

République algérienne démocratique et populaire  
Ministère de l'enseignement supérieur et de la recherche scientifique

UNIVERSITE ABDERAHMANE.MIRA



Faculté de science exacte

DEPARTEMENT INFORMATIQUE

En vue de l'obtention du diplôme de master en informatique

Option : administration et sécurité des réseaux

**Thème :**

**Conception et réalisation d'un système de communication VOIP  
au sein du réseau de l'université de Bejaïa.**

Réalisé par :

M<sup>elle</sup> HAMMICHE Chahrazad.

M<sup>elle</sup> BENSAIDOUNE Naziha.

Les jurys :

Le président : M<sup>r</sup> Achroufene. Achour

Le promoteur : M<sup>r</sup> Larbi. Ali

L'examinatrice : M<sup>elle</sup> SABRI. Salima

L'examinatrice : M<sup>elle</sup> BOUADEM .Nassima

2011/2012

## *DEDICACES*

*Je dédie ce modeste travail à mes très chers parents qui  
soutenu tout au long ce travail et tout au long de mon  
parcourt d'étude, dieu les protège*

*A ma plus grande joie, mes sœurs : Lília, Sara, Amína, Sylía  
et la petite Melissa*

*A mon très cher frère Salim*

*Que dieu nous garde les uns pour les autres*

*A mes très chères copines : Hassíba, Hanane, Djemy et  
Dihía*

*Pour leur soutien et leur compréhension*

*A ma binôme et à toute sa famille*

*A toute ma famille et tous mes amis qui m'ont aidé de loin  
ou de prés*

*Et tous ceux qui m'ont réservé une place dans leurs cœurs*

*A toute la promotion 2012.*

*NAZIHA*

## *DEDICACES*

*Je dédie ce modeste travail à :*

*Mes très chers parents qui m'ont toujours soutenu.*

*Ma chère sœur et mon cher frère.*

*Mes deux grand-mères et mon grand-père.*

*A toute ma famille paternelle et maternelle.*

*A tous mes amis*

*Tout particulièrement à ma binôme zahzouha et à tahar.*

*Que dieu me les garde les protège*

*Ainsi à toute la promotion 2012*

*Chahrazad.*

## *REMERCIEMENT*

*Notre gratitude et notre grande reconnaissance pour notre promoteur Mr ALI. Larbi pour ses précieux conseils, sa patience, sa compréhension afin d'aboutir à ce travail ainsi aux membres du jury d'avoir accepté de juger notre présent travail.*

*Nos forts remerciements vont aussi à la directrice du centre du calcul et pour tout le reste du personnel, pour avoir mis à notre disposition les ressources matérielles.*

*Et enfin, nous remercions tous ceux qui ont participé de loin ou de prêt à réaliser ce modeste travail.*

## *Table des matières*

INTRODUCTION GENERALE.....	1
 CHAPITRE 1 LA VOIX ET LA TELEPHONIE SUR IP	
I. Introduction .....	2
II. La téléphonie sur IP(TOIP).....	3
1. Définition .....	3
2. Les types de la TOIP.....	3
2.1. Téléphonie PC à PC.....	3
2.2. Téléphonie entre PC et poste téléphonique.....	4
2.3. Téléphonie entre poste téléphonique.....	4
3. La signalisation.....	5
4. Le codage audio et vidéo .....	5
III. La VOIP .....	6
1. Définition de la VOIP.....	6
2. Les protocoles associés à VOIP .....	7
2.1. Le protocole IP.....	7
2.2. Les protocoles TCP/UDP.....	8
2.3. Les protocoles SIP/H.323.....	11
2.4. Les protocoles RTP/RTCP.....	12
3. Contraintes de la VOIP.....	14
4. Sécurité.....	16
5. Avantages et inconvénients de la VoIP.....	16
IV. Conclusion .....	17
 CHAPITRE II : LES PROTOCOLES DE SIGNALISATION	
I. Introduction.....	18
II. Le protocole H.323.....	18
1. Historique .....	18
2. Fonctionnement .....	19
3. Architecture et fonctionnalités du H.323.....	19
3.1. Architecture .....	19
A. Le terminal H323.....	19
B. Getkeeper .....	19
C. Gateway.....	20
D. Le MCU.....	20
3.2. La pile protocolaire .....	21
3.3. Un exemple de communication complète .....	21
4. Les avantages et les inconvénients du H .323.....	24
III. Le protocole SIP.....	25
1. Historique .....	25
2. Architecture .....	25
3. Adressage SIP.....	28
4. Message SIP.....	29
5. Le fonctionnement .....	29
6. Les services de SIP.....	33

	7. Avantages et inconvénients.....	34
	8. Sécurité .....	34
IV.	La comparaison H.323/SIP.....	35
V.	Conclusion .....	36

### Chapitre III : LA CONCEPTION DU SYSTEME DE COMMUNICATION

I.	Introduction.....	37
II.	Le langage UML.....	37
	1. Définition.....	37
	2. Diagramme d'UML.....	38
III.	System de communication.....	39
	a) Diagramme de cas d'utilisation.....	39
	b) Diagrammes de séquence.....	41
	c) Digramme de classe.....	44
IV.	Conclusion.....	45

### Chapitre IV : IMPLEMENTATION SU SYSTEME DE COMMUNICATION

I.	Introduction.....	46
II.	Asterisk.....	46
	1. Définition.....	46
	2. Historique.....	46
	3. Intérêt du choix asterisk.....	47
	4. Ses principales fonctionnalités.....	47
	5. Son architecture .....	47
III.	Les différentes étapes de réalisation de notre système de communication VoIP.48	
	1. Installation du serveur asterisk.....	48
	2. Configuration du serveur asterisk.....	48
	a) La configuration su fichier sip.conf.....	49
	b) La configuration du fichier extension .conf.....	50
	3. Les fonctionnalités implémentées.....	50
	a) Le transfert d'appel.....	51
	b) La mise en attente.....	51
	c) La messagerie vocale.....	51
	d) Le menu interactif.....	52
IV.	Le softphone.....	52
V.	Conclusion .....	57
	CONCLUSION GENERALE ET PERSPECTIVES.....	58

## Liste des abréviations

### -A-

<b>AAA</b>	Authentications, Autorization, Accounting
<b>ACF</b>	Admission ConFirm
<b>Admin</b>	Administrateur
<b>ARQ</b>	Admission ReQuest
<b>ATM</b>	Asynchronous transfer mode

### -C-

<b>CLC:</b>	Close Logical Channel
<b>CLC-ACK:</b>	Close Logical Channel-ACKnowledge
<b>CODEC</b>	COdeurDECodeur
<b>CR-LF</b>	Carriage Return-Line Feed
<b>CSRC</b>	Contributing Source

### -D-

<b>DCF</b>	Desengage ConFirm
<b>DHCP</b>	Dynamic Host Configuration Protocol
<b>DRQ</b>	Desengage ReQuest
<b>DSP</b>	Digital Signal Processors

### -G-

<b>GPL</b>	General Public License
------------	------------------------

### -I-

<b>IAP</b>	Internet Access Provider
<b>IAX</b>	Inter-Asterisk eXchange
<b>IETF</b>	Internet Engineering Task Force
<b>IHL</b>	Internet Header Length
<b>IP</b>	Internet Protocol
<b>IPX</b>	Internetwork Paquet eXchange/

**-M-**

<b>MC</b>	Multipoint Controller
<b>MCU</b>	Multipoint Control Unit
<b>MMUSIC</b>	Multi party Multimedia Session Control
<b>MOS</b>	Mean Operatinal Score
<b>MP</b>	Multipoint Processor
<b>MTA</b>	Mail Transfer Agent

**-N-**

<b>NETBEUI</b>	NETwork Basic Input-Output System Extended User Interface
<b>NAT</b>	Network Address Translation

**-O-**

<b>OLC</b>	Open Logical Channel
<b>OLC-ACK</b>	Open Logical Channel- ACKnowledge
<b>OSI</b>	Open System interconnection

**-P-**

<b>P</b>	Padding
<b>PABX</b>	Private Automatic Branch eXchange
<b>PCM</b>	Pulse Code Modulation
<b>POO</b>	Programmation Orienté Objet
<b>PSH</b>	PUSH
<b>PT</b>	Payload Type

**-R-**

<b>RAS</b>	Registration Admission Status
<b>RNIS</b>	Réseau numérique à Intégration de Services
<b>RST</b>	Rapid Spanning Tree Protocol
<b>RSVP</b>	Ressource reSerVation Protocol
<b>RTC</b>	Réseau Téléphonique Commuté
<b>RTCP</b>	Real time Transport Control Protocol
<b>RTP</b>	Real time Transport Protocol

**-S-**

**SDP** Session Description Protocol

**SIP** Session Initiation Protocol

**SPX** Sequenced Paquet eXchange

**SSRC** Synchronization Source

**-T-**

**TCP** Transmission Control Protocol

**TCS** Terminal Capacity Set

**TCS-ACK** Terminal Capacity Set-ACKnowledge

**TOIP** Telephony Over IP

**TTL** Time To Live

**-U-**

**UA** User Agent

**UAC** User Agent Client

**UAS** User Agent Server

**UDP** User Datagram Protocol

**UIT** Union Internationale de Télécommunication

**UML** Unified Modeling Language

**URG** Urgency

**-V-**

**V** Version

**VOIP** Voice Over IP

**-X-**

**X** Extension

## Liste des figures

### CHAPITRE I

<b>Figure 1</b> : Téléphonie de PC à PC en grand public.....	3
<b>Figure 2</b> : Téléphonie de PC à PC en entreprise .....	3
<b>Figure 3</b> : Téléphonie entre un PC et un centre téléphonique.....	4
<b>Figure 4</b> : Téléphonie entre postes téléphoniques.....	4
<b>Figure 5</b> : Entête du protocole IP .....	7
<b>Figure 6</b> : L'entête TCP .....	8
<b>Figure 7</b> : L'entête du protocole UDP .....	10
<b>Figure 8</b> : L'entête RTP .....	12
<b>Figure 9</b> : L'entête RTCP.....	13

### CHAPITRE II

<b>Figure 10</b> : Architecture de H.323 .....	20
<b>Figure 11</b> : La pile protocolaire de H.323.....	21
<b>Figure 12</b> : Exemple d'une communication complète .....	22
<b>Figure 13</b> : Les composants du protocole SIP .....	26
<b>Figure 14</b> : La fonction du terminal .....	26
<b>Figure 15</b> : Forma d'un message SIP .....	29
<b>Figure 16</b> : Session SIP simple .....	30
<b>Figure 17</b> : Session SIP faisant appel à un serveur proxy.....	31

### CHAPITRE III

<b>Figure 18</b> : Diagramme de cas d'utilisation des fonctions de l'Administrateur et d'un utilisateur au niveau de notre système.....	40
<b>Figure 19</b> : Diagramme de cas d'utilisation d'ajout d'un utilisateur avec succès.....	41
<b>Figure 20</b> : Diagramme de séquence pour le cas d'utilisation « appeler le serveur et demander le département d'informatique ».....	41
<b>Figure 21</b> : Diagramme de séquence « laisser un message vocal ».....	42
<b>Figure 22</b> : Diagramme de classe.....	43

### CHAPITRE IV

<b>Figure 23</b> : Architecture d'Asterisk .....	48
<b>Figure 24</b> : Fichier de configuration d'Asterisk.....	49
<b>Figure 25</b> : Menu interactif .....	52
<b>Figure 26</b> : Ekiga.....	53
<b>Figure 27</b> : Ajout d'un compte SIP.....	54
<b>Figure 28</b> : Usr1 appel usr2.....	54
<b>Figure 29</b> : L'appel en cour entre usr1 et usr2.....	55

<b>Figure 30:</b> La messagerie instantanée .....	55
<b>Figure 31:</b> Connexion au serveur vocal .....	56

**Liste des tableaux**

<b>Tableau1 : Codec audio.....</b>	<b>6</b>
<b>Tableau 2: Comparaison SIP/H.323 .....</b>	<b>35</b>

## Introduction générale

Le monde moderne est très demandeur de nouveaux services, notamment les réseaux de télécommunication qui sont utilisés dans tout les secteurs, que soit dans les entreprises, dans les laboratoires de recherche ou au niveau des universités. La téléphonie sur IP est une technologie qui s'est imposée dans le milieu de la télécommunication, grâce au large éventail de solutions offertes, elle est considérée comme une technique de pointe qui répond aux mieux aux besoins des utilisateurs, elle consiste à assurer des communications téléphoniques via un réseau IP. Elle peut être implémentée avec des plateformes de logiciels et d'équipements variées.

L'apparition récente de la transmission de la voix et de la vidéo sur IP représente une avancée technologique marquante dans le domaine du multimédia.

Notre projet permet de réaliser une application de communication VoIP au sein notre université, en effet les systèmes de communication peuvent continuer à fonctionner même si certaines parties du réseau présentent des problèmes de connectivité. Notre objectif est l'étude des protocoles de téléphonie sur IP et le développement d'une solution VOIP pour le réseau de l'université, pour cela on a suivi le plan de travail suivant :

- Le premier chapitre représente les deux techniques VOIP et TOIP ainsi que les protocoles associés soit ceux de temps réel (RTP/RTCP) ; soit ceux de signalisation (H.323/SIP).
- Le deuxième chapitre présente une étude détaillée sur les deux protocoles intervenants, à savoir ; ceux qui sont liés au transport de paquet (RTP/RTCP), et ceux qui sont liés à la signalisation (H.323 et SIP): leur fonctionnement et leur architecture.
- Le troisième chapitre aura pour but de présenter la conception de notre système de communication en utilisant UML.
- Le dernier chapitre fera l'objet de la réalisation et la mise en œuvre de notre système sous asterisk en utilisant un softphone Ekiga.

# ***CHAPITRE I***

## ***La voix et la téléphonie sur IP***

## **I-1. Introduction**

Aujourd'hui, le réseau désigné pour transporter la voix, la vidéo et les données est le réseau IP (Internet Protocol), ceci nous permet de converger vers un réseau de communication unique.

La voix sur IP (VOIP) désigne le transport de la voix sur un réseau de données, et la téléphonie sur IP (TOIP) vise le transport de la voix dans le cadre de la téléphonie, elle est bien plus, car la téléphonie c'est aussi et surtout un riche éventail de possibilités et de fonctionnalités offertes à l'utilisateur, comme la mise en attente, l'identification ou la conférence d'appels et bien d'autres services encore, devenus de nos jours indispensables.

Cette première partie vise à présenter un panorama large des mécanismes de la VoIP et ToIP et à étudier les composantes de ces deux technologies ainsi que les protocoles associés.

## **I-2. La téléphonie sur IP (ToIP)**

### **I-2.1. Définition**

La téléphonie sur IP ou la ToIP consiste à mettre en place des services téléphoniques sur un réseau IP en utilisant la technique de la voix sur IP. Les communications vocales sont alors transmises via ce réseau. Un téléphone IP doit être capable de numériser la voix pour la transmettre sur des réseaux IP, le téléphone de l'autre bout doit rassembler les paquets entrants pour interpréter la voix reçue. La téléphonie sur IP circule sur des réseaux privés ou publics [1]. Donc la ToIP est bien plus qu'un simple transport de la voix, puisqu'il devient question de définir une suite de protocoles adéquats et dédiés à la gestion de flux multimédia pour satisfaire les exigences et de faire de la ToIP une technologie dominante du monde des réseaux.

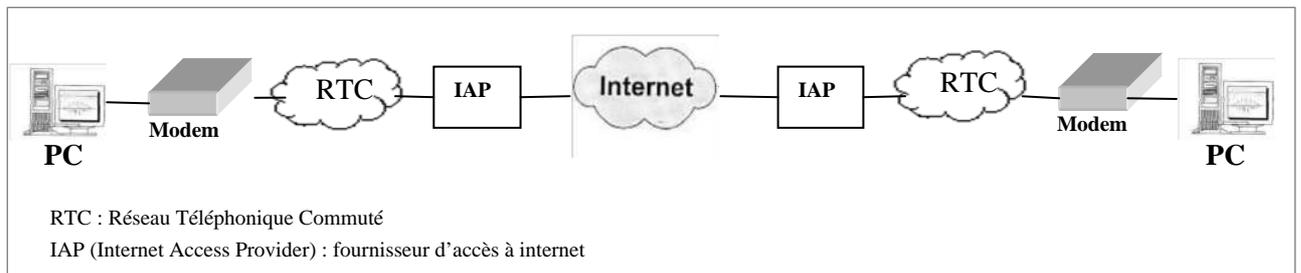
La ToIP offre une gamme de services tels que la redirection d'appel, le répondeur téléphonique et la conférence-audio. Dans ce qui suit nous détaillerons les types de la ToIP et les protocoles à mettre en œuvre.

### **I-2.2. Types de la ToIP**

Au niveau de l'utilisateur, on distingue trois scénarios pour accéder aux services, selon les équipements terminaux et les types de réseaux mis en œuvre.

