionnement du disque dur :

-si nous installons FreeBSD sur un support physique on choisi la partition ou nous voulant le faire. Bien qu'il y a deux possibilités :

-installer FreeBSD tout seul sur la machine : on peut choisir de lui allouer tout l'espace disque en appuyant sur **A**. ou bien de lui créer une partition dédié en appuyant sur **C** et on choisi après la taille de la partition.

-installer FreeBSD comme système d'exploitation invité : soit on lui alloue une partition. Soit on procède à la défragmentation du disque dur ou de la partition ou se trouve le système d'exploitation hôte. Cette dernière opération consiste à regroupé les fichiers du l'OS hôte initialement reparti un peu partout sur le support, dans une partie du support de stockage et laisser ainsi l'espace restant à l'autre. Il existe un logiciel performant : **Partition Magic[X]** sous **Windows** et **GParted** sous **Linux.**

-si nous installons FreeBSD sur VirtualBox c'est plus simple on lui alloue tout le disque dur virtuel (figure C.6).

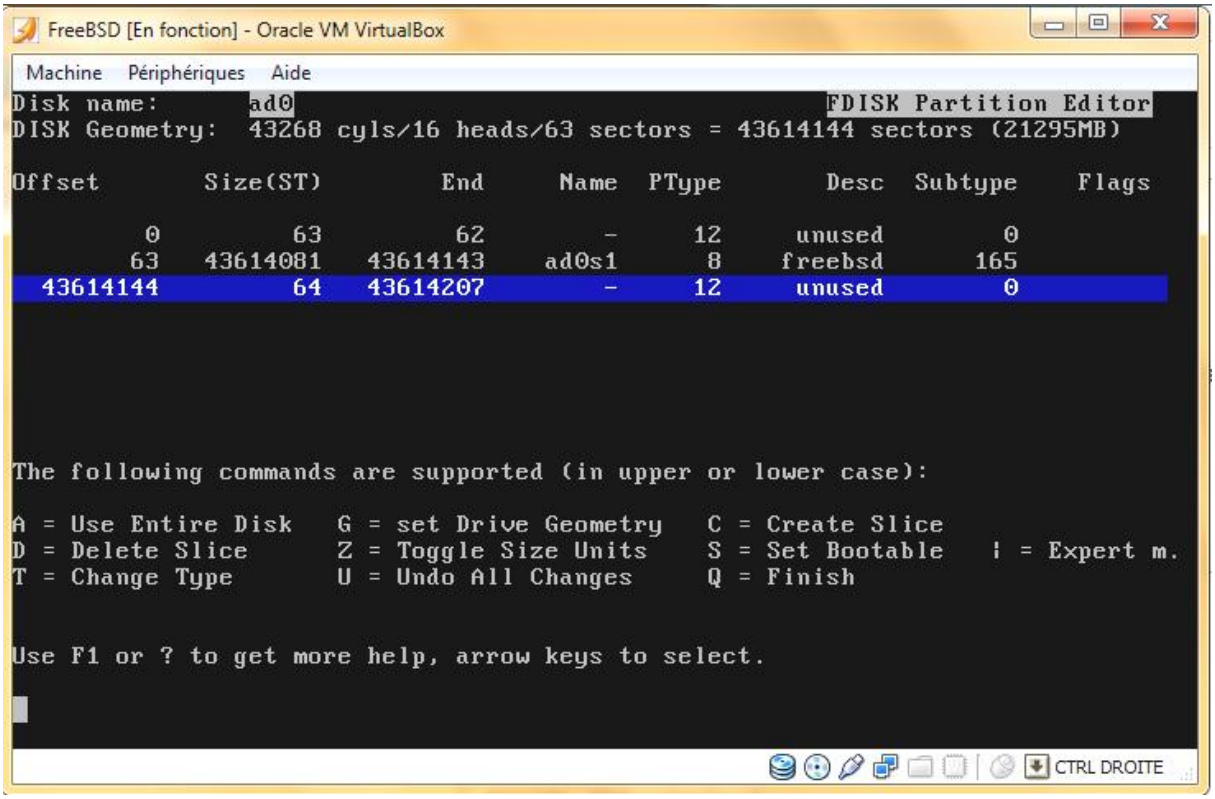


Figure C.6 : choix d'une partition pour l'installation de **FreeBSD.**

Ø pour quitter il faux appuyer sur **q.**

La boite de dialogue ci après (figure 2.7) nous demande que faire à propos du boot manager.

Le **Boot Manager** est un programme qui se lance au démarrage de l'ordinateur et nous permet de choisir quel système d'exploitation va être utilisé. Celui de FreeBSD s'appelle **boot0** et celui de Linux s'appelle **GRUB**.

Le **Boot Manager** n'est pas installé sur l'une des partitions du disque dur mais dans une zone particulière du disque dur qu'on appelle le **Master Boot Record (MBR)**. Il s'agit des 512 premiers octets du disque dur, ceux que l'ordinateur lit en premier quand il boote. Le problème, c'est qu'on ne peut installer qu'un seul **Boot Manager** sur le **MBR**. Il va donc falloir faire un choix : soit on installe le boot managé standard soit celui de FreeBSD.

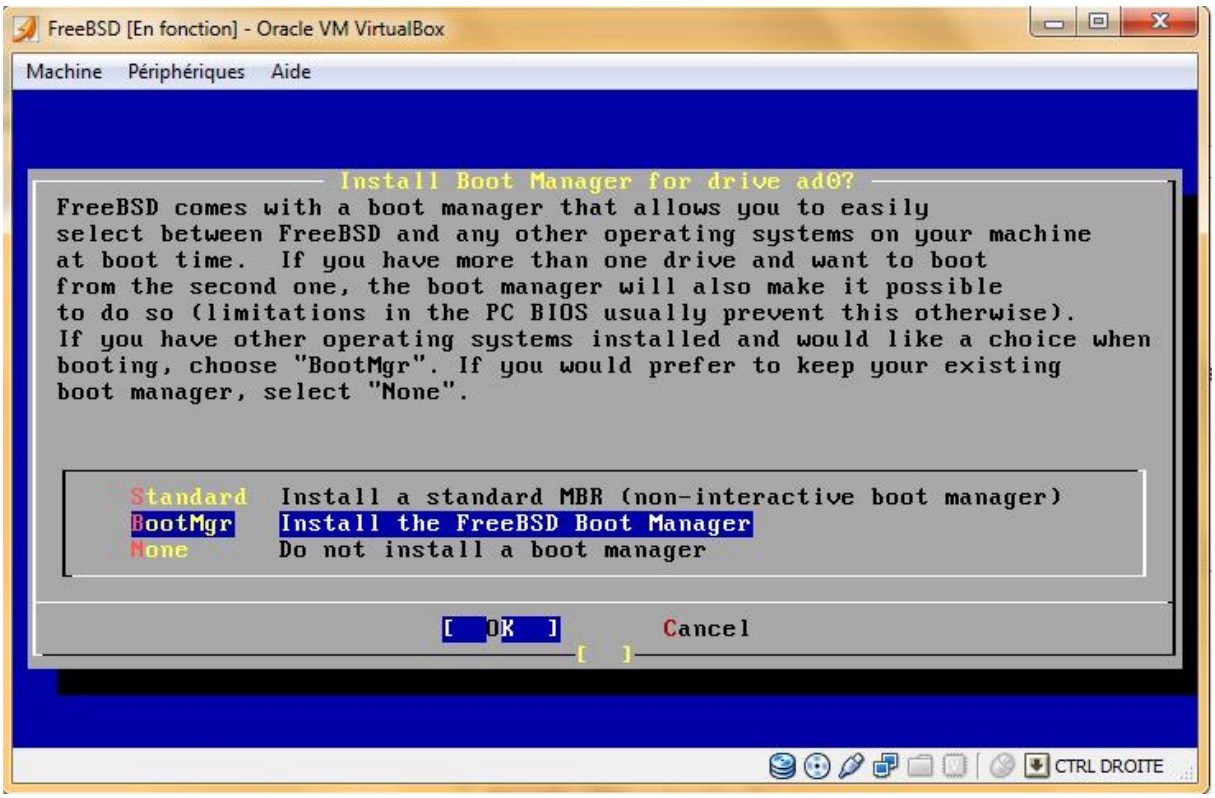


Figure C.7 : choix du bootmanager**.**

- après avoir **partitionner** le disque dur (virtuel ou réel selon notre choix) avec le logiciel **FDisk**, c'est à dire y délimiter des **partitions**, des zones aux propriétés différentes. Nous allons maintenant créer des **répartitions logiques**, chacune étant consacrée à un ou plusieurs **dossiers** (**répertoires**).

En générale FreeBSD a besoin de cinq partitions :

-/ : la racine. Elle contiendra le noyau, tous les programmes de démarrage et tous les autres dossiers. Il est préférable que sa taille soit un peu prés de 4% de la taille totale

-swap. Lorsque la mémoire RAM est saturée, certaines informations sont stockées sur cette partie du disque dur. Cela ralentit les programmes car il est moins rapide de lire sur le disque que dans la RAM. C'est pour les raisons citer précédemment que la taille de la partition swap devra être d'environs 2 fois la taille de la RAM.

-/tmp et /var. Ils sont des dossiers dont le contenu change tout le temps donc il est plus pratique que le système d'exploitation sache ou les trouver. **/tmp** et **/var** doit avoir respectivement plus de 4% et 10% de la taille totale.

-/usr. Il contient les dossiers et fichier des utilisateurs : texte, image, vidéo…l'espace libre restant sera consacrer au dossier **/usr.**

La procédure de création d'une partition logique :

Nous allons créer la racine : le dossier **/.**

- On appui sur **C.** pour créer une nouvelle partition.
- On lui donne une taille en faisant bien attention de mettre **M** devant le chiffre. Comme le montre la figure suivante (figure C.8) :

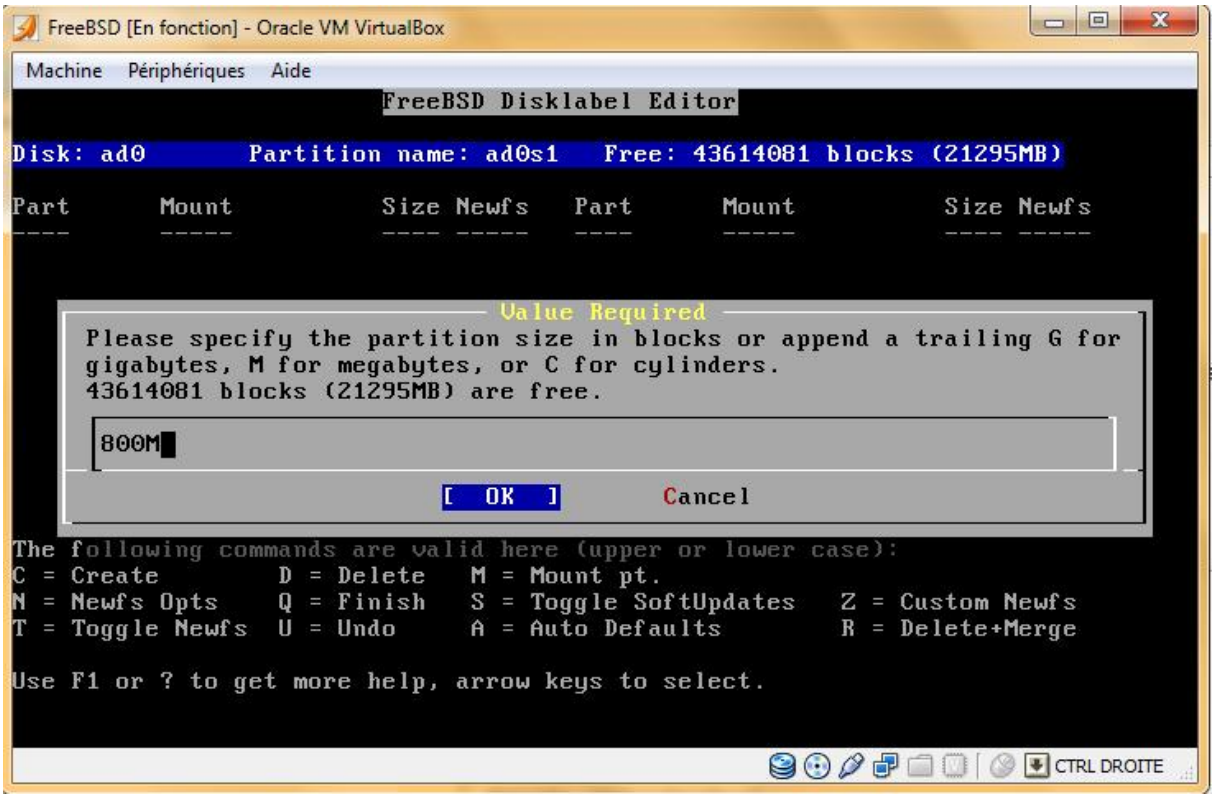


Figure C.8 : donner une taille à la répartition logique**.**

- On choisi le type de la partition (figure 2.9). '/ ' est de type **FS : file system (**systèmes de fichiers**)**:

