

République Algérienne Démocratique et Populaire  
Université Abou Bakr Belkaid– Tlemcen  
Faculté des Sciences  
Département d'Informatique

Mémoire de fin d'études

pour l'obtention du diplôme de Licence en Informatique

*Thème*

## Déploiement d'une application de TOIP

Réalisé par :

- BACHIRI Soumia
- BELARBI Baraka

*Présenté le 01 juin 2015 devant la commission d'examination composée de MM.*

- *DiDi fedoua* (Encadreur)
- *Iles Nawel* (Examineur)
- *Belhocine Amine* (Examineur)

Année universitaire: 2014-2015.

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ  
{وَقُلْ اَعْمَلُوا فِیْ سَبِیْلِ اللّٰهِ عَمَلِكُمْ وَرَسُوْلَهُ وَ الْمُؤْمِنُوْنَ }  
صدق الله العظيم

## اهداء

تنحني عبارات الشكر للذي من بنوره نهدي، للذي بالعلم شرح صدري ومُقفلي، الذي لا يطيب الليل إلا بشكره و لا يطيب النهار إلا بطاعته، إليك "رَبِّي حَسْبِي وَ مَطْلَبِي".  
إلى من بنوره تلاً الكون وابتسم، والبشر بروياه تكلل وارتسم إليه "صلى الله عليه وسلم".  
إلى أنسي ورفيقي مصحفي نور طريقي.

إلى من كلفه الله بالهيبــــــــــــــــة و الوقار، و حملت اسمه بكل افتخار "أبي الغالي".  
إلى من أرضعتني الحبّ و الحنان، إلى بلسم حياتي و سرّ نجاحي "أمي الحبيبة".

كما أتوجه بالشكر الخاص إلى العمّ الغالي "مراد" الذي علّمني معنى التّفاؤل و المضي إلى الأمام.  
إلى كافة أعمامي و عمّاتي : البشير، مصطفى، زوليخة، مليكة، نورية، حليلة.

إلى القلوب الطاهرة إلى رياحين حياتي "أخواتي" : حفصة، نسيبة، كوثر، نانلة، رجاء.  
"و أخصّ بالذكر روح البراءة : ابن أختي "عاصم".

إلى صاحبة الحنجرة الذهبية و النوايا الصادقة معلّمة القرءان "الأخت سعاد".  
إلى كلّ من أحببتهم وأحبّوني صديقاتي : نعيمة، توحه، فاطمة، بركة.

إلى أبناء أعمامي: هداية - محمّد، إخلاص، إسراء، كرام، لقمان، أبو بكر، خديجة.  
إلى عائلتي بشيري وهدراش.

إلى كل من ساعدني أو شجّعني أو دعا لي بظهر الغيب.

بشيري سميرة

# Dédicace

Je dédie ce travail A

Mes très chers parents qui ne serai jamais comment exprimer mes sentiments pour leurs tendresse qui l'ont toujours accomplis avec dévouements pour me permettre de réussir ma vie qu'ils trouvent ici l'expression de mos profond respect.

Mon très chère marie : Boucif Abdelilah qui fait tout la possibilité pour finir mon étude à bon résultat et pour réussir dans ma vie.

Mes très chers frères : Abdullah et Abderrahmane.

Mes très chers sœurs : Meriem et son marie et ses enfants : Nouhe et ferdaws, Zineb et son marie et son enfant : Islem Mahmoud, Chifaà et Safia.

A tout la famille : Belarbi et Boucif.

A tout mes amis : Faten, Meriem, Imane, Akila et Ilham.

A mon binôme : Bachiri Soumia et sa famille.

Baraka

## *Remerciements*

Avant tout, nous remercions **ALLAH** pour ses faveurs et ses grâces, et de nous avoir donné le courage, la santé et la patience pour terminer notre mémoire.

Nos vifs remerciements à Madame **DIDI Fedoua**, notre encadreur, pour nous avoir fait l'honneur de nous encadrer, et nous guider par ses conseils avisés et son aide très précieuse.

Nos remerciements s'adressent également à l'administration et aux enseignants de département d'informatique pour les moyens qu'ils ont mis à notre disposition afin d'élaborer notre projet.

Nous nous devons aussi remercier les membres du jury qui ont accepté d'examiner ce travail.

Pour finir, Nos sincères remerciements à tous ceux qui nous ont aidés de loin ou de près à réaliser ce modeste travail.

## Table des matières

Abréviation.....	3
Liste des figures.....	4
Introduction Générale .....	5
Chapitre 1 : Généralités sur la Téléphonie sur IP.....	8
<b>1 Historique .....</b>	<b>9</b>
<b>2 La téléphonie par circuit et par paquets .....</b>	<b>9</b>
<b>3 Définition de la ToIP.....</b>	<b>11</b>
<b>4 Principe de fonctionnement de la ToIP .....</b>	<b>11</b>
<b>5 Définition de VoIP .....</b>	<b>13</b>
<b>6 Les architectures de la ToIP .....</b>	<b>13</b>
6.1 Architecture de la téléphonie classique d'entreprise.....	14
6.2 Architecture VoIP d'entreprise « architecture hybride ».....	14
6.3 Architecture VoIP d'entreprise « architecture Full-IP ».....	14
<b>7 Les scénarios de la ToIP.....</b>	<b>15</b>
7.1 Téléphonie entre deux ordinateurs (pc to pc).....	15
7.2 Téléphonie entre PC et poste téléphonique (pc to phone).....	16
7.3 Téléphonie entre PC et deux postes téléphonique (phone to phone)...	16
<b>8 La différence entre la ToIP et la VoIP.....</b>	<b>17</b>
<b>9 Les avantages de la ToIP.....</b>	<b>18</b>
<b>10 Les problèmes clés de la ToIP.....</b>	<b>19</b>
<b>11 Les différences élément pouvant composer un réseau.....</b>	<b>20</b>
<b>12 Protocoles liés à la ToIP.....</b>	<b>23</b>
12.1 Les protocoles de signalisation.....	23
12.1.1 Le protocole H323.....	24
12.1.2 Le protocole SIP.....	26
12.1.3 Le protocole MGCP.....	30
12.2 Les protocoles de transport .....	30
12.2.1 Le protocole RTP.....	30
12.2.2 Le protocole RTCP .....	31
<b>13 Comparaison entre les protocoles .....</b>	<b>31</b>
<b>14 Conclusion.....</b>	<b>32</b>
Chapitre 2 : Déploiement du service de la TOIP.....	34

<b>1</b>	<b>Introduction.....</b>	<b>34</b>
<b>2</b>	<b>Présentation d'Asterisk.....</b>	<b>34</b>
<b>3</b>	<b>Fonctionnalités d'Asterisk .....</b>	<b>35</b>
<b>4</b>	<b>Installation d'Asterisk.....</b>	<b>36</b>
<b>5</b>	<b>Configuration d'Asterisk.....</b>	<b>38</b>
<b>6</b>	<b>Configuration de FreePBX.....</b>	<b>38</b>
<b>7</b>	<b>Déploiement de la ToIP en utilisant un softphone commercial.....</b>	<b>41</b>
	7.1 Présentation de softphone Jitsi.....	41
	7.2 Fonctionnalités de Jitsi.....	42
	7.3 Configuration de Jitsi.....	42
<b>8</b>	<b>Configuration de téléphone IP.....</b>	<b>43</b>
<b>9</b>	<b>Tests et résultats.....</b>	<b>45</b>
<b>10</b>	<b>Déploiement de la ToIP en utilisant notre interface.....</b>	<b>45</b>
	10.1 Réalisation de l'interface graphique.....	46
	10.2 Intégration des fonctions d'appel à l'interface graphique.....	47
<b>11</b>	<b>Conclusion.....</b>	<b>47</b>
	Conclusion générale.....	48
	Bibliographie.....	50
	Résumé.....	52
	Annexe.....	53

# Abréviations

<b>ADSL</b>	Asymmetrical Data Subscriber Line.
<b>DNS</b>	Domain Name System.
<b>GNU</b>	GNU is not Unix.
<b>GPL</b>	General Public Licence.
<b>HTTP</b>	HyperText Transfer Protocol.
<b>IETF</b>	Internet Engineering Task Force.
<b>IP</b>	Internet Protocol.
<b>IPBX</b>	Internet Protocol Private Branch eXchange.
<b>IPv6</b>	Internet Protocol version.
<b>LAN</b>	Local Area Network.
<b>MGCP</b>	Media Gateway Control Protocol.
<b>PSTN</b>	Public Switched Telephone Network.
<b>RFC</b>	Request For Comment.
<b>RTC</b>	Réseau téléphonique commuté.
<b>RTCP</b>	Real-Time Control Protocol .
<b>RTP</b>	Real-Time transport Protocol.
<b>SDP</b>	Session Description Protocol.
<b>SIP</b>	Session Initiation Protocol.
<b>SIPS</b>	Session Initiation Protocol Secure.
<b>ToIP</b>	Telephony over Internet Protocol.
<b>UAC</b>	User Agent Client.
<b>UAS</b>	User Agent Server.
<b>UDP</b>	User Datagram Protocol.
<b>VoIP</b>	Voice over Internet Protocol.

## Liste des figures

<b>Figure 1.1:</b> La technique de transfert de paquets.....	10
<b>Figure 1.2 :</b> Un flot de paquets téléphonique.....	11
<b>Figure 1.3 :</b> Équipement à traverser par une communication téléphonique sur IP....	12
<b>Figure 1.4 :</b> Schéma général de la plateforme. ....	13
<b>Figure1.5 :</b> Architecture du réseau de téléphonie classique d'entreprise .....	14
<b>Figure 1.6 :</b> Architecture VoIP d'entreprise « architecture Full-IP ».....	15
<b>Figure 1.7 :</b> Architecture VoIP « architecture type centrex ». ....	15
<b>Figure 1.8 :</b> Pc to Pc.....	16
<b>Figure 1.9 :</b> Pc to phone.....	16
<b>Figure 1. 10:</b> Phone to phone.....	17
<b>Figure 1. 11:</b> Phone to phone «boitier ».....	17
<b>Figure 1.12 :</b> La différence entre la ToIP et la VoIP.....	18
<b>Figure 1.13 :</b> Protocolaire Pile « H.323 ».....	24
<b>Figure 1.14 :</b> Communication H.323.....	26
<b>Figure 1.15 :</b> la pile SIP.....	27
<b>Figure 1.16 :</b> Syntaxe d'une adresse SIP.....	28
<b>Figure 1.17:</b> Architecture SIP.....	30
<b>Figure 2. 1:</b> Topologie du réseau test.....	34
<b>Figure 2.2:</b> Fonctionnalités offertes par Asterisk . ....	35
<b>Figure 2.3 :</b> Capture ecran de notre interface graphique.....	46



# Introduction Générale

Les technologies de l'information se développent régulièrement en ouvrant de nouvelles possibilités de communications. Depuis l'émergence et l'extraordinaire développement de l'Internet, les mentalités changent et on s'habitue au principe de réseau informatique et de son accès forfaitaire. Bien que l'Internet se développe rapidement, le téléphone reste encore le favori du public en matière de communication. Plus convivial car le contact est presque réel, il reste en plus simple à utiliser. Pourtant, il fusionne de plus en plus avec le matériel informatique. L'existence de deux réseaux parallèles (réseaux téléphoniques et Internet) et la double facturation (pour le téléphone et pour l'accès à Internet) ont fait naître une nouvelle technologie : la téléphonie sur IP (Telephony over Internet Protocol).

Par cette technologie on entend le fait que deux personnes ou plus puissent se parler entre elles à partir de différents « terminaux » et par le biais de n'importe quel réseau informatique ayant un protocole IP comme les réseaux LAN. Aujourd'hui, la téléphonie sur IP est devenue un service réseau comme le web, la messagerie, le transfert de fichiers, etc. Cependant la complexité de ce service et la nécessité de prise en charge de différentes architectures (pc-pc, téléphone-pc, téléphone- téléphone), et la qualité exigée dans de telles applications, tout cela a conduit au développement de protocoles spécifiques pour la gestion de ce service.

L'objectif de notre travail était d'implémenter le service ToIP sur un petit réseau LAN constitué de deux terminaux et un autocommutateur. Nous avons choisi de réaliser ce travail en deux étapes :

1<sup>ière</sup> étape : En utilisant des softphones commerciaux (Jitsi, Xlite...).

2<sup>ième</sup> étape : En remplaçant le softphone commercial par une interface graphique que nous réalisons nous même. Cette interface se limite à la simple fonction d'appel et de réponse à un appel téléphonique.

Dans les deux cas on essaye d'établir un appel sur IP entre deux terminaux téléphoniques selon deux architectures :

- Une liaison PC-PC (c'est-à-dire un appel à partir d'un softphone installé sur un PC vers un softphone installé sur un autre PC).
- Une liaison Téléphone-PC (c'est-à-dire un appel à partir d'un téléphone IP vers un softphone installé sur un PC ou vice versa).

Cette liaison sera matérialisée par le biais d'un autocommutateur téléphonique appelé: Asterisk, il s'agit d'un logiciel libre très fiable permettant d'implémenter toutes les fonctionnalités d'un PABX moderne.

Ce mémoire se présente en deux chapitres, le premier chapitre dresse le cadre théorique de notre travail, le deuxième constitue la partie application, il comprend une description détaillée de ce que nous avons menée dans le cadre de ce projet.

# **Chapitre 1**

## **Généralités**

### **Sur la Téléphonie sur IP.**

Dans ce chapitre nous présentons les notions de base utiles à la compréhension de la ToIP ainsi que les concepts généraux des architectures réseau nécessaires pour mettre en œuvre un système de ToIP. Les protocoles standardisés utilisables pour transporter des flux multimédia et de données seront détaillés ensuite. Ces éléments communs étant définis, les solutions majeures disponibles seront brièvement exposées : H.323, SIP, MGCP.