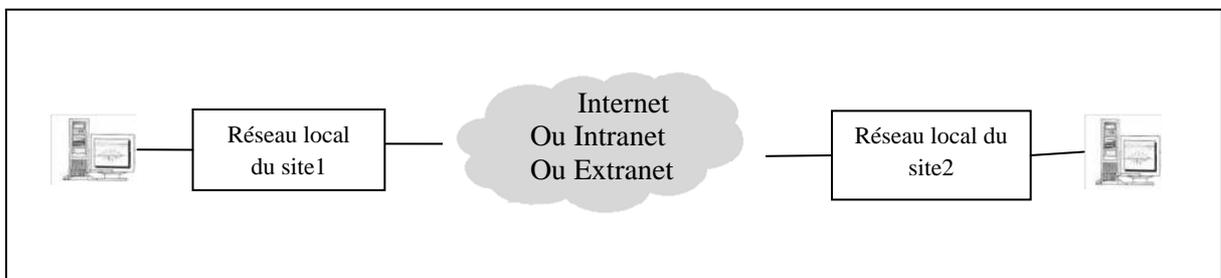
**a) PC à PC**

Dans ce cas, les correspondants utilisent des ordinateurs, chacun muni des microphones, des haut-parleurs, modems, des cartes réseaux et bien sur des logiciels de la voix sur IP compatibles de chaque coté. La voix est comprimée et décomprimée par des logiciels de compression. Dans ce mode de fonctionnement, les correspondants doivent être connectés en permanence à internet et pouvoir consulter un annuaire dynamique afin de permettre à l'appelant de connaître l'adresse de l'appelé [2], et cela est illustré dans la figure qui suit :



**Figure 1 : Téléphonie de PC à PC en grand public.**

Et dans le cas de l'entreprise on peut passer par internet, intranet, voire aussi extranet comme le définit cette figure :

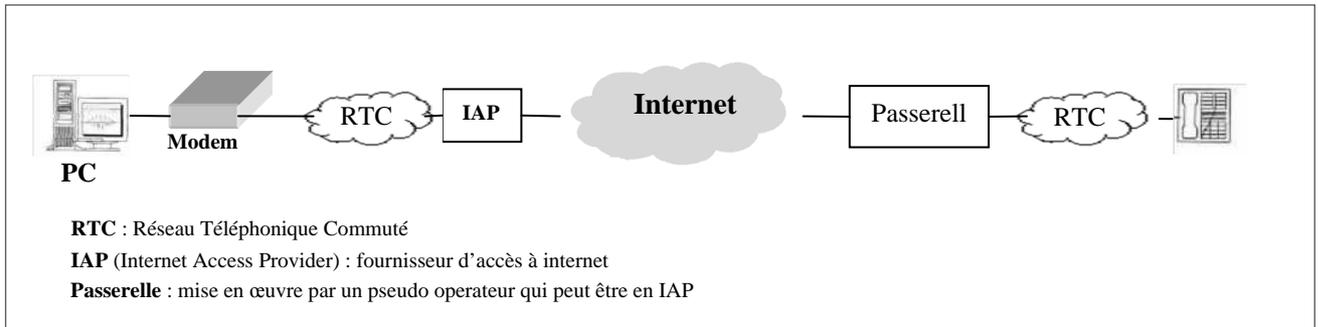


**Figure 2 : Téléphonie de PC à PC en entreprise.**

**b) Téléphonie entre un PC et un poste téléphonique**

Dans ce scenario, l'un des correspondants utilise un PC rattaché à un réseau internet par un fournisseur d'accès et l'autre utilise un téléphone rattaché à un réseau téléphonique. Une passerelle doit être mise en œuvre entre les deux réseaux (réseau internet et réseau téléphonique), elle se charge de l'appel du correspondant et de l'ensemble de la signalisation relative à la communication téléphonique du coté du correspondant demandé.

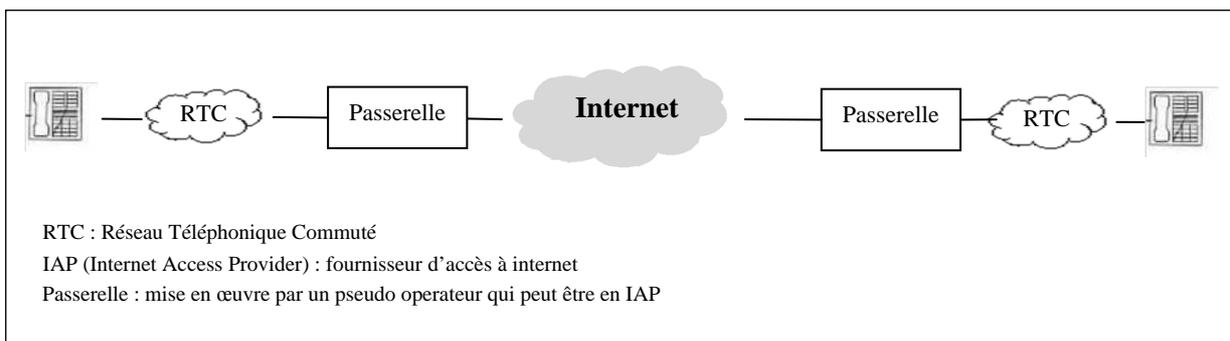
La signalisation d'appel et la négociation des paramètres de communication multimédia se font au niveau du PC [2]. La figure ci-dessous illustre ce scénario :



**Figure 3 : Téléphonie entre un PC et un centre téléphonique.**

**c) La téléphonie entre les postes téléphoniques :**

Dans ce cas, les correspondants utilisent chacun un téléphone qui est relié à un réseau téléphonique commuté (RTC). Le correspondant appelle le numéro d'une passerelle (qui se trouve entre le réseau téléphonique et le réseau internet) et lui demande le numéro du correspondant qu'il cherche à joindre. La passerelle a pour objectif d'assurer la communication entre les deux correspondants, elle permet la signalisation et la conversion de la voix IP en Voix et vice versa [2]. Ce type de téléphonie peut être illustré comme suit :



**Figure 4 : Téléphonie entre postes téléphoniques**

**I-2.3. La signalisation**

Une signalisation dans un environnement ToIP est nécessaire, pour activer et coordonner les différents éléments de cet environnement, afin qu'un appel puisse être effectué et reçu, elle sert effectivement à l'échange d'informations entre les nœuds du réseau et elle peut être gérée par plusieurs types de protocoles ; citons les plus importants : SIP et H.323 qui permettent notamment de :

- Initier et de fermer la session.
- Enregistrer les terminaux.

Dans le chapitre qui suit nous détaillerons plus profondément chacun des deux protocoles, leurs architectures, fonctionnalités et méthodes.

#### I-2.4. Le codage audio et vidéo

La voix est un signal analogique, pour que ce signal soit transporté sur le réseau IP, il doit être numérisé et compressé.

**d) La numérisation :** C'est le fait de convertir le signal analogique sous forme numérique suivant le format PCM (Pulse Code Modulation), les programmes permettant d'effectuer cette conversion sont appelés CODEC (COdeur - DECodeur). Pour numériser cette voix, nous utilisons l'échantillonnage. Cette technique consiste à capturer certains échantillons de la voix à intervalles réguliers. Plus la fréquence de capture est grande, plus la qualité est bonne.

D'une façon générale en téléphonie, la voix est échantillonnée 8000 fois par seconde (8kHz). Chaque capture étant codée sur 8 bits, cela signifie que la bande passante nécessaire est de 64 Kbits/s. Ce codec développé par l'UIT (Union Internationale des Télécommunications) est plus connu sous le nom de G.711. [3]

**a) La compression/décompression:** Après avoir numérisé le signal, ce dernier va être compressé pour diminuer le débit nécessaire. Dans la VOIP, on utilise généralement les codeurs UIT pour le codage de la parole. Les algorithmes des codec sont les plus souvent réalisés par un DSP (Digital Signal Processors) afin d'augmenter l'efficacité du transport de la voix.

La décompression est l'opération inverse de la compression. C'est-à-dire redonner au fichier compressé (donnée informatique, image, son..) sa taille initiale et les informations dans le codage initial.

Pour le codage video, les protocoles les plus utilisés sont : H261, H262, H263, H264.

Les codecs concernant le codage audio sont représentés dans le tableau qui suit :

Le score d'écoute est défini selon le critère MOS(Mean Operatinal Score).

CODEC	DEBIT	MOS
G.711 (PCM)	64KB/S	82%
G.726(ADPCM)	32KB/S	77%
G.728(LD-CELP)	16KB/S	72%
G.729(CS-ACELP)	8KB/S	78%
G.723.1(MPLQ)	6.3KB/S	78%
G.723.1(ACELP)	5.3KB/S	73%

**Tableau1 : Codec audio**

Le tableau montre que le codec classique G.711 est le plus efficace avec un débit de 64KB/S et un MOS de 82%.

### **I-3. La VoIP**

#### **I-3.1. Définition de la VOIP**

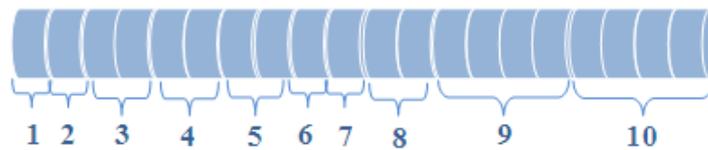
La VoIP (Voice over IP) est une technologie de communication via Internet qui permet à ses utilisateurs d'effectuer des appels sans avoir recours à des lignes téléphoniques RTC. Ce système assure une qualité de voix similaire, voire meilleure que celle du téléphone, pour un minimum de coût.

La VoIP exploite le protocole IP pour numériser, compresser et router les voix sous forme de paquets à travers Internet. C'est également ce protocole qui est utilisé pour les différentes fonctionnalités téléphoniques du système VoIP.

#### **I-3.2. Les protocoles associés à VOIP**

##### **I-3.2.1. Le protocole IP**

IP signifie "Internet Protocol", il représente le protocole réseau le plus répandu, il permet de découper l'information à transmettre en paquets, de les adresser, de les transporter indépendamment les uns des autres et de recomposer le message initial à l'arrivée. Ce protocole utilise une technique dite de commutation de paquets, il apporte, en comparaison à IPX (Internetwork Paquet eXchange)/SPX (Sequenced Paquet eXchange) et Netbeui (NETwork Basic Input-Output System Extended User Interface), l'adressage en couche 3 qui permet, par exemple, la fonction principale de routage. [4]



**Figure 5 : Entête du protocole IP**

La figure représente l'entête du protocole IP constitué de plusieurs champs que voici :

[1] : est codé sur 8 bits dont 4 bits représentent le champ Vers (version) et les 4 autres bits représentent le **champ IHL**:

- ✓ 'Le champ Vers' : représente le numéro de version du protocole IP
- ✓ 'Le champ IHL' : (Internet Header Length) représente la longueur en mots de 32 bits de l'entête IP. Par défaut, il est égal à 5 (20 octets), cependant, avec les options de l'entête IP, il peut être compris entre 6 et 15.

[2] : 'Service' : (Type Of Service) est codé sur 8 bits, il permet la gestion d'une qualité de service traitée directement en couche 3 du modèle OSI.

[3] : 'Longueur totale' : est codé sur 16 bits et représente la longueur du paquet incluant l'entête IP et les Data associées.

[4] : 'Identification' : est codé sur 16 bits et constitue l'identification utilisée pour reconstituer les différents fragments.

[5] : est codé sur 16 bits dont 3 bits pour le champ Flags et les 13 bits restants pour le **champ Position fragment**:

- ✓ 'Le champ Flags' : indique l'état de la fragmentation.
- ✓ 'Position fragment' : indique la position du fragment par rapport à la première trame

[6] : 'TTL' : (Time To Live) est codé sur 8 bits et indique la durée de vie maximale du paquet.

[7] : 'Protocole' : est codé sur 8 bits et représente le type de Data qui se trouve derrière l'entête IP.

[8] : **‘Checksum’** : est codé sur 16 bits et représente la validité du paquet de la couche 3.

[9] : **‘Adresse IP source’** : est codé sur 32 bits et représente l'adresse IP source ou de réponse.

[10] : **‘Adresse IP destination’** : est codé sur 32 bits et représente l'adresse IP destination.

### I-3.2.2. TCP (Transmission Control Protocol)/UDP (User Datagram Protocol)

Même si l'appellation courante de la modélisation n'a retenu que TCP comme protocole de transport, il existe deux protocoles distincts suivant le type de réseau utilisé. Le premier étant protocole TCP qui fonctionne en mode connecté et assure un service fiable tandis que le deuxième étant le protocole UDP qui assure un service de datagramme en mode non connecté sans aucune garantie de fiabilité.

#### a) Le protocole TCP

Le protocole TCP (Transmission Control Protocol), c'est le premier protocole de transport développé pour l'Internet. TCP intervient au niveau de la couche 4(transport) du modèle OSI [5], ses fonctionnalités sont :

- ✓ La segmentation des données des applications.
- ✓ L'établissement d'une connexion de bout-en-bout.
- ✓ L'émission des segments d'un hôte à un autre.
- ✓ L'assurance de fiabilité de transport des segments entre les hôtes connectés.

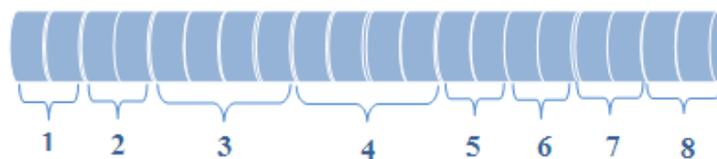


Figure 6 : L'entête TCP

La figure précédente représente l'entête du protocole TCP ainsi que les champs qui lui sont appropriés.

[1] : **‘Port source’** : est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine source.

[2] : **‘Port destination’** : est codé sur 16 bits et correspond au port relatif à l'application en cours sur la machine de destination.

[3] : **‘Numéro de séquence’** : est codé sur 32 bits et correspond au numéro du paquet. Cette valeur permet de situer à quel endroit du flux de données le paquet, qui est arrivé, doit se situer par rapport aux autres paquets.

[4] : **‘Numéro de l'accusé de réception’** : est codé sur 32 bits et définit un acquittement pour les paquets reçus. Cette valeur signale le prochain numéro de paquet attendu.

[5] : est codé sur 16 bits, ils sont répartis comme suit :

- ✓ **‘Le champ Offset’** : est codé sur 4 bits et définit le nombre de mots de 32 bits dans l'entête TCP. Ce champ indique donc où les données commencent.
- ✓ **‘Le champ Réserve’** : est codé sur 6 bits et il servira pour des besoins futurs.
  - **‘Flags’** qui représentent les champs suivants :
    - ✓ **‘Le champ URG’** est codé sur 1 bit et indique que le champ pointeur de donnée urgente est utilisé.
    - ✓ **‘Le champ ACK’** est codé sur 1 bit et indique que le numéro de séquence pour les acquittements est valide.
    - ✓ **‘Le champ PSH’** est codé sur 1 bit et indique au récepteur de délivrer les données à l'application et de ne pas attendre le remplissage des tampons.
    - ✓ **‘Le champ RST’** est codé sur 1 bit et demande la réinitialisation de la connexion.
    - ✓ **‘Le champ SYN’** est codé sur 1 bit et indique la synchronisation des numéros de séquence.
    - ✓ **‘Le champ FIN’** est codé sur 1 bit et indique fin de transmission.

[6] : **‘Le champ Fenêtre’**: (Windows) est codé sur 16 bits et correspond au nombre d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir. Le destinataire ne doit donc pas envoyer les paquets après Numéro de séquence + Windows.

[7] : **‘Le champ Checksum’**: est codé sur 16 bits et représente la validité du paquet de la couche 4 TCP.

[8] : **‘le champ pointeur’**: est codé sur 16 bits et indique la position d’une données urgente, il doit pointer sur l’octet suivant la donnée urgente.

### b) Le protocole UDP

Le protocole UDP (User Datagram Protocol) intervient dans la couche 4 (transport) du modèle TCP/IP, non orienté connexion. Ce protocole est très simple étant donné qu’il ne fournit pas de contrôle d’erreurs .Il permet aux applications d’accéder directement à un service de transmission de Datagrammes rapide.

UDP est utilisé pour transmettre de faibles quantités de données où le coût de la création de connexions et du maintien de transmissions fiables s’avère supérieur aux données à émettre. UDP peut également être utilisé pour les applications satisfaisant à un modèle de type "question réponse". La réponse étant utilisée comme un accusé de réception à la question. Il est aussi utilisé dans un second cas, tel que la voix sur IP. L’envoi en temps réel est primordial, donc si une trame n’arrivait pas, la retransmission serait inutile. [6]



**Figure 7 : L’entête du protocole UDP**

La figure montre les différents champs du protocole UDP et leur signification est comme suit :

[1] : **‘Port source’**: est codé sur 16 bits et correspond au port relatif à l’application en cours sur la machine source.

[2] : **‘Port destination’**: est codé sur 16 bits et il correspond au port relatif à l’application en cours sur la machine de destination.

[3] : **‘Longueur’**: est codé sur 16 bits et il représente la taille de l’entête et des données.

[4] : 'Checksum' : est codé sur 16 bits et représente la validité du paquet de la couche 4 UDP.

### **I-3.2.3. Les protocoles H.323 et SIP :**

#### **a) Le protocole H.323 :**

La norme H.323 englobe les protocoles de communications audio, vidéo et de transmission de données à travers les réseaux à commutation de paquets IP : réseaux LAN, Intranet, Extranet et Internet. Elle a été développée par l'ITU qui a approuvé les spécifications H.323 depuis 1996.

Cette norme couvre largement tout type de dispositifs, allant des studios de visioconférence autonomes jusqu'aux ordinateurs personnels aussi bien en mode point à point qu'en conférences multipoints.

#### **b) Le protocole SIP :**

Session Initiation Protocol (SIP) est un protocole normalisé et standardisé par l'IETF qui a été conçu pour établir, modifier et terminer des sessions multimédia. Il se charge de l'authentification et de la localisation des multiples participants. Il se charge également de la négociation sur les types de média utilisables par les différents participants en encapsulant des messages SDP (Session Description Protocol).

On verra plus en détail ces deux protocoles dans le chapitre suivant.

### **I-3.2.4. Les protocoles RTP (Real-time transport Protocol)/RTCP (Real-Time controle Protocol)**

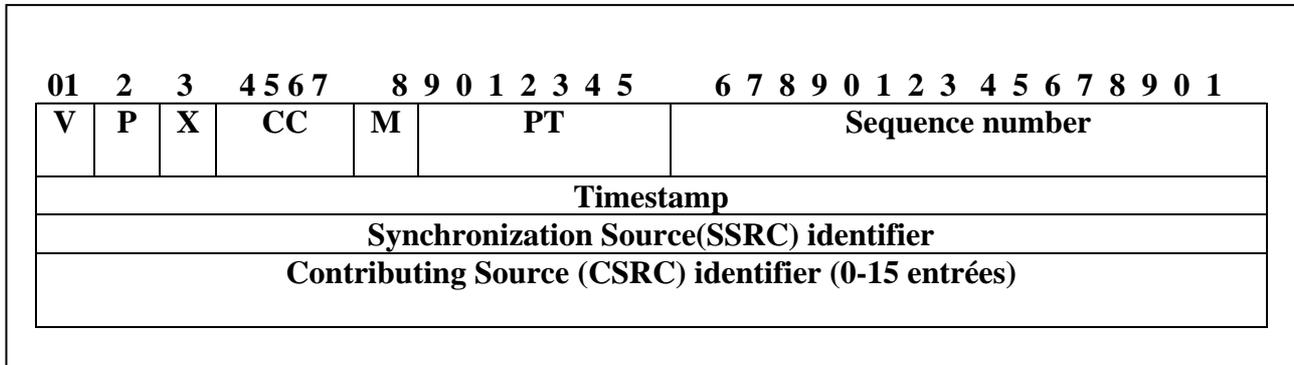
Pour le transport de données temps réel telles que la voix ou la vidéo, il est nécessaire d'utiliser deux protocoles supplémentaires : RTP et RTCP.