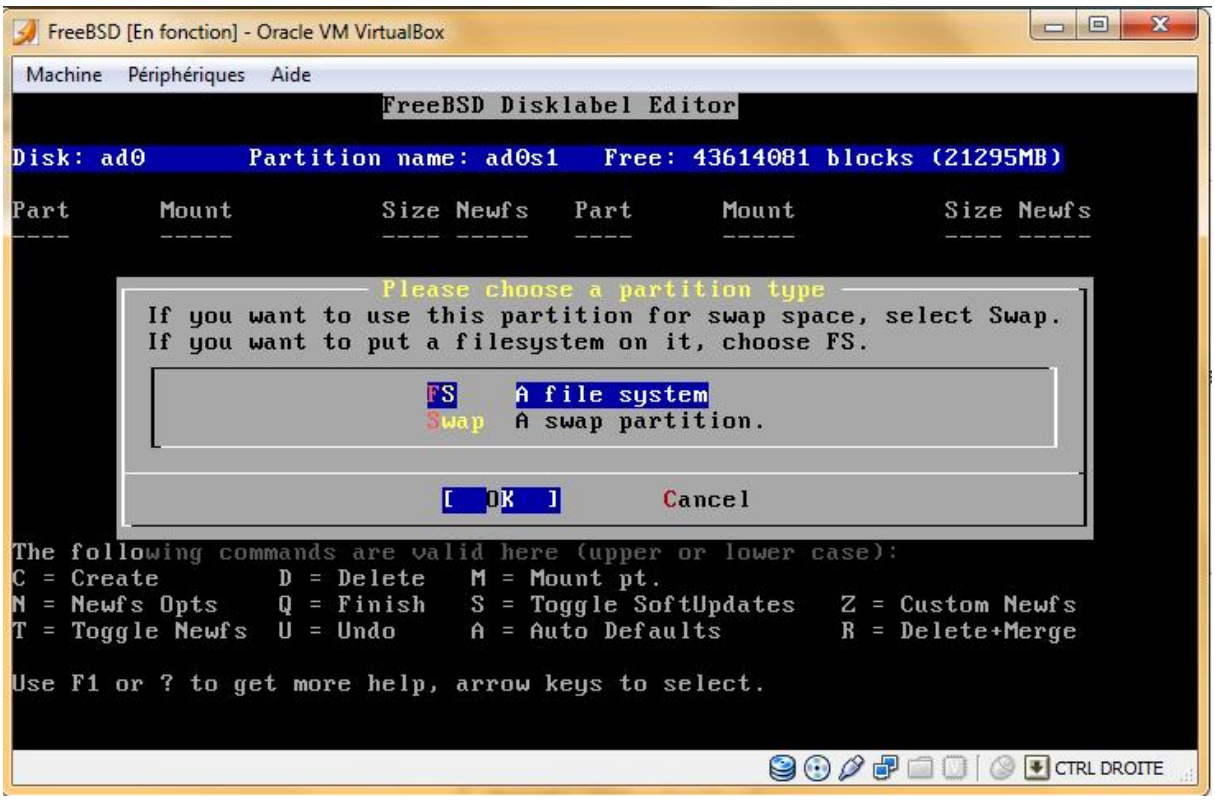
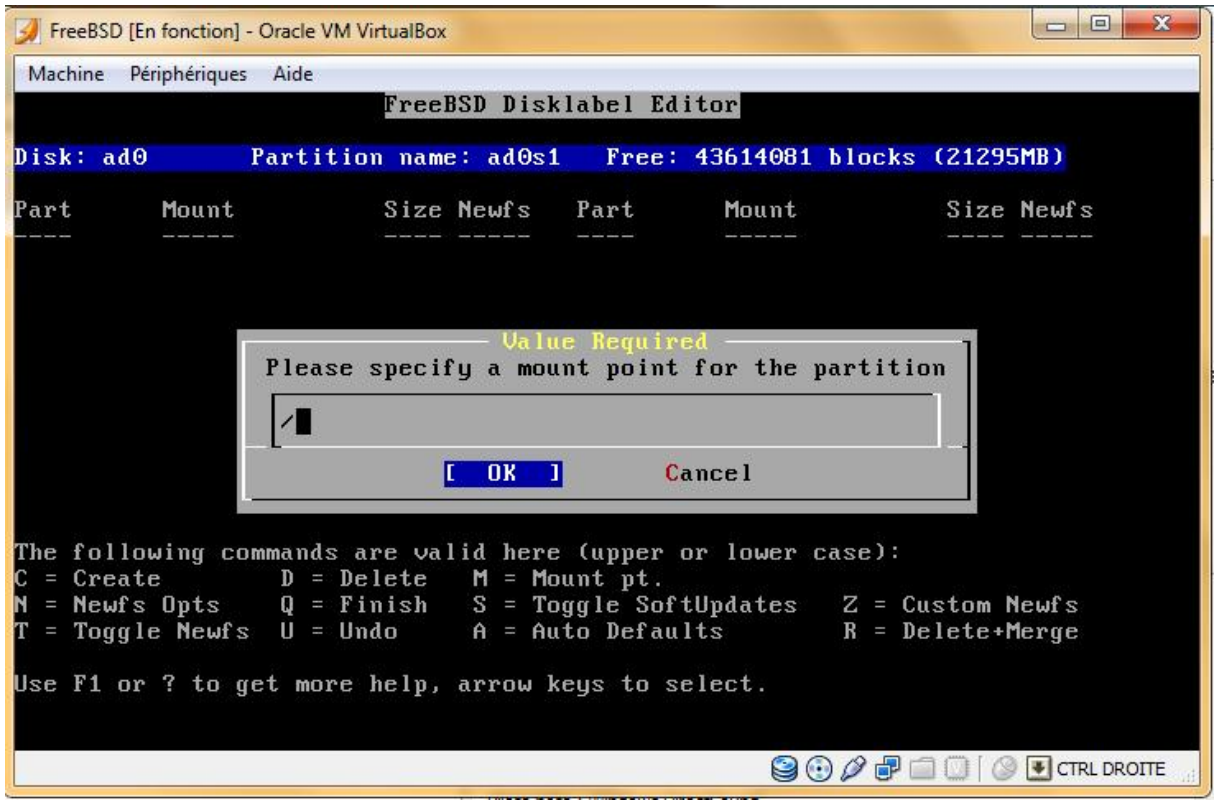


Figure C.9 : choix du type de la répartition**.**

- On donne un mon à la partition qui est ‘/ ’ dans notre cas (figure C.10) :



Ø on notera que toutes les partitions se créer de la même façon sauf **swap** qui de type **swap** (figure C.11) :

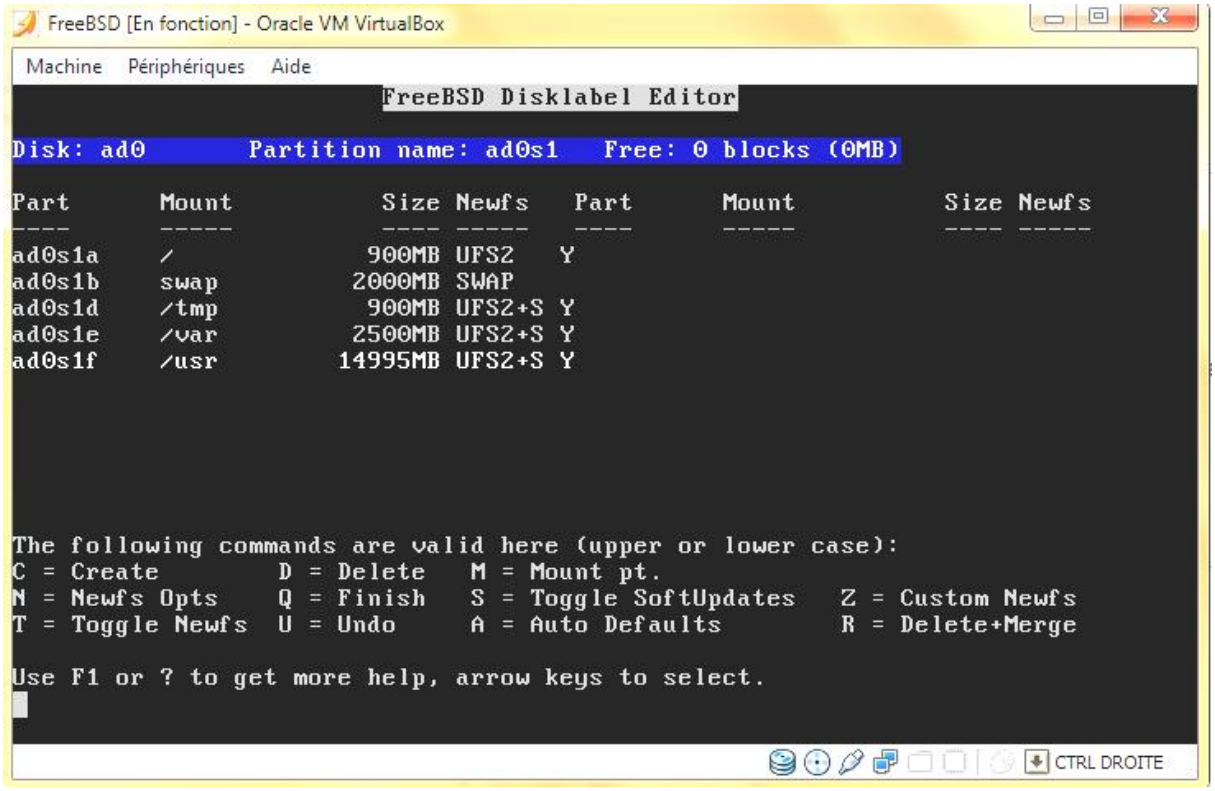


Figure C.11 : les listes des répartitions logiques**.**

Ø Lorsque la création des partitions logiques est achevée il faut appuyer sur **Q**.