## 1. Historique

Durant les années 1870, essayant de comprendre le son et les communications sonores, l'inventeur d'origine écossaise Alexander Graham Bell eut l'idée d'un équipement qui transmettrait le son sur de longues distances, en convertissant le son en signal électrique. Cet équipement fut ensuite appelé téléphone .De nombreux autres développements furent réalisés autour de cet équipement à la fin de l'année 1870. Bell fut à l'origine de l'écouteur (inducteur), et Thomas Edison fut le concepteur du microphone (à base de carbone). L'incorporation de ces améliorations réalisa du téléphone un objet utilisable pratiquement.[8]

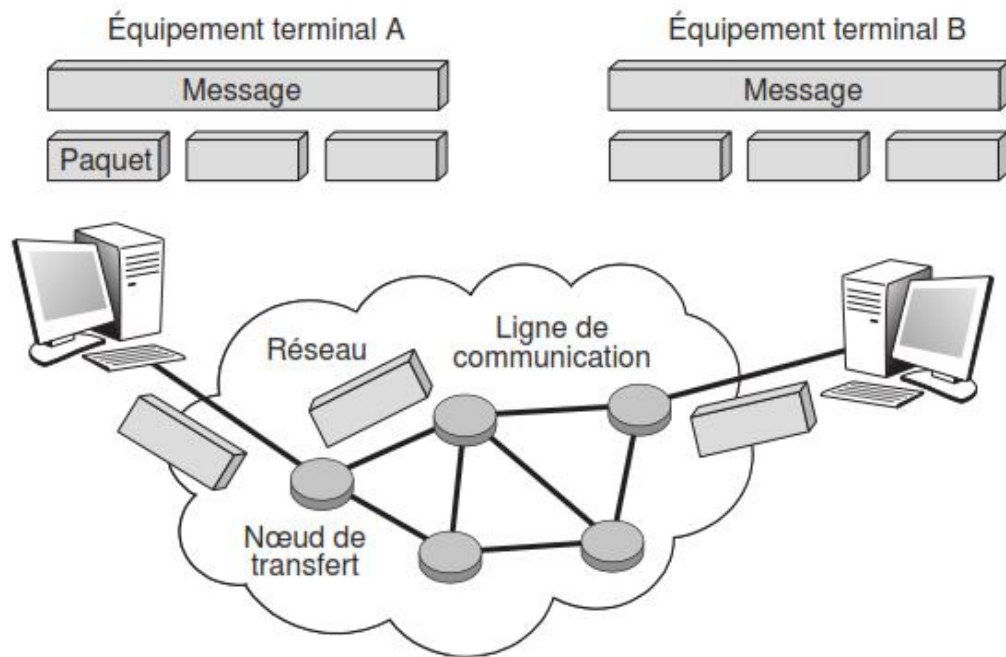
Les réseaux téléphoniques sont passés sur plusieurs changements depuis cette époque. Les protocoles utilisés étaient principalement le SS7 ou l'ISDN. Le premier se déclinant en deux grandes variantes : ITU (Europe) et ANSI (Amérique). Avant l'arrivée de la téléphonie sur IP, le réseau ainsi réalisé set appelé « Réseau téléphonique commuté RTC (PSTN). Cependant la plus part des techniques restent les mêmes. Jusqu'au milieu des années 90, les organismes de normalisation ont tenté de transmettre les données de manière toujours plus efficace sur des réseaux téléphonique. A partir de cette date, il y a eu changement de paradigme. C'est sur les réseaux de données, en particulier sur l'Internet, que l'on s'est évertué à convoier la parole. [1]

## 2. La téléphonie par circuit et par paquets

Dans la communication à transfert de paquets, toutes les informations à transporter sont découpées en paquets pour être acheminées d'une extrémité à une autre du réseau. Cette technique est illustrée à la figure 1.1.[4]

L'équipement terminal A souhaite envoyer un message à B. Le message est découpé en trois paquets, qui sont émis de l'équipement terminal vers le premier nœud du réseau, lequel les envoie à un deuxième nœud, et ainsi de suite, jusqu'à ce qu'ils arrivent à

l'équipement terminal B. Dans l'équipement terminal les paquets rassemblés reconstituent le message de départ.



**Figure 1.1:** La technique de transfert de paquets.

Le paquet peut en fait provenir de différents médias. Sur la figure 1.1, nous supposons que la source est un message composé de données, comme une page de texte préparée au moyen d'un traitement de texte. Le terme message est cependant beaucoup plus vaste et recoupe toutes les formes sous lesquelles de l'information peut se présenter. Cela va d'une page Web à un flot de parole téléphonique représentant une conversation.

Dans la parole téléphonique, l'information est regroupée pour être placée dans un paquet, comme illustré à la figure 1.2. Le combiné téléphonique produit des octets, provenant de la numérisation de la parole, c'est-à-dire le passage d'un signal analogique à un signal sous forme de 0 et de 1, qui remplissent petit à petit le paquet. Dès que celui-ci est plein, il est émis vers le destinataire. Une fois le paquet arrivé à la station terminale, le processus inverse s'effectue, restituant les éléments binaires régulièrement à partir du paquet pour reconstituer la parole téléphonique.

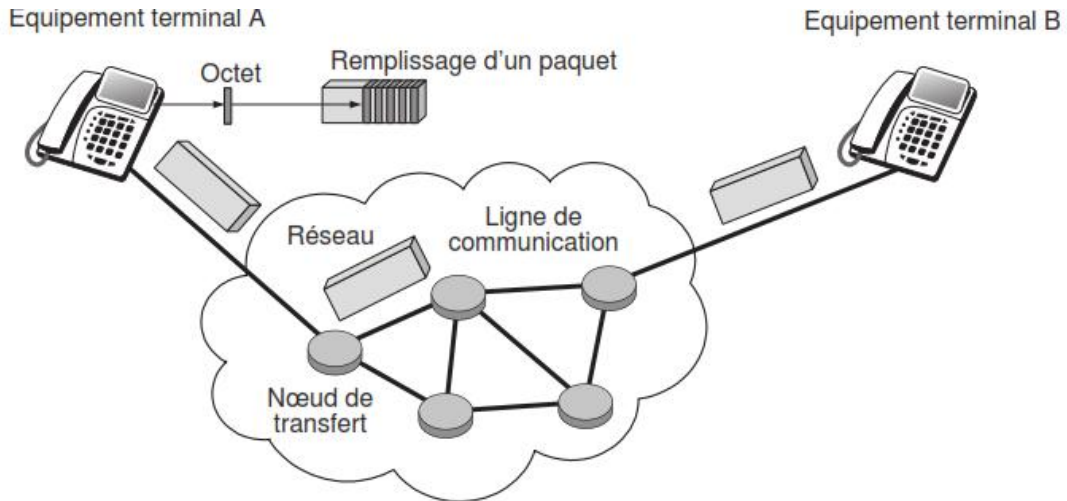


Figure 1.2 : Un flot de paquets téléphonique.

### 3. Définition de la ToIP

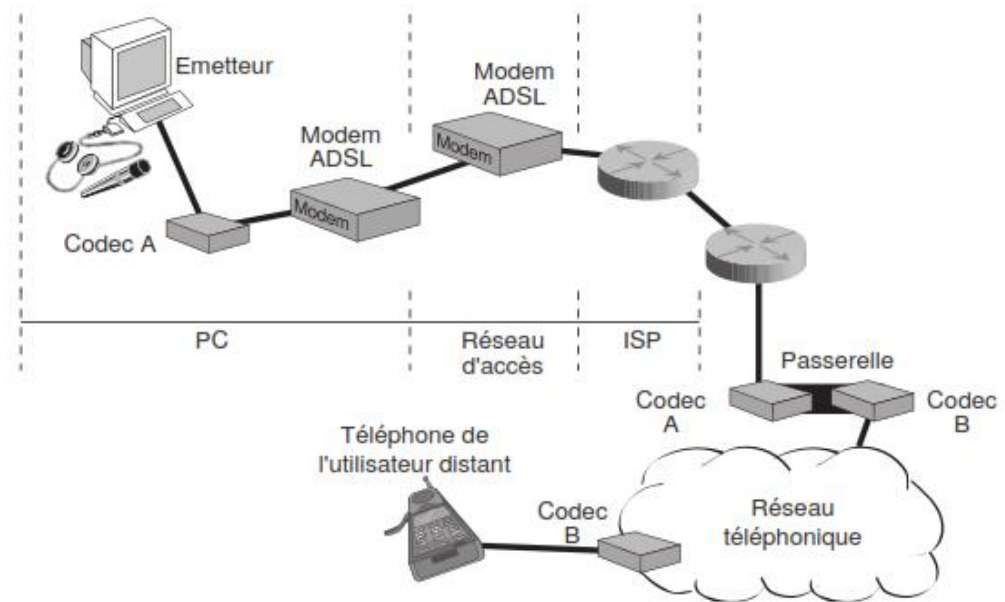
La téléphonie sur IP correspond à la transmission de la voix et des données sur une seule infrastructure IP. L'objectif donc est d'utiliser un réseau existant IP (intranet, LAN, WAN, etc..) qui n'est pas dédié à la téléphonie pour effectuer des conversations vocales grâce au protocole IP. A la différence du réseau RTC qui fonctionne par transmission des signaux sur un réseau de commutation de circuit, la téléphonie sur IP utilise la commutation de paquets. Ainsi le signal numérique obtenu par numérisation de la voix est découpé en paquets qui sont véhiculés sur le réseau IP jusqu'à sa destination, ou une application se chargera de la transformation inverse (paquets vers voix). Au lieu de disposer à la fois d'un réseau téléphonique commuté (RTC) et d'un réseau informatique, l'entreprise peut donc tout fusionner sur un même réseau. [2]

### 4. Principe de fonctionnement de la ToIP

Le déroulement d'une communication téléphonique sur IP parcourt les cinq grandes étapes suivantes : [4]

- a) **Mise en place de la communication** : Une signalisation démarre la session. Le premier élément à considérer est la localisation du récepteur (*User Location*). Elle s'effectue par une conversion de l'adresse du destinataire (adresse IP ou adresse téléphonique classique) en une adresse IP d'une machine qui puisse joindre le destinataire (qui peut être le destinataire lui-même). Le récepteur peut être un combiné téléphonique classique sur un réseau d'opérateur télécoms ou

une station de travail (lorsque la communication s'effectue d'un combiné téléphonique vers un PC). Le protocole DHCP (Dynamic Host Configuration Protocol) et les passerelles spécialisées (*gatekeeper*) sont employés à cette fin.



**Figure 1.3** : Équipement à traverser par une communication téléphonique sur ip.

- b) **Établissement de la communication.** : Cela passe par une acceptation du terminal destinataire, que ce dernier soit un téléphone, une boîte vocale ou un serveur Web. Plusieurs protocoles de signalisation sont utilisés pour cela, en particulier le protocole SIP (Session Initiation Protocol) de l'IETF. Comme son nom l'indique, SIP est utilisé pour initialiser la session. Une requête SIP contient un ensemble d'en-têtes, qui décrivent l'appel, suivis du corps du message, qui contient la description de la demande de session. SIP est un protocole client-serveur, qui utilise la syntaxe et la sémantique de HTTP. Le serveur gère la demande et fournit une réponse au client.
- c) **Transport de l'information téléphonique** : Le protocole RTP (Real-time Transport Protocol) prend le relais pour transporter l'information téléphonique proprement dite. Son rôle est d'organiser les paquets à l'entrée du réseau et de les contrôler à la sortie de façon à reformer le flot avec ses caractéristiques de départ (vérification du synchronisme, des pertes, etc.). C'est un protocole de niveau transport, qui essaye de corriger les défauts apportés par le réseau.



- d) **Changement de réseau** : Un autre lieu de transit important de la ToIP est constitué par les passerelles, qui permettent de passer d'un réseau à transfert de paquets à un réseau à commutation de circuits, en prenant en charge les problèmes d'adressage, de signalisation et de transcodage que cela pose. Ces passerelles ne cessent de se multiplier entre FAI et opérateurs télécoms.
- e) **Arrivée au destinataire** : De nouveau, le protocole SIP envoie une requête à la passerelle pour déterminer si elle est capable de réaliser la liaison circuit de façon à atteindre le destinataire. En théorie, chaque passerelle peut appeler n'importe quel Numéro de téléphone. Cependant, pour réduire les coûts, mieux vaut choisir une passerelle locale, qui garantit que la partie du transport sur le réseau téléphonique classique est le moins cher possible. [4]

## 5. Définition de VoIP

La voix sur IP, ou « VoIP » pour Voice over IP, est une technique qui permet de communiquer par la voix (ou *via* des flux multimédia : audio ou vidéo) sur des réseaux compatibles IP, qu'il s'agisse de réseaux privés ou d'Internet, filaire (câble/ADSL/optique) ou non (satellite, Wi-Fi, GSM, UMTS ou LTE). La VoIP concerne le transport de la voix sur un réseau IP. Cette technologie est complémentaire de la téléphonie sur IP (« ToIP » pour Telephony over Internet Protocol). [2]

## 6. Les architectures de la ToIP

La téléphonie sur IP peut être déployée en entreprise de plusieurs manières, en fonction du degré de convergence désiré et en tenant compte de certaines mesures (budget, équipement, etc.).

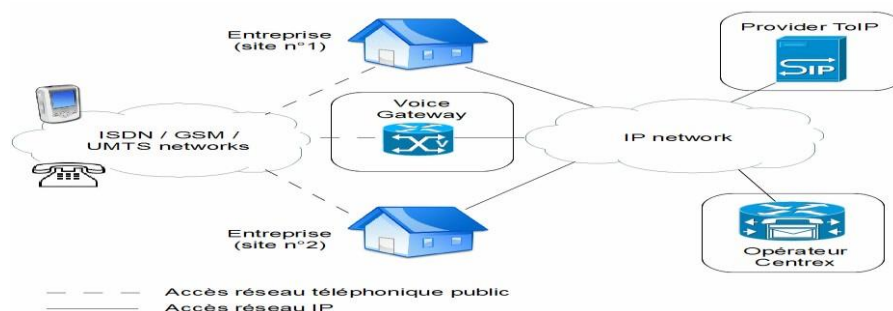
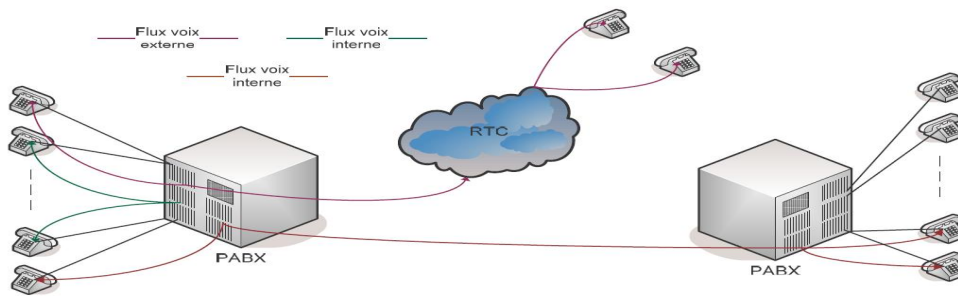


Figure 1.4 : Schéma général de la plateforme.