RTP et RTCP sont des protocoles qui se situent au niveau de l'application et s'appuient sur le protocole de transport UDP. RTP et RTCP peuvent utiliser aussi bien le mode Unicast (point à point) que le mode Multicast (multipoint). [7]

#### **a) Le protocole RTP :**

RTP est un protocole qui a été développé par l'IETF afin de faciliter le transport temps réel de bout en bout des flots de données audio et vidéo sur les réseaux IP. Le rôle principal de RTP consiste à mettre en œuvre des numéros de séquence de paquets IP pour reconstituer les informations de voix ou vidéo même si le réseau sous-jacent change l'ordre des paquets. Ce dernier permet :

- ✓ D'identifier le type de l'information transportée.
- ✓ D'ajouter des marqueurs temporels et des numéros de séquence à l'information transportée.
- ✓ De contrôler l'arrivée à destination des paquets.



**Figure 8 : L'entête RTP**

‘Version (V)’ (2 bits) : indique le numéro de version RTP utilisée.

‘Le champ Padding (P)’ (1 bit) : si P est égal à 1, le paquet contient des octets additionnels de bourrage (Padding) pour finir le dernier paquet.

‘Le champ Extension X’ (1 bit) : si X=1 l'en-tête est suivie d'un paquet d'extension.

‘Le champ CSRC count CC (4bit)’ : contient le nombre de CSRC qui suivent l'entête

‘Marker M’ (1 bit) : son interprétation est définie par un profil d'application (profile).

‘Le champ Payload Type PT’ (7 bits) : ce champ identifie le type du payload (audio, vidéo, image, texte, html, etc.).

‘Le champ séquence number’ (16 bits) : sa valeur initiale est aléatoire et il s'incrémente de 1 à chaque paquet envoyé, il peut servir à détecter des paquets perdus.

‘Le champ timestamp’ (32 bits) : reflète l'instant où le premier octet du paquet RTP à été échantillonné.

‘Le champ SSRC’ (32 bits) : identifie de manière unique la source, sa valeur est choisie de manière aléatoire par l'application.

‘Le champ CSRC’ (32 bits) : identifie les sources contribuant.

### b) Le protocole RTCP :

Le protocole RTCP (Real-Time Control Protocol) est fondé sur la transmission périodique de paquets de contrôle à tous les participants d'une session. C'est le protocole UDP qui permet le multiplexage des paquets de données RTP et des paquets de contrôle RTCP. Le protocole RTP utilise le protocole RTCP pour surveiller la qualité des services offerts et à fournir des informations concernant les partenaires de la conversation.



**Figure 9: L'entête RTCP**

La figure représente l'en-tête du protocole RTCP et la signification des champs est comme suit :

[1] : est codé sur 8 bits qui sont répartis comme suit :

- ✓ **Le champ V** : est codé sur 2 bits, permet d'indiquer la version de RTP, qui est la même que dans les paquets RTCP. Actuellement,  $V=2$ .
- ✓ **Le champ P** : indique que, s'il est à 1, les données possèdent une partie de bourrage.
- ✓ **Le champ RC** : est codé sur 5 bits, indique le nombre de blocs de rapport de réception contenus en ce paquet. Une valeur de zéro est valide.

[2] : **le champ PT** : est codé sur 1 octet, est fixé à 200 pour identifier ce datagramme RTCP comme SR.

[3] : **Longueur** : est codé sur 2 octets, représente la longueur de ce paquet RTCP incluant l'entête et le bourrage.

[4] : **SSRC** : Basé sur 4 octets, il représente l'identification de la source pour le créateur de ce paquet SR.

### **I-3.3. Contraintes de la VOIP**

La qualité de service d'un réseau désigne sa capacité de transport des données dans de bonnes conditions, mais il existe quelques aspects qui viennent perturber cette dernière.

Nous citons :

- ✓ La latence
- ✓ La perte de paquets
- ✓ La gigue

#### **I-3.3.1. La latence**

Le temps de latence, c'est le temps qui s'écoule entre l'entrée d'un paquet dans le réseau et son temps de sortie du réseau (le temps de propagation). Pour une bonne qualité de conversation il ne faut pas que la latence soit supérieure à 150 ms.

On rencontre aussi un autre problème qui est la perception d'écho. Un écho est un signal qui revient dans l'oreille de l'émetteur, ce dernier qui repart en sens inverse est numérisé par un codec (codeur/décodeur) et traverse sans problème un réseau numérique. La valeur normalisée de la latence de l'écho est de 56 ms, pour que l'écho ne soit pas gênant à l'oreille, il ne faut pas que le temps de transit de la communication dépasse 28 ms dans un sens.

#### **I-3.3.2. La perte de paquets**

C'est un problème lié au réseau IP, il est d'autant plus important quand on n'utilise pas de protocole de transport tel que TCP.

Lorsque les buffers du réseaux IP sont encombrés , ils libèrent automatiquement de la bande passante en se débarrassant de certaines proportions des paquets entrant, ce qui permet d'envoyer un signal aux terminaux TCP, qui diminuent d'autant leur débit d'après les acquittements négatifs émis par le destinataire qui ne reçoit plus les paquets.

Mais pour les paquets de voix, qui sont véhiculés au dessus d'UDP, aucun mécanisme de contrôle de flux ou de retransmission des paquets perdus n'est offert au niveau du transport. D'où l'importance des protocoles RTP et RTCP qui permettent de déterminer le taux de perte de paquet, et d'agir en conséquence au niveau applicatif.

Si aucun mécanisme performant de récupération des paquets perdus n'est mis en place, alors la perte de paquet IP se traduit par des ruptures au niveau de la conversation Cette dégradation est bien sûr accentuée si chaque paquet contient un long temps de parole.

Par ailleurs, les codeurs à très faible débit sont généralement plus sensibles à la perte d'information, et mettent plus de temps à reconstruire un codage fidèle.

### **I-3.3.3. La gigue**

La gigue n'est autre que la variation de délais de transmission. Ce dernier peut être constant ce qui va préserver la synchronisation du signal entre l'émetteur et le récepteur.

En d'autres termes, elle mesure la variation temporelle entre le moment où deux paquets auraient dû arriver et le moment de leur arrivée effective. Cette irrégularité d'arrivée des paquets est due à de multiples raisons nous citons : l'encapsulation des paquets IP dans les protocoles supportés, la variation des chemins empruntés dans le réseau...etc.

La dégradation de la qualité de service due à la présence de gigue, se traduit par une combinaison des deux facteurs cités précédemment: le délai et la perte de paquets; puisque d'une part on introduit un délai supplémentaire de traitement lorsque l'on décide d'attendre les paquets qui arrivent en retard, et que d'autre part on finit tout de même par perdre certains paquets lorsque ceux-ci ont un retard qui dépasse le délai maximum autorisé par le buffer.

### **I-3.4. Sécurité :**

Vue que la VOIP est un service du réseau IP alors les menaces liées à la sécurité sont celle du réseau IP plus celle du réseau vocal. L'implémentation d'une solution VOIP fait appel à des éléments supplémentaires tel que : Gateway, proxy, serveur d'enregistrement, terminaux ...etc.

La VOIP comprend deux types de vulnérabilités principales. La première est celle directement dépendante des protocoles utilisés dans l'implémentation et la deuxième est associées au système d'exploitation. Chaque protocole (H323, SIP) ou service a ses propres vulnérabilités. A ces deux types de vulnérabilités peuvent s'ajouter les vulnérabilités d'infrastructure et humaines.

### I-3.5. Avantages et inconvénients de la VoIP

#### ❖ **Avantage :**

- Minimiser les couts : Cette technologie mène à la plus grande épargne financière. Ceci se produit parce qu'il existe un seul réseau portant la voix et les données fournies par seulement un fournisseur.
- La portabilité : on peut passer et recevoir des appels de téléphone partout où il y a un raccordement à bande large. Ceci rend VoIP aussi pratique que l'email.
- Dispositifs : À la différence du service de téléphone normal qui charge habituellement plus pour les dispositifs supplémentaires, VOIP vient avec une foule de dispositifs avancés de communication, par exemple, le renvoi d'appel, l'attente d'appel, le courrier vocal.
- Flexibilité : possibilité d'utiliser un numéro de téléphone dans n'importe quelle place au monde et même d'utiliser d'autres indicatifs.
- vidéoconférence : VoIP permet des possibilités avancées de largeur de bande et la vidéoconférence améliorée.

#### ❖ **Inconvénients :**

- Aucun service pendant une coupure électrique : Pendant une panne d'électricité un téléphone normal est maintenu dans le service par le courant fourni par la ligne téléphonique. Ce n'est pas possible avec des téléphones d'IP.
- Fiabilité : la VOIP se fonde sur une connexion internet, donc la qualité de VOIP est affectée par la qualité et la fiabilité du service d'Internet à bande large et parfois par les limitations du PC. Les connexions internet et la congestion pauvres peuvent avoir comme conséquence la qualité de voix déformée ou tordue.
- Qualité de voix de VoIP : la qualité du service de la VOIP dépend de tant de facteurs : le raccordement à bande large, le matériel, le service fourni par le fournisseur, la destination de votre appel etc.

**I.4. Conclusion :**

Dans ce chapitre nous avons présenté les deux mécanismes de transfert de la voix sur IP: la ToIP et la VoIP , nous avons également vu la façon dont la voix est traitée (numérisation, codage, compression, réassemblage....) et aussi les protocoles associés à la VoIP tel que les protocoles de contrôle en temps réel comme RTP, RTCP, et brièvement les protocoles de signalisation comme H.323 et SIP que nous allons voir en détail dans le chapitre suivant.

# ***CHAPITRE II***

## ***Les protocoles de signalisation***

## II-1. Introduction

Le transport de la voix nécessite certaines exigences pour pouvoir surpasser du trafic traditionnel du réseau voix (réseau à commutation de circuits). Le facteur le plus décisif est la contrainte temporelle, donc il faut bien penser à implémenter un mécanisme de signalisation permettant d'assurer la connexion entre les utilisateurs.

Plusieurs organismes ont participé à l'élaboration de plusieurs normes, les plus utilisées de nos jours sont le H.323 par l'UIT (Union Internationale des Télécommunications) dans le milieu de télécommunication, et le SIP (Session Initiation Protocol) par l'IETF (Internet Engineering Task Force) dans le milieu informatique.

## II-2. Le protocole H323

### II-2.1. Historique

L'année 1996 a été l'année de transition et un tournant important pour l'essor de la VoIP, c'est à partir de là qu'elle commença à prendre un nouveau départ. L'handicap majeur qui l'a empêché avant réside dans l'incompatibilité des protocoles utilisés dans ce secteur, par exemple lorsque deux utilisateurs distincts utilisent des normes de communication différentes, il est impraticable que les deux utilisateurs utilisant ces deux réseaux différents de communiquer. C'est à partir de cet ennui que l'UIT (Union Internationale de Télécommunication) commença à doubler ses efforts et proposa une famille de protocoles H32x, le plus utilisé dans la VoIP c'est le H323, il a pu convaincre tous les acteurs du multimédia.

En janvier 1998, l'UIT-T a prévu pour la version 2 en améliorant les fonctions du H323, en particulier les délais d'établissement d'une communication. [B1]

### II-2.2. Fonctionnement

Le protocole H323 est un support d'actions réciproques en temps réel, entre autre la visioconférence (signalisation, l'enregistrement, contrôle d'admission, négociation, transport et codage). C'est le leader du marché de la VoIP. Il est conçu pour des applications audio et vidéo sur des réseaux locaux à commutation de paquets. Les principales fonctions destinées au protocole H323 sont :

- ✓ Etablissement, contrôle et suivi d'appel.
- ✓ Gestion des flux multimédias (liste de codec à utiliser).
- ✓ Gestion des conférences multipoint.

- ✓ Gestion de la bande passante.
- ✓ Interconnexion à d'autres réseaux : ATM, RNIS

Afin d'établir une communication H323, il est indispensable de passer par les phases suivantes :

- ✓ Signalisation et établissement d'appel par le protocole Q.931.
- ✓ Echange de capacités et de performances, plus la réservation de la bande passante par le protocole RSVP (Ressource reSerVation Protocol).
- ✓ Etablissement d'une communication audio-visuel.
- ✓ Invocation des services.
- ✓ Libération de l'appel.

### II-2.3. Architecture et fonctionnalités du H323 :

#### II-2.3.1. Architecture :

L'architecture H323 est considérée comme fortement centralisée, elle s'articule sur quatre entités interagissantes dans un réseau en mode de paquet: le terminal H323, Getkeeper, Gateway et le MCU. Dans ce qui suit nous détaillerons chacun de ces composants ainsi que leurs tâches.

##### a) Le terminal H323 :

C'est un mécanisme multimédia permettant la communication en temps réel, on distingue deux types de terminal: matériel (codec, PC, micro et divers accessoires) et logiciel (NetMeeting dans l'environnement Windows). Un terminal H323 doit être muni des protocoles suivants :

- **H225** : Pour la signalisation de l'appel, la synchronisation, la mise en paquets des données et l'enregistrement auprès d'un Getkeeper.
- **H245** : Pour la négociation des codecs ainsi l'ouverture et la fermeture des canaux.
- **RAS (Registration/Admission/Status)** : Pour la communication avec le Getkeeper, il permet le contrôle d'admission et la gestion de la bande passante.
- **RTP/RTCP** : Pour l'enchaînement des paquets audio.

##### b) Getkeeper :

Les utilisateurs dans le réseau peuvent s'identifier entre eux par des noms auxquels il faut attribuer des adresses IP, le Getkeeper assure cette traduction d'adresses afin de localiser les abonnés dans le réseau IP. Dans le cas où deux utilisateurs se trouvent sur des réseaux différents, le Getkeeper localise l'entité intermédiaire à joindre pour établir l'appel. La

présence du Getkeeper dans un réseau nécessite que tous les terminaux s'y enregistrent et y sollicitent l'autorisation d'effectuer des appels. Plusieurs fonctionnalités sont rajoutées au Getkeeper, les plus avantageux :

- Le contrôle d'admission en vérifiant la bande passante.
- AAA (Authentication, Autorization, Accounting) : L'authentification pour connaître l'identité de l'utilisateur, l'autorisation pour les droits attribués aux connectés.
- Gestion de flux ou de la bande passante (allocation, limitation et rejet des abonnés)

**c) Gateway (passerelle) :**

C'est un équipement permettant à des utilisateurs du réseau IP de joindre d'autres utilisateurs sur des réseaux non-IP comme le RTC, ATM, RNIS.

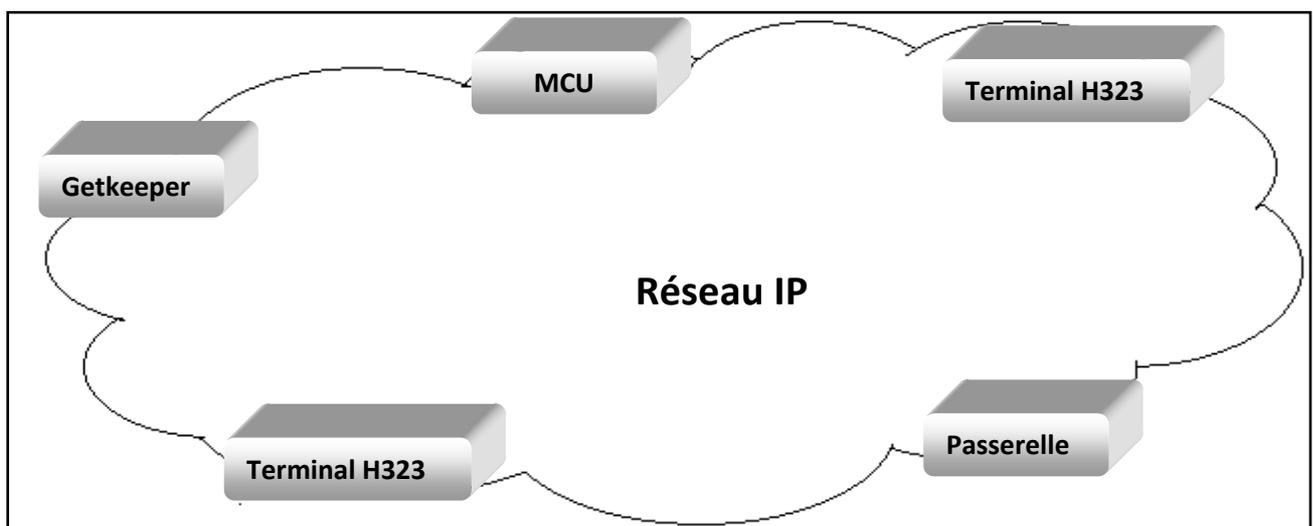
**d) le MCU (Multipoint Control Unit)**

Unité de contrôle multipoint, c'est une station sur le réseau qui permet à plusieurs interlocuteurs de participer à des conférences, par conséquent il permet :

- La gestion des communications multimédia.
- Négociation avec les terminaux de différents codecs à utiliser.
- Contrôle de flux de données.

Le MCU se compose d'un contrôleur obligatoire MC (Multipoint Controller) et des processeurs optionnels MP (Multipoint Processor), le MC pour la signalisation et le MP dédié à la transmission proprement dite.