A présent, le programme **sysinstall** va nous demander quels éléments nous voulons installer (figure 2.12). Le plus simple est de tout prendre en choisissant **All**. Pour essayer de gagner un peu d'espace disque, nous pouvons aussi choisir **Custom**, l'installation à la carte :

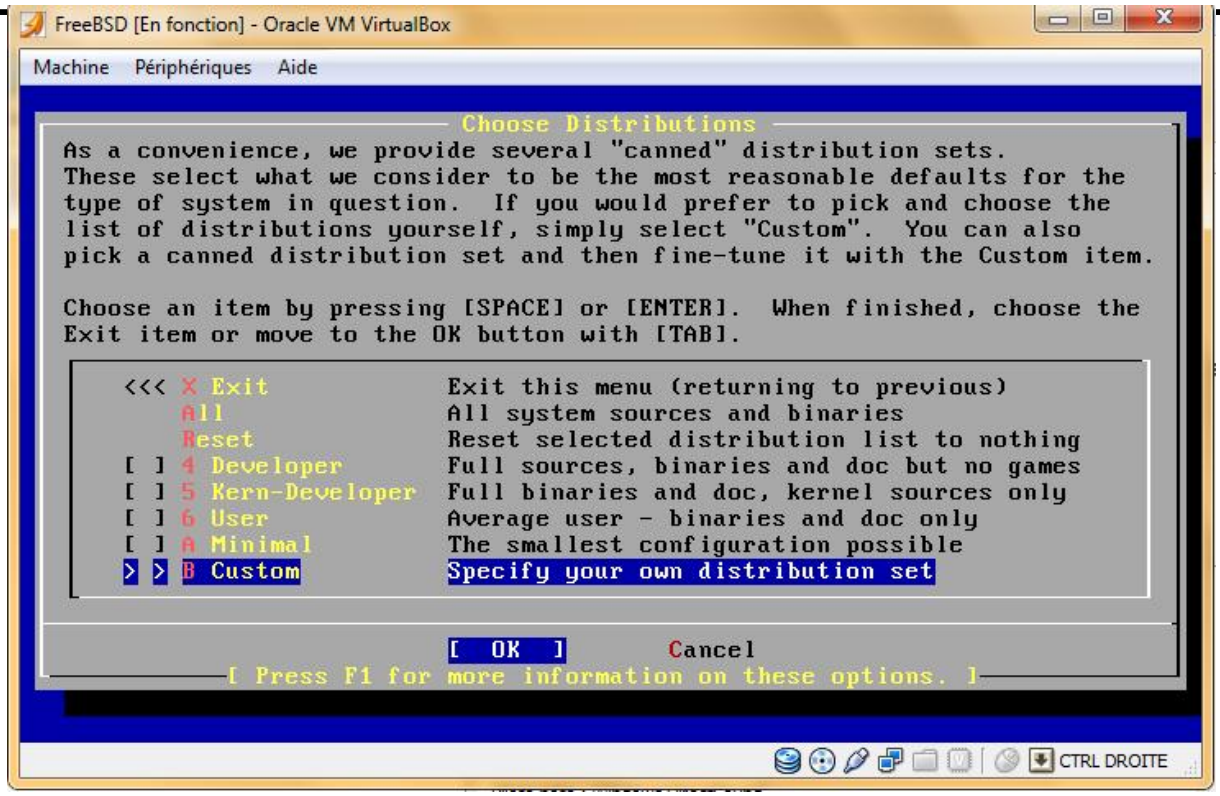


Figure C.12 : choix d'une distribution**.**

La boite de dialogue ci-dessus (figure 2.13) nous demande de choisir quels éléments installer :

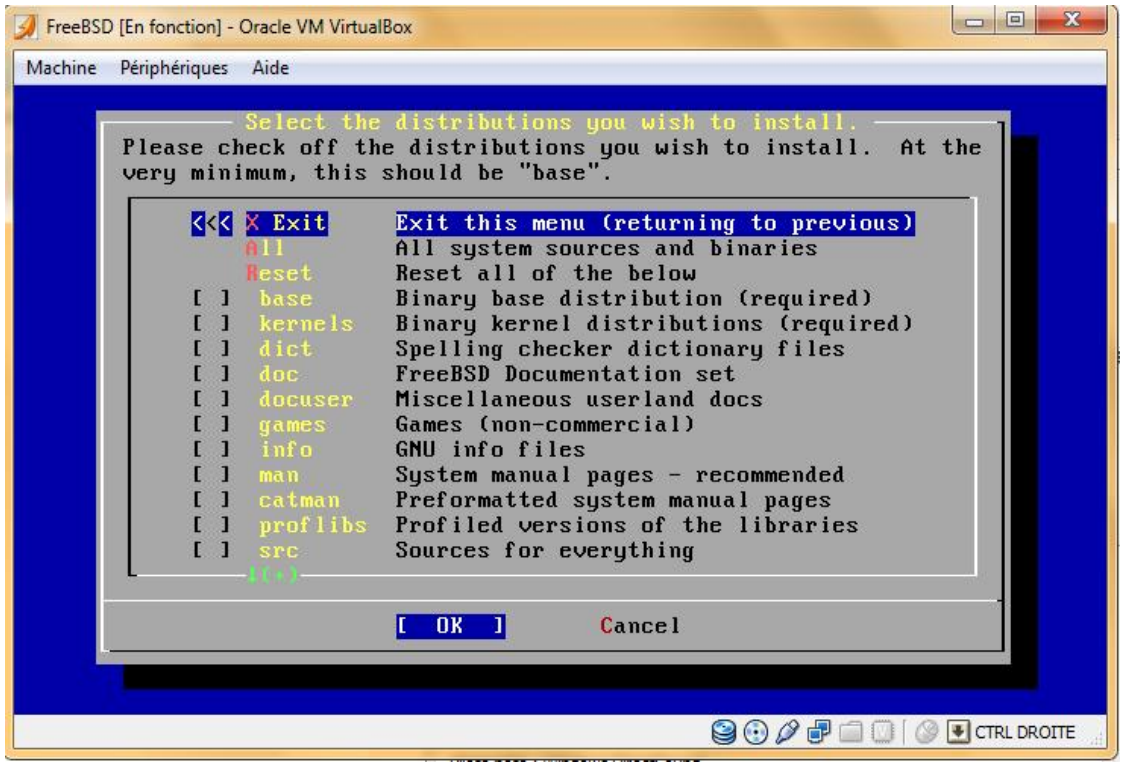


Figure C.13 : choix des éléments à installer**.**

Voyons ce que cette "carte" propose:

- Le système de base. On prend, bien sûr.
- Les noyaux (kernels). On choisit le noyau GENERIC (figure 2.14).

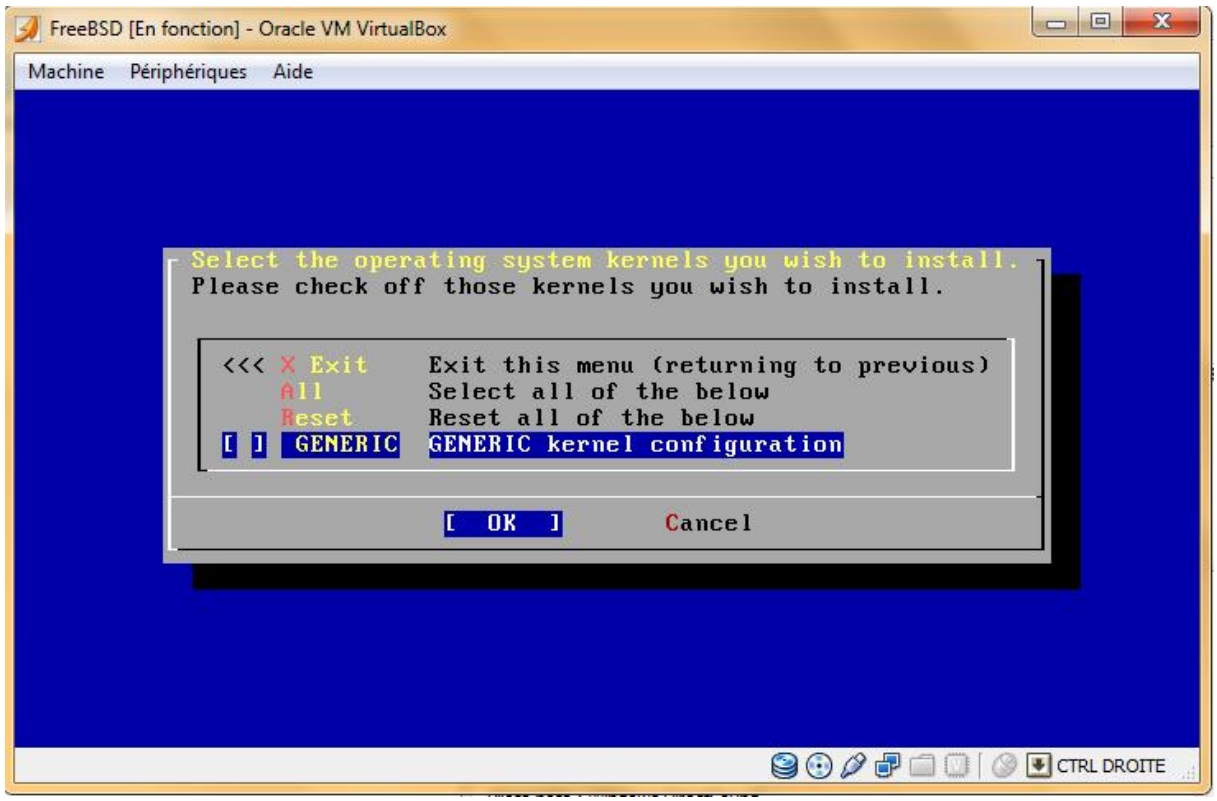


Figure C.14 : choix du noyau**.**

- Le vérificateur orthographique. On prend bien sûre c'est utile.
- La doc en Français (figure 2.15).

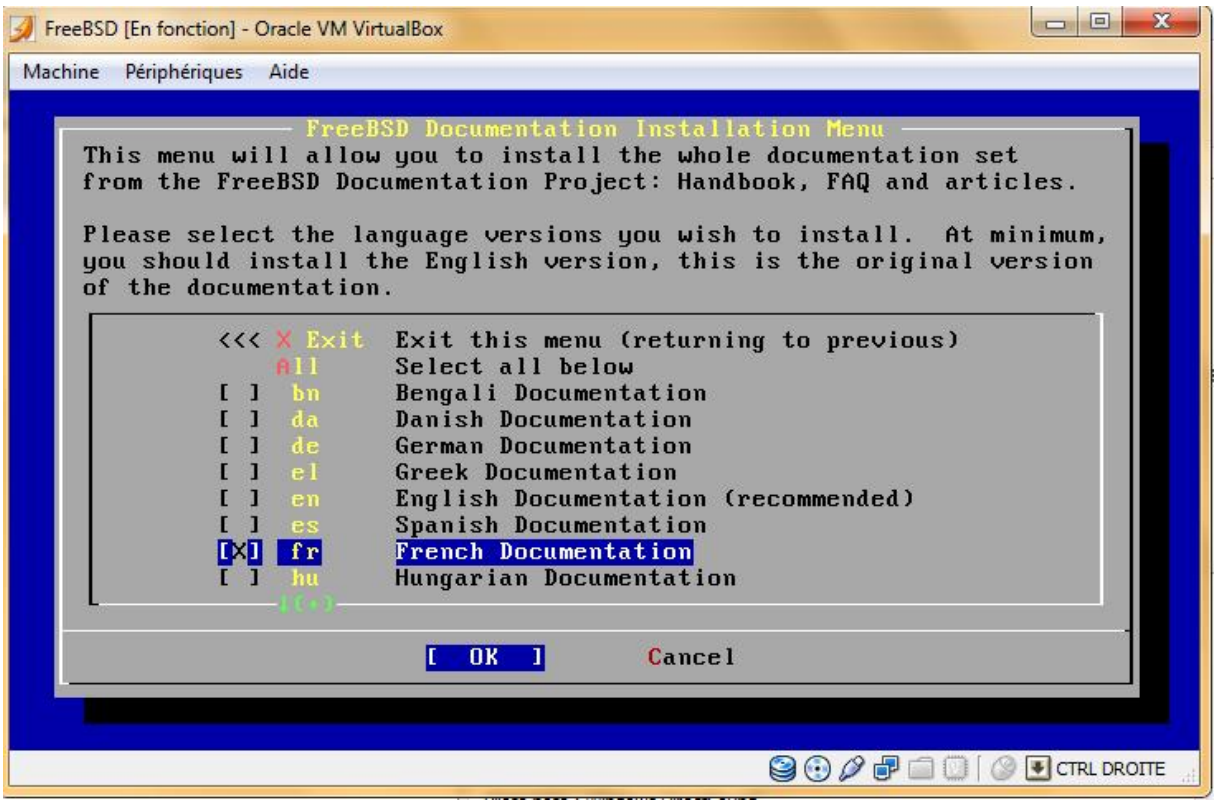


Figure C.15 : choix de la documentation en français**.**

- docuser : Même principe
- Les jeux. Pas très ludiques, mais il y a des choses utiles, comme les programmes **grdc** et **fortune**. On les prend donc.
- Les infos GNU. Des infos sur les programmes de la **Fondation pour le Logiciel Libre**.
- Le manuel (man et catman) A prendre absolument.
- Les bibliothèques profilées : des fonctions préprogrammées utilisées par plusieurs logiciels. Très utile.
- Les codes-source : on les prend tous, nous en aurons besoin (figure C.16).