## 6.1 Architecture de la téléphonie classique d'entreprise

En architecture de la téléphonie classique, l'ensemble des flux voix et signalisation est centralisé au niveau du PABX de chaque site, pendant toute la durée d'une communication. Cette architecture est la plus répandue dans la grande majorité des contextes « entreprises ». Les architectures de téléphonie sur IP qui suivent sont à mettre en parallèle avec le réseau existant afin d'envisager les évolutions à conduire dans le cadre d'une migration vers le déploiement d'une solution full-IP.



**Figure 1.5 :** Architecture du réseau de téléphonie classique d'entreprise [3]

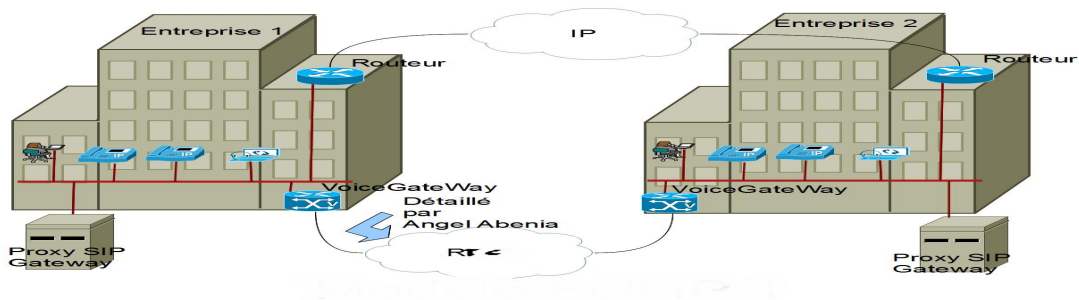
## 6.2 Architecture VoIP d'entreprise « architecture hybride »

Cette solution présente comme avantage de ne pas remettre en cause l'infrastructure existante tout en bénéficiant des avantages du transport de la voix sur IP pour les communications inter-site. La mise en œuvre de cette solution peut se faire soit par l'ajout d'un boîtier « Voice Gateway » externe au PABX, soit par un recours aux fonctionnalités de Gateway intégrées aux routeurs de nouvelle génération (sous forme de carte).

Généralement les fonctionnalités de téléphonie liées aux protocoles de signalisation propre au PABX sont perdues lors du passage par la Voice Gateway. Ce déploiement peut concerner, dans un premier temps, seulement le transport inter-site, et peut constituer la première étape de la migration vers le full-IP.

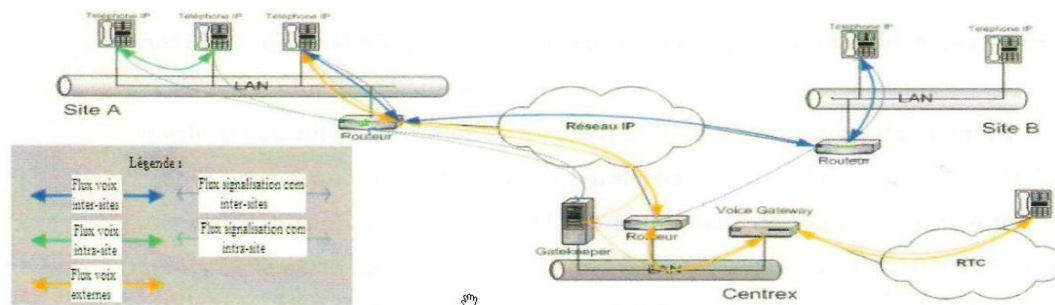
## 6.3 Architecture VoIP d'entreprise « architecture Full-IP »

Plus lourde qu'une solution hybride, l'architecture full-IP présente une migration totale vers la téléphonie sur IP de l'ensemble de l'entreprise, incluant les terminaux téléphoniques utilisateurs. Cette migration s'accompagne de nombreux bénéfices en posant les bases de la convergence entre le système informatique et la téléphonie de l'entreprise.



**Figure 1.6 :** Architecture VoIP d'entreprise « architecture Full-IP »

La Voice Gateway sera la passerelle d'accès vers le RTC, et lors d'une communication inter ou intra-site seuls les flux de signalisation transitent par le Gatekeeper. L'entreprise peut aussi choisir de diminuer son investissement en choisissant d'externaliser les fonctions « Gatekeeper/Voice Gateway » chez un fournisseur centrex IP. Ainsi l'intelligence sera déportée dans le cœur du réseau. Puisque les échanges avec le Gatekeeper sont limités aux flux de signalisation, l'externalisation n'implique plus, comme précédemment, un transit systématique des flux voix par le site du fournisseur. Ceci est illustré dans le schéma suivant:



**Figure 1.7 :** Architecture VoIP « architecture type centrex ».

## 7. Les scénarios de la ToIP

Selon le type de terminal utilisé, on distingue trois scénarios possibles de téléphonie sur IP : [3,5]

### 7.1 Téléphonie entre deux ordinateurs (pc to pc)

Dans ce scénario le but sera de transformer son ordinateur en un poste téléphonique en lui ajoutant une carte son full-duplex pour garantir une conversation simultanée, un micro et un logiciel de voix sur IP compatible. Le correspondant quant à lui, doit disposer des mêmes outils et surtout du même logiciel de téléphonie. A cet instant, le

poste numérique, compresse et encapsule les échantillons de voix dans des paquets IP avant de les envoyer sur Internet. L'accès se fait via un fournisseur d'accès à internet IAP/ISP.

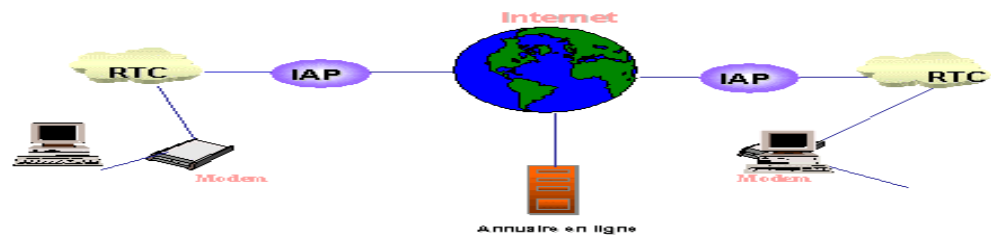


Figure 1.8: Pc to Pc

## 7.2 Téléphonie entre PC et poste téléphonique (pc to phone)

Ce sont donc à la fois le réseau Internet et le réseau téléphonique commuté qui sont utilisés dans ce mode de communication. Le service n'est plus gratuit puisque le réseau RTC est généralement facturé à l'usage et non forfaitairement. Grâce à ces crédits, les utilisateurs peuvent communiquer partout dans le monde, à des tarifs très avantageux, une bonne partie de la communication transitant sur le réseau IP, y compris la partie qui relie l'abonné appelant à son opérateur.

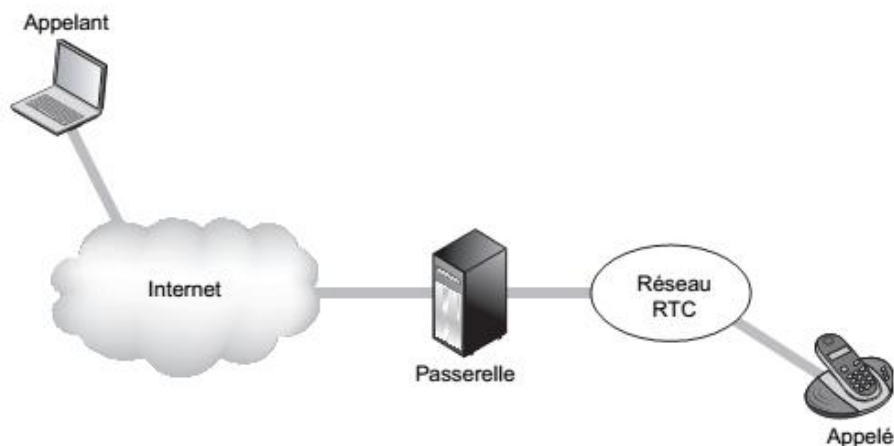


Figure 1.9 : Pc to phone [4]

## 7.3 Téléphonie entre PC deux postes téléphonique (phone to phone)

Dans ce cas l'appelant et l'appelé sont tous les deux des abonnés du réseau téléphonique commuté public (RTCP) et utilisent de manière classique leur appareil téléphonique pour la communication vocal.

On peut distinguer deux méthodes pour faire dialoguer deux postes téléphoniques ordinaires via un réseau IP ou internet :

**a) En utilisant des passerelles**

Dans ce cas, les passerelles ainsi que le réseau IP géré pourraient appartenir à des acteurs différents selon qu'il s'agit:

- D'un usage purement interne de la voix sur IP au sein du réseau d'un opérateur téléphonique unique (usagers A et B ainsi gérés).
- De la fourniture d'un service de voix longue distance par un opérateur longue distance utilisant la technologie de la voix sur IP (les usagers A et B appartenant alors à des réseaux distincts).



Figure 1.10: Phone to phone

**b) En utilisant des boîtiers d'adaptation**

Pour faire bénéficier de ce service, un certain nombre de sociétés commercialisent des boîtiers ressemblant à des modems et qui s'interpose entre le poste téléphonique de l'utilisateur et son branchement au réseau téléphonique public commuté.

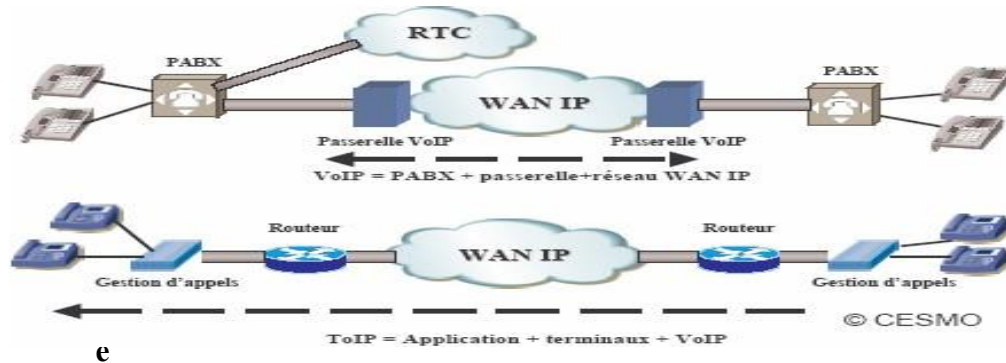


Figure 1. 11: Phone to phone «boitier».

## 8. La différence entre la ToIP et la VoIP

Nous faisons souvent un amalgame entre la téléphonie sur IP et la voix sur IP. Cela est normal, car les deux concepts sont très proches. La nuance réside dans le fait que la VoIP est incluse dans la ToIP.

La VoIP représente seulement la technologie de transport de voix sur le protocole Internet. La ToIP, représente la VoIP en addition de toutes les applications téléphoniques qu'il peut y avoir. Le schema ci-dessous explique cette différence:



1.12 : La différence entre la ToIP et la VoIP.

## 9. Les avantages de la ToIP

La téléphonie n'a jamais été une application simple. Les contraintes temps réel et de synchronisation pèsent lourdement sur sa mise en œuvre, et la téléphonie par paquet ne fait que compliquer le transport. Cependant, plusieurs raisons expliquent le succès de la téléphonie par paquet, et plus spécifiquement de la téléphonie sur IP : [4]

- **Convergence** : Quel que soit le type de données véhiculées, le réseau est unique : les flux de voix, de vidéo, de textes et d'applicatifs transitent sur le même réseau. Les communications deviennent plus riches, et sans avoir besoin de multiplier les canaux de transport. Les utilisateurs peuvent, par exemple, envoyer un compte rendu d'activité en même temps qu'ils téléphonent à leur correspondant. Pour les utilisateurs, la convivialité est accrue. En entreprise, la productivité est améliorée. Pour les administrateurs, un seul réseau est à administrer, ce qui simplifie grandement la gestion.
- **Coût de transport quasiment nul** : Grâce à l'intégration de la téléphonie parmi de nombreuses autres applications, le coût du transport devient pratiquement nul. Le réseau permettant d'effectuer le transport est le réseau cœur des opérateurs, celui qui effectue tous les transports de données. Ces opérateurs, qui étaient auparavant obligés de maintenir au moins deux réseaux, celui de téléphonie et celui de données, n'en ont plus qu'un seul à maintenir.

L'intégration supplémentaire de la télévision dans le réseau de données fait également chuter les coûts de transport de cette application.

- **Services exclusifs :** Certains services sont propres aux réseaux IP. Par exemple, le service de présence, consistant à détecter si un utilisateur est connecté au réseau ou non, ne nécessite aucune réservation de ressources dans un réseau IP, à la différence du réseau RTC. De façon analogue, pour le nomadisme des utilisateurs, il est plus simple de passer, partout dans le monde, par le réseau IP plutôt que par le réseau RTC.
- **Flexibilité:** Les solutions de téléphonie sur IP sont conçues pour assumer une stratégie de migration à faible risque à partir de l'infrastructure existante. La transition de la solution actuelle vers la téléphonie sur IP peut donc s'effectuer en douceur. De plus, la communication par Internet offre la gratuité des communications intersites ainsi qu'une facilité d'intégration des sièges distants. Egalement, les standards ouverts (interopérabilité) permettent de changer de prestataire et d'interconnecter du matériel de fournisseurs différents.

## 10. Les problèmes clés de la ToIP

La téléphonie sur IP va inéluctablement remplacer la téléphonie numérique classique. Les enjeux sont considérables puisque l'ensemble des entreprises aura adopté cette technologie dans les dix années à venir.

Les cinq problèmes clés auxquels il est important de réfléchir avant de décider de passer à la ToIP. Ces problèmes sont les suivants : [4 , 3]

- **Sécurité :** Dans les versions classiques de la téléphonie, la sécurité est fortement garantie par un réseau spécifique, lequel ne peut être attaqué par l'émission de paquets d'attaque puisque le réseau n'est pas à transfert de paquets. Dans la ToIP, la confidentialité est assez simple à garantir par le biais de tunnels. Reste le problème de l'authentification de l'utilisateur, qui mérite réflexion.
- **Disponibilité :** Dans la téléphonie classique, la disponibilité est aux 5 « neuf », c'est-à-dire que le système est en état de marche 99,999 % du temps. Dans la ToIP, elle passe aux 3 « neuf », soit 99,9 %, avec un bon fournisseur de service

IP et plutôt moins en général. La question est de savoir comment prendre en compte cette problématique pour revenir à des disponibilités plus acceptables.