On peut résumer l'architecture de H.323 dans la figure suivante :



**Figure 10: Architecture de H.323**

II-2.3.2. La pile protocolaire :

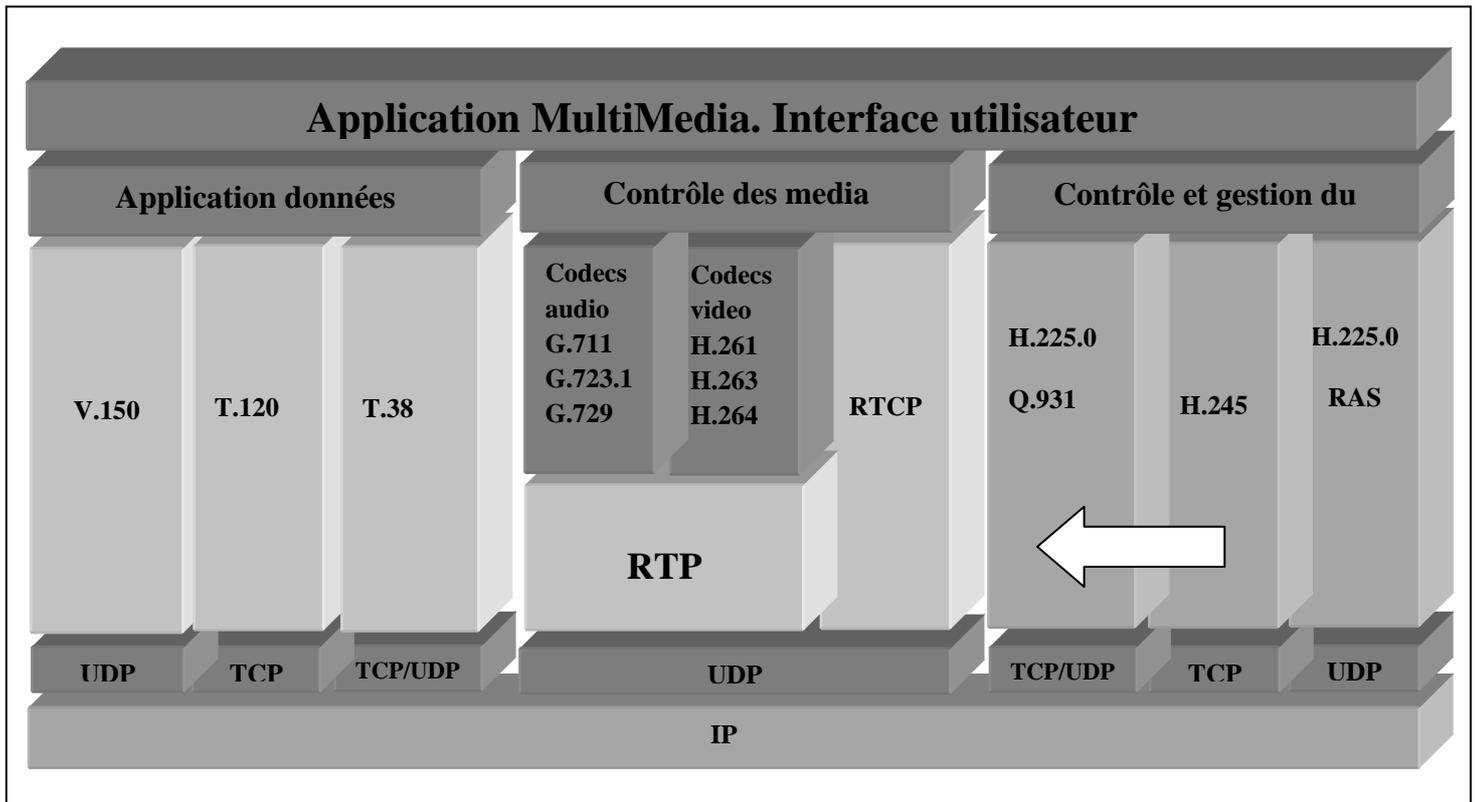


Figure 11 : La pile protocolaire de H.323

Cette figure illustre la pile protocolaire de H.323.

II-2.3.3. Un exemple d’une communication complète :

Le schéma suivant illustre les messages échangés entre deux terminaux appelant A et appelé B.

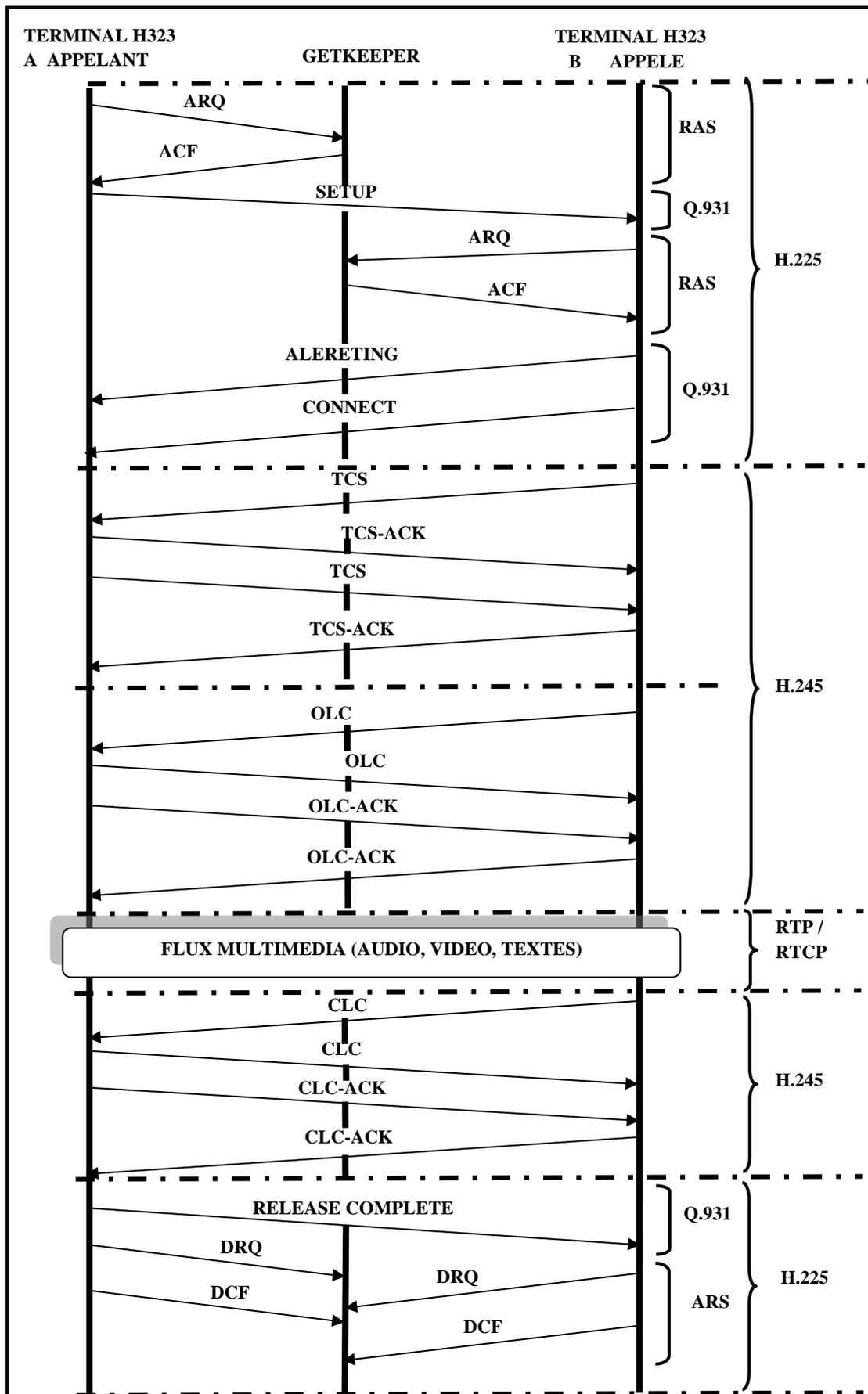


Figure 12 : Exemple d'une communication complète

Voici une brève explication du schéma :

✓ **Dans le canal H.225**

-Le terminal A H323 (appelant) s'assure d'être autorisé d'émettre un appel en envoyant un message ARQ (Admission ReQuest) au Getkeeper en signalant qu'il souhaite appeler un autre terminal B H323 (appelé).

-Le Getkeeper vérifie que le terminal A est autorisé, puis il localise le terminal B et envoie après une réponse ACF (Admission ConFirm) au terminal A qui contient l'autorisation d'appel ainsi la localisation de B. Le terminal peut contacter directement le terminal B en envoyant un message SETUP.

-Le terminal B a son tour envoie ARQ au Getkeeper qui lui répond d'un ACF. A ce niveau la les deux terminaux sont enregistrés auprès du Getkeeper. la communication donc est établie en envoyant les messages ALERTING indiquant que sa sonne chez l'appelé (le terminal A) et CONNECT indiquant que l'appelé a répondu à l'appel.

✓ **Dans le canal H.245**

-Chaque terminal envoie la liste de ses capacités dans un message TCS (Terminal Capacity Set) et reçoit un accusé de réception TCS-ACK.

-Les messages OLC (Open Logical Channel) permettent d'ouvrir un canal logique en indiquant le type de données transmises et les codecs utilisés, les deux terminaux s'échangent des OLC et reçoivent des OLC-ACK.

Cela généralement se fait sur le port TCP, mais ça peut se faire sur des ports UDP.

✓ **sur le canal RTP/RTCP**

Echange de flux MultiMedia (audio et vidéo).

✓ **Dans le canal H.245**

Pour clôturer le canal, des messages CLC (Close Logical Channel) sont échangés entre les terminaux ainsi que des acquittements CLC-ACK.

✓ **Dans le canal H.225**

Lorsque l'appelant veut terminer l'appel, il envoie RELEASE COMPLETE à l'appelé, d'un autre côté les deux terminaux envoient des DRQs (Disengage ReQuest) au Getkeeper et de confirmer la fin de la communication par des DCFs.

**II-4. Avantages et inconvénients du H323 :****❖ Les avantages :**

- Codecs standards : le H323 fournit des codecs standards de compression / décompression des flux intermédiaires, ce qui assure la communication même sur des supports différents.
- L'interopérabilité : le problème de l'incompatibilité des terminaux destinataires est éliminé, car le H323 est doté d'un mécanisme commun d'établissement d'appel permettant à des utilisateurs de dialoguer entre eux.
- Indépendance vis à vis des plates formes et des applications : le H323 ne dépend d'aucune plate forme.
- Support multipoint : le H323 assure la communication entre plus de deux terminaux sans avoir recours à une unité de contrôle spécialisée.
- Gestion de la bande passante : le H323 permet la gestion de la bande passante ce qui évite l'encombrement occasionné par le trafic audio et vidéo.
- Support multicast : le H323 optimise le réseau en envoyant le flux sur le réseau et ce dernier va le dupliquer afin de l'envoyer pour tous les destinataires.

**❖ Les inconvénients:**

- Protocole complexe : crée originalement pour des conférences multimédias, il combine des méthodes exagérées dans la téléphonie ce qui a de conséquences sur les terminaux.
- Il intègre plusieurs alternatives susceptibles d'être implémentées de façon différentes, donc impose de problèmes d'interopérabilité.

Le protocole H323 été le premier standard utilisé par les industriels dans le monde IP, il a été envisagé pour les opérations téléphoniques et la visioconférence sur IP, il offre une série de taches puissantes dans le domaine de la TOIP vu la signalisation. Cependant il été employé par les grands systèmes (Microsoft).

D'autre part, le H323 reste un protocole complexe et rude à mettre en œuvre, sa faiblesse se montre dans les écarts par rapport à la définition stricte donnée dans les recommandations de l'UIT et le protocole a perdu son rôle de fédérateur comme prévu en perdant les deux fonctionnalités principales la compatibilité et l'interopérabilité, ce qui rend son utilisation un peu rigide.

Le H323 se marginalise davantage et laisse le terrain pour un autre protocole plus qualifié qui a pu imposer sa part dans le monde IP, c'est le protocole SIP qu'on va étudier en détail.

### II-3. Le protocole SIP

#### II-3.1. Historique :

SIP (Session Initiation Protocol) est un protocole de signalisation permettant l'établissement, la libération et la modification de sessions multimédias (RFC 3261).

A l'origine, SIP a été spécifié par le groupe de travail MMUSIC (Multi party MultiMedia Session Control) de l'IETF (Internet Engineering Task Force) dans la RFC 2543 en mars 1999. En septembre 1999 l'IETF a créé le groupe de travail SIP pour poursuivre ces travaux de spécification. C'est un protocole conçu pour être un protocole de pure signalisation, suivant ainsi le credo de l'IETF «Un problème, un protocole».

Le protocole SIP est utilisé dans les télécommunications multimédias. Il est depuis 2007 le protocole le plus utilisé pour la téléphonie par Internet. [8], [B1]

#### II-3.2. Architecture :

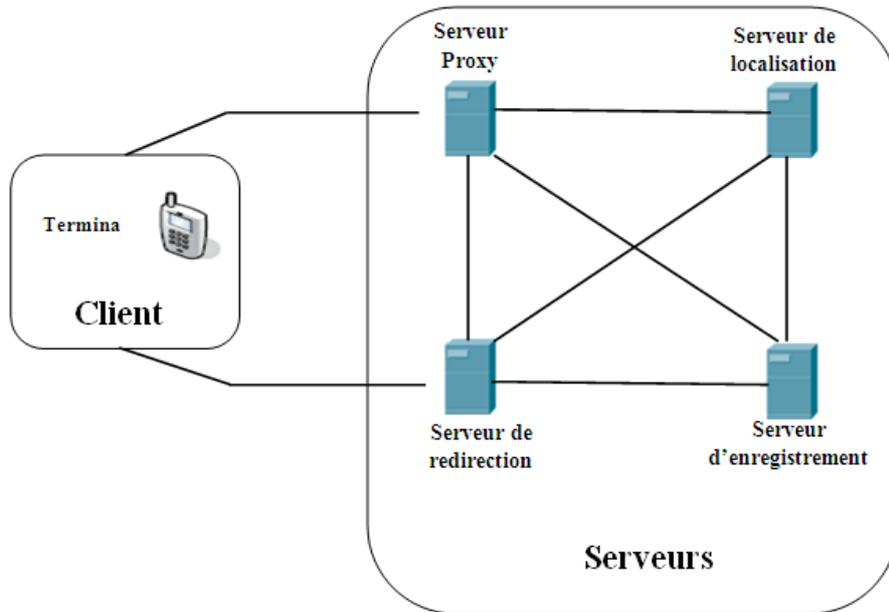
Contrairement à H.323, largement fondé sur une architecture physique, le protocole SIP s'appuie sur une architecture purement logicielle.

Son architecture est composée de cinq entités suivantes :

- ✓ Terminal utilisateur.
- ✓ Serveur d'enregistrement.
- ✓ Serveur de localisation.
- ✓ Serveur de redirection.
- ✓ serveur proxy.

La figure ci-dessous illustre ces cinq composants :

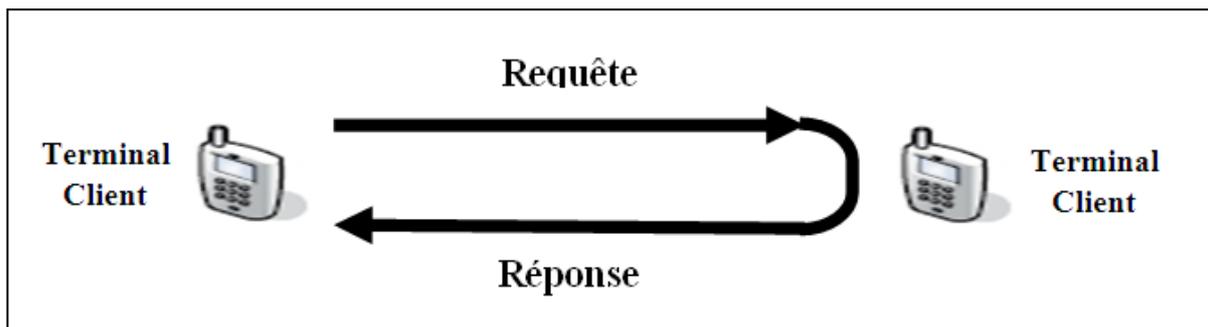
Sachant qu'avec un seul terminal (client), il nous est impossible d'effectuer une session.



**Figure 13: Les composants du protocole SIP**

Nous allons voir en quelques lignes, la fonctionnalité de chacun de ces composants, nous allons commencer par le terminal (le client) :

Le terminal à comme fonction l'émission des requêtes et la réception de réponses, on illustre cela sur la figure suivante :



**Figure 14 : Le dialogue entre deux terminaux**

Le terminal est l'élément qui permet à l'utilisateur d'appeler et/ou d'être appelé. Il peut se présenter sous la forme d'un composant matériel (un téléphone) ou d'un composant logiciel (un programme lancé à partir d'un ordinateur).

Le terminal est appelé UA (User Agent). Il est constitué de deux sous-entités, comme le montre la figure précédente :

- ✓ Une entité cliente, appelée UAC (User Agent Client), chargée d'émettre les requêtes. C'est l'UAC qui initie un appel.
- ✓ Une entité serveur, appelée UAS (User Agent Server), qui est en écoute, reçoit et traite les requêtes. C'est l'UAS qui répond à un appel.

L'association des deux entités constitue un dialogue.

Nous remarquons que c'est le même scénario qui se produit avec le protocole http dans une application web : un internaute utilise un navigateur comme client pour l'envoi de ces requêtes à un serveur ou une autre machine sauf qu'en téléphonie le terminal est utilisé pour joindre un utilisateur et pour appeler donc nous déduisons que le terminal dans ce cas joue le rôle du client et du serveur en même temps.

Nous allons voir maintenant les différents serveurs qui sont :

Serveur d'enregistrement, de localisation, de redirection et le serveur de proxy.

✓ **Serveur d'enregistrement (Registrar Server):**

Deux terminaux peuvent communiquer entre eux sans passer par un serveur d'enregistrement, à la condition que l'appelant connaisse l'adresse IP de l'appelé et comme un utilisateur peut être mobile et donc peut ne pas avoir d'adresse IP fixe et aussi cette dernière peut être fournie de manière dynamique par un serveur DHCP alors le serveur d'enregistrement (Registrar Server) entre en jeu puisqu'il offre un moyen de localiser un correspondant avec souplesse, tout en gérant la mobilité de l'utilisateur. Il peut en outre supporter l'authentification des abonnés.

L'enregistrement d'un utilisateur est constitué par l'association de son identifiant et de son adresse IP. Un utilisateur peut s'enregistrer sur plusieurs serveurs d'enregistrement en même temps.

✓ **Serveur de localisation (Location Server):**

Il joue un rôle complémentaire par rapport au serveur d'enregistrement en permettant la localisation de l'abonné. Ce serveur contient la base de données de l'ensemble des abonnés qu'il gère et qui est renseignée par le serveur d'enregistrement. Chaque fois qu'un utilisateur s'enregistre auprès du serveur d'enregistrement, ce dernier en informe le serveur de localisation. Le serveur de localisation et le serveur d'enregistrement sont implémentés au sein d'une même entité. On parle alors de service de localisation d'un serveur d'enregistrement vu ces fonctionnalités qui sont proches et dépendantes.

✓ **Serveur de redirection (Redirect Server):**

Il agit comme un intermédiaire entre le terminal client et le serveur de localisation. Il est sollicité par le terminal client pour contacter le serveur de localisation afin de localiser un utilisateur.