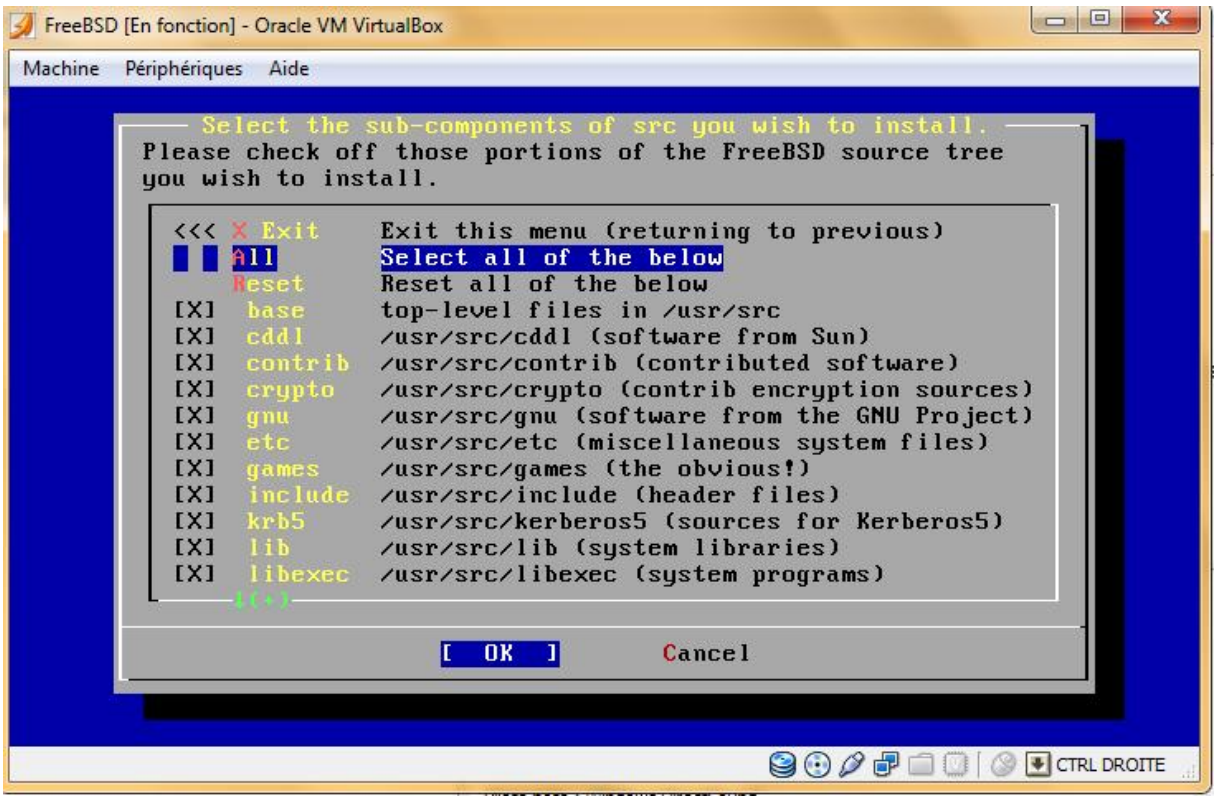


Figure C.16 : le package des codes sources**.**

- Les ports : Essentiels pour installer des programmes. Mais ceux du CD-ROM ne sont plus à jour. Inutile donc de cocher ça : nous installerons les ports autrement.

- local : Des ajouts locaux. On les prend au cas où.

Lorsqu'on est sur de nous choix on appui sur **ok** (figure C.17)**.**

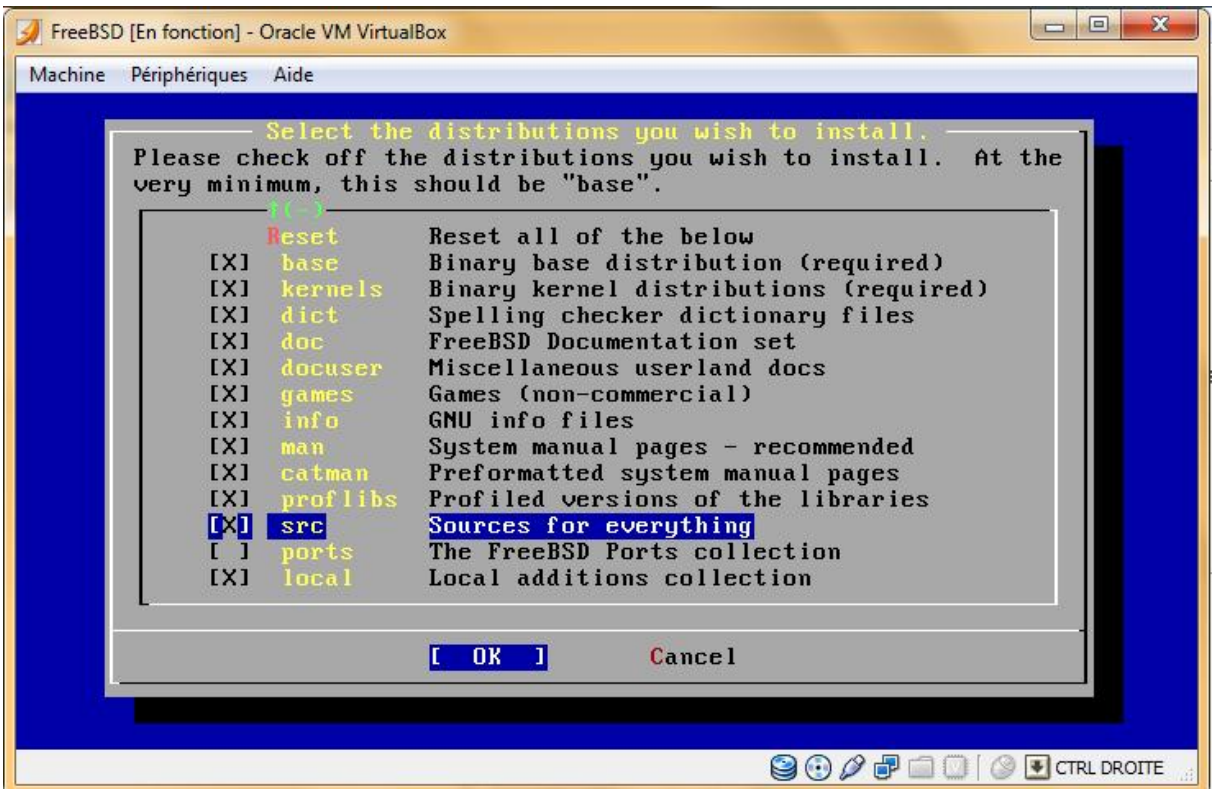


Figure C.17 : la sélection complète.

- à ce stade de l'installation on choisi **exit** pour retourner au menu précédant puis **exit** pour sortir.

Maintenant nous allons choisir le support à partir duquel nous allons poursuivre notre installation (figure 2.18). C'est le lecteur **CD /DVD** bien évidemment :

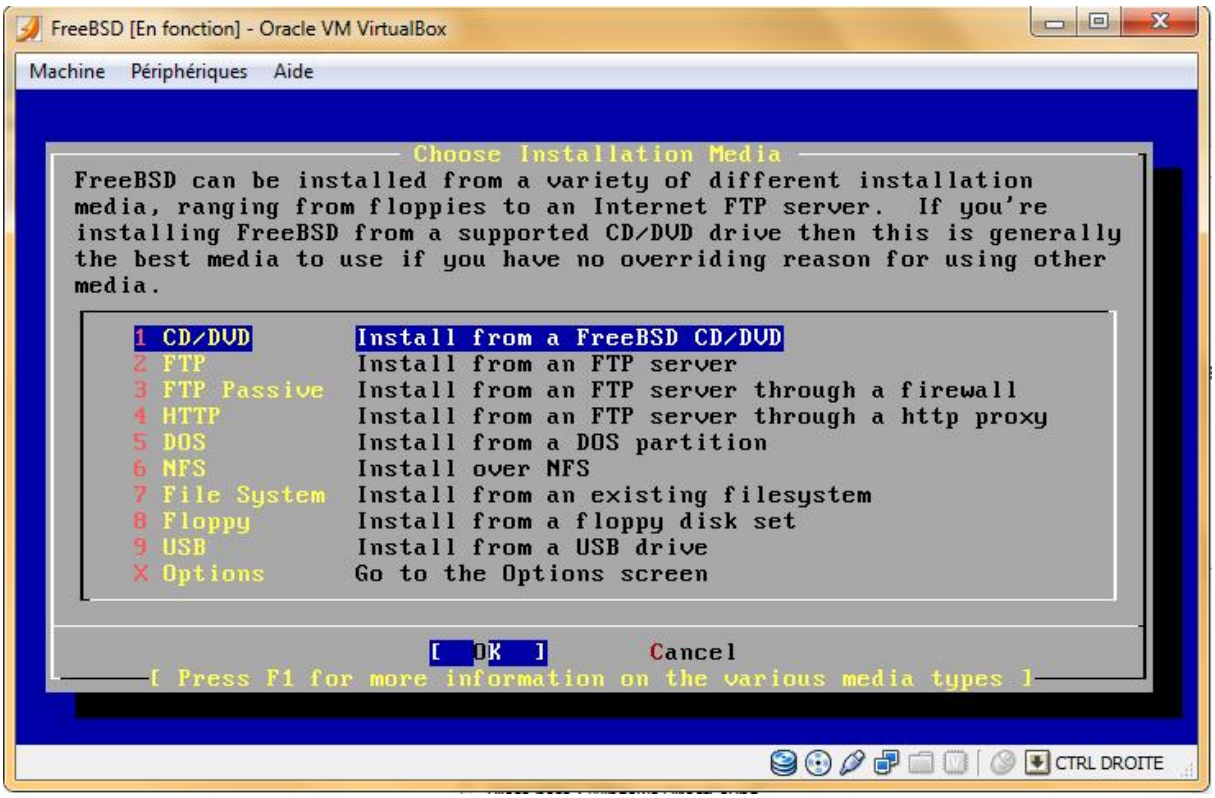


Figure 2.19: choix du media de démarrage**.**

Un dernier message d'avertissement quant au donné stockées sur la partition utiliser (figure C.20) :

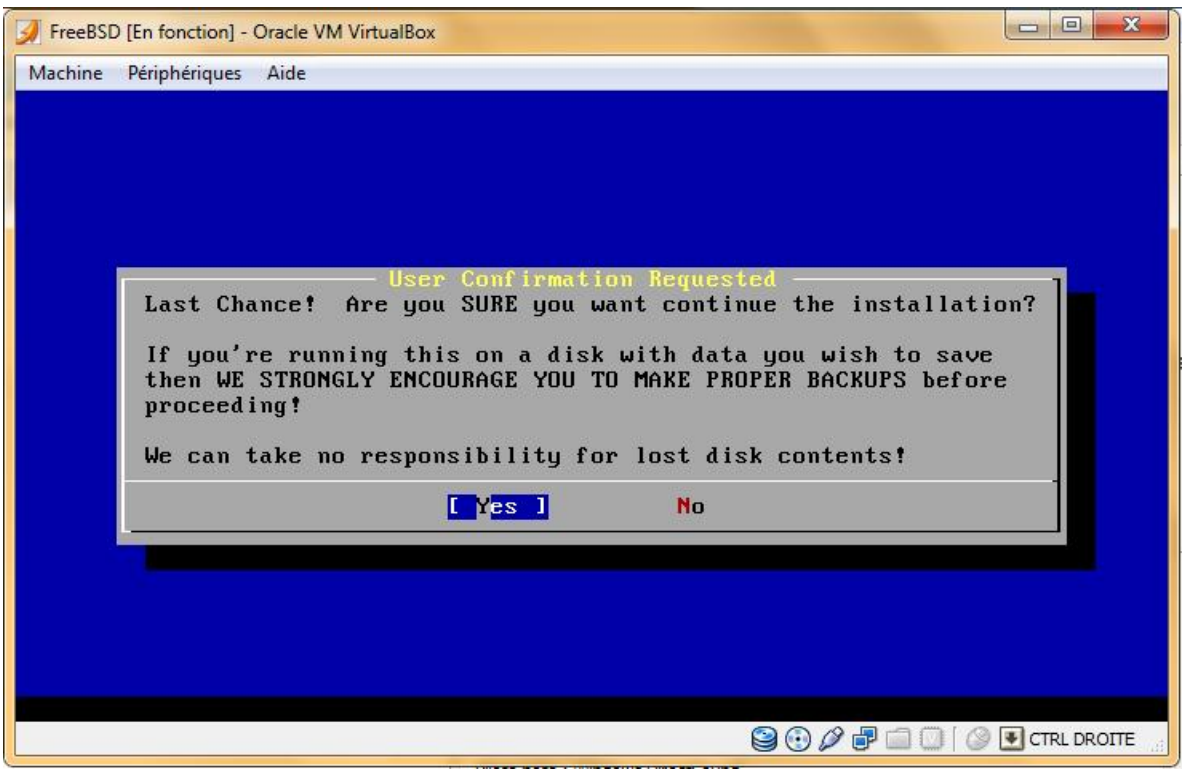


Figure 2.20 : confirmation de l'installation**.**

Ø Le programme d'installation **sysinstall** va nous aider à faire des configurations rapides :