- **Gestion:** la gestion du réseau téléphonique commuté est relativement simple, puisqu'elle consiste à maintenir des circuits téléphoniques. Avec l'intégration de la ToIP dans le réseau de données, la gestion de l'environnement téléphonique devient beaucoup plus complexe. Comment la nouvelle génération de réseaux intégrant la ToIP va-elle pouvoir répondre à cette question ?
- **Contrôle :** comme la gestion, le contrôle de la téléphonie est assez simple dans l'environnement unique des circuits numériques. L'intégration de la ToIP dans un réseau de données global complexifie grandement le contrôle, alors même qu'il s'agit d'un service crucial compte tenu des contraintes temps réel de l'application de téléphonie.
- **Qualité de service :** la téléphonie par paquets est une application complexe, pour laquelle une excellente qualité de service est nécessaire. Cette problématique ayant été amplement commentée tout au long de l'ouvrage, nous n'y reviendrons que pour en résumer l'essentiel.

## 11. Les différents éléments pouvant composer un réseau

- **Le PABX-IP,** c'est lui qui assure la commutation des appels et leurs autorisations, il peut servir aussi de routeur ou de Switch dans certains modèles, ainsi que de serveur DHCP. Il peut posséder des interfaces de type analogiques (fax), numériques (postes), numériques (RNIS, QSIG) ou opérateurs (RTC-PSTN ou EURO-RNIS). Il peut se gérer par IP en intranet ou par un logiciel serveur spécialisé que ce soit en interne ou depuis l'extérieur. Il peut s'interconnecter avec d'autres PABX-IP ou PABX non IP de la même marque (réseau homogène) ou d'autres PABX d'autres marques (réseau hétérogène).





- **Le serveur de communications** (exemple : Call Manager de Cisco), il gère les autorisations d'appels entre les terminaux IP ou softphones et les différentes signalisations du réseau. Il peut posséder des interfaces réseaux opérateurs (RTC-PSTN ou RNIS), sinon les appels externes passeront par la passerelle dédiée à cela (Gateway).



- **La passerelle (Gateway)**, c'est un élément de routage équipé de cartes d'interfaces analogiques et/ou numériques pour s'interconnecter avec soit d'autres PABX (en QSIG, RNIS ou E&M), soit des opérateurs de télécommunications local, national ou international. Plusieurs passerelles peuvent faire partie d'un seul et même réseau, ou l'on peut également avoir une passerelle par réseau local (LAN). La passerelle peut également assurer l'interface de postes analogiques classiques qui pourront utiliser toutes les ressources du réseau téléphonique IP (appels internes et externes, entrants et sortants).



- **Le routeur**, il assure le routage des paquets d'un réseau vers un autre réseau.



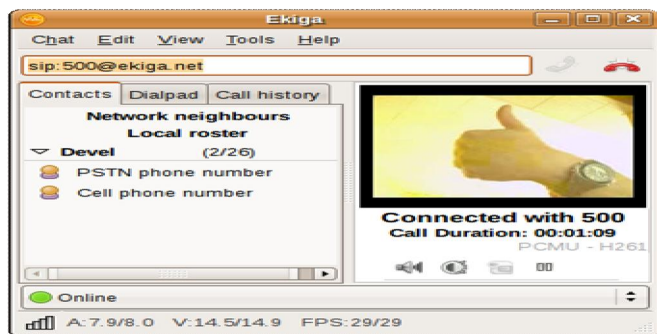
- **Le Switch**, il assure la distribution et commutation de dizaines de port Ethernet à 10/100 voire 1000 Mbits/s. Suivant les modèles, il peut intégrer la télé alimentation des ports Ethernet à la norme 802.3af pour l'alimentation des IP-phones ou des bornes



- **Le Gatekeeper**, il effectue les translations d'adresses (identifiant H323 et @ IP du référencement du terminal) et gère la bande passante et les droits d'accès. C'est le point de passage obligé pour tous les équipements de sa zone d'action.
- **Le MCU**, est un élément optionnel et gère les conférences audio vidéo.
- **L'IP-Phone**, c'est un terminal téléphonique fonctionnant sur le réseau LAN IP à 10/100 avec une norme soit propriétaire, soit SIP, soit H.323. Il peut y avoir plusieurs codecs pour l'audio, et il peut disposer d'un écran monochrome ou couleur, et d'une ou plusieurs touches soit programmables, soit préprogrammées. IL est en général doté d'un hub passif à un seul port pour pouvoir alimenter le PC de l'utilisateur (l'IP-PHONE se raccorde sur la seule prise Ethernet mural et le PC se raccorde derrière l'IP-PHONE).



- **Le Softphone** c'est un logiciel qui assure toutes les fonctions téléphoniques et qui utilise la carte son et le micro du PC de l'utilisateur, et aussi la carte Ethernet du PC. Il est géré soit par le Call Manager, soit par le PABX-IP.



## 12. Protocoles liés à la ToIP

La téléphonie sur IP (ToIP) est un service de téléphonie qui transporte les flux voix des communications téléphoniques sur un réseau IP. A la différence de la VoIP où l'on ne fait qu'établir une communication « voix », la ToIP intègre l'ensemble des services associés à la téléphonie : double appel, messagerie, renvoi d'appel, FAX, etc.

Afin de rendre possibles les communications ToIP, les solutions proposées dopent la couche IP par des mécanismes supplémentaires nécessaire au flux voix de types temps réel, en plus de l'intelligence nécessaire à l'exécution de services. A cet effet, il existe deux types de protocoles principaux utilisés dans la ToIP :

- Protocoles de signalisation.
- Protocoles de transport. [6]

### 12.1 Les protocoles de signalisation

La signalisation correspond à la gestion des sessions de communication (ouverture, fermeture, etc.). Le protocole de signalisation permet de véhiculer un certain nombre d'informations notamment: enregistrement d'un utilisateur, invitation à une session multimédia, annulation d'un appel, réponse à une requête, etc.

Plusieurs normes et protocoles ont été développés pour la signalisation ToIP, quelques uns sont propriétaires et d'autres sont des standards. Ainsi, les principales propositions disponibles pour l'établissement de connexions en ToIP sont :

- SIP (Session Initiation Protocol) qui est un standard IETF (Internet Engineering Task Force) décrit dans le RFC 3261.
- H323 englobe un ensemble de protocoles de communication développés par l'UIT-T (Union Internationale des Télécommunications – secteur de la normalisation des Télécommunications).
- MGCP (Media Gateway Control Protocol) standardisé par l'IETF (RFC 3435).

### 12.1.1 Le protocole H323

H.323 est un protocole de communication englobant un ensemble de normes utilisées pour l'envoi de données audio et vidéo sur internet. Il existe depuis 1996 et a été initié par l'IUT. Concrètement, il est utilisé dans des programmes tels que Microsoft NetMeeting, ou encore dans des équipements tels que les routeurs Cisco. Il existe un projet « Open h.323 » qui développe un client H.323 en logiciel libre afin qu'on puisse avoir accès à ce protocole sans avoir à déboursé beaucoup d'argent.

#### a) Briques d'architecture H.323

L'infrastructure H.323 repose sur des éléments réseaux suivants :

- Les portiers (gk : Gatekeeper)
- Les passerelles (gw : Gateway)
- Les terminaux: Dans un contexte de téléphonie sur IP, deux types de terminaux H.323 sont aujourd'hui disponibles :
  - ✓ Un poste téléphonique IP raccordés directement au réseau Ethernet de l'entreprise.
  - ✓ Un PC multimédia sur lequel est installée une application compatible H.323.[3]

#### b) Pile protocolaire H.323

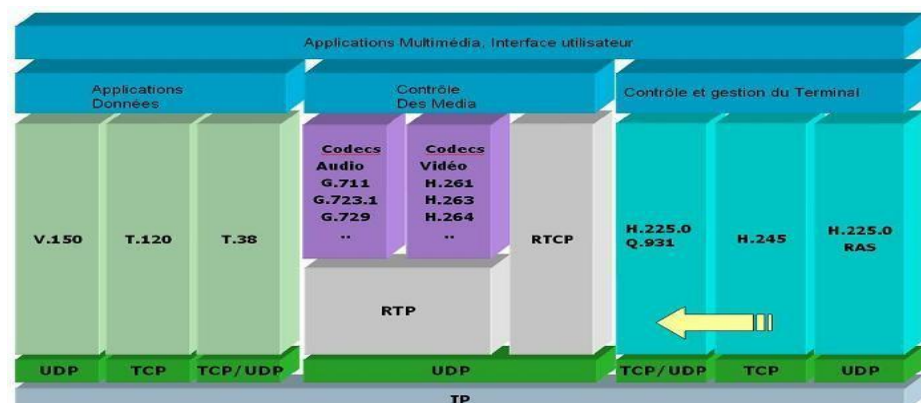


Figure 1.13 : protocolaire Pile « H.323 ».

### c) Signalisation

Dans un contexte de téléphonie sur IP, la signalisation a pour objectif de réaliser les fonctions suivantes :

- Recherche et traduction d'adresses.
- Contrôle d'appel.
- Services supplémentaires: déviation, transfert d'appel, conférence, ...etc.

Trois protocoles de signalisation sont spécifiés dans le cadre de H.323, à savoir :

- RAS (registration, admission and status) : Ce protocole est utilisé pour communiquer avec un Gatekeeper. Il sert notamment aux équipements terminaux pour découvrir l'existence d'un Gatekeeper et s'enregistrer auprès de ce dernier ainsi que pour les demandes de traduction d'adresse. La signalisation RAS utilise des messages H.225.06 transmis sur un protocole de transport non fiable (UDP par exemple).
- Q.931 : H.323 utilise une version simplifiée de la signalisation RNIS Q.931 pour l'établissement et le contrôle d'appels téléphonique sur IP. Cette version simplifiée est également spécifiée dans la norme H.225.
- H.245 : Ce protocole est utilisé pour l'échange de capacités entre deux équipements terminaux. Par exemple, il est utilisé pour s'accorder sur le type de codec à activer. Il peut également servir à mesurer le retard aller-retour (Round Trip Delay) d'une communication.

### d) Etablissement d'une communication H.323

Une communication H.323 se déroule en 5 phases :

- Établissement d'appel.
- Echanges de capacité et réservation éventuelle de la bande passante à travers le protocole RSVP (Ressources Réserveation Protocol)
- Etablissement de la communication audio visuelle.
- Invocation éventuelle de services en phase d'appel (transfert d'appel. changement de bande passante. etc.).
- Libération de l'appel.

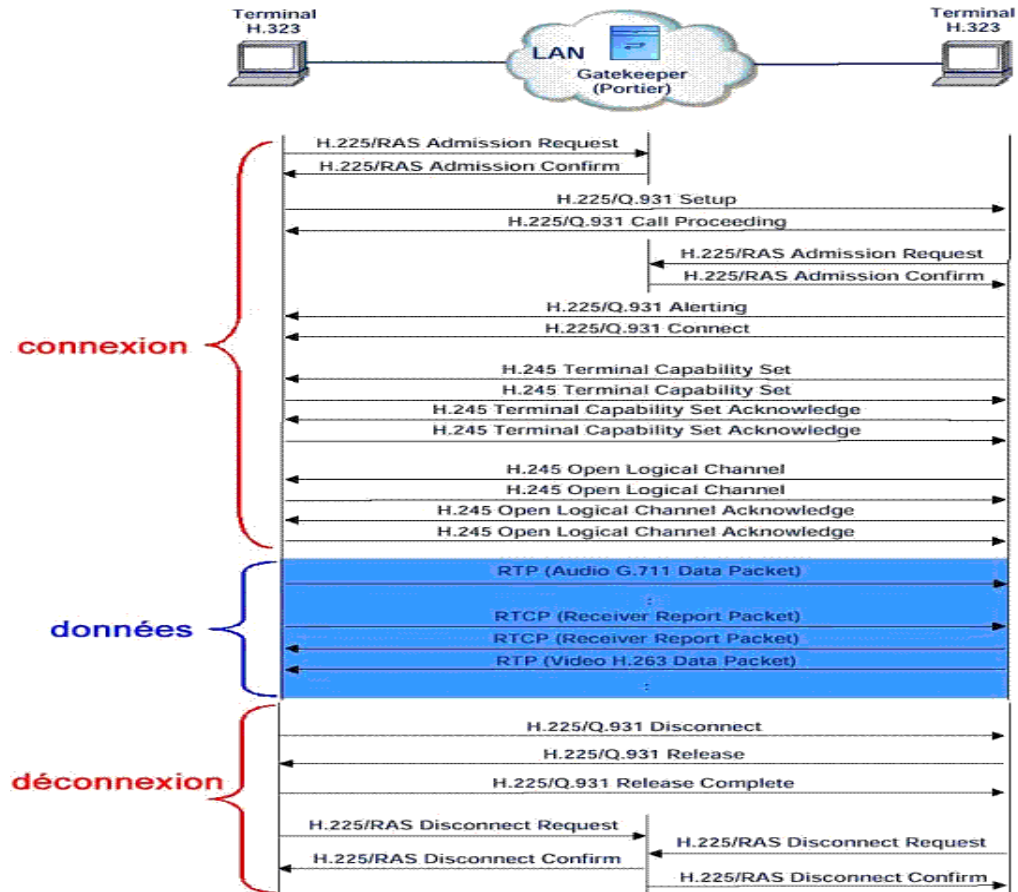


Figure I. 14: Communication H.323.

### 12.1.2 Le protocole SIP

Le protocole SIP (Session Initialisation Protocol) a été initié par le groupe MMUSIC (Multiparty Multimedia Session Control) [RFC 2543] et désormais repris et maintenu par le Groupe SIP de l'IETF [RFC 3261]. SIP est un protocole de signalisation appartenant à la couche application du modèle OSI. Il a été conçu pour l'ouverture, le maintien et la terminaison de sessions de communications interactives entre des utilisateurs. De telles sessions permettent de réaliser de l'audio, de l'enseignement à distance et de la voix (téléphonie) sur IP essentiellement. Pour l'ouverture d'une session, un utilisateur émet une invitation transportant un descripteur de session permettant aux utilisateurs souhaitant communiquer de négocier sur les algorithmes et codecs à utiliser. SIP permet aussi de relier des stations mobiles en transmettant ou redirigeant les requêtes vers la position courante de la station appelée.