✓ **Serveur de proxy :**

Il permet d'initier une communication à la place de l'appelant. Il joue le rôle d'intermédiaire des interlocuteurs et agit pour le compte de ces derniers. Le serveur proxy s'occupe de:

- ✓ localiser un correspondant.
- ✓ réaliser éventuellement certains traitements sur les requêtes.
- ✓ initier, maintenir et terminer une session vers un correspondant.

### II-3.3. Adressage SIP :

Le protocole SIP est conçu pour les réseaux à commutation de paquet de type IP. Ses utilisateurs peuvent contacter des terminaux connectés à d'autres réseaux différents et cela via des passerelles (Gateway), qui assure la correspondance des signaux (la communication) entre les différents réseaux. Cette dernière se fait à l'aide des adresses qui sont des identifiants uniques désignant chaque terminal ou utilisateur dans chaque réseau.

Tout utilisateur SIP possède un identifiant unique qui lui permet d'être localisé et qu'on appelle aussi adresse SIP qui est sous la forme suivante :

```
SIP: identifiant [:mot_de_passe] @serveur [? paramètre]
```

L'explication des champs :

**SIP** : il spécifie le protocole à utiliser pour la communication.

**Identifiant** : c'est le nom ou le numéro de l'utilisateur et il est unique.

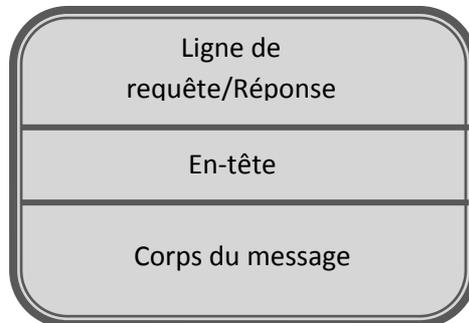
**Mot\_de\_passe** : il est facultatif et est utilisé pour l'authentification au près du serveur et aussi pour des fins de facturations.

**Serveur** : c'est le serveur chargé du compte SIP son identifiant précède l'arobase (@).

**Paramètre** : comme le mot de passe ce champ aussi est facultatif, il permet de modifier le comportement par défaut tel que le protocole de transport ou bien de spécifier des informations complémentaires tel que l'objet d'un appel.

### II-3.4. Message SIP :

Les messages SIP peuvent à la fois être des requêtes d'un utilisateur vers un serveur ou des réponses du serveur à l'utilisateur et ils sont sous la forme suivante :



**Figure 15 : Forma d'un message SIP**

➤ **Signification des trois parties :**

**La première partie :** Est soit une ligne de requête quand il s'agit d'une requête, soit d'une ligne d'état quand il s'agit d'une réponse.

**La deuxième partie :** Est l'en-tête du message SIP, qui a le rôle de fournir des informations sur le message et permet son traitement. Cet en-tête peut être un en-tête général, un en-tête de requête, un en-tête de réponse ou bien un en-tête d'entité.

**La troisième partie :** Représente le corps du message qui contient des informations permettant la participation à une session multimédia. Les informations comportent : le codec, la destination (adresse IP)...etc. le corps du message est codé à l'aide du protocole SDP (Session Description Protocol).

**Remarque :**

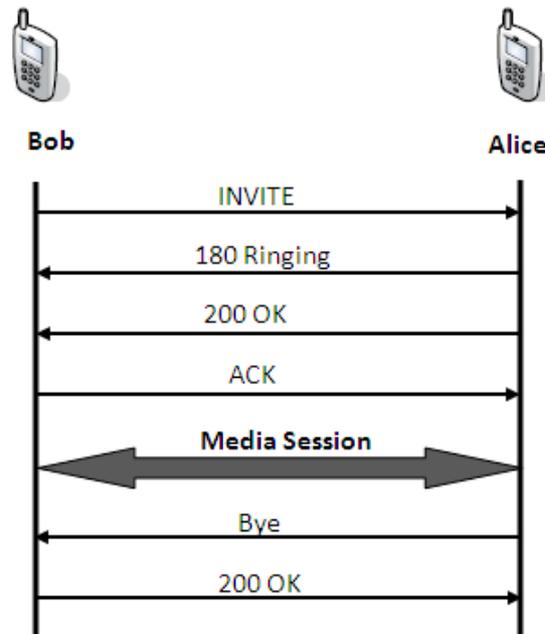
Entre la deuxième et la troisième partie il peut y avoir une partie qui est optionnelle c'est une ligne vide « CR-LF » (Carriage Return-Line Feed) il désigne la fin du champ en-tête et le début du corps du message.

### II-3.5. Le fonctionnement

Le protocole SIP repose sur un modèle requête/réponse, la session peut être directement entre deux utilisateurs SIP si chacun d'eux connaît l'adresse de l'autre ou bien via un serveur proxy si l'adresse du destinataire est inconnue.

Nous allons illustrer ces deux cas dans les figures suivantes et par la suite nous allons procéder à l'explication des différentes étapes suivies.

- ✓ Etablissement d'une session SIP simple, ça nous ramène au premier cas où l'adresse destination est connue :



**Figure 16 : Session SIP simple**

Ce cas reflète la simplicité du protocole et on a besoin de 4 étapes pour la mise en place de l'appel :

- 1) Bob envoie un message (requête INVITE) à Alice en lui proposant d'initier une communication.
- 2) Dès qu'Alice reçoit l'appel informe Bob et lui indique que l'appel s'effectue par le message (réponse 180 Ringing).
- 3) Alice accepte l'appel et avertis Bob par un message (réponse 200 OK).
- 4) Bob répond à Alice par un acquittement (requête ACK) et la prévient que la communication peut débiter.

Pour mettre fin à cette discussion (appel) faut juste que l'un des deux Bob ou Alice envoie un message de requête Bye et l'autre par un message de réponse 200 OK et la communication s'achève.

Maintenant nous allons voir le cas où l'adresse du destinataire est inconnue alors on va faire appel à un serveur proxy :

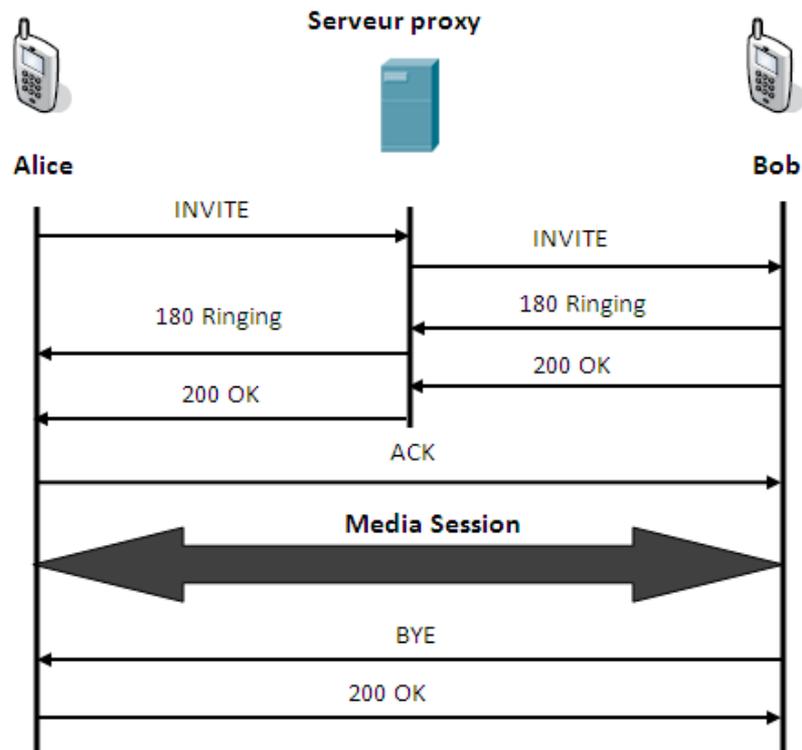


Figure 17: Session SIP faisant appel à un serveur proxy

Dans ce cas, ce qui diffère est que au début de l'appel Alice ne connaît pas l'adresse ou numéro de Bob alors elle fait appel au serveur proxy qui lui se charge de mettre Alice en contacte avec Bob et une fois cela établi alors la communication entre eux se fera de la même manière que le cas précédent.

➤ **Requêtes et Réponses :**

Une session qu'elle soit simple ou non elle a toujours recours à des requêtes et des réponses, dans ce qui suit nous allons voir quelques unes d'elles :

a) **Requêtes :** ou ce qu'on appelle aussi par méthodes, c'est les messages envoyés par un utilisateur SIP à un serveur SIP (ou directement à un autre utilisateur SIP dans le cas d'une session simple).

**INVITE :** permet d'initier une communication en invitant un correspondant à y participer.

**ACK :** Confirme que l'appelant a reçu une réponse à sa requête « INVITE ».

**BYE:** Met fin à une connexion entre utilisateurs ou refuse l'invitation à la participation d'une session

**CANCEL :** Annule une requête.

**OPTIONS** : Sollicite du proxy server afin qu'il précise ses capacités à contacter le terminal appelé.

**REGISTER** : Cette méthode est utilisée par le client pour enregistrer l'adresse listée dans le champ TO par le serveur auquel il est relié.

**b) Réponses :**

Ces réponses sont associées aux requêtes citées précédemment.

**1XX** : messages d'informations (100 – essai, 180 – sonnerie, 183 – en cours)

**2XX** : succès de la requête (200 –OK)

**3XX** : Redirection de l'appel, la demande doit être dirigée ailleurs

**4XX** : Erreur du client (La requête contient une syntaxe erronée)

**5XX** : Erreur du serveur (le serveur n'a pas réussi à traiter une requête correcte)

**6XX** : Echec général (606 – requête non acceptable par aucun serveur)

Nous allons voir maintenant le contenu du message requête INVITE qui a été envoyé à par **ALICE** au **serveur de proxy** :

```
INVITE sip: B.Bob@home.com SIP /2.0
VIA: SIP/2.0/UDP 100.123.145.124:5060; branch=z9hG4bKmp13a
Max-Forwards: 89
TO: Bob<B.Bob@home.com>
From: A.Alice <sip: A.Alice@mais.com >;tag=56
Call-ID: 74639856443
CSeq: 1INVITE
Subject: salut!!!!
Contact: <sip: A.Alice@mais.com >
Content-Type: application/sdp
Content-Length: 156
v=0
o= A.Alice 2546374896520915 65783643435 IN IP4 100.123.145.124
s= Phone Call
t= 0 0
c= IN IP4 100.123.145.124
m= audio 49170 RTP/AVP 0
a= rtpmap:0 PCMU/8000
```

**Explication des champs:**

**INVITE:** La première ligne du message de la demande contient la méthode spécifiée suivie de la requête URI.

**Via:** Il sert à retracer la route et à chaque nœud intermédiaire un nouveau champ Via est ajouté.

**Max-Forwards :** Il est utilisé pour chaque requête pour limiter le nombre de saut (éviter le bouclage), c'est un nombre qui se décrémente par chaque nœud qui reçoit et expédie la demande.

**To :** Il indique la destination et il est le même dans les réponses.

**FROME :** Correspond à une valeur unique permettant d'identifier l'émetteur sans équivoque.

**Call-ID :** C'est un identifiant unique généré à chaque nouvelle requête.

**CSeq :** Il contient un nombre suivi par le nom de la méthode et ce nombre est incrémenté à chaque envoi d'une nouvelle requête.

**Contact :** Il fournit un identifiant (URI) dont la valeur dépend le type de requête ou de réponse.

**Content-Type :** Décrit le type d'information contenu dans le corps du message.

**Content-Length :** Indique le nombre de caractères contenu dans le message

**II-3.6. Les services de SIP :**

Comme nous venons de le voir précédemment, SIP établit, modifie et termine des communications multimédias et parmi ces services populaires nous citons la messagerie instantanée et la visioconférence.

- ✓ **La messagerie instantanée :** désigne l'échange instantané des messages textuels entre deux ou plusieurs utilisateurs connectés à un même réseau de communication et cela peut être complété par une communication vidéo et cela revient à le définir comme étant une visioconférence.

- ✓ **La visioconférence** : Ce mot vient d'une combinaison de deux mots qui sont vision et conférence. La conférence est une forme de conversation entre personnes et dans la visioconférence que SIP nous offre c'est un jumelage entre l'audio le visuel et le textuel et cela ce fait via des applications de messagerie tel que Messenger ou bien Skype ...etc.

### II-3.7. Avantages et inconvénients

Le protocole SIP initie et termine des sessions multimédia en réseau mais il ne transporte pas les données multimédias échangées ce qui a l'avantage de le rendre très flexible et évolutif. Nous distinguons les avantages suivants :

- ✓ Flexible : utilisé pour tout type de multimédias (image, son, donnée informatique texte... etc.).
- ✓ Evolutif : c'est un protocole standardisé par l'IETF donc toujours en évolution et en amélioration.
- ✓ Ouvert : protocole et document détaillés ouvert et accessible à tous.
- ✓ Et aussi il est simple similaire au protocole http.

Et pour les inconvénients on trouve que :

SIP est un protocole du réseau IP donc nous rencontrons des soucis tel que : les adresses IP car SIP ne traverse pas les NATs et aussi le problème de débit avec la qualité de la connexion, revenons à parler de la bande passante et donc sa nous arrive d'avoir une mauvaise qualité de service qui est du aux critères cités.

### II-3.8. Sécurité :

Les messages SIP peuvent contenir des données confidentielles, en effet le protocole SIP possède 3 mécanismes de cryptage :

- ✓ Cryptage de bout en bout du Corps du message Sip et de certains champs d'en-tête sensibles aux attaques.
- ✓ Cryptage au saut par saut (hop by hop) à fin d'empêcher des pirates de savoir qui appelle qui.
- ✓ Cryptage au saut par saut du champ d'en-tête Via pour dissimuler la route qu'a empruntée la requête.

De plus, à fin d'empêcher à tout intrus de modifier et retransmettre des requêtes ou réponses SIP, des mécanismes d'intégrité et d'authentification des messages sont mis en place. Et pour des messages SIP transmis de bout en bout, des clés publiques et signatures sont utilisées par SIP et stockées dans les champs d'en-tête Autorisation.

#### II-4. Comparaison entre les deux protocoles SIP et H.323 :

Le tableau suivant nous illustre les différences entre le protocole SIP et le protocole H.323

	SIP	H.323
Nombre de messages échanges pour établir une connexion	1,5 aller-retour	6à7 allers-retours
Maintenance du code protocolaire	Simple par sa nature textuelle à l'exemple de HTTP	Complexe et nécessitant un compilateur
Evolution du protocole	Protocole ouvert à des nouvelles fonctions	Ajout d'extensions propriétaires sans concertation entre vendeurs
Fonction de conférence	Distribué	Centralisé par l'unité MC
Fonction de téléservices	Oui, par défaut	H.323 v2+H.450
Détection d'un appel en boucle	Oui	Inexistante sur la version 1 un appel routé sur l'appelant provoque une infinité de requêtes
Signalisation multicast	Oui, par défaut	Non

**Tableau 2: Comparaison SIP/H.323**

**II-5. Conclusion**

La téléphonie a pu franchir les obstacles posés par la téléphonie traditionnelle en s'appuyant sur de nouvelles technologies (le transport de la voix sur un réseau IP) et adopter des standards qui sont le H323 et le SIP. Dans ce chapitre nous avons vu les deux signalisations : leur fonctionnement, architecture et les différents messages échangés pour établir une communication multimédia. On a conclu par une comparaison de ces deux derniers. Dans le chapitre suivant nous allons mettre en œuvre les outils précédemment cités pour la conception d'un système de communication VoIP.

# ***CHAPITRE III***

## ***La conception du système de communication***

### III.1. Introduction

De nos jours les réseaux informatiques deviennent de plus en plus complexe, que ce soit en termes d'équipements qui les composent ou bien la quantité d'informations (flux) qu'ils véhiculent. Un administrateur doit veiller sur la maintenance et la performance du réseau.

Dans cette optique, notre travail consiste à réaliser une application de communication sous Asterisk qui doit fournir les services suivants :

- ✓ Permettre de passer un appel pour joindre le serveur ou bien un autre utilisateur.
- ✓ Laisser un message écrit ou vocal.
- ✓ Consulter une boîte vocale.

Pour le développement de notre application, nous avons adopté une approche orientée objet. La programmation orientée objet (POO) est un paradigme de programmation informatique élaboré par Alan Kay dans les années 1970. Il s'agit de passer du monde réel, complexe et confus, au monde informatique où les structures et les propriétés des objets doivent être définies et identifiées aussi que l'interaction entre ces objets. Un objet représente un concept ou une entité physique qui possède une structure interne et un comportement lui permettant de communiquer avec d'autres objets. La communication entre les objets via leurs relations permet de réaliser les fonctionnalités attendues, de résoudre les problèmes. Cette tâche est essentielle dans la conception du schéma d'une base de données.

### III.2. Le langage UML :

#### III.2.1. Définition :

UML (Unified Modeling Language) se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. [B2]

UML présente des caractéristiques particulières nous citons :

- a) Il est générique et expressif.
- b) Il possède une syntaxe et une sémantique définies.
- c) Il est flexible : On peut adapter le langage à des domaines particuliers sans ajouter de nouveaux types de diagrammes, on peut également introduire de

nouvelles notions en la définissant comme extension d'une notion existante.

### III.2.2. Les diagrammes UML

UML s'articule maintenant autour de 13 diagrammes différents (UML2), il modélise le système suivant deux modes de représentation : la représentation statique et la représentation dynamique. [B2]

- Le mode de représentation statique s'appuie sur les 7 diagrammes suivants :
  - ✓ Le diagramme de cas d'utilisation : Il met en relief des fonctions du système du point de vue des utilisateurs.
  - ✓ Le diagramme de classes : Il a pour objet de mettre en évidence les classes d'un système avec les relations qui les associent.
  - ✓ Le diagramme de packages : il consiste à utiliser un diagramme de classe pour y représenter la hiérarchie des modules (catégories) d'un projet.
  - ✓ Le diagramme d'objets : sert à illustrer des structures de classes compliquées en montrant des exemples d'instances.
  - ✓ Le diagramme de structure composite : il décrit la composition d'un objet complexe lors de son exécution.
  - ✓ Le diagramme de composants : il représente les concepts connus de l'exploitant pour installer et dépanner le système.
  - ✓ Le diagramme de déploiements : Répartition des processus sur les dispositifs matériels.
  