Puisque nous allons configurer que les options de routages on aura besoin de :

1- configurer un réseau Ethernet et les options qui lui sont liées (figure C.21) :

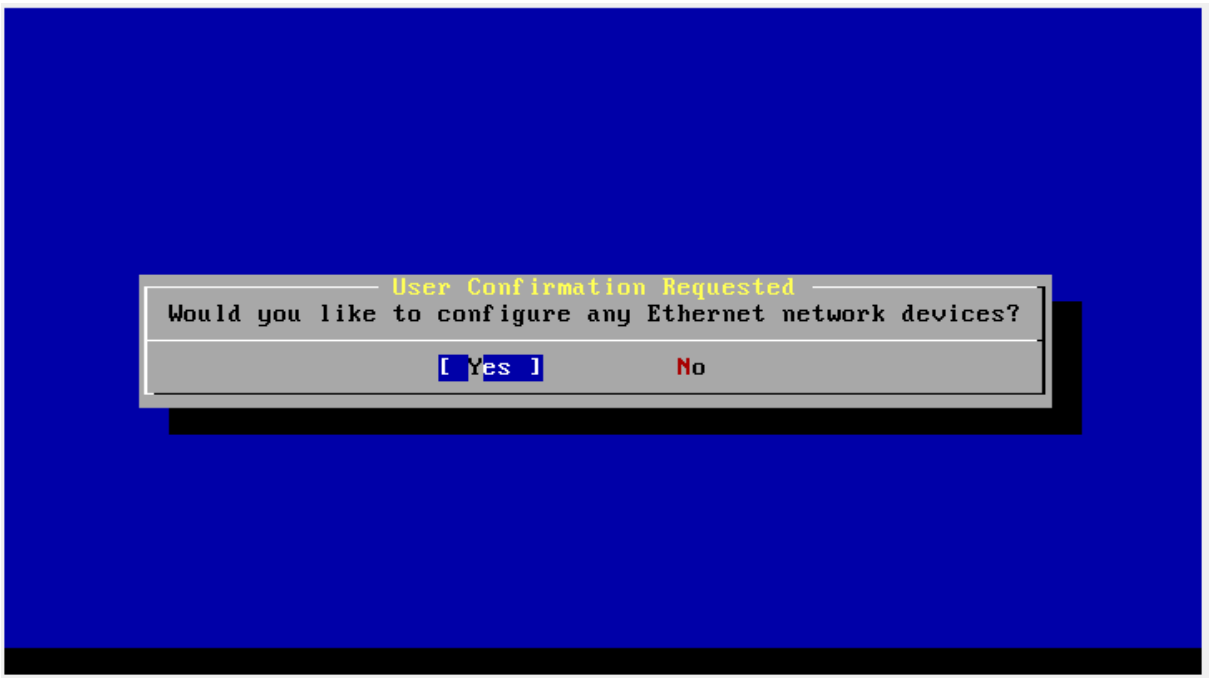


Figure C.21 : configurer un réseau Ethernet.

- configurer la machine comme passerelle (figure C.22) :

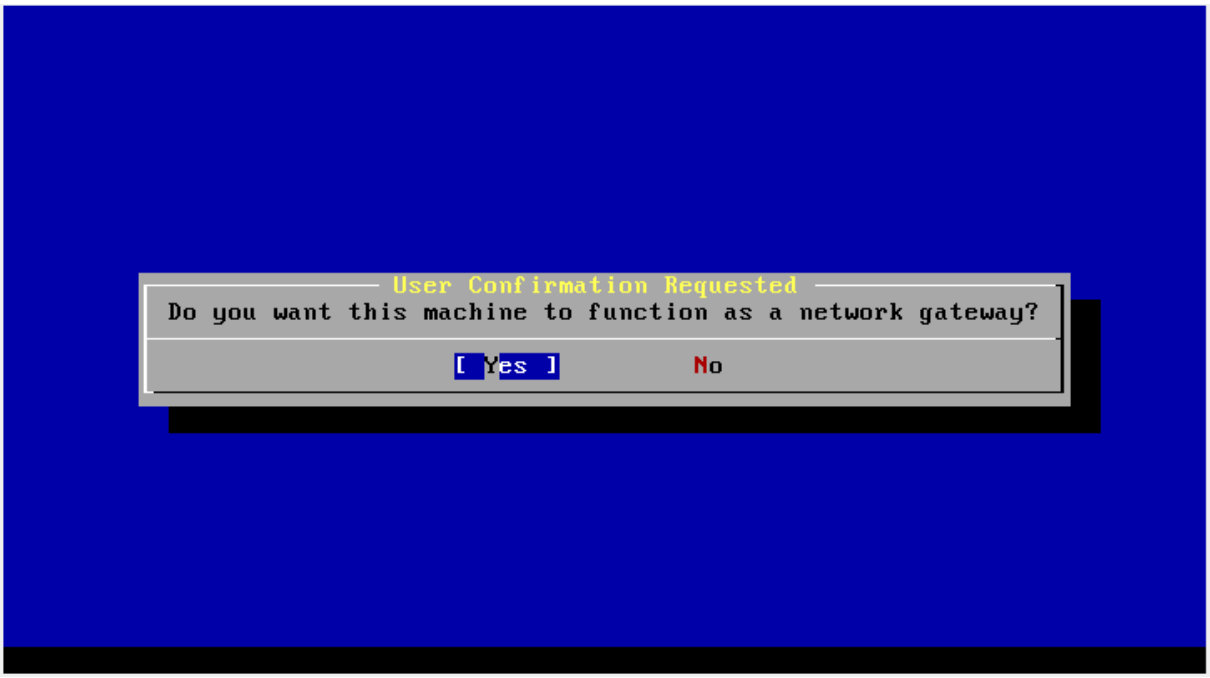


Figure C.22 : configuration de la passerelle.

-charger le daemon **inetd** (figure 2.23):

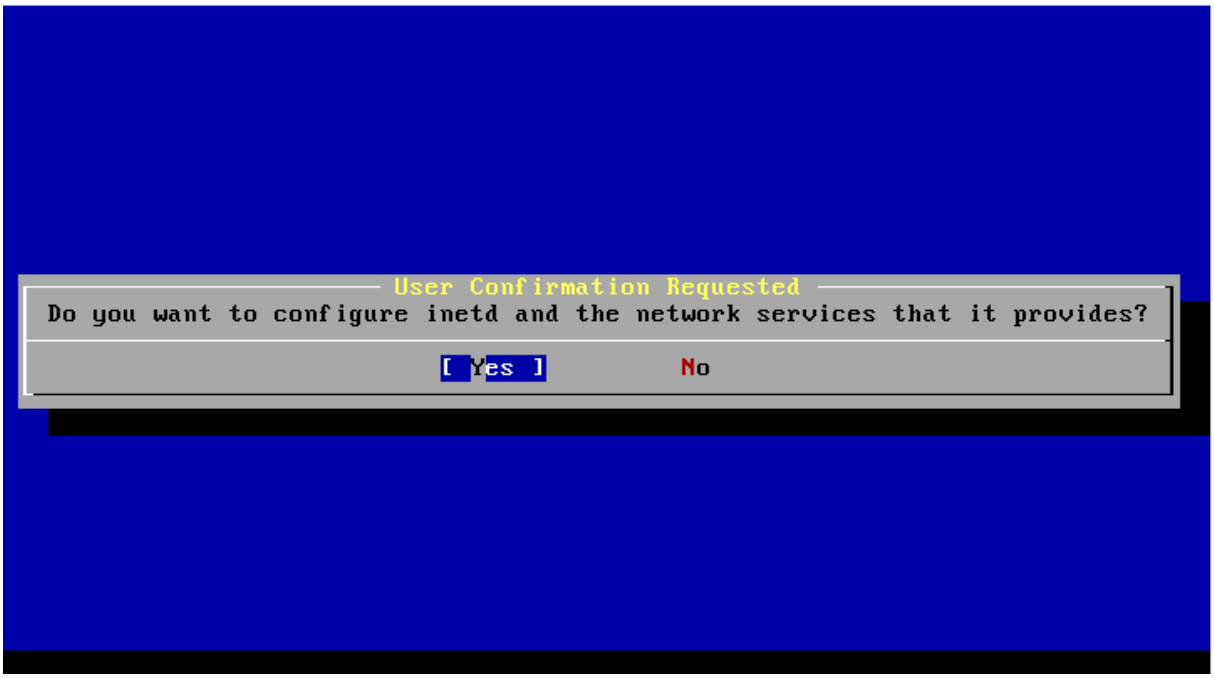


Figure C.23 : configuration des services réseaux de inetd.

- optimiser quelque peu notre console (figure 2.24) :

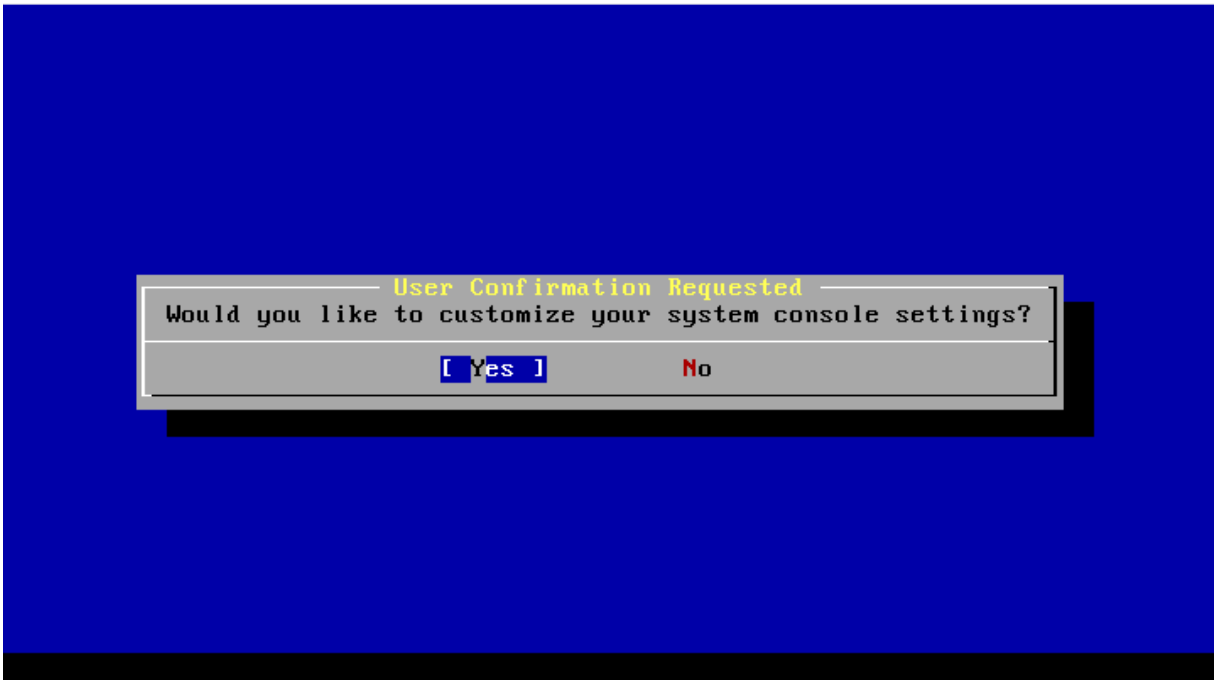


Figure C.24 : optimisation de la console.

