

VoIP pour le développement

- Un guide pour construire un infrastructure de
télécommunication en région émergente



Auteurs:

Alberto Escudero-Pascual

Louise Berthilson

Traduit par:

Johan Bilien

Table des abréviations

Abréviation	Définition
ATA	Adaptateur pour Téléphone Analogique
DECT	Digital Enhanced Cordless Telecommunications
FXO	Foreign Exchange Office
FXS	Foreign Exchange Station
GSM	Global System for Mobile communication
IAX (IAX2)	Inter-Asterisk eXchange protocol (version 2)
IETF	Internet Engineering Task Force
ITU	International Telecommunications Union
IVR	Interactive Voice Response
NAT	Network Address Translator
PBX (PABX)	Private (Automatic) Branch Exchange
PCM	Pulse Code Modulation
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RFC	Request For Comment
RTP	Real-time Transport Protocol
SCCP	Skinny Call Control Protocol
SIP	Session Initiation Protocol
SS7	Signalling System 7
TA	Adaptateur Téléphonique
UDP	User Data Protocol
VoIP	Voix sur IP
VSAT	Very Small Aperture Terminal

Description des illustrations



PBX utilisant Asterisk



ATA



IAXy



Routeur sans fil (wireless router)



Passerelle sans fil (wireless bridge)



Switch



Routeur



Téléphone analogique



Téléphone DECT



Téléphone VoIP



Softphone



Antenne parabolique



VSAT



Satellite

Table des matières

1. Introduction.....	1
2. La potion magique.....	2
2.1 VoIP.....	2
2.2 Standards ouverts et logiciel libre.....	3
2.3 Asterisk.....	4
3. La recette.....	4
3.1 PBX.....	4
3.2 PSTN.....	5
3.3 Signalisation en téléphonie traditionnelle.....	6
3.3.1 Signalisation analogique.....	7
3.3.2 Signalisation entre centraux téléphoniques.....	7
3.4 Signalisation en téléphonie IP.....	7
3.4.1 Protocole d'Initiation de Session (Session Initiation Protocol, SIP).....	8
3.4.2 Serveurs Proxy.....	8
3.4.3 Protocole de transport temps-réel (Real-time Transport Protocol, RTP) et NAT.....	9
3.4.4 Protocole d'Échange Inter-Asterisk (Inter-Asterisk eXchange, IAX).....	9
3.5 Matériel de VoIP.....	10
3.5.1 Téléphone VoIP.....	10
3.5.2 Logiciels de VoIP.....	11
3.5.3 Cartes d'interface vers le PSTN.....	11
3.5.4 Adaptateurs pour Téléphone Analogique.....	12
3.6 Codecs.....	12
3.7 Qualité de Service.....	13
3.7.1 Latence.....	13
3.7.2 Gigue.....	13
4. En pratique – Construire votre PBX.....	15
4.1 De quoi ai-je besoin ?.....	15
4.1.1 Conseils d'installation.....	15
5. Installer Asterisk.....	16
5.1 Compiler Asterisk.....	17
5.2 Se procurer les paquets Asterisk.....	18
5.3 Commandes Asterisk de base.....	18
5.4 Fichiers de configuration.....	19
5.5 Pairs, utilisateurs et amis.....	20
6. SCÉNARIO A.....	22
- Réseau téléphonique privé dans une communauté rurale.....	22
6.1 Contexte.....	22
6.2 Configurer les clients de VoIP.....	23
6.2.1 Bibliothèque communautaire.....	23
6.2.2 Hôpital.....	24
6.2.3 École primaire.....	25
6.2.4 Collective agricole.....	26
6.3 Configurer Asterisk.....	27
7. SCÉNARIO B.....	29
- Joindre le PSTN.....	29
7.1 Ajouter le support de la carte TDM400P.....	29

7.2 Gérer les appels en provenance du PSTN.....	31
7.2.1 Rendre le PSTN accessible depuis le plan de numérotation.....	31
7.3 Connecter un téléphone analogique au PBX.....	32
7.4 Mettre à jour le plan de numérotation.....	33
8. SCÉNARIO C :.....	34
- Interconnecter des communautés par la VoIP.....	34
8.1 Problèmes inhérents aux connexions par satellite.....	34
8.2 Inter-connecter 2 serveurs Asterisk.....	35
8.2.1 Télécentre.....	35
8.2.2 Centre de formation.....	36
8.2.3 La commande « register ».....	37
9. Aller plus loin.....	37
10. Conclusion.....	37
11. Remerciements.....	38