- Le mode de représentation dynamique s'appuie sur les 6 diagrammes suivants :
  - ✓ Le diagramme d'états : il représente le cycle de vie commun aux objets d'une même classe.
  - ✓ Le diagramme d'activité : il représente les règles d'enchaînement des activités et actions dans le système.
  - ✓ Les diagrammes de communication et de séquences : ils représentent les échanges de messages entre objets, dans le cadre d'un fonctionnement particulier du système.

- ✓ Le diagramme global d'interactions (overview interaction) : il propose d'associer les notations du diagramme de séquence avec celles du diagramme d'activité.
- ✓ Le diagramme de temps (timing diagram) : il provient de techniques connues de l'ingénierie système et répond à des besoins de modélisation très spécifique lorsque l'interaction entre plusieurs objets exige des contraintes temps-réel extrêmement précise et non équivoques. [B2]

Pour la conception de notre application, nous avons utilisé les trois diagrammes suivants : diagramme cas d'utilisation, diagramme de séquence et diagramme de classe, puis qu'ils permettent de mettre en évidence les fonctionnalités à implémenter.

### III.3. Le système de communication :

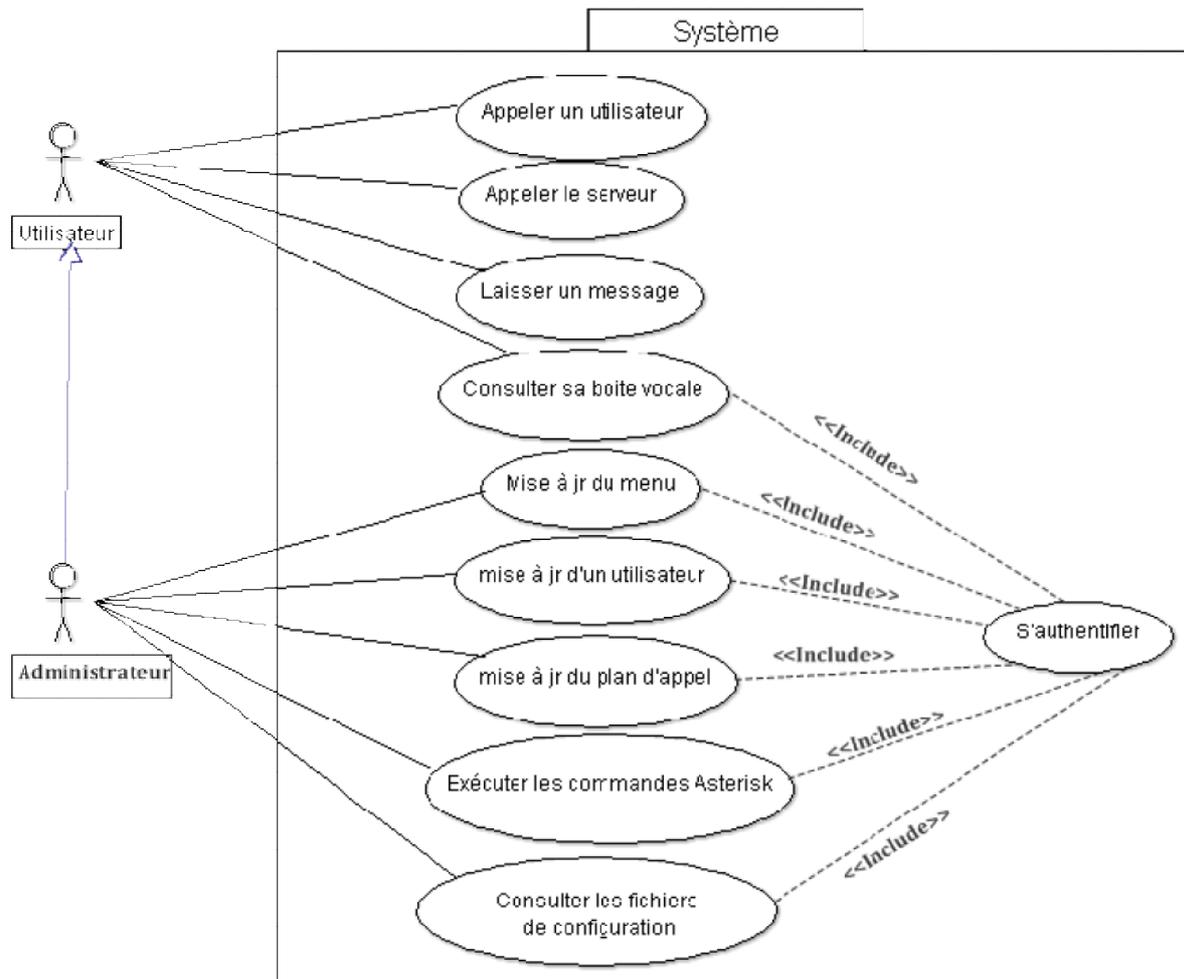
#### a) Diagramme de cas d'utilisation :

##### ➤ Détermination des cas d'utilisation :

Un cas d'utilisation permet de modéliser les besoins des acteurs d'un système. Avant de décrire ces cas d'utilisation il faut bien déterminer les acteurs interagissant avec notre système ; nous avons comme acteur :

- **Administrateur** : C'est la personne qui se charge de la gestion et la configuration du système.
- **Utilisateur** : C'est la personne qui bénéficie et utilise les fonctionnalités du système.
- **Département** : C'est un utilisateur qui joue le rôle du département dans notre serveur vocal (menu interactif).

Pour le diagramme de cas d'utilisation, nous illustrons dans la figure qui suit le système qui permet à l'utilisateur de bénéficier des différentes fonctionnalités qu'Asterisk offre ainsi que les différentes tâches qu'un administrateur effectue.

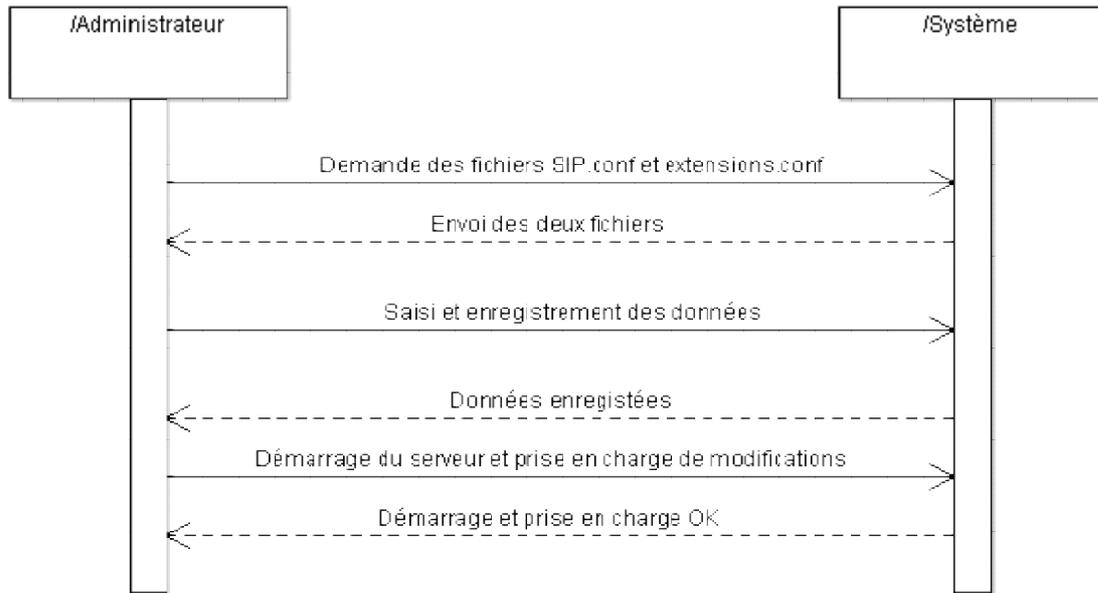


**Figure 18 : Diagramme de cas d'utilisation des fonctions de l'Administrateur et d'un utilisateur au niveau de notre système.**

#### b) Diagrammes de séquence :

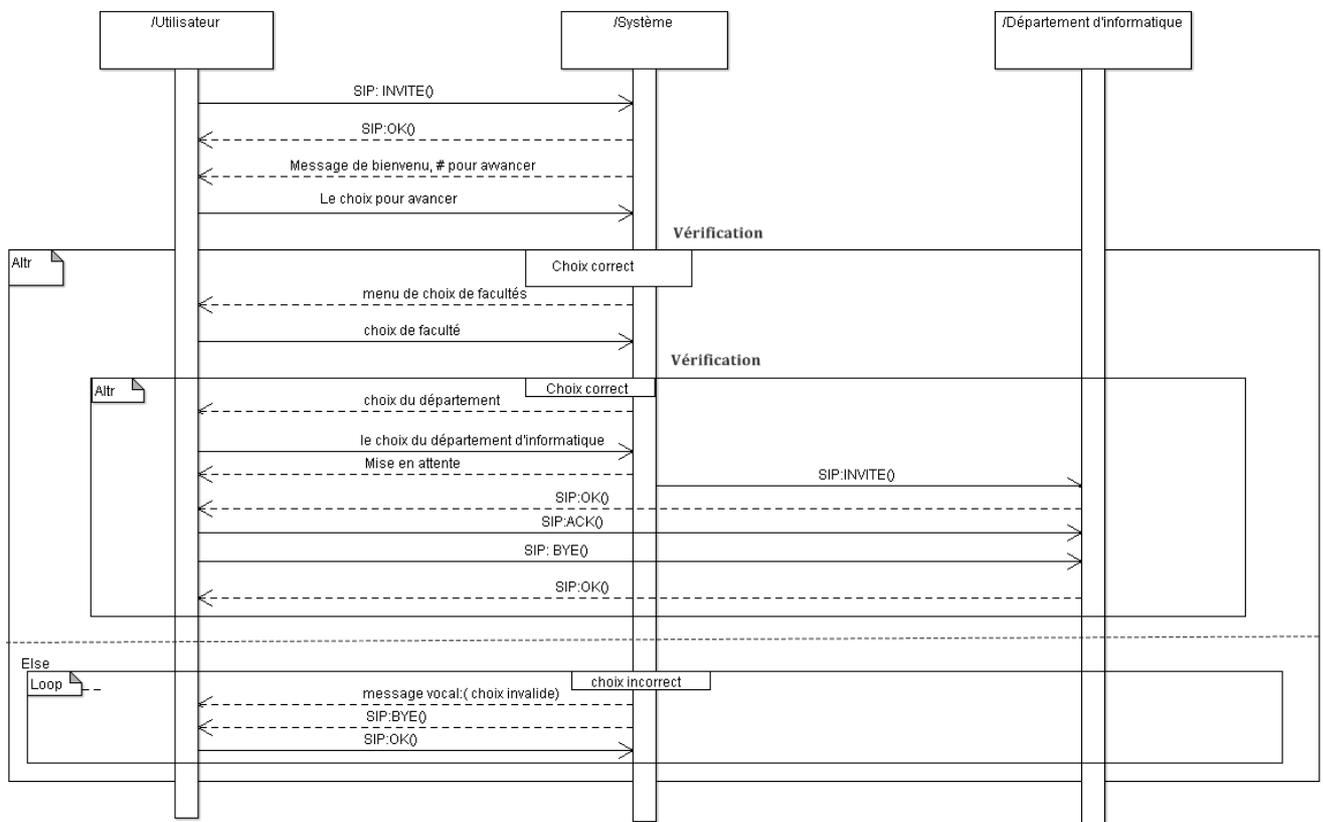
Pour les principaux cas d'utilisation du système, nous allons décrire avec des diagrammes de séquence les différentes étapes et leurs enchaînements chronologiques.

- ❖ L'ajout d'un utilisateur, d'un menu, ou la configuration du plan d'appel c'est le même diagramme la seule chose qui diffère c'est le fichier de configuration (sip.conf ou bien extensions.conf ou voicemail.conf). Nous allons voir le diagramme de séquence d'ajout d'un utilisateur qui est illustré dans la figure suivante :



**Figure 19: Diagramme de séquence d’ajout d’un utilisateur avec succès.**

❖ La figure suivante illustre le diagramme de séquence pour le cas d’utilisation joindre le département d’informatique en passant par le serveur vocale :



**Figure 20 : Diagramme de séquence pour le cas d’utilisation « appeler le serveur et demander le département d’informatique ».**

- ❖ La figure qui suit montre le diagramme de séquence du cas d'utilisation « laisser un message vocal » :

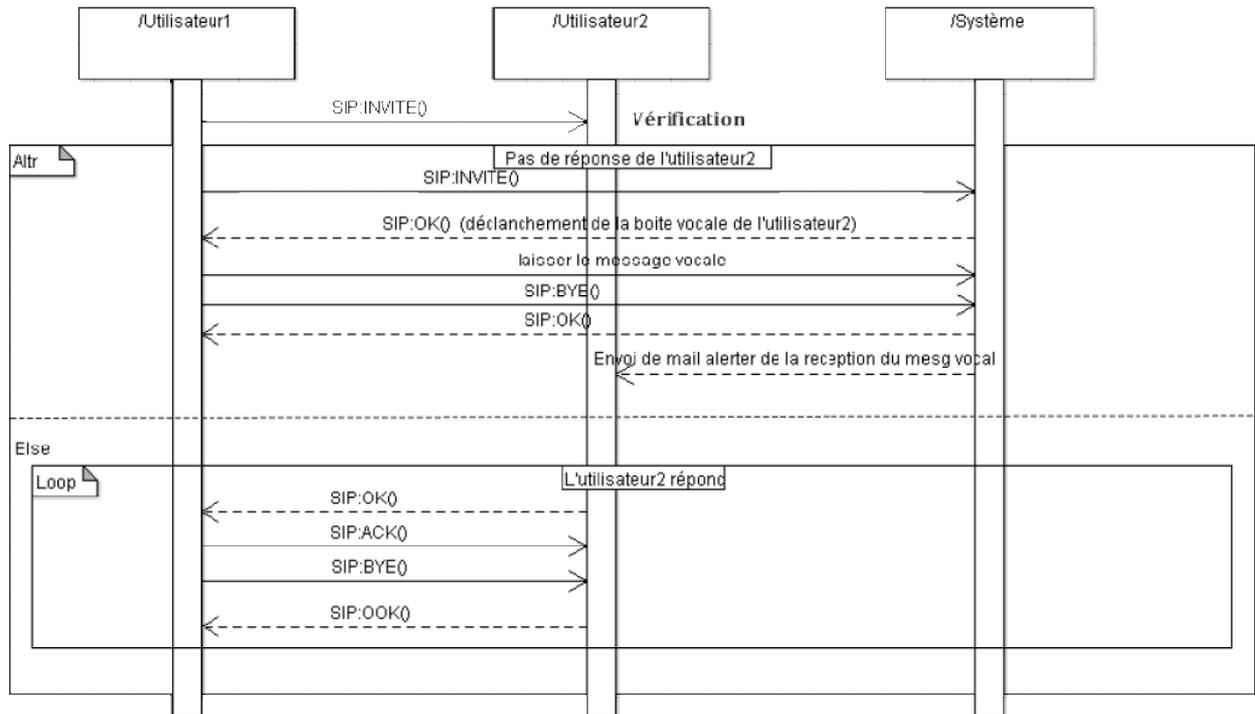


Figure 21: Diagramme de séquence « laisser un message vocal»

c) Diagramme de classe :

La figure suivante montre le diagramme de classe de notre système :

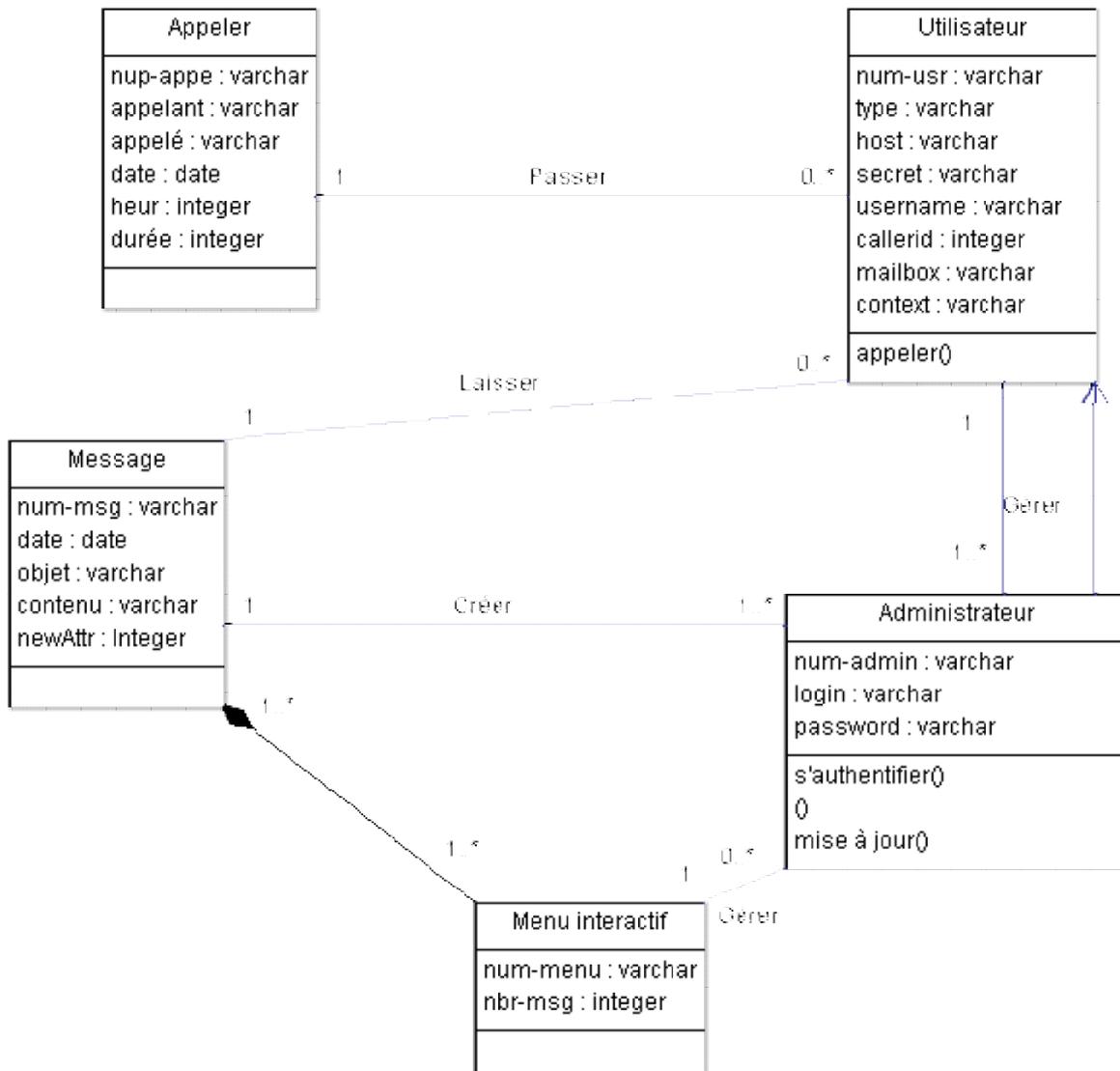


Figure 22: Diagramme de classe.