-on qu'à choisir selon le goût l'interface que nous préférons (figure C.25) :

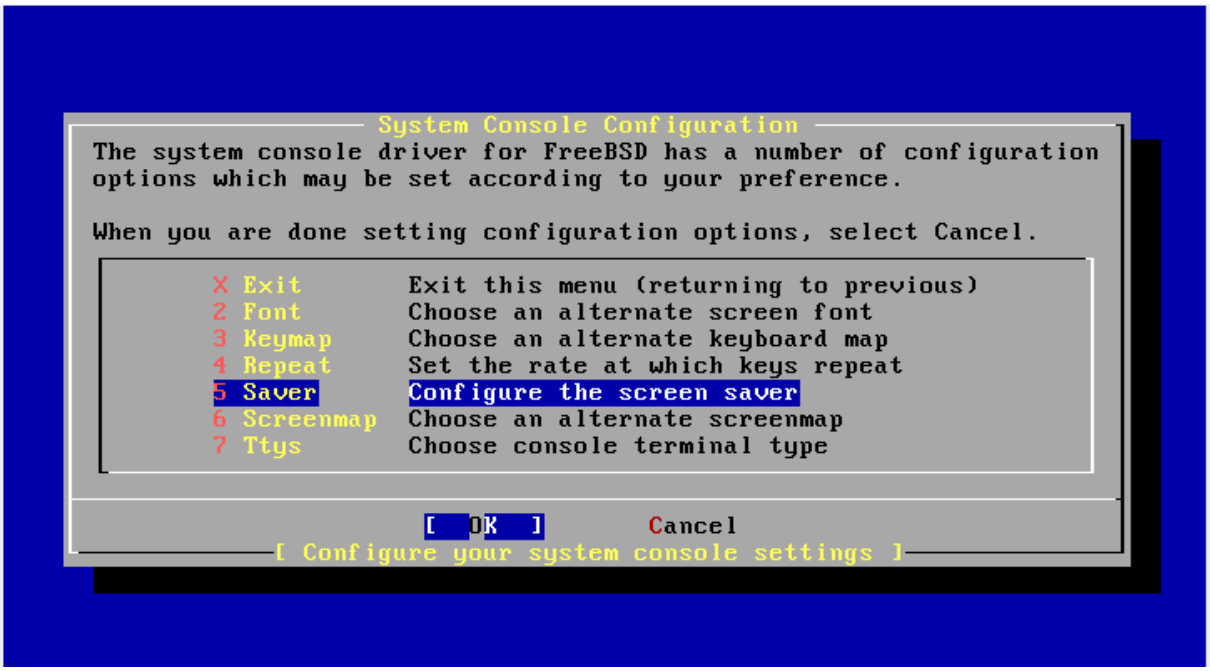


Figure C.25 : configuration de la console**.**

-configurer la zone d'heure (figure C.26) :

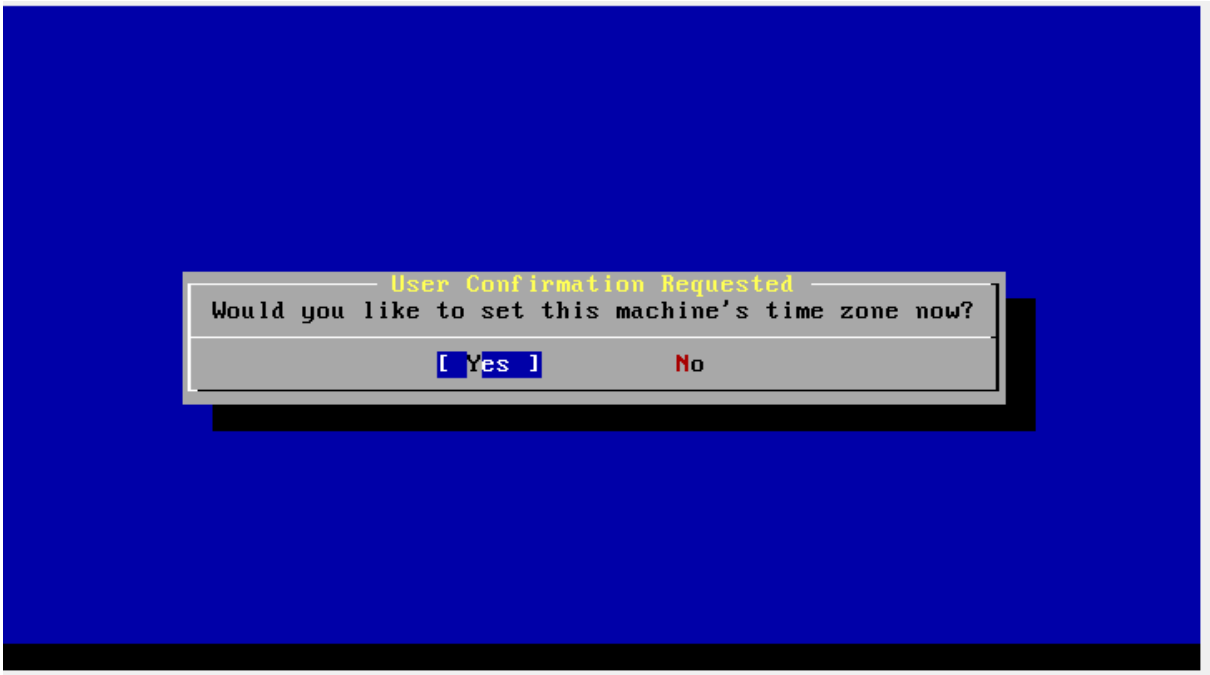


Figure C.26 : activer la zone d'heure**.**

-puisque on est UTC+1(figure C.27). Choisir Afrique (figure C.28) puis Algérie (figure C.29) :

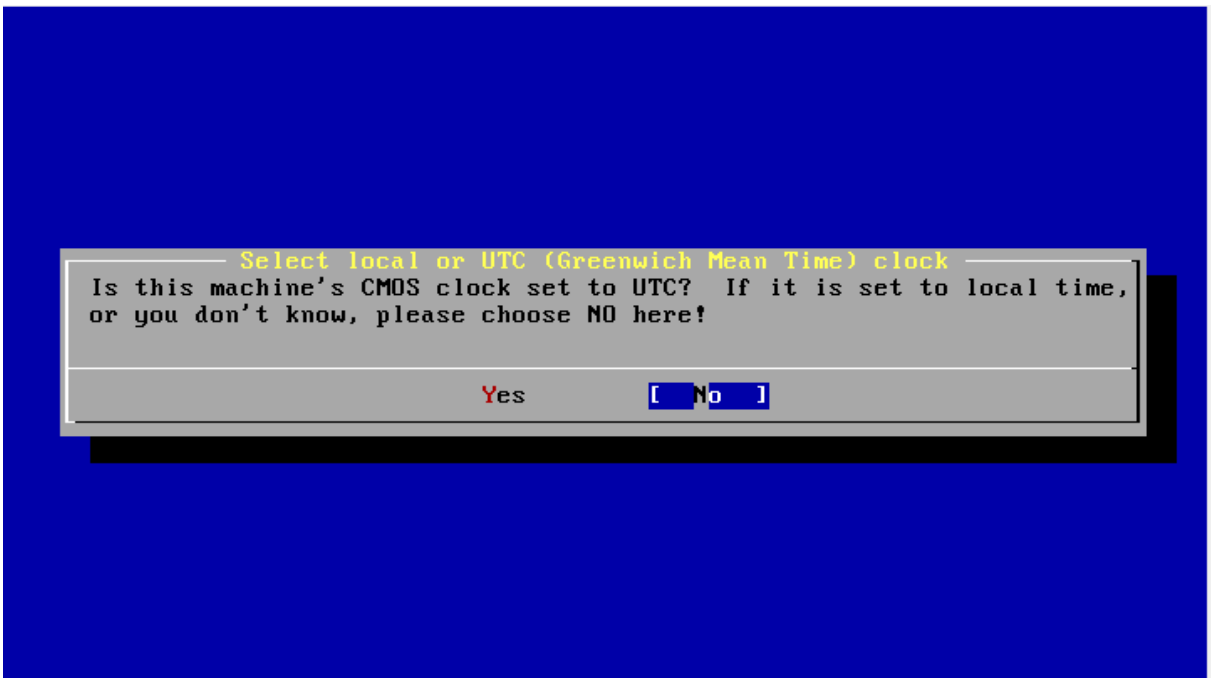


Figure 2.27 : choix de la zone d'heure**.**

Figure C.28 : choix du continent**.**

Figure C.29 : choix du pays**.**

-créer le compte " super utilisateur " (figure C.30).

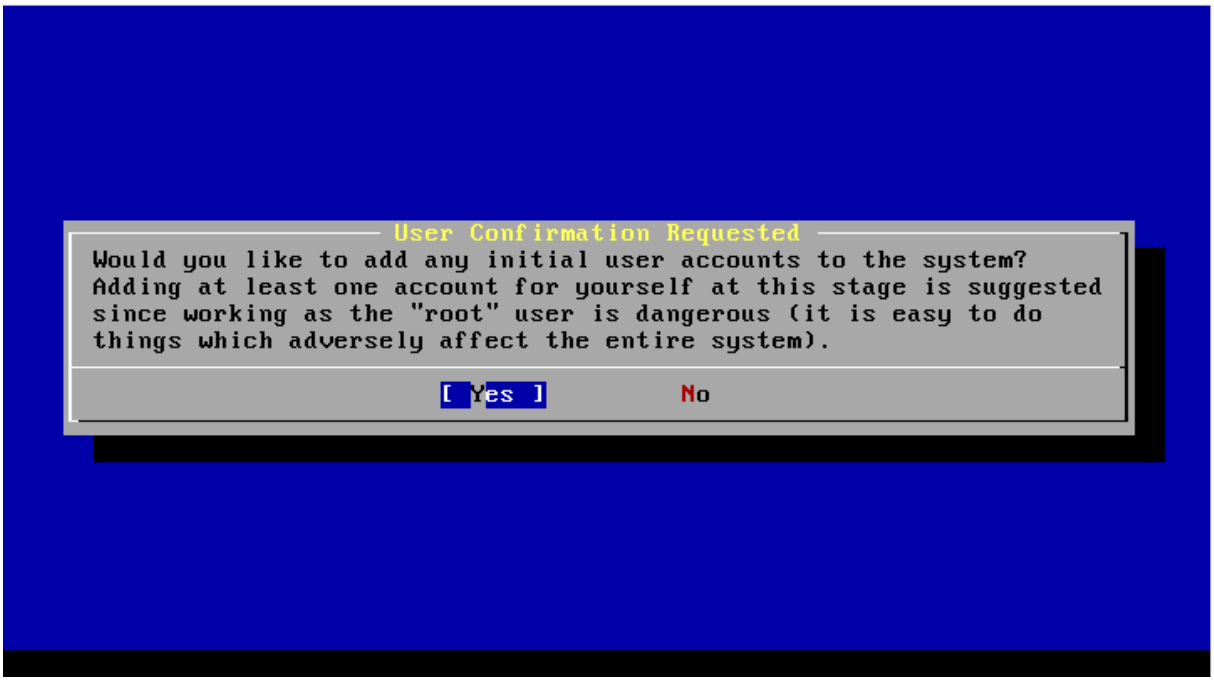


Figure C.30 : ajouter un nouvel utilisateur**.**

- pour ajouter un nouvel utilisateur on choisi **add a new user to the système (figure C.31)** :

Figure C.31 : procédure de création d'un nouveau compte utilisateur**.**

-Entrer l'identifiant de l'utilisateur (figure 2.32), un mon de passe. Il est important de mettre **Wheel** comme **member group**. C'est ce dernier qui permet d'accéder au système en tant que root :