Enfin, SIP est indépendant du médium utilisé et aussi du protocole de transport des couches basses.

#### a) Architecture protocolaire

SIP est un protocole indépendant des couches de transport, il appartient aux couches applications du modèle OSI. Le SIP gère la signalisation et l'établissement des sessions interactives de communication multimédias et multipartites. Il est aussi basé sur le concept Client / Serveur pour le contrôle d'appels et des services multimédias. Conçu selon un modèle de type IP, il est hautement extensible et assez simple en conception architecturale, de sorte qu'il peut servir de base à la création d'applications et de services. Il est basé sur le protocole HTTP et peut utiliser UDP ou TCP [6].

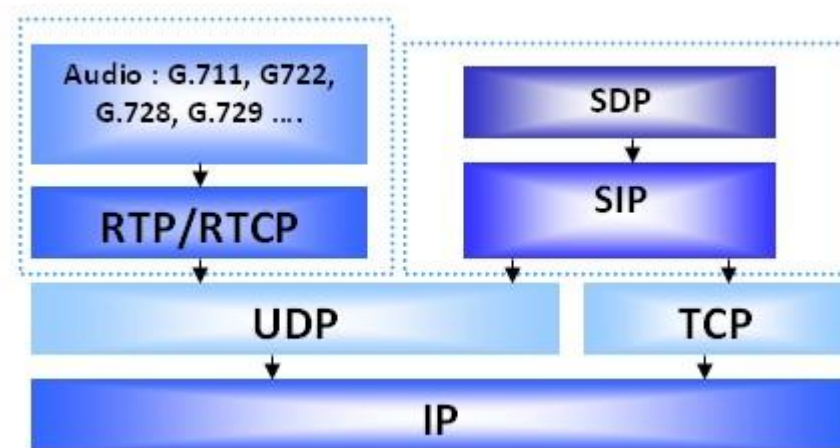


Figure1.15 : la pile SIP

#### b) Les éléments architecturaux de SIP

Contrairement à H.323, largement fondé sur une architecture physique, le protocole SIP s'appuie sur une architecture purement logicielle. [7]

##### (1) Les agents utilisateurs (User Agent UA)

Deux types d'agents utilisateurs sont à considérer :

- **Les agents utilisateurs client (UAC):** ce sont les terminaux qui initient la session par envoi de requête.
- **Les agents utilisateurs serveurs (UAS) :** Ils répondent à la demande d'établissement de session.

Un UA peut être un UAC ou UAS selon le cas où c'est lui qui initie la session ou pas ; mais dans une session déjà établie un UA peut être soit un UAC ou un UAS à la fois.

**(2) Les serveurs :**

Il existe trois types de serveurs :

- **Les serveurs d'enregistrement :** Reçoivent les mises à jour concernant la localisation actuelle des utilisateurs.
- **Les serveurs proxy :** Un serveur proxy Reçoit les requêtes des UA et les redirige vers un autre serveur proxy, un UAS ou un serveur de redirection. Il a accès aussi à une base de données ou à un service de localisation (location service) pour déterminer vers où rediriger la requête reçue. L'interface entre le serveur proxy et la base de données/le service de localisation n'est pas définie par le protocole SIP.

Les serveurs proxy peuvent travailler en deux modes :

- i) **Stateless proxy :** dans ce cas le proxy est un simple élément de redirection, il ne garde pas d'information concernant la requête reçue et redirigée.
- ii) **Stateful proxy :** contrairement à un stateless proxy, un stateful proxy mémorise toutes les informations concernant les requêtes entrantes et essaie plusieurs localisations de l'utilisateur et renvoie la meilleure réponse.

- **Les serveurs de redirection :**

Quand un serveur de redirection reçoit une requête, au lieu de la rediriger il renvoie l'adresse du prochain serveur au client qui devra le contacter directement.

**c) Format des adresse SIP**

Tout utilisateur SIP dispose d'un identifiant unique. Cet identifiant constitue l'adresse de l'utilisateur permettant de le localiser. Le format d'une adresse SIP se présente sous la forme illustrée à la Figure I.16.

**sip: infos\_utilisateur @ domaine**

**Figure 1.16:** Syntaxe d'une adresse SIP.[9]



Avec :

**infos\_utilisateur** = nom de l'utilisateur ou numéro de téléphone : mot de passe.

**domaine** = nom de domaine ou adresse IP : port

Sachant que : Le mot de passe et port sont facultatif.

#### d) Requêtes et Réponses SIP

SIP est un protocole de type client serveur. A cet effet, les échanges entre un terminal appelant et un terminal appelé se font par l'intermédiaire de requêtes et réponse SIP.

Voici une liste exhaustive des requêtes SIP :

- INVITE : Cette requête indique que l'application (ou utilisateur) correspondante à l'Url SIP spécifié est invitée à participer à une session.
- ACK : Cette requête permet de confirmer que le terminal appelant a bien reçu une réponse définitive à une requête INVITE.
- BYE : Cette requête est utilisée par le terminal de l'appelé pour signaler qu'il souhaite mettre un terme à la session.
- CANCEL : Cette requête est envoyée par un terminal ou un serveur mandataire afin d'annuler une requête non validée par une réponse finale.
- REGISTER : Cette méthode est utilisée par le client pour enregistrer l'adresse listée dans le champ TO par le serveur auquel il est relié.
- OPTIONS : Un serveur mandataire en mesure de contacter le terminal appelé, doit répondre à une requête OPTIONS en précisant ses capacités à contacter le même terminal.

A ces requêtes sont associées des réponses qui sont dans le même format que celles du protocole HTTP. Voici les plus importantes d'entre elles :

- 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).

Le schéma suivant illustre le scénario d'une communication SIP :

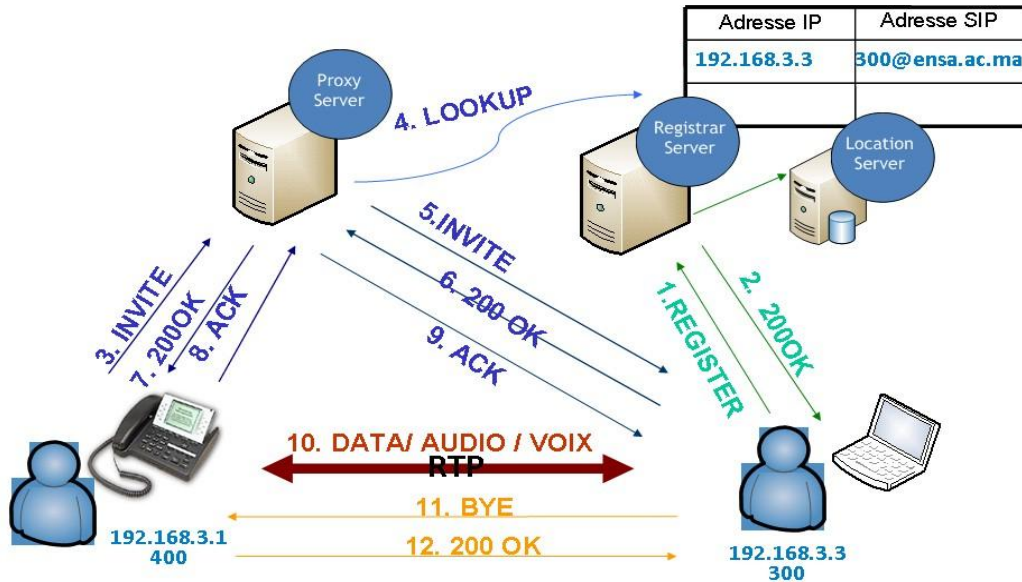


Figure 1.17: Architecture SIP.[6]

### 12.1.3 Le protocole MGCP

Le protocole MGCP se base principalement sur les couches 3 et 4 du modèle OSI, Il définit l'architecture d'un réseau de passerelles, MGCP se base sur la notion des « User Agent » ou « Media Gateway Controller » qui est un organe permettant le contrôle d'appel et pilotant une ou plusieurs passerelles (MG), celles-ci permettent la conversation audio sur le réseau téléphonique vers audio sur le réseau paquet.

Les téléphones MGCP ne peuvent pas s'appeler entre eux mais doivent passer obligatoirement par un contrôleur central : la Gateway.[3]

## 12.2 Les protocoles de transport :

### 12.2.1 Le protocole RTP

Le groupe de l'IETF a développé en 1993 le protocole de transport en temps réel dont le but est de transmettre sur Internet des données qui ont des propriétés temps réel (audio, vidéo ...). C'est un protocole de la couche application du modèle OSI et utilise les protocoles de transport TCP ou

UDP, mais, généralement, il utilise UDP qui est mieux approprié à ce genre de transmission.[5]

### 12.2.2 Le protocole RTCP

Le RTCP est un protocole de contrôle utilisé conjointement avec RTP pour contrôler les flux de données et la gestion de la bande passante. Il permet de contrôler le flux RTP, et de véhiculer périodiquement des informations de bout en bout pour renseigner sur la qualité de service de la session de chaque participant à la session. Des quantités telles que le délai, la gigue, les paquets reçus et perdus sont très important pour évaluer la qualité de service de toute transmission et réception temps réelles.

C'est le protocole sous-jacent (UDP par exemple) qui permet grâce à des numéros de ports différents et consécutifs (port pair pour RTP et port impair immédiatement supérieure pour RTCP) le multiplexage des paquets de données RTP et des paquets de contrôle RTCP.

## 13. Comparaison entre les protocoles :

	<b>H323</b>	<b>SIP</b>	<b>MGCP</b>
<b>Inspiration</b>	Téléphonie	HTTP	
<b>Nombres d'échange pour établir la connexion</b>	6 à 7 aller-retour	1 à 5 aller-retour	3 à 4 aller-retour
<b>Complexité</b>	Elevée	Faible	Elevée
<b>Adaptabilité / Modularité protocolaires</b>	Faible	Elevée	Modérée

<b>Implémentation de nouveaux services</b>	NON	OUI	NON
<b>Adapté à internet</b>	NON	OUI	NON
<b>Protocole de Transport</b>	TCP	TCP ou UDP	TCP ou UDP
<b>Coût</b>	Elevé	Faible	Modéré
<b>Avantages</b>	<ul style="list-style-type: none"> <li>-Maturité du protocole: Actuellement version 4 pour la définition.</li> <li>-Les premières mises en œuvre de V3 commencent juste à apparaître</li> <li>Beaucoup de constructeurs</li> </ul>	<ul style="list-style-type: none"> <li>-Simple à mettre en œuvre, messages écrits en clair</li> <li>-Interopérabilité très bonne</li> <li>Grâce à CPL (Call Processing Language) qui utilise XML, il est très facile d'ajouter des</li> </ul>	<ul style="list-style-type: none"> <li>-Permet d'utiliser des téléphones « idiots »</li> <li>Indépendant des protocoles de signalisation supérieurs (H323, SIP)</li> <li>Bien pour les opérateurs voulant faire du RTC-IP-RTC</li> </ul>

**Tableau 1.1 :** Comparaison entre les différents protocoles.[3]

## 14. Conclusion

Dans ce chapitre, on a passé en revue les concepts liés à la ToIP, ce qui nous permet de mieux maîtriser cette technologie et mieux la comprendre et enfin envisager un déploiement même sur un petit réseau LAN test, et le choix des protocoles à utiliser s'en trouve facilité après cette introspection des plus connus. On a donc décidé d'utiliser que du matériel utilisant le protocole SIP, suite à la comparaison faite précédemment.

# Chapitre 2

## Déploiement du service de la ToIP.

## 1. Introduction

Après avoir passé en revue les architectures de la TOIP, on a opté pour la configuration PC-PC et téléphone-PC dans un réseau LAN constitué d'un Switch, d'un PABX (logiciel installé sur un PC), de deux clients sous Windows et d'un téléphones IP. On a aussi essayé de programmer une interface graphique d'un softphone, mais on n'a pas réussi à la finir à temps. Par contre dans la figure suivante une capture d'écran du résultat auquel on est arrivé, puis le code source sera donné en annexe pour les futurs étudiants. Donc, dans ce chapitre nous détaillons toutes les étapes de l'installation et de la configuration d'un PABX (Asterisk) et des clients (jitsi sous Windows + iPhone). Et nous validons l'application sur un petit réseau LAN (voir Figure 2.1), en expliquant tout ce que nous avons fait, pour se faire.

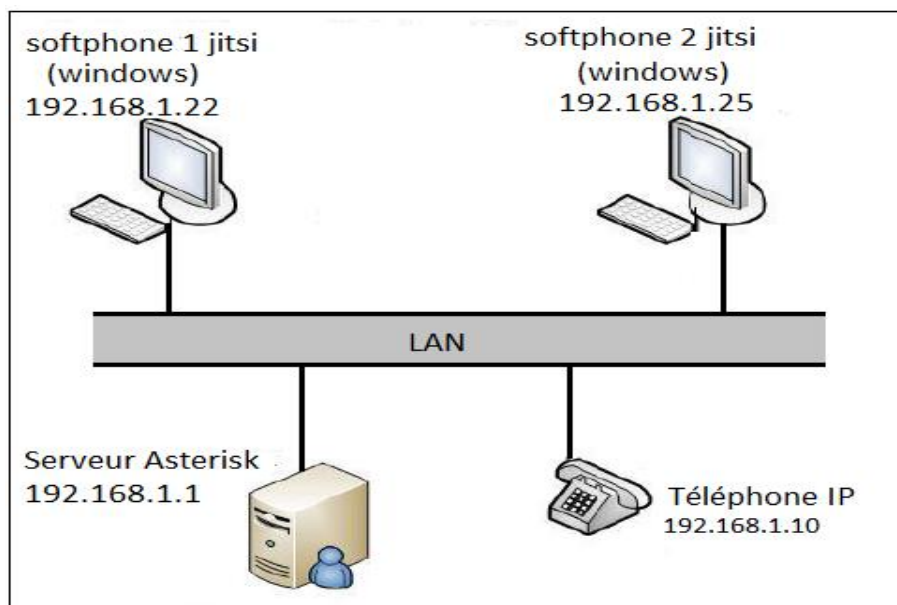


Figure 2. 1: Topologie du réseau test

## 2. Présentation d'Asterisk

Asterisk est un projet démarré en 1999 par Mark Spencer. Son objectif était alors de fournir à Linux un commutateur téléphonique complet et totalement libre. Aujourd'hui Asterisk est un PABX (Private Automatic Branch eXchange) d'une rare puissance et souplesse, capable de gérer la téléphonie analogique, mais surtout, et c'est ce qui nous intéresse, la voix sur IP.