**III.4. Conclusion :**

Dans ce chapitre, nous avons pu faire une conception de notre système en utilisant trois sortes de diagrammes d'UML qui sont : le diagramme de cas d'utilisation, diagramme de séquences et le diagramme de classe pour mettre en œuvre cette architecture, nous allons l'implémenter sous Asterisk,

Nous nous allons voir dans le chapitre suivant, l'explication et la mise en œuvre de notre application.

# ***CHAPITRE IV***

## ***Implémentation du système de communication***

### IV.1. Introduction

Dans les chapitres précédant, nous avons vu les deux techniques de transfert de la voix sur un réseau IP, qui sont la ToIP et la VoIP, les protocoles associés ainsi que la modélisation de notre système de communication VoIP.

Dans ce chapitre nous allons procéder à la réalisation de notre système de communication VoIP qui sera implémenté sous la plateforme d'Asterisk .

### IV.2. Asterisk

#### IV.2.1. Définition

Asterisk est un logiciel PABX (Private Automatic Branch eXchange) libre, multiplateforme, publié sous licence GPL (General Public License) par Mark Spencer de la société Digium.

Asterisk permet, entre autres, la messagerie vocale, la conférence, les serveurs vocaux, la distribution des appels, il implémente aussi les protocoles H323 et SIP, ainsi qu'un protocole spécifique nommé IAX (Inter-Asterisk eXchange). Ce protocole IAX permet la communication entre client et serveur Asterisk ainsi qu'entre deux serveurs Asterisk. Il peut également jouer le rôle de passerelle avec les réseaux publics (RTC, GSM, ...etc.).

#### IV.2.2. Historique

Asterisk a vu le jour quand Mark Spencer a voulu acquérir un PBX traditionnel pour sa société en 1999, en trouvant que le prix d'acquisition était démesuré alors il initia un projet open-source et commença à développer Asterisk.

Son équipe s'est rapidement rapprochée de celle de Jim Dixon (Zapata Telephony Project), dans le but de construire des cartes d'interface « bon marché » avec le réseau téléphonique classique. Leur but était de construire des cartes compatibles avec des plateformes basées sur Intel et ainsi de permettre à n'importe quel PC, avec le système d'exploitation Linux, une carte d'interface et le logiciel Asterisk, d'avoir un PBX entièrement fonctionnel. Pour information, le nom Asterisk fait référence au symbole « \* » qui signifie « Wildcard » en ligne de commande Unix et DOS. Ce choix a été fait car Asterisk est conçu dans le but d'offrir une très grande souplesse dans les réseaux de transmission de la voix.

### IV.2.3. Intérêt du choix d'Asterisk

Asterisk joue un rôle important dans le monde de la téléphonie, c'est un commutateur privé, il a été conçu en premier lieu pour sa compatibilité avec les équipements numériques et analogiques de la ToIP de base standard ainsi pour ses moindres coût et en deuxième lieu pour sa flexibilité. C'est un serveur qui évolue régulièrement en fournissant davantage de nouvelles fonctionnalités, il supporte tout les protocoles de la téléphonie et fonctionne sur plusieurs plateformes (Linux, Windows, Mac).

### IV.2.4. Ses principales fonctionnalités

Asterisk compte un nombre très élevé de fonctions permettant l'intégration complète pour répondre à la majorité des besoins en téléphonie nous citons :

- ✓ Messagerie vocale
- ✓ Conférence téléphonique
- ✓ Répondeur vocal interactif
- ✓ Mise en attente d'appels
- ✓ VoIP ...etc.

### IV.2.5. Son architecture

Asterisk a une architecture très simple ; il est considéré comme un élément primordial permettant de connecter des applications et des technologies de téléphonie. Les APIs spécifiques sont définies autour d'un PABX central, ce noyau permet de manipuler l'interconnexion interne du PABX proprement soustrait des protocoles spécifiques, des codecs et des interfaces matérielles des applications de téléphonie. Cela permet à Asterisk d'utiliser n'importe quel matériel approprié et technologie disponible pour exécuter ces fonctions essentielles, en connectant le matériel et les applications.

Le schéma suivant illustre l'architecture d'Asterisk que nous avons implémentée. Nous disposons d'un serveur, de trois ordinateurs dont chacun d'eux est doté d'un softphone Ekiga ses 4 entités sont connectées au réseau de notre université que ce dernier est connecté à internet et via leur adresses ils peuvent communiquer entre eux.

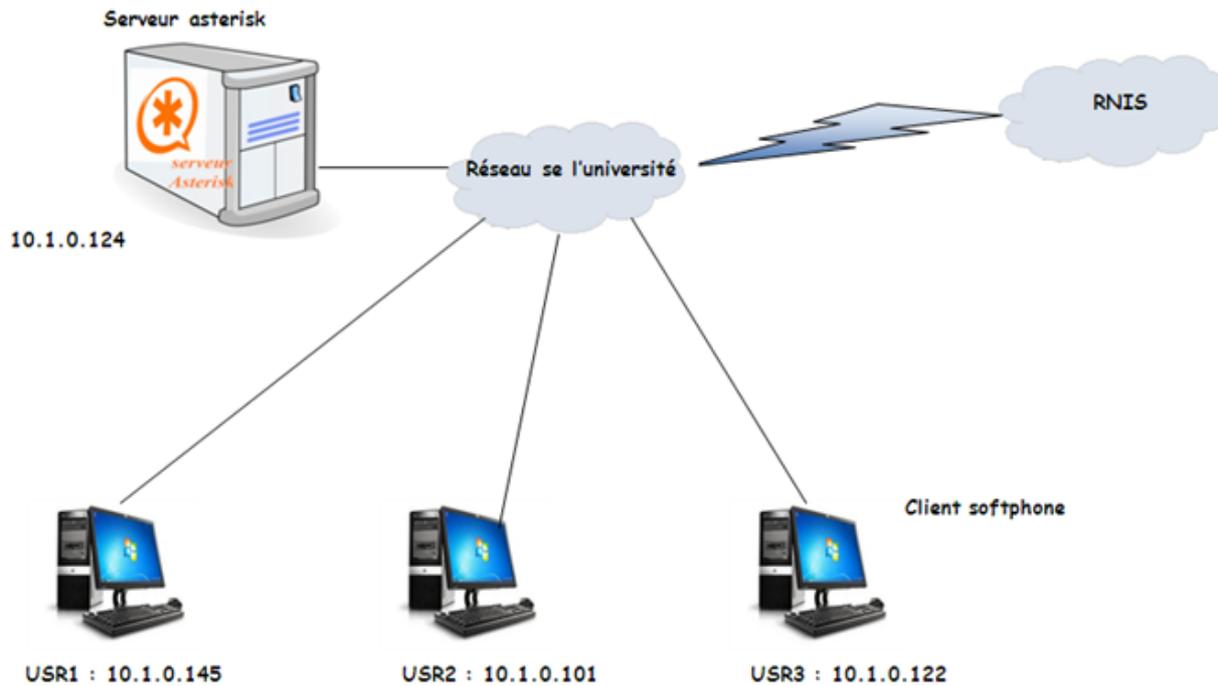


Figure 23: Architecture d'Asterisk

### IV.3. Les différentes étapes de réalisation de notre système de communication VoIP

#### IV.3.1. Installation du serveur d'Asterisk

Un serveur Asterisk peut s'installer sur différents OS (Windows, Linux, Mac), nous avons de porter notre étude et installation sur le système Linux exactement UBUNTU version 11.10. Pour procéder à l'installation d'Asterisk, faut d'abord se connecter en tant que super utilisateur et lancer l'installation du serveur avec la commande **apt-get install asterisk** ensuite taper sur O ou Y selon la langue du système et Asterisk s'installe sans problème, il faut juste préciser qu'on doit être connecté à un réseau internet.

Pour démarrer Asterisk faut taper la commande **asterisk -r** et on aura la version d'Asterisk installée qui apparaît, sur ma machine c'est : Asterisk 1.8.4.4~dfsg-2ubuntu1

#### IV.3.2. Configuration du serveur Asterisk

Pour le bon fonctionnement de notre système, Asterisk a besoin d'un certain nombre de fichiers de configuration dont quelques uns sont illustrés dans la figure suivante :

```

root@ubuntu: ~
root@ubuntu:~# ls /etc/asterisk
adsi.conf          dnsmgr.conf      osp.conf
agents.conf       dsp.conf         oss.conf
ais.conf          dundi.conf      phone.conf
alarmreceiver.conf enum.conf       phoneprov.conf
alsa.conf         extconfig.conf  queuerules.conf
amd.conf          extensions.ael   queues.conf
app_mysql.conf   extensions.conf  res_config_mysql.conf
asterisk.adsi    extensions.conf~ res_config_sqlite.conf
asterisk.conf    extensions.lua   res_curl.conf
calendar.conf    extensions_minimv.conf res_fax.conf
ccss.conf        extensions-sauv.conf res_ldap.conf
cdr_adaptive_odbc.conf extensions-sauv.conf~ res_odbc.conf
cdr.conf         features.conf    res_pgsql.conf
cdr_custom.conf  features.conf~  res_pktccops.conf
cdr_manager.conf festival.conf    res_snmp.conf
cdr_mysql.conf   followme.conf   res_stun_monitor.conf
cdr_odbc.conf    func_odbc.conf  rpt.conf
cdr_pgsql.conf   gtalk.conf      rtp.conf
cdr_sqlite3_custom.conf h323.conf      say.conf
cdr_syslog.conf http.conf       sip.conf
cdr_tds.conf     iax.conf       sip.conf~
cel.conf         iaxprov.conf   sip_notify.conf
cel_custom.conf  indications.conf sip-sauv.conf
cel_odbc.conf    jabber.conf    skinny.conf
cel_pgsql.conf   jingle.conf    sla.conf
cel_sqlite3_custom.conf logger.conf    smdi.conf
cel_tds.conf     manager.conf   telcordia-1.adsi
chan_dahdi.conf  manager.d      udptl.conf
chan_mobile.conf meetme.conf    unistim.conf
chan_ooh323.conf mgcp.conf     usbradio.conf

```

Figure 24 : fichiers de configuration d'Asterisk

Pour notre système nous avons décidé d'implémenter une application de communication VoIP sous Asterisk et elle est caractérisé par :

Ces Trois utilisateurs qui, en premier lieu joueront le rôle de simples utilisateurs qui auront la possibilité de s'appeler les uns les autres et en deuxième lieu, l'un deux jouera le rôle d'un appelant et les deux autres le rôle de département vue qu'on veut réaliser un serveur vocale pour l'université. Le but c'est d'appeler le serveur qui va nous diriger vers l'un des départements en passant par une liste de choix à faire. Pour ceux nous aurons besoin de modifier quelque fichiers de configuration d'Asterisk nous citons : sip.conf, extensions.conf.

Les deux fichiers se trouvent dans le répertoire /etc/asterisk.

#### a) La configuration du fichier sip.conf

Dans ce fichier nous allons déclarer les différents utilisateurs comme suit :

**[usr1]**

**Type**= friend

**Host** = dynamic

**User** =usr1

**Secret**= user1

**Context**= default

### ❖ Explication des champs :

- ✓ **[usr1]** : C'est le début du bloc sip et tout ce qui suit ce bloc c'est les caractéristiques du usr1
- ✓ **Type = friend** : permet d'appeler et d'être appelé.
- ✓ **Host= dynamic** : l'adresse IP de la machine est défini par le DHCP, si l'adresse IP est fixe on la défini directement.
- ✓ **User= usr1** : c'est le nom de l'utilisateur.
- ✓ **Secret= user1** : c'est le mot de passe en clair de l'utilisateur.
- ✓ **Context= default** : c'est le contexte auquel le compte est associé dans le dialplan (extensions.conf)

### b) La configuration du fichier extensions.conf

Dans ce fichier extensions.conf et qu'on appel aussi dialplan, qui veut dire le plan de numérotation, on explicite l'action qui déroule au moment de composer un numéro donné.

Pour cela nous allons ajouter quelques lignes dans la section [default]. Cette section [default] c'est ce qu'on appel un contexte.

La ligne qui associe un numéro de téléphone à un utilisateur donné, par exemple pour l'utilisateur usr1 nous allons lui associer le numéro 555 alors la ligne sera comme suit :

**Exten => 555, 1, Dial (SIP/usr1).**

Le 1 c'est le numéro de séquence et Dial (SIP/usr1) est l'action à effectuer entre autre Dial () c'est la fonction qui déclenche l'appel.

Une fois les deux fichiers modifiés, Asterisk doit les prendre en charge. Pour cela il faut démarrer le serveur d'Asterisk avec la commande **asterisk -r** et ensuite nous allons taper sip reload et dialplan reload pour charger les deux fichiers sip.conf et extensions.conf.

### IV.3.3. Fonctionnalités implémentées

Parmi les fonctionnalités qu'Asterisk dispose, nous allons voir le transfert d'appel, la mise en attente et la messagerie vocale.

### a) Le transfert d'appel

Pour activer cette fonction avant toute chose, il faut penser à activer les tonalités DTMF (Dual-Tone Multi-Frequency). Pour cela nous avons modifié cette ligne dans le fichier `extensions.conf` :

```
Exten => 555, 1, Dial (SIP/usr1,, tT)
```

Après `usr1` nous avons ajouté `tT` qui autorisent à l'appelé et à l'appelant de transféré l'appel en appuyant sur `#` durant l'appel.

### b) La mise en attente

La mise en attente permet de mettre une communication en pose et pour cela nous avons modifié le fichier `extensions.conf` en ajoutant la ligne suivante dans la section `[default]` :

```
Include => parkedcalls
```

Pour mettre une communication en attente il faut taper `#700`.

### c) La messagerie vocale

Par défaut Asterisk dispose d'un utilisateur 1234 avec une boîte vocale configurée qui possède le mot de passe 4242. Pour y accéder il faut juste appeler le numéro 8500 et s'authentifier.

Nous pouvons installer les messages vocaux en français en installant les packages appropriés avec la commande : **`apt-get install asterisk-prompt-fr`** ainsi nous avons remplaçant la ligne `;language=en` par `language=fr` dans le fichier `sip.conf`.

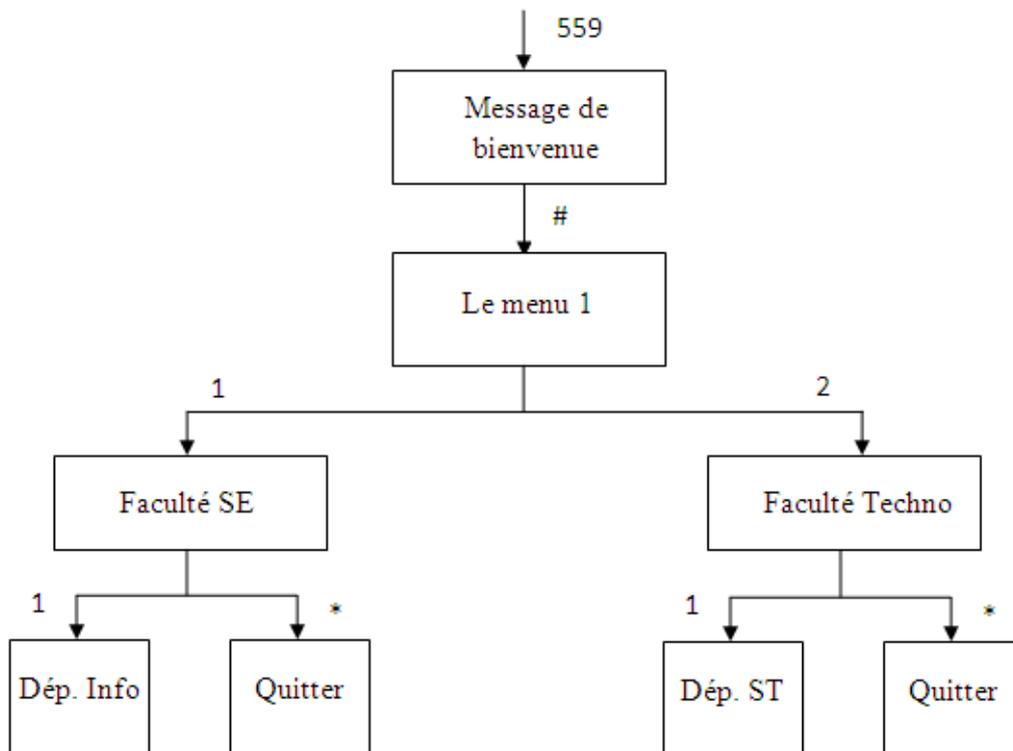
Ensuite nous avons créé la boîte vocale en la déclarant dans le fichier `voicemail.conf` dans la section `[default]`, en lui associant un numéro et un mot de passe sans oublier bien sûr de la déclarer dans le fichier `extensions.conf`.

Si un utilisateur a reçu un message vocal dans sa boîte vocale alors Asterisk via un serveur MTA (Mail Transfer Agent), que nous avons choisi va envoyer ce message vocal en pièce jointe par mail à l'utilisateur ainsi il pourra l'écouter immédiatement.

Une fois notre MTA installé et configuré, le serveur mail se relance et Asterisk peut prévenir l'utilisateur concerné par mail en ayant le message vocal comme pièce jointe.

**d) Le menu interactif**

La figure suivante illustre notre menu :



**Figure 25 : menu interactif.**

En composant le numéro 559, nous allons atteindre le serveur vocal d’Asterisk et on aura le message bienvenue et le code suivant l’illustre :

```

exten => 559,1, Answer()
exten => 559,3, Goto(accueil,s,1)
[accueil]
exten => s,1,Background(/usr/share/asterisk/sounds/msgbienvenue)
exten => s,2, WaitExten
exten => #,1,Goto(menu1,s,1)
exten => i,2,Playback(/usr/share/asterisk/sounds/choix-erreur)
    
```

Pour continuer nous tapons # et nous aurons un menu à nous de faire un choix et ainsi de suite, et si nous choisissons une touche qui diffère de celles proposées dans le menu nous aurons un message disant touche invalide.