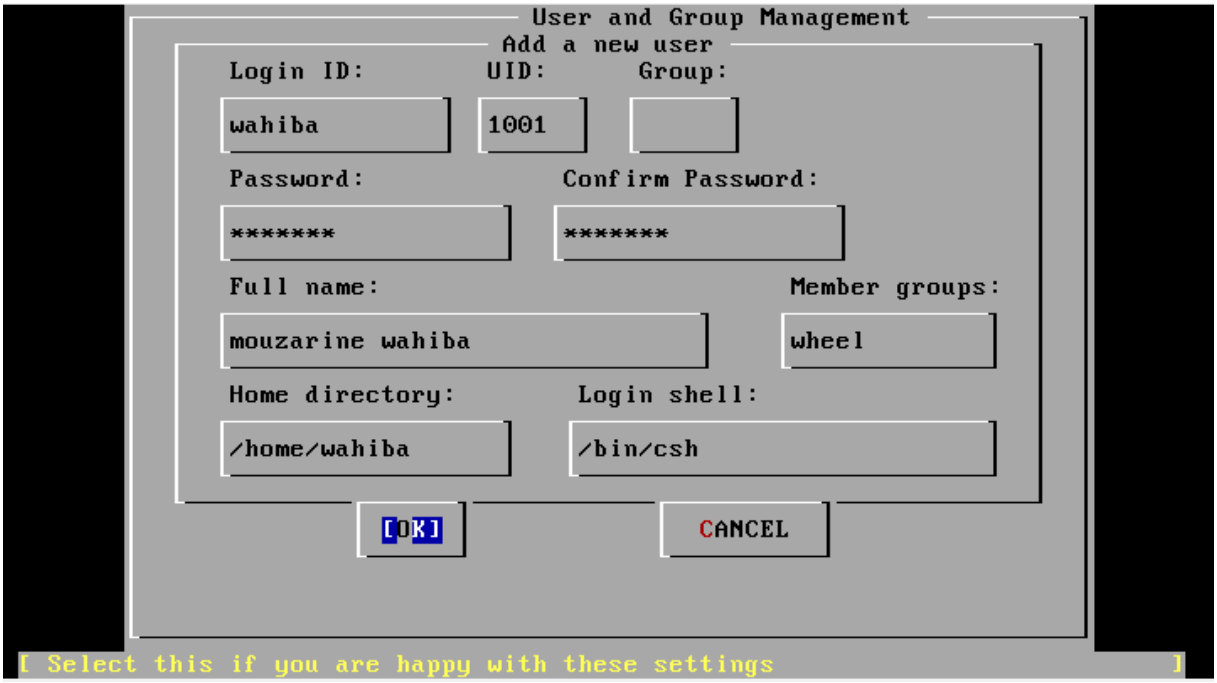


Figure C.32 : remplissage du formulaire**.**

-le programme d'installation sysinstall nous propose de revoir une dernière fois notre configuration (figure C.33) :

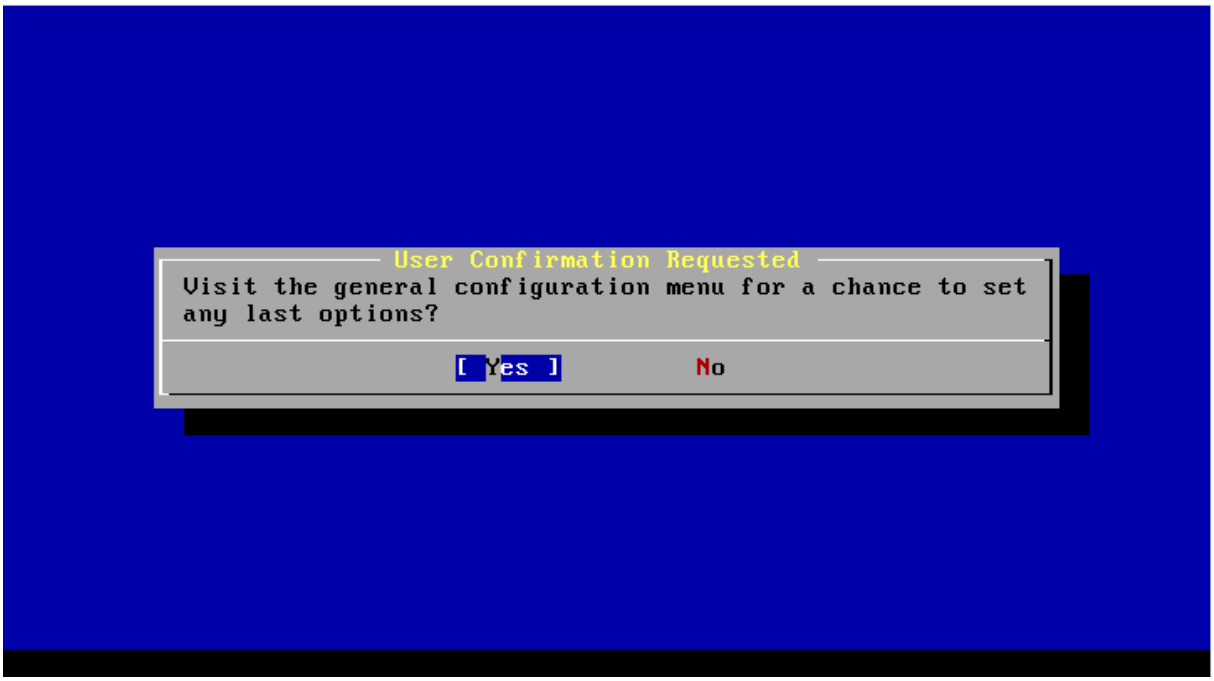


Figure C.33 : revoir la configuration générale

-Puis il nous demander d'entrer le mot de passe de compte root (figure C.34 et figure C.35) :

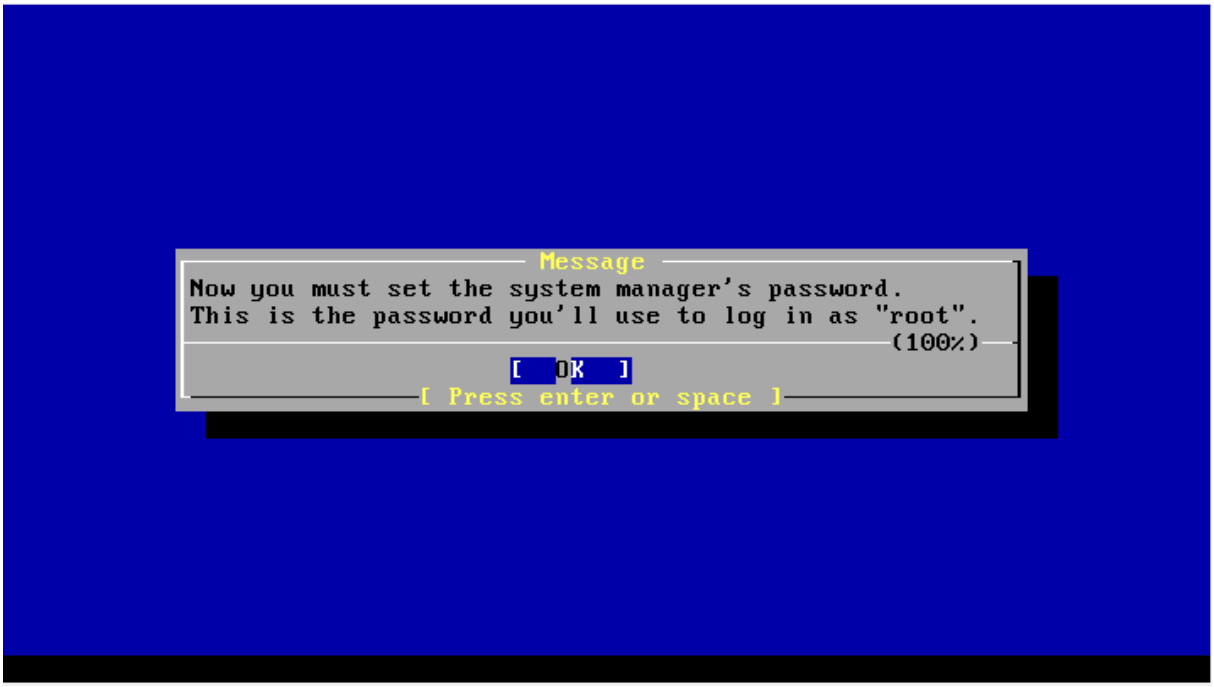


Figure C.34

Figure C.35 : insertion du mot de passe pour le compte root.

L'installation standard étant terminée. On appui sur **Exit Install** (figure C.36) :

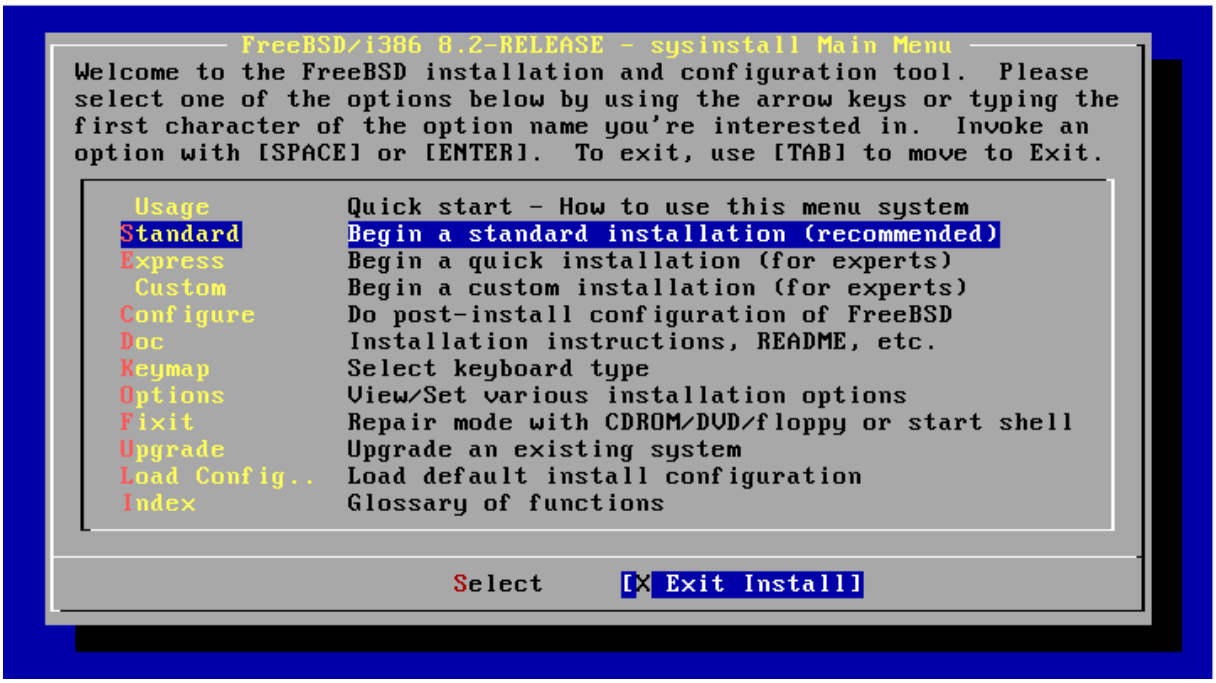


Figure C.36 : quitter sysinstall**.**

- sysinstall nous averti du fait que le système va rebooter choisir **yes** (figure 2.37)**:**

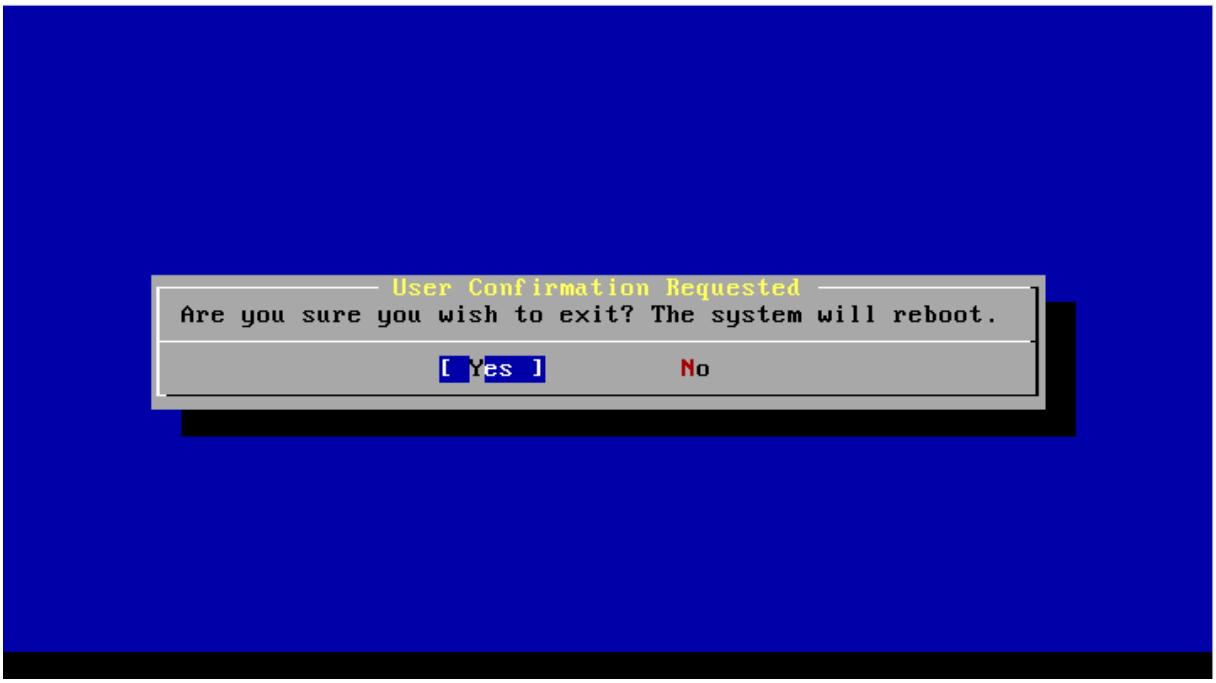


Figure C.37 : message d'avertissement sur le redémarrage**.**

-Sysinstall nous demande ainsi de retirer le media d'installation (figure C.38) :