La VoIP sur Asterisk passe entre autre par la prise en charge d'un protocole standard, ouvert et très largement utilisé : le SIP (Session Initiation Protocol), est un protocole très proche d'HTTP qui n'est pas limité seulement à la voix mais qui prend aussi en charge la vidéo et la messagerie instantanée.

Parmi les caractéristiques d'Asterisk on peut citer les suivantes :

- Il est multi-plateforme : Linux, MAC OS X, Windows
- Son utilisation est très économique comparé à l'achat d'un PABX classique
- Grande souplesse de configuration (mais une configuration complexe)
- Extensibilité par des scripts ou des modules en langage Perl, C, Python, PHP, et Ruby
- Interopérabilités avec la téléphonie traditionnelle
- Permet de remplacer totalement, par le biais de cartes FXO/FXS, un PABX, et d'y adjoindre des fonctionnalités de VoIP pour le transformer en PBX IP.

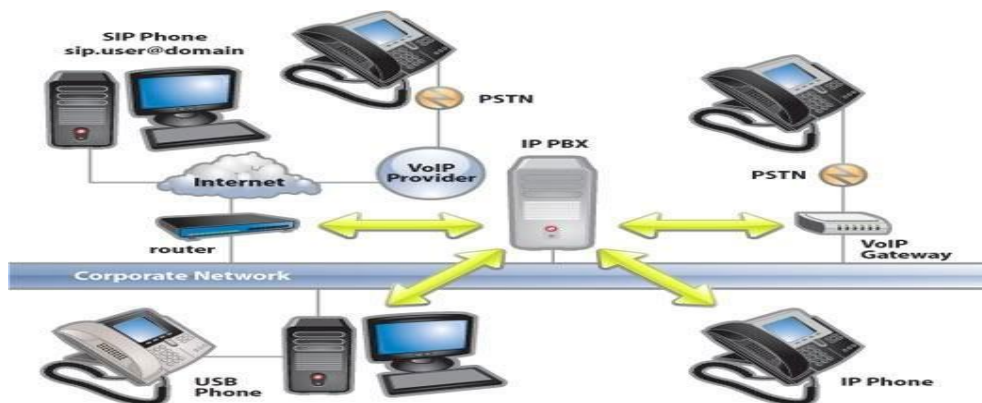


Figure 2.2: Fonctionnalités offertes par Asterisk

### 3. Fonctionnalités d'Asterisk

D'un point de vue fonctionnalité, Asterisk permet de faire tout ce que l'on peut attendre d'un PABX moderne, parmi ses nombreuses fonctionnalités on peut citer les plus essentielles:

- Authentification des utilisateurs appelants
- Standard vocal automatique ou IVR (Interactive Voice Response)
- Transfert d'appel
- Filtrage des appels
- Service de messagerie vocale
- Gestion des conférences
- Mise en attente
- Facturation détaillée
- Journalisation des appels
- Enregistrement des appels

Une liste exhaustive des fonctionnalités est fournie sur le site d'Asterisk<sup>1</sup>

#### **4. Installation d'Asterisk**

Nous avons installé Asterisk en utilisant un package proposé avec le système d'exploitation linux connu sous le nom de « AsteriskNow », Ce package est disponible sur le site d'Asterisk<sup>2</sup>.

AsteriskNow est téléchargé sous forme d'une image ISO, Pour l'installer il faut le graver sur un DVD

Une fois le PC démarré du DVD, nous voyons apparaitre à l'écran deux options pour installer le PBX avec ou sans interface graphique, comme suit :

Nous avons préféré l'option 1 avec interface graphique, ceci lancera l'opération d'installation graphique automatisé.

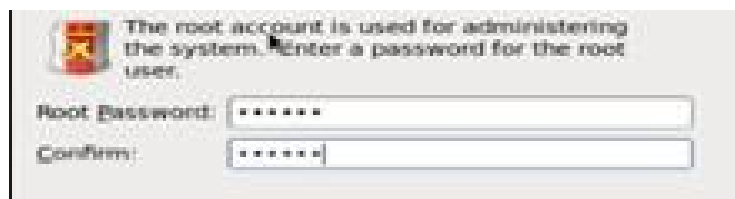
---

<sup>1</sup> <http://www.asterisk.org/get-started/features>

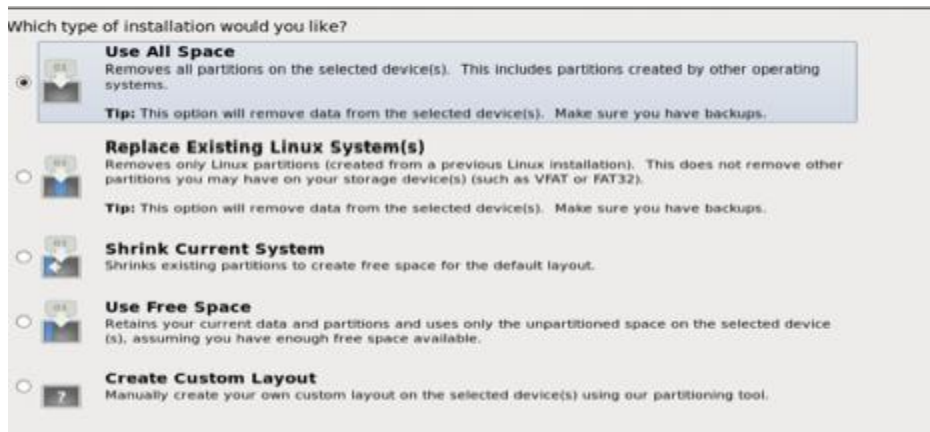




Ensuite, nous choisissons un mot de passe



L'écran suivant Nous demande de choisir la partition du disque dur sur laquelle on va installer le PABX,



Dès qu'on valide notre choix l'installation démarre.



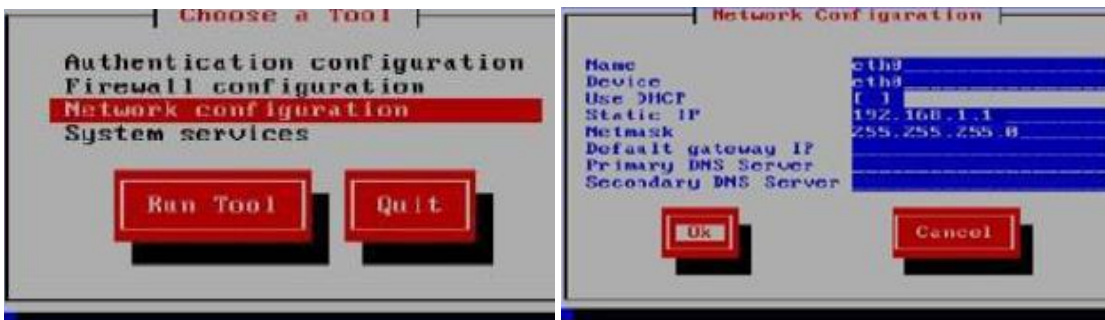
## 5. Configuration d'Asterisk :

Voilà l'interface en ligne de commande d'Asterisk, ici on voit l'adresse IP pour l'interface graphique et cette adresse a été attribuée par un serveur DHCP mais on va donner à notre serveur une adresse fixe.

```

machine écran renpeniques Aide
AsteriskNOW 3.8.0
To configure AsteriskNOW with FreePBX, point your web browser to http://10.0.2.15/
localhost login: root
Password:
root@localhost ~]# loadkeys fr_
    
```

Ensuite on va configurer l'adresse IP du serveur. Faites setup et allez dans « network configuration » pour donner une adresse fixe à votre serveur :



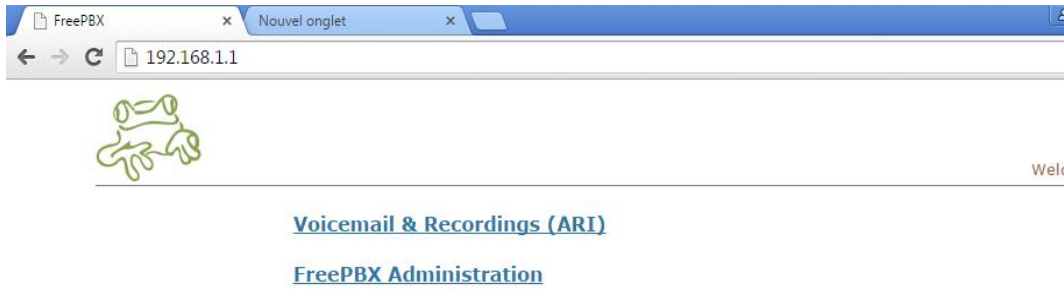
L'adresse de notre serveur est (192.168.1.1).

Nous sommes maintenant prêtes à passer à la configuration d'AsteriskNow à partir de l'interface Web FreePBX.

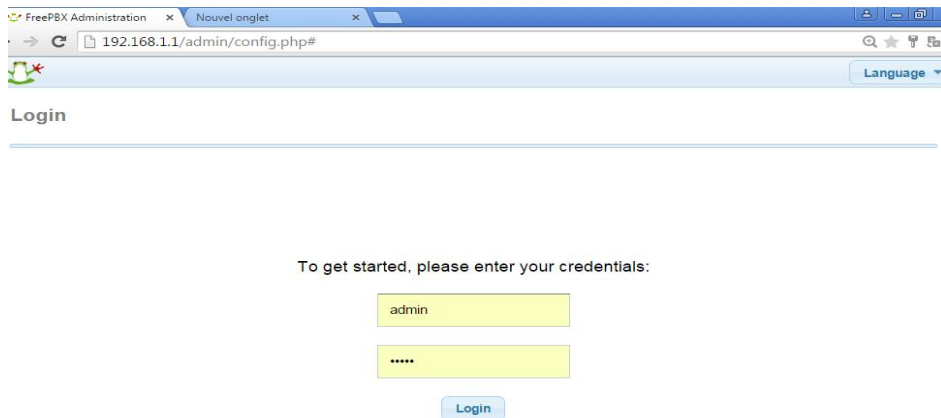
Après avoir terminé la configuration on redémarre l'ordinateur pour que notre configuration prenne son effet.

## 6. Configuration de FreePBX

Pour configurer notre FreePBX, nous avons ouvert un navigateur sur un autre PC en réseau avec notre serveur et avons introduit l'adresse IP de notre serveur :

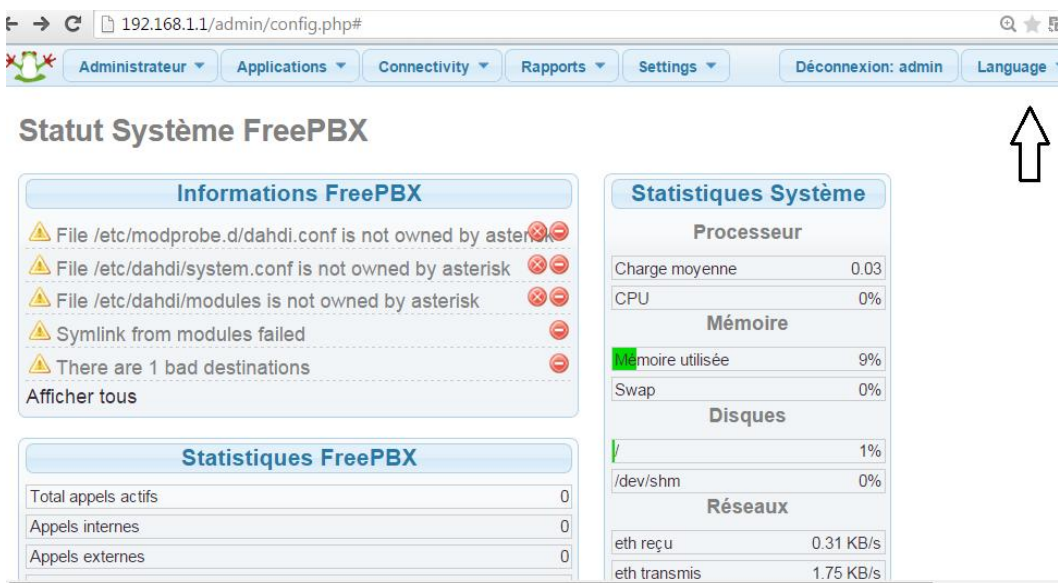


En cliquant sur « FreePBX administrator », la fenêtre qui permet de nous identifier apparaît :



Le nom d'utilisateur et le mot de passe sont par défaut « admin ».

Après avoir connecté avec succès dans FreePBX, on a commencé par changer la langue :



## ✓ Création des comptes utilisateurs

Pour créer un compte utilisateur il faut aller dans l'onglet « Applications » puis « Extensions », choisir l'outil « Périphérique SIP » et on valide par le bouton « Soumettre ».

Dans notre cas nous avons besoins de créer trois comptes (deux softphone, un IP phone). il suffit de remplir les champs illustrés suivants des trois utilisateurs :

Extension Utilisateur : c'est le nom ou le numéro utilisé pour faire les conversations.

Nom Affiché : c'est nom de l'extension utilisé pour l'affichage.

Secret : c'est le mot de passe de votre compte .

### - Ajout Extension

Extension Utilisateur <sup>?</sup>	122
Nom Affiché (CID) <sup>?</sup>	soumia
Numéro alias du CID <sup>?</sup>	
Alias SIP <sup>?</sup>	

Ce périphérique utilise la technologie sip

secret <sup>?</sup>

dtmfmode <sup>?</sup>

nat <sup>?</sup>

Enfin faire « Soumettre » puis nous cliquons sur le bouton rouge de rechargement « Apply Config » .