Ainsi nous venons de finir avec la configuration de notre serveur ainsi que ces différentes fonctionnalités, pour mettre en œuvre ces dernières nous allons installer un softphone qui est Ekiga.

### IV.4. Le softphone

Un softphone est un logiciel qu'on utilise pour faire de la téléphonie sur Internet depuis son ordinateur. Il existe plusieurs softphone tel que : twinkle, X-lite, Ekiga notre choix se porte sur Ekiga, préalablement nommé GnomeMeeting c'est un logiciel libre de téléphonie et de visioconférence par Internet (voix sur IP), pour Linux et Windows (beta) dont l'interface a été développée avec les bibliothèques de l'environnement GNOME.

Ekiga utilise les protocoles de communication standards et ouvert H323 et SIP, ce qui le rend compatible et interopérable avec les autres logiciels et appareil basés sur ces mêmes protocoles. La figure suivante illustre le softphone Ekiga :

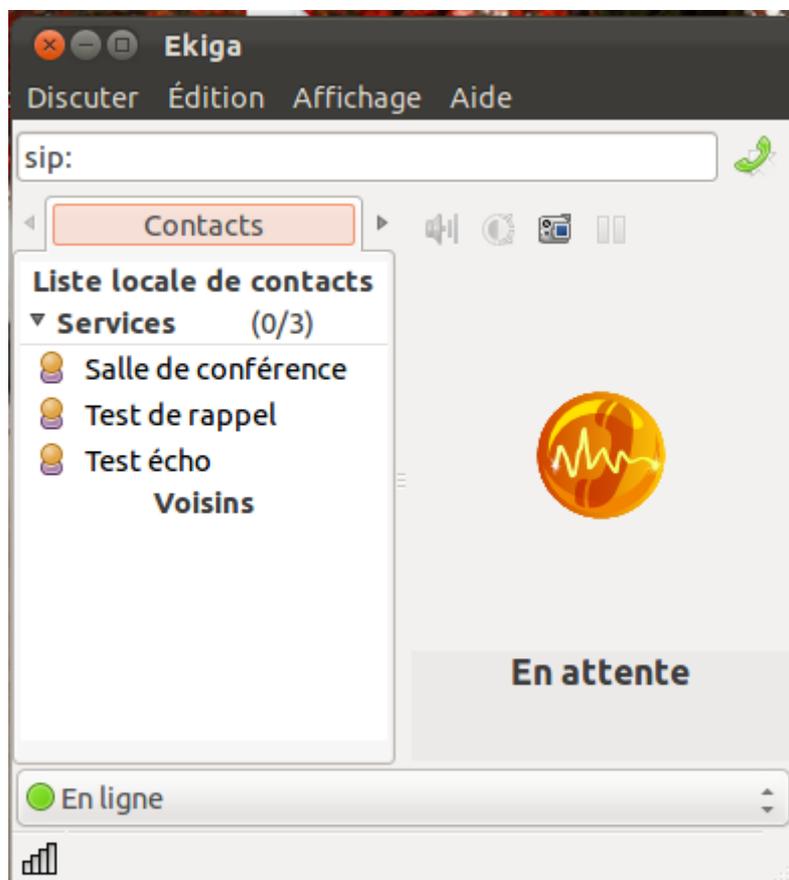


Figure 26: Ekiga

La figure ci dessous illustre l'ajout d'un compte sip en insérant les différents champs :

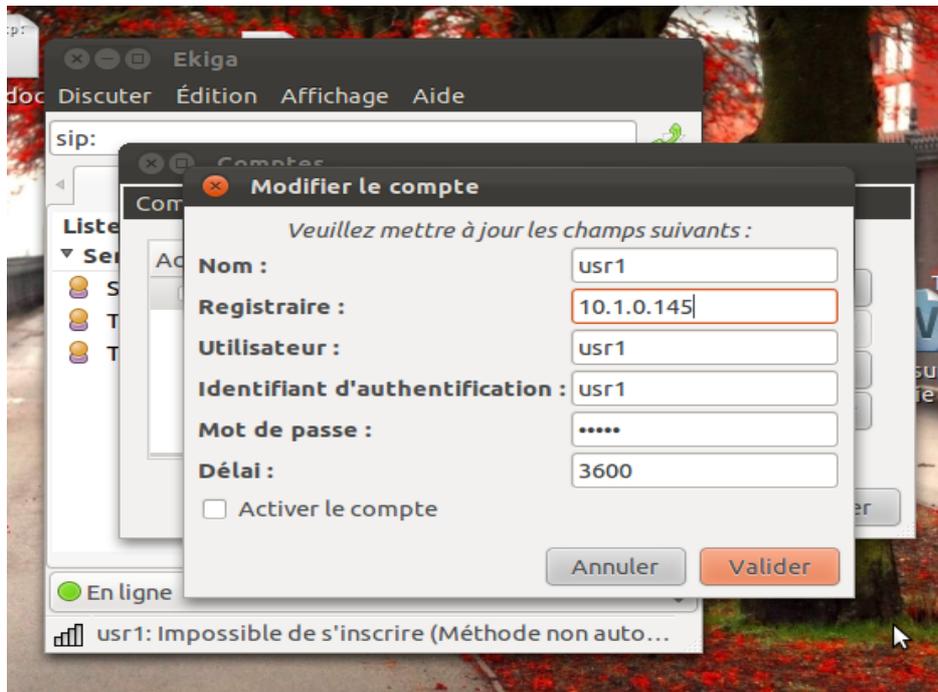


Figure 27: ajout d'un compte SIP.

Nous allons voir maintenant une communication entre deux utilisateurs usr1 et usr2 qui s'appellent en premier lieu et ensuite ils s'envoient des messages instantanés.

L'utilisateur usr1 appelle l'utilisateur usr2 la figure suivante illustre l'alerte chez usr2 :

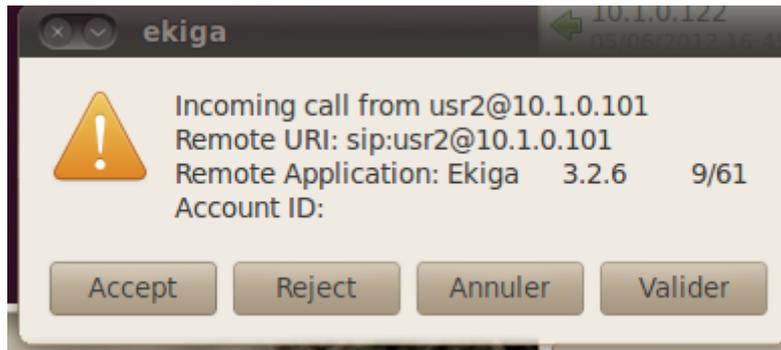


Figure 28: usr1 appel usr2

Une fois que l'utilisateur usr2 décroche :



Figure 29: l'appel en cour entre usr1 et usr2

La figure suivante illustre l'envoi de message de l'usr2 à usr1 :

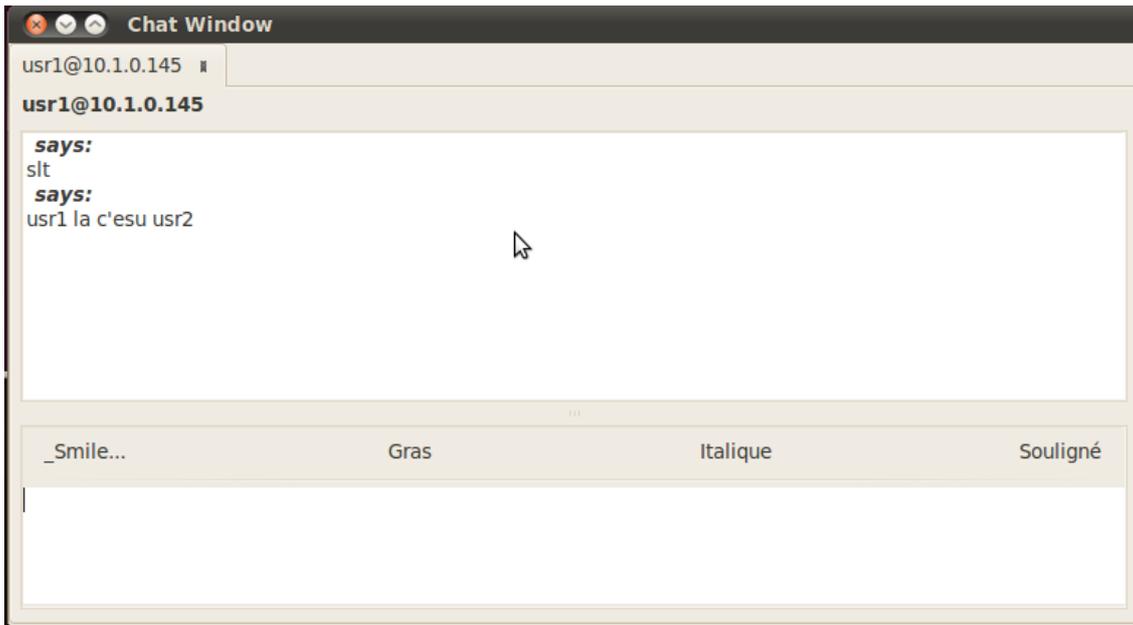
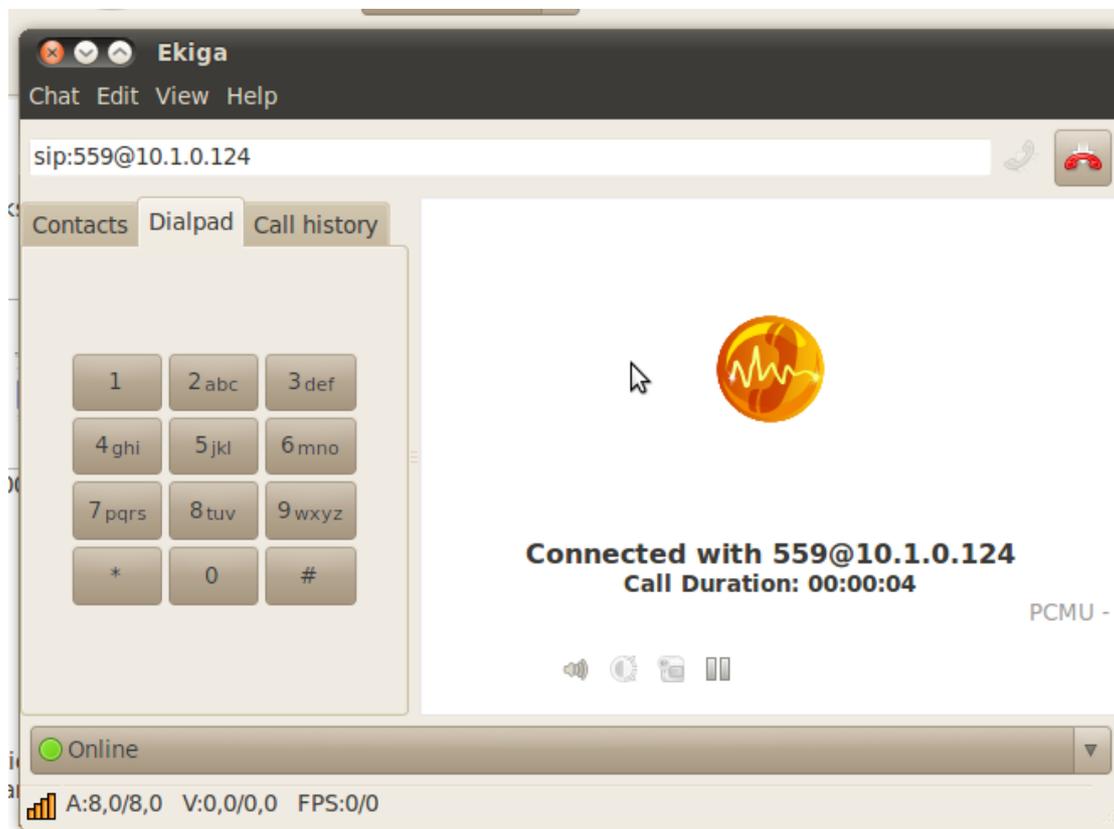


Figure 30: la messagerie instantanée.

Maintenant que nous avons vu une communication entre les deux utilisateurs, la figure qui suit montre la connexion au serveur vocal le menu interactif:



**Figure 31 : connexion au serveur vocal****IV.5. Conclusion**

Dans ce chapitre, nous avons présenté Asterisk, son architecture et ses principales fonctionnalités, par la suite, nous avons procédé à son installation ainsi que sa configuration. Pour notre système de communication nous avons utilisé le softphone Ekiga qui nous a permis de réaliser les fonctions suivantes : le transfert d'appel, la messagerie vocale et la mise en attente, et cela en adoptant le protocole SIP.

## CONCLUSION GENERALE

L'émergence de la VOIP rendue possible grâce au protocole IP. En effet la souplesse et l'ouverture d'un réseau IP permettent l'intégration de nouvelles applications et services dans un large éventail de capacités multimédia comme la téléphonie, la visioconférence et la messagerie instantané.

Tous ces services sont intégrés dans une seule technologie qui est la téléphonie sur IP. Cette technologie est entrain de transformer profondément la manière dont les universités et les laboratoires de recherche la manière dont les utilisateurs de leur réseau interagissent entre eux et offre de nouvelle possibilité pour réduire les coûts et améliorer le service.

Dans le cadre de notre travail, nous avons introduit les concepts de la technologie de la voix sur IP ainsi que ses applications et services et l'étude deux techniques de transmission de la voix qui sont téléphonie sur IP (ToIP) et la voix sur IP (VoIP) ainsi les protocoles associés nous citons : TCP/IP, UDP, RTP/RTCP

Notre projet est scindé en deux grands volets principaux :

Le premier présente la définition des deux techniques de transmission de la voix qui sont : la téléphonie sur IP (ToIP) et la voix sur IP (VoIP) ainsi les protocoles associés nous citons : TCP/IP, UDP, RTP/RTCP et les deux protocoles de signalisation H323 et SIP.

Le deuxième traite de la conception et la mise en ouvre de notre système de communication VoIP ainsi qu'un serveur interactif joue le rôle d'un standard téléphonique pour l'université en utilisant le protocole SIP et les logiciels Asterisk et Ekiga qui joue le rôle d'une passerelle entre les différents utilisateurs ainsi qu'entre les utilisateur et le serveur.

Notre projet nous a permis de développer nos connaissances concernant le système d'exploitation LINUX qui nous a aidé à mettre en place une solution qui permet aux utilisateurs connectés sur le réseau de l'université de communiquer entre eux gratuitement.

## BIBLIOGRAPHIE

[B1] : Laurent Ouakil et Guy Pujolle. La téléphonie sur IP. 2<sup>ème</sup> Edition. EYROLLES, 484.

[B2] : Pascal Roques et Franck Vallée. UML 2 en action. De l'analyse des besoins à la conception. 4<sup>ème</sup> édition.EYROLLES. 2007. 394.

[1]: L'INTERNAUTE. High-Tech. Dictionnaire informatique. [en ligne]  
[http://encyclopedia.linternaute.com/definition/1036/9/telephony\\_over\\_internet\\_protocol.shtml](http://encyclopedia.linternaute.com/definition/1036/9/telephony_over_internet_protocol.shtml).  
[mars/2012].

[2]: Ka Kit Yick.Téléphonie IP. [en ligne].  
<http://charon.minilab.bdeb.gc.ca/inf850/yickkk/type.htm>. [mars/2012].

[3]:Université Paris-Est Marne-la-Vallée. L'essentiel de la VoIP. [ en ligne]. [http://www-igm.univ-mlv.fr/~dr/XPOSE2008/touali\\_VoIP/numerisation.voix.php](http://www-igm.univ-mlv.fr/~dr/XPOSE2008/touali_VoIP/numerisation.voix.php). [mars/2012].

[4]: Sébastien Fontaine.FrameIP TcpIP.[en ligne]. <http://www.frameip.com/entete-ip/>.  
[mars/2012].

[5]: Sébastien Fontaine .FrameIP TcpIP.[en ligne]. <http://www.frameip.com/entete-tcp/>.  
[mars/2012].

[6]: Sébastien Fontaine .FrameIP TcpIP.[en ligne]. <http://www.frameip.com/entete-udp/>.  
[mars/2012].

[7]: Sébastien Fontaine .FrameIP TcpIP.[en ligne]. <http://www.frameip.com/entete-rtcp/>.  
[mars/2012].

[8]:VoIPfr.org. Robert Viseur. Portail VoIP. [ en ligne].  
<http://www.voipfr.org/glossaire/byname,sip.php>. [avril/2012]

[9]: Asterisk. [en ligne] [.http://asterisk.open-source.fr](http://asterisk.open-source.fr). [mai,juin2012]

**Résumé:**

La téléphonie sur IP (ToIP, Telephony over IP) est une technologie qui s'est imposée dans tous les secteurs. Elle consiste à transiter les communications téléphoniques sur un réseau IP.

Actuellement, cette technologie est la plus répandue au sein des universités et laboratoires de recherche connectés au réseau national de télécommunication et cela grâce à ses moindres coûts en comparant à la téléphonie traditionnelle. De plus la téléphonie sur IP utilise moins de bande passante que la téléphonie traditionnelle.

Une telle technologie au sein de notre université ne peut être que bénéfique. En effet, les systèmes de communications VoIP peuvent toujours fonctionner séparément même si certaines parties du réseau présente des soucis de connectivité et pour le cas de coupure de lien internet, il est toujours possible de prévoir une solution de secours pour la connexion Internet grâce à l'utilisation des passerelles sur le réseau RTC.

**Abstract:**

IP telephony (VoIP, Telephony over IP) is a technology that has emerged in all sectors. It is to pass the telephone calls over an IP network.

Currently, this technology is the answer in universities and research laboratories connected to the national telecommunications network and thanks to its lower cost in comparison to traditional telephony. More IP telephony uses less bandwidth than traditional telephony.

Such technology within our university can be beneficial. Indeed, VoIP communications systems can still operate separately even though parts of this network and connectivity issues for the case of power internet link, it is always possible to provide a solution of succor for the Internet connection through the use of bridges over the PSTN.

The aim of our project is the study of this technology, we associate the different protocols are: TCP / IP, RTP / RTCP, UDP, SIP and H323 and the development of a VoIP communication system for the network of university.