Figure C.38 : sysinstall demande de retirer le media d'installation**.**

-pour le faire il faut aller à **périphériques lecteur CD /DVD Ejecter du disque du lecteur virtuel** (figure 2.39) si nous travaillant sur VirtualBox ou éjecter directement le CD /DVD du lecteur si on travail sur un support physique réel**.**

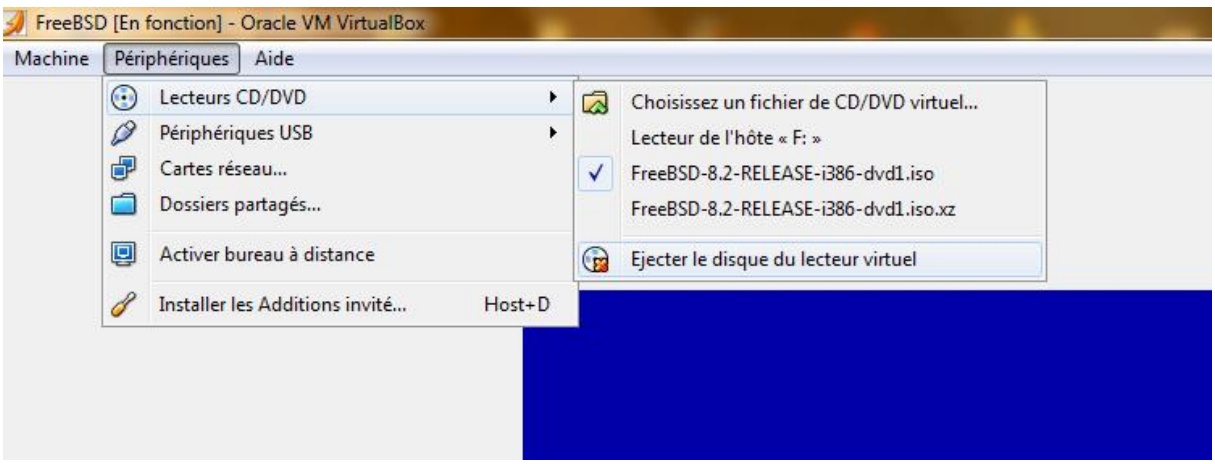


Figure C.39 : éjecter le media**.**

Cisco Systems
Cisco 2811 Integrated Services Router - Routeur - DSL - EN, Fast EN - Cisco IOS 12.3(8)T - 1U

Cisco Systems révolutionne le routage pour entreprise et PME/PMI avec une nouvelle gamme de routeurs à services intégrés spécifiquement conçus pour délivrer à la vitesse du câble et de manière sécurisée des services convergents vidéo, voix, données. Fruit de vingt années d'innovation et d'expertise dans le domaine des technologies Internet, la gamme de routeurs à services intégrés Cisco 2800 intègre de façon intelligente au sein d'un même système de routage, des services de téléphonie, de routage multi protocole et de sécurité d'entreprise. Une solution complètement intégrée, parfaitement adaptée aux besoins des entreprises en termes de services, de performances et de sécurité. Grâce à une nouvelle architecture interne qui intègre de façon native la sécurité, la gamme Cisco 2800 apporte aux entreprises une agilité business maximale tout en protégeant leurs investissements.

Description du produit	Cisco 2811 Integrated Services Router - routeur
Type de périphérique	Routeur
Facteur de forme	Externe - modulaire - 1U
Dimensions (LxPxH)	43.8 cm x 41.7 cm x 4.5 cm
Poids	6.4 kg
RAM	256 Mo (installé) / 760 Mo (maximum)
Mémoire flash	64 Mo (installé) / 256 Mo (maximum)
Protocole de liaison de données	Ethernet, Fast Ethernet
Protocole réseau / transport	IPSec
Protocole de gestion à distance	SNMP 3
Protocole de signal numérique	ADSL
Caractéristiques	Cisco IOS 12.3(8)T , protection par firewall, chiffrement matériel, VPN, prise en charge de MPLS, filtrage de l'URL
Conformité aux normes	IEEE 802.3af

Spécifications détaillées

Général

Type de périphérique	Routeur
Facteur de forme	Externe - modulaire - 1U
Largeur	43.8 cm
Profondeur	41.7 cm
Hauteur	4.5 cm
Poids	6.4 kg

Mémoire

RAM	256 Mo (installé) / 760 Mo (maximum)
Mémoire flash	64 Mo (installé) / 256 Mo (maximum)

Réseaux

Technologie de connectivité	Filaire
Protocole de liaison de données	Ethernet, Fast Ethernet
Protocole réseau / transport	IPSec
Protocole de gestion à distance	SNMP 3
Caractéristiques	Protection par firewall, chiffrement matériel, VPN, prise en charge de MPLS, filtrage de l'URL

Conformité aux normes	IEEE 802.3af

Communications

Type	Modem DSL
Protocole de signal numérique	ADSL

Extension / connectivité

Nombre total de connecteurs d'extension (disponibles)	1 (1) x NME 4 (3) x HWIC 2 (2) x AIM 2 (2) x PVDM Mémoire 1 Carte CompactFlash
Interfaces	2 x réseau - Ethernet 10Base-T/100Base-TX - RJ-45 2 x USB 1 x gestion - console 1 x série - auxiliaire 1 x modem - ADSL

Divers

Algorithme de chiffrement	DES, Triple DES, AES
Conformité aux normes	CISPR 22 Class A, CISPR 24, EN 60950, EN 61000-3-2, VCCI Class A ITE, IEC 60950, EN 61000-3-3, EN55024, EN55022 Class A, UL 60950, EN50082-1, CSA 22.2 No. 60950, AS/NZ 3548 Class A, JATE, FCC Part 15, ICES-003 Class A, CS-03, EN 61000-6-2

Alimentation

Périphérique d'alimentation	Alimentation - interne

Logiciels / Configuration requise

Système d'exploitation fourni	Cisco IOS 12.3(8)T

Caractéristiques d'environnement

Température de fonctionnement mini	0 °C
Température de fonctionnement maxi	40 °C
Taux d'humidité en fonctionnement	5 - 95%

Onduleurs et Périphériques d'alimentation

PWR-2811-AC=	Cisco - Alimentation

RAM

MEM2811-256D=	Cisco - Mémoire - 256 Mo - DDR

Mémoire flash

MEM2800-64CF=	Cisco - Carte mémoire flash - 64 Mo - CompactFlash Card

Modems

VIC-2DID=	Cisco - Carte d'interface vocale - module enfichable - Logement pour extension / 2 port(s) analogique (s)
WIC-1DSU-T1-V2=	Cisco - DSU/CSU - module enfichable - 1.5 Mbits/s - T-1
HWIC-4SHDSL=	Cisco High-Speed WAN Interface Card 4-pair G.SHDSL - Modem DSL - module enfichable - HWIC - 2.304 Mbits/s / 4 port(s) analogique(s)
	Cisco Interface Module 4-port ISDN-BRI - Modem (numérique) - module enfichable - 4 port(s)

NM-4B-S/T=	numérique(s)
NM-8B-S/T=	Cisco Interface Module 8-port ISDN-BRI - Modem (numérique) - module enfichable - 8 port(s) numérique(s)
WIC-1B-S/T-V3=	Cisco WAN Interface Card - Adaptateur de terminal RNIS - module enfichable
WIC-1SHDSL-V3=	Cisco WAN Interface Card - Modem DSL - module enfichable - 4.6 Mbits/s
HWIC-1ADSL=	Cisco WAN Interface Card High-Speed - Modem DSL - module enfichable - HWIC - 24 Mbits/s

Concentrateurs & Commutateurs

HWIC-4ESW=	Cisco EtherSwitch HWIC - Commutateur - 4 ports - EN, Fast EN - 10Base-T, 100Base-TX - module enfichable
HWIC-D-9ESW=	Cisco EtherSwitch HWIC - Commutateur - 9 ports - EN, Fast EN - 10Base-T, 100Base-TX - module enfichable

Périphériques réseau

HWIC-AP-G-E=	Cisco HWIC Access Point Interface Card - Borne d'accès sans fil - 802.11b, 802.11g - module enfichable

Adaptateurs réseau

NM-16A/S=	Cisco - Adaptateur série - série - 16 ports
NM-1T3/E3	Cisco - Module d'extension - ATM, HDLC, Frame Relay, PPP - T-3/E-3
WIC-2T=	Cisco - Module d'extension - Frame Relay, PPP - 2 ports
NM-HD-2V=	Cisco High Density VoiceFax Network Module - Module voix DSP
HWIC-4T=	Cisco High-Speed WAN Interface Card - Adaptateur série - HDLC, RS-232, PPP, RS-530, X.21, V.35, RS-449, SLIP, RS-530A - 4 ports
HWIC-4A/S=	Cisco High-Speed WAN Interface Card - Adaptateur série - HDLC, RS-232, PPP, RS-530, X.21, V.35, RS-449, SLIP, RS-530A - 4 ports
HWIC-16A=	Cisco High-Speed WAN Interface Card - Adaptateur série - RS-232 - 16 ports
HWIC-1FE=	Cisco High-Speed WAN Interface Card - Module d'extension - HWIC - EN, Fast EN - 10Base-T, 100Base-TX
VWIC-2MFT-E1=	Cisco Multiflex Trunk - Carte d'interface vocale - 2 ports - E-1
VWIC-1MFT-T1=	Cisco Multiflex Trunk - Module d'extension - T-1
VWIC2-2MFT-T1/E1=	Cisco Multiflex Trunk Voice/WAN Interface Card - Module d'extension - 2 ports - T-1/E-1
VWIC-1MFT-E1=	Cisco Multiflex Trunk Voice/WAN Interface Card - Module d'extension - fractional E-1/E-1
VWIC2-1MFT-T1/E1=	Cisco Multiflex Trunk Voice/WAN Interface Card 2nd Generation - Module d'extension - T-1/E-1
VWIC-2MFT-G703=	Cisco Multiflex Trunk Voice/WAN Interface Card G.703 - Module d'extension - 2 ports - E-1
VWIC-1MFT-G703=	Cisco Multiflex Trunk Voice/WAN Interface Card G.703 - Module d'extension - E-1
VWIC2-2MFT-G703=	Cisco Multiflex Trunk Voice/WAN Interface Card G.703 2nd Generation - Module d'extension - 2 ports - T-1/E-1
VWIC2-1MFT-G703=	Cisco Multiflex Trunk Voice/WAN Interface Card G.703 2nd Generation - Module d'extension - T-1/E-1
VWIC-2MFT-E1-DI=	Cisco Multiflex Trunk Voice/WAN Interface Card with Drop and Insert - Module d'extension - 2 ports - fractional E-1/E-1
WIC-1T=	Cisco WAN Interface Card 1-Port Serial - Module d'extension - RS-232, RS-530, X.21, V.35, RS-449

Câbles réseau

CAB-HD8-ASYNC=	Cisco - Câble RS-232 série - RJ-45 (M) - RJ-45 (M) - 3 m

Accessoires réseau

ACS-2811RM-23=	Cisco - Kit de montage pour rack
ACS-2811RM-19=	Cisco - Kit de montage pour rack

*** Tech Data n’est pas responsable des erreurs dans la documentation des produits.**