Administrateur Applications Connectivity Rapports S

- Options **Apply Config**

No Answer <sup>?</sup>

CID Prefix <sup>?</sup>

Occupé <sup>?</sup>

CID Prefix <sup>?</sup>

Not Reachable <sup>?</sup>

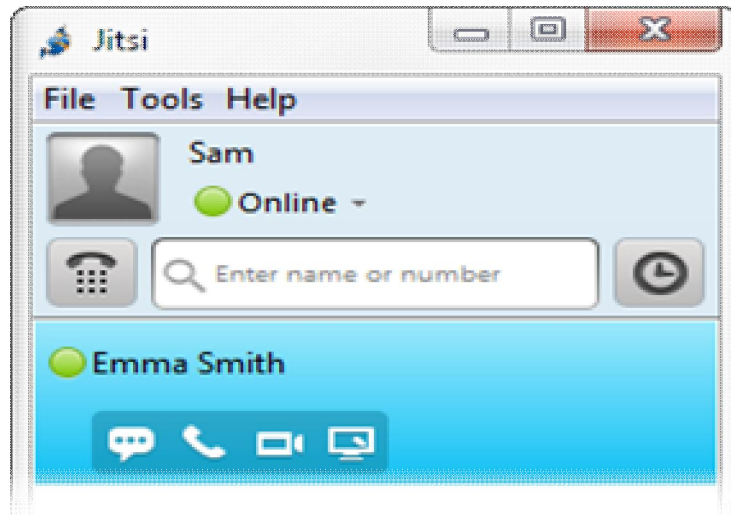
CID Prefix <sup>?</sup>

**Soumettre**

## 7. Déploiement de la ToIP en utilisant un softphone commercial

### 7.1 Présentation de softphone Jitsi

Jitsi (anciennement SIP Communicator) est une application développée en java par Emil Ivov pour répondre aux besoins de l'Université Louis Pasteur de Strasbourg en 2003, qui permet à ses utilisateurs d'établir des conversations audio et vidéo sur Internet via le protocole SIP , Elle intègre également des fonctions de messagerie instantanée en prenant en charge quelques-uns des réseaux les plus populaires : SIP/SIMPLE, Jabber (XMPP), AIM/ICQ, MSN (Windows Live Messenger), Yahoo!, Facebook Chat, Google Talk.[10]



## 7.2 Fonctionnalité de Jitsi

- jitsi est un logiciel pour :[12]
  - ✓ Messagerie instantanée.
  - ✓ Appels téléphoniques.
  - ✓ Partage d'écrans.
  - ✓ Visioconférence et pont vidéo.
- Il veut être une alternative libre et sécurisée au logiciel bien connu : Skype
- Il est publié sous licence libre LGPL
- Il est multi-plateformes : Windows, Mac OS X, GNU/Linux, Android.
- C'est un logiciel qui ne vous espionne pas.
- Une notification sonore et visuelle est activée dès qu'une connexion est établie.
- Le cryptage du réseau est effectué par ZRTP.

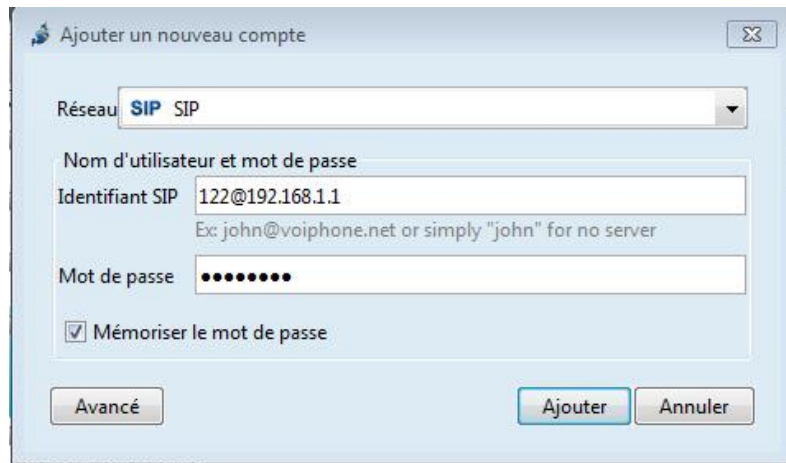
## 7.3 Configuration de Jitsi

### ✓ Création d'un compte SIP

Un compte SIP sur jitsi appelé « SIP registrarless » qui vous permet de contacter d'autres personnes sur le même réseau local.[11]


Pour mettre en place un compte SIP , nous sommes allées dans le menu Fichier> Ajouter un nouveau compte , choisir le protocole "SIP" .

les champs ont été remplis comme précisé dans la figure ci-dessous :



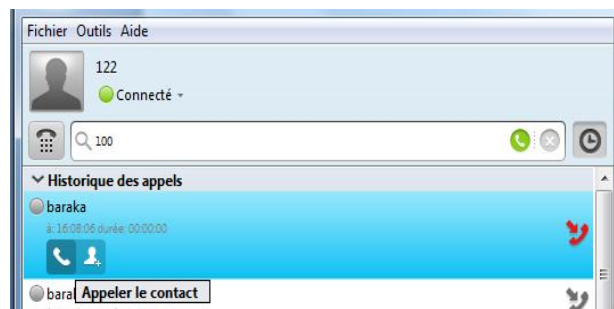
Identifiant SIP = le numéro de softphone @l'adresse IP de serveur .

Notez que : le numéro et le mot de passe sont identiques à ceux entrés dans le serveur.

Dans la fenêtre principale de jitsi l'icône suivante «  » doit apparaitre comme signe que le soft phone s'est bien enregistré auprès du serveur et qu'il est actif.

### ✓ Appeler avec jitsi

L'appelant et le correspondant sont tous les deux sur le même serveur de réseau. Pour appeler un correspondant, il suffit de connaître son numéro et de le composer. Sur la figure suivante on a entré le numéro de téléphone c'est : 100.



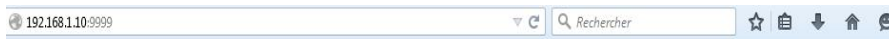
## 8. Configuration de téléphone IP

Après avoir branché l'alimentation et le câble Ethernet, il faut accéder aux menus et configurer le téléphone.

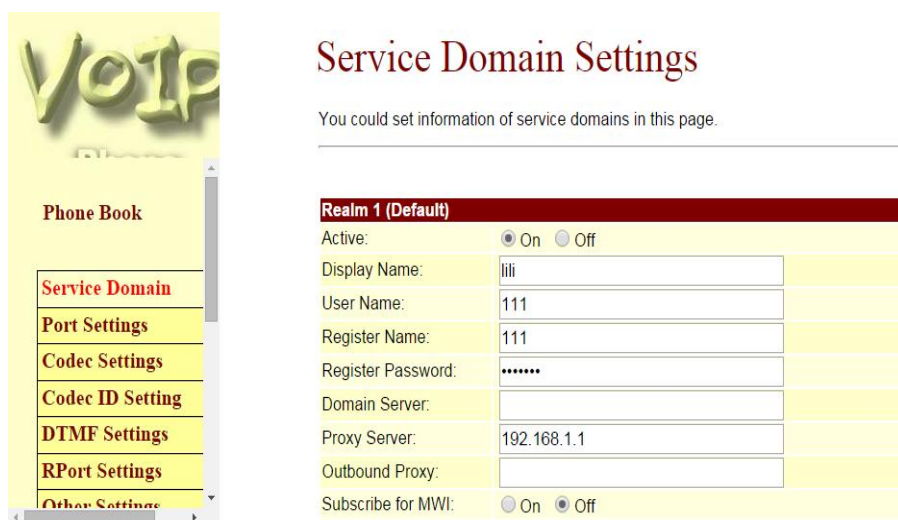


Par contre On peut régler le téléphone par navigateur, en entrant l'adresse IP de téléphone dans le navigateur et ajouter 9999 derrière adresse IP : <http://192.168.1.10:9999>.

ci-dessous nous donnons juste quelques captures d'écran de l'interface graphique de téléphones IP, à laquelle on accède par un navigateur. Il est à noter que le téléphone IP peut se configurer de manière identique directement par l'intermédiaire de son clavier, en utilisant un menu (ajouter / supprimer des numéros de téléphone ,des noms ...).

A screenshot of a web browser window displaying a "Login VoIP" interface. The interface has a yellow background and a red header. The text "Enter your username and password to login" is displayed above the "VoIP server" label. There are two input fields: "Username" with the value "root" and "Password" with four dots. Below the fields are "Login" and "Clear" buttons.





## 9. Tests et résultats

Après avoir la configuration du serveur Asterisque et les clients (softphone , téléphone IP ), ont été essayés d'établir des appels sur IP par l'intermédiaire d'un Switch selon les deux architecture cités précédemment .

les tests ont été faits une vingtaine de fois avec succès , sauf que nous avons signé dans un petit problème de son concernant le modèle pc-pc.

## 10. Déploiement de la ToIP en utilisant notre interface

Cette partie du travail nous a coûté beaucoup de temps parce que nous l'avons considérée la partie la plus essentielle de ce projet, mais malheureusement nous n'avons pas réussi à l'achevé. Cette interface a été réalisée en utilisant le langage Java , un langage de programmation orienté objet assurant aux programmes une portabilité sur plusieurs systèmes d'exploitation (Windows, Linux.....), chose qui est très souhaitable dans une application de téléphonie. Etant donnée que nous ne connaissons pas ce langage nous avons essayé de gagner du temps en utilisant un petit manuel sur Java pour se lancer dans le travail, mais après peu de temps il s'est avéré que cette approche est insuffisante et qu'une apprentissage systématique du langage est inévitable pour mener à bien notre travail.

Ceci justifie d'un côté le retard que nous avons mis dans la réalisation de ce projet, mais d'un autre côté nous a permis d'acquérir un langage très utile et très puissant qui va nous aider en tant qu'informaticiens à développer nos compétences de programmation.

L'implémentation des fonctionnalités d'un softphone se fait :

- Soit par l'utilisation d'une bibliothèque spéciale au protocole de communication adopté, dans le cas du protocole SIP on peut utiliser la bibliothèque JainSIP.
- Soit par l'utilisation des bibliothèques prédéfinies dans Java en utilisant la technique de sockets.

## 10.1 Réalisation de l'interface graphique

La figure ci-dessous montre une capture d'écran de l'interface graphique que nous avons réalisé. Elle est très simplifiée, le but était de pouvoir tester le fonctionnement d'un appel entre deux PC en déployant la technique de la ToIP.



Figure 2.3 :capture ecran de notre interface graphique.

On avait besoin, bien entendu, de :

- Un champ pour saisir l'adresse du destinataire.
- Un bouton pour lancer les appels
- Un bouton pour répondre aux appels

Le code source de l'interface graphique est donné dans l'annexe ...

## 10.2 Intégration des fonctions d'appel à l'interface graphique

Pour implémenter les fonctions d'appels de notre interface nous avons choisi l'utilisation de la bibliothèque JainSIP. Pour comprendre le principe de fonctionnement de cette bibliothèque et comment nous pouvons l'utiliser nous avons essayé d'examiner le code source du softphone Jitsi, qui utilise la même bibliothèque (Jitsi est écrit en Java et son code source est disponible sur son site officiel). Malheureusement la lecture du code source s'est avérée un peu compliquée et ça nécessite une certaine familiarité avec les codes open sources. Pour surmonter cette difficulté on a essayé de lire un livre sur l'architecture des logiciels libres [13] mais finalement il ne reste plus de temps pour terminer et on s'est trouvé contraint de passer à la rédaction.

## 11. Conclusion

Dans ce chapitre, nous avons dressés la partie application de la technologie TOIP, en installant un PABX open source, des clients sous Windows et des téléphones IP, en réseau. On a pu tester ce service selon les deux architectures (pc-pc, pc- phone).

Par contre, on n'a pas pu tester ce service avec notre application à cause des raisons citées précédemment.



# Conclusion Générale

La téléphonie sur IP est un mode de téléphonie utilisant le protocole de télécommunications créé pour Internet (IP pour Internet Protocol). La voix est numérisée puis acheminée sous forme de paquets comme n'importe quelles autres données.

Cette technologie possède plusieurs avantages, elle est plus souple, conviviale, coûte moins chère, propose de nouveaux services et en plus elle est basée sur des standards ouverts ce qui permet l'interaction avec les équipements téléphoniques standards. C'est pour cette raison que toute entreprise qui veut être compétitive et moderne aujourd'hui, jette son dévolu sur la téléphonie sur IP pour gérer ses communications tant internes qu'externes.

Actuellement, il est évident que la téléphonie sur IP va continuer à se développer dans les prochaines années à cause de la véritable opportunité économique que possède cette technologie au niveau des réseaux Intranet. En effet, les entreprises peuvent adopter une stratégie de communication unifiée de la voix, la vidéo et les données sur son réseau privé.

Dans ce projet de fin d'étude, on a voulu justement tester la faisabilité et la simplicité de cette technologie, en installant un PABX open source, des clients sous Windows et des téléphones IP, en réseau. On a pu tester ce service entre deux clients sous Windows en utilisant le softphone Jitsi, et, entre un client sous Windows et un téléphone IP. Malheureusement, on n'a pas pu tester ce service avec notre application à cause des raisons citées précédemment. Mais malgré ceci, le travail sur ce projet nous a permis de découvrir un domaine très intéressant qui ne cesse de se développer ces dernières années.

Outre les connaissances acquises durant ce travail, cette expérience nous a confronté à un certain nombre de problèmes auxquels on doit faire face pour développer nos compétences scientifiques, comme la recherche de la documentation, l'organisation du travail, la méthodologie de rédaction et la communication oral et écrite.

## Bibliographies

- [1] BABAHADJ Abdeldjabbar, OUAYNI Tayeb, (2005). Etude et simulation d'un réseau de téléphonie sur IP. Mémoire de fin d'étude pour l'optimisation de diplôme d'ingénieur d'état en informatique, université de science et de la technologie d'Oran.
- [2] La Telephonie Sur IP , Docstoc.com , consulté le 24/05/2015 <http://www.docstoc.com/docs/109707824/La-telephonie-sur-IP>.
- [3] La Téléphonie sur IP , (ToIP) Didi Souhila , Guerriche meryem (2014), Mémoire de fin d'études, Pour l'obtention du diplôme de Licence en Informatique.
- [4] ,Telephonie\_sur\_IP , Laurent\_Ouakil,\_Guy\_Pujolle , Groupe Eyrolles, 2007, ISBN : 978-2-212-12099-8 , [www.editions-eyrolles.com](http://www.editions-eyrolles.com) . consulté le 11/05/2015.
- [5] développement d'un environnement de communication multicast (voix et vidéo) sur internet , MOURAD EL ALLIA (2002) , Mémoire présenté à l'école de technologie supérieur.
- [6] Etude et mise en œuvre du service pilote ,Mr Mohamed El Mahdi BOUMEZZOUGH, ToIP RENATER (2009), Pour l'obtention du diplôme d'Ingénieur d'Etat En Génie Réseaux et Systèmes.
- [7] Session Initiation Protocol SIP , related:www.irisa.fr/armor/lesmembres/Alia/SIP/Session%20Initiation%20Protocol%20SIP.doc format des messages SIP, consulté le 26 Avril 2015.
- [8] Etude et simulation d'un réseau de téléphonie sur IP ( TOIP) , TAHRA Zahia (2008) , MEMOIRE DE FIN D'ETUDES en vue de l'obtention du Diplôme d'ingénieur d'état en Informatique.

[9] Le Protocole SIP : RFC 2543 , [http://www-igm.univ-mlv.fr/~dr/ XPOSE2002 /](http://www-igm.univ-mlv.fr/~dr/XPOSE2002/) DEBOURDEAU / Consulté le , 24/04/2015.

[10] le site officiel de Wikipédia , <http://www.wikipedia.org/> , consulté le 06/04/2015.

[11] le site officiel de jitsi , [www.jitsi.org](http://www.jitsi.org) , consulté le 06/04/2015 .

[12] le logiciel jitsi <http://www.agenux.org/wp-content/uploads/2014/01/Logiciel-Jitsi-.pdf> , consulté le 12/04/2015.

[13] The Architecture of Open Source Applications , Amy Brown and Greg Wilson, Lulu.com, 2011.



## ملخص

الاتصال عبر بروتوكول الإنترنت (ToIP) هي تكنولوجيا بدأت تفرض نفسها تدريجيا في جميع القطاعات و تتمثل في تمرير المكالمات الهاتفية بواسطة الشبكة IP . تشغيل هذه الخدمة يستند أساسا على محول هاتفى برمجي يسمى (Asterisk) يدير المكالمات بين المستخدمين عن طريق البرتوكول (SIP). هاتف برمجي متعدد الانظمة (Jitsi) وجهاز هاتف IP، وكلها متصلة بواسطة شبكة محلية (LAN) برابط من نوع (Switch). وفي هذا السياق تم إنجاز مشروع تخرجنا.

## Résumé

La téléphonie sur IP (ToIP) est une technologie qui s'impose progressivement dans tous les secteurs, elle consiste à faire transiter les communications téléphoniques par le réseau IP. La mise en œuvre de ce service repose principalement sur un autocommutateur logiciel appelé Asterisk qui utilise le protocole SIP (Session Initiation Protocol), un soft phone multiplateforme (Jitsi) et des téléphones IP, le tout relié en réseau local par un Switch. C'est dans ce contexte que nous avons réalisé notre projet de fin d'études.

## Abstract

Telephony over IP (ToIP) is a technology that gradually imposed in all sectors, it consists in doing telephone calls over the IP network .The implementation of this service is based primarily on PABX software, called Asterisk, that uses the SIP protocol (Session Initiation Protocol), a multiplatform soft phone (Jitsi) and an IP phones, all connected by a LAN Switch. It was in this context that we have made our final project studies.

## Annexe

---

```
package siptéléphone;
import java.awt.*;
import java.awt.event.*;

import javax.swing.*;
import javax.swing.text.*;

public class SipGUI implements MouseListener, ActionListener
{

    private JFrame fenetre;

    private JTextPane textPane;
    private StyledDocument doc;
    private JPanel optionPanel , numPanel;
    private JMenuBar menu;
    private JMenu menuFichier;
    private JMenu menuEdition;
    private JMenu menuAide;
    // private UA ua;
    private JLabel labelBas , font ;
    private String adresseContact;
    private String callId="";

    public SipGUI() throws Exception {

        fenetre = new JFrame(" SipPhone");
        JFrame.setDefaultLookAndFeelDecorated(true);
        JPanel panFond = new JPanel();
        font = new JLabel ("images/person.png");
        panFond.setOpaque(false);
        panFond.setBackground(Color.white);
        fenetre.setContentPane(panFond);
        fenetre.setLocation(600, 200);
        fenetre.setSize(240,490);
        fenetre.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        fenetre.getContentPane().setBackground(Color.white);
        fenetre.getContentPane().setLayout(new
BoxLayout(fenetre.getContentPane(),
BoxLayout.PAGE_AXIS));
        fenetre.getRootPane().setBackground(Color.white);
        // création barre de menu
        menu = new JMenuBar();
        fenetre.setJMenuBar(menu);
        menuFichier = new JMenu("Fichier");
        JMenuItem itemAppeler = new JMenuItem("Appeler");
        menuFichier.add(itemAppeler);
        JMenuItem itemRaccrocher = new JMenuItem("Raccrocher");
        menuFichier.add(itemRaccrocher);
```

```

JMenuItem itemQuitter = new JMenuItem("Quitter");
menuFichier.add(itemQuitter);
menuEdition = new JMenu("Edition");
JMenuItem itemAjouter = new JMenuItem("Ajouter un contact");
itemAjouter.addActionListener(this);
menuEdition.add(itemAjouter);
JMenuItem itemHistorique = new JMenuItem("Historique des
appels");
itemHistorique.addActionListener(this);
menuEdition.add(itemHistorique);
JMenuItem itemPreferences = new JMenuItem("Preferences");
itemPreferences.addActionListener(this);
menuEdition.add(itemPreferences);
// menu Aide
menuAide = new JMenu("Aide");
JMenuItem itemApropos = new JMenuItem("A propos");
menuAide.add(itemApropos);
menu.add(menuFichier);
menu.add(menuEdition);
menu.add(menuAide);
fenetre.setJMenuBar(menu);
JPanel panelFond = new JPanel();
panelFond.setOpaque(false);
panelFond.setMaximumSize(new Dimension(228,103));
textPane = new JTextPane();
textPane.setPreferredSize(new Dimension(225,103));
textPane.setOpaque(false);
textPane.setFont(new Font("Trebuchet MS",Font.BOLD,12));
textPane.setForeground(Color.black);
textPane.setEditable(false);
doc = textPane.getStyledDocument();
fenetre.getContentPane().add(panelFond,
BorderLayout.LINE_START);
JPanel optionPanel2 = new JPanel();
optionPanel2.setBackground(Color.white);
optionPanel2.setMaximumSize(new Dimension(200,40));
JButton audioButton = new JButton(new
ImageIcon("images/audio.png"));
audioButton.addMouseListener(this);
audioButton.setName("audioButton");
audioButton.setPreferredSize(new Dimension(30,30));
audioButton.setBorderPainted(false);
JButton confButton = new JButton(new
ImageIcon("images/conf.png"));
confButton.addMouseListener(this);
confButton.setName("confButton");
confButton.setPreferredSize(new Dimension(30,30));
confButton.setBorderPainted(false);
optionPanel = new JPanel();
optionPanel.setBackground(Color.white);
optionPanel.setMaximumSize(new Dimension(200,50));
optionPanel.setOpaque(false);
JButton callButton = new JButton(new
ImageIcon("images/Call.png"));

```

```

        callButton.setName("callButton");
        callButton.addMouseListener(this);
        callButton.setBorder(BorderFactory.createEmptyBorder());
        callButton.setBorderPainted(false);
        callButton.setBackground(Color.white);
        callButton.setPreferredSize(new Dimension(71,36));
        callButton.setOpaque(false);
        optionPanel.add(callButton, BorderLayout.EAST);
        JButton byeButton = new JButton(new
ImageIcon("images/bye.png"));
        byeButton.setName("byeButton");
        byeButton.addMouseListener(this);
        byeButton.setBorder(BorderFactory.createEmptyBorder());
        byeButton.setBorderPainted(false);
        byeButton.setBackground(Color.white);
        byeButton.setPreferredSize(new Dimension(71,36));
        byeButton.setOpaque(false);
        optionPanel.add(byeButton, BorderLayout.WEST);
        JTextField ch_saisie = new JTextField(20);
        ch_saisie.setBorder(BorderFactory.createEmptyBorder());
        ch_saisie.setBorder(BorderFactory.createEmptyBorder());
        optionPanel.add(ch_saisie, BorderLayout.WEST);

        fenetre.getContentPane().add(optionPanel,
BorderLayout.CENTER);

        numPanel = new JPanel();
        numPanel.setBorder(BorderFactory.createLineBorder(Color.black));
        numPanel.setBackground(Color.white);
        numPanel.setMaximumSize(new Dimension(150,200));
        numPanel.setLayout(new GridLayout(4,3,5,5));
        numPanel.setOpaque(false);
        for(int i=1;i<10;i++) {
            JButton numButton = new JButton();
            numButton.addMouseListener(this);
            numButton.setBorder(BorderFactory.createEmptyBorder());
            numButton.setBorderPainted(false);
            numButton.setBackground(Color.white);
            numButton.setMaximumSize(new Dimension(44,44));
            numButton.setIcon(new ImageIcon("images/"+i+".png"));
            numButton.setOpaque(false);
            numPanel.add(numButton);
        }
        JButton etoileButton =
            new JButton();

        etoileButton.setBorder(BorderFactory.createEmptyBorder());
        etoileButton.setBorderPainted(false);
        etoileButton.setBackground(Color.white);
        etoileButton.setMaximumSize(new Dimension(44,44));
        etoileButton.setIcon(new
ImageIcon("images/etoile.png"));
        numPanel.add(etoileButton);
        JButton OButton = new JButton();

```

```

OButton.setBorder(BorderFactory.createEmptyBorder());
    OButton.setBorderPainted(false);
    OButton.setBackground(Color.white);
    OButton.setMaximumSize(new Dimension(44,44));
    OButton.setIcon(new ImageIcon("images/0.png"));
    numPanel.add(OButton);
    JButton dieseButton = new JButton();

    dieseButton.setBorder(BorderFactory.createEmptyBorder());
    dieseButton.setBorderPainted(false);
    dieseButton.setBackground(Color.white);
    dieseButton.setMaximumSize(new Dimension(44,44));
    dieseButton.setIcon(new
ImageIcon("images/diese.png"));
    numPanel.add(dieseButton);
    fenetre.getContentPane().add(numPanel);

    JPanel textLogo = new JPanel(new
FlowLayout(FlowLayout.LEFT));

    textLogo.setBorder(BorderFactory.createBevelBorder(1,
Color.GRAY,Color.LIGHT_GRAY));
    textLogo.setBackground(Color.white);
    labelBas = new JLabel("Connexion...");
    //textLogo.add(new JLabel(new
ImageIcon("clipboard_resize.png"));
    textLogo.add(labelBas);

    //textLogo.setMaximumSize(new Dimension(250,30));
    fenetre.getContentPane().add(textLogo,
BorderLayout.PAGE_END);
    fenetre.setVisible(true);

}

public static void main(String[] args) throws Exception {

    new SipGUI();
}
}
-----
import javax.sip.*;
import javax.sip.message.*;
import javax.sip.header.*;
import javax.sip.address.*;
import java.net.*;
import java.util.*;

public class example2Listener implements SipListener{

private SipFactory mySipFactory;
private SipStack mySipStack;
private ListeningPoint myListeningPoint;

```

```

private SipProvider mySipProvider;
private MessageFactory myMessageFactory;
private HeaderFactory myHeaderFactory;
private AddressFactory myAddressFactory;
private Properties myProperties;
private String myIP;
private int myPort;
Example2GUI myGUI;

    public example2Listener(int port,Example2GUI GUI) throws Exception {

        myPort=port;
        myGUI=GUI;
        myIP=InetAddress.getLocalHost().getHostAddress();
        myGUI.jLabel5.setText("Initialized at IP "+ myIP+", port
"+myPort);

        mySipFactory=SipFactory.getInstance();
        mySipFactory.setPathName("gov.nist");

        myMessageFactory=mySipFactory.createMessageFactory();
        myHeaderFactory=mySipFactory.createHeaderFactory();
        myAddressFactory=mySipFactory.createAddressFactory();

        myProperties=new Properties();
        myProperties.setProperty("javax.sip.STACK_NAME", "myStack");
        mySipStack=mySipFactory.createSipStack(myProperties);

        myListeningPoint=mySipStack.createListeningPoint(myIP, myPort,
"udp");
        mySipProvider=mySipStack.createSipProvider(myListeningPoint);
        mySipProvider.addSipListener(this);
    }

public void processRequest(RequestEvent requestReceivedEvent) {
    Request myRequest=requestReceivedEvent.getRequest();
    myGUI.display("<<< "+ myRequest.toString());
}
public void processResponse(ResponseEvent responseReceivedEvent) {
}
public void processTimeout(TimeoutEvent timeoutEvent) {
}
public void processTransactionTerminated(TransactionTerminatedEvent
tevent) {
}
public void processDialogTerminated(DialogTerminatedEvent tevent) {
}
public void processIOException(IOExceptionEvent tevent) {
}

public void userInput(String destination,String aor) throws Exception
{

```

```

    Address destinationAddress =
myAddressFactory.createAddress(destination);
    javax.sip.address.URI myRequestURI=destinationAddress.getURI();
    Address addressOfRecord = myAddressFactory.createAddress(aor);
    Address contactAddress =
myAddressFactory.createAddress("sip:"+myIP+": "+myPort);

    ArrayList viaHeaders=new ArrayList();
    ViaHeader myViaHeader = myHeaderFactory.createViaHeader(myIP,
myPort, "udp",
    "z9hG4bKnashds7");
    viaHeaders.add(myViaHeader);
    MaxForwardsHeader myMaxForwardsHeader =
myHeaderFactory.createMaxForwardsHeader(70);
    CallIdHeader myCallIdHeader = mySipProvider.getNewCallId();
    CSeqHeader
myCSeqHeader=myHeaderFactory.createCSeqHeader(1L, "REGISTER");
    FromHeader
myFromHeader=myHeaderFactory.createFromHeader(addressOfRecord, "456248"
);
    ToHeader
myToHeader=myHeaderFactory.createToHeader(addressOfRecord, null);

    Request myRequest =
myMessageFactory.createRequest(myRequestURI, "REGISTER",
    myCallIdHeader, myCSeqHeader, myFromHeader, myToHeader,
viaHeaders, myMaxForwardsHeader);

    ContactHeader myContactHeader =
myHeaderFactory.createContactHeader(contactAddress);
    myRequest.addHeader(myContactHeader);

    mySipProvider.sendRequest(myRequest);
    myGUI.display(">>> "+myRequest.toString());
}
}
}
-----

```