1. Introduction

Ce n'est qu'au printemps 2004 que j'ai pour la première fois entendu parler d'un logiciel qui réponde parfaitement à mes besoins. Je vivais alors en Tanzanie, avec une ligne téléphonique peu fiable et des tarifs prohibitifs. Dès que je pus accéder à Internet, j'utilisai un logiciel propriétaire de téléphonie bien connu pour appeler à l'étranger, mais me voyais bientôt frustré par son manque de flexibilité. J'aurais vraiment voulu pouvoir utiliser ma connexion à Internet en Tanzanie pour pouvoir passer des coups de téléphone par l'intermédiaire de mon bureau en Suède, et d'offrir ce service à d'autres.

L'idée d'utiliser Internet comme une alternative au réseau téléphonique n'était pas nouvelle, mais le logiciel qui rendait cela possible était certainement « révolutionnaire ». Ce programme qui permet de convertir un quelconque ordinateur en un central téléphonique se nomme « *Asterisk* »¹. Rapidement je réalisai que ce programme libre me permettait de faire bien plus que je n'eusse pu imaginer, et ouvrait la porte à d'innombrables opportunités pour les communautés des pays développés comme celles des pays émergents. Ce sentiment se rapprochait beaucoup de celui éprouvé lors de mon premier contact avec le World Wide Web en 1994.

Sans aucun doute l'apprentissage de l'installation d'*Asterisk* fut difficile, et bien que de très bons livres soient disponibles aujourd'hui, je n'ai pas rencontré de documents qui présentent quelques notions fondamentales de manière intuitive au commun des mortels. Ce précis tente de présenter ces concepts essentiels à la téléphonie par Internet, et donne quelques exemples de son potentiel dans les régions émergentes. Le but étant de conserver ce document succinct mais utile, un effort particulier a été porté sur sa pédagogie.

Soyez patient. La persévérance est la clef du succès pour l'autodidacte.

Avant de décrire l'installation de votre propre système de téléphonie, nous présentons les bases de la téléphonie sur Internet (Section 2 et 3). Prenez le temps de parcourir ces sections, à long terme maîtriser les concepts de base est bien plus important que d'installer le logiciel.

Pour ceux qui aiment mettre la théorie en pratique, les deux sections suivantes couvrent l'installation d'un PBX, comprenant l'assemblage des composants matériels (Section 4) et l'installation du logiciel (Section 5).

Plutôt qu'une liste exhaustive des commandes et innombrables options de configuration, nous avons sélectionné 3 scénarios concrets comme exemple de base. Souvenez-vous que le but est de vous mettre sur le bon chemin. Les trois scénarios sont les suivants :

Téléphonie privée dans une communauté rurale
Connecter le réseau local de téléphonie au PSTN
Inter-connecter 2 communautés distantes
Nous concluons par une liste de références pour continuer votre apprentissage (Section 9).

¹ Asterisk est un projet opensource basé sur des idées du projet « Zapata Telephony »
<http://www.asteriskdocs.org/modules/tinycontent/index.php?id=10>

2. La potion magique

Trois ingrédients magiques vous permettront de déployer une infrastructure de téléphonie flexible et auto-gérée : la *VoIP*, les *standards ouverts* et les *logiciels libres*.

2.1 VoIP

La Voix sur IP (aussi appelée téléphonie IP) est la possibilité de transporter les conversations téléphoniques dans des paquets IP. Par Voix sur IP nous désignons téléphonie par Internet au sens large. Nous ne nous référons pas à un mécanisme concret de transport de la voix sur Internet en particulier. Parmi ces mécanismes on peut distinguer 2 catégories : systèmes fermés / propriétaires et ouverts / libres. Dans la première catégorie nous trouvons le très populaire Skype et le légendaire protocole de CISCO, Skinny (SCCP)². La seconde catégorie regroupe les protocoles ouverts basés sur SIP³, H.323⁴ et IAX2⁵.

Le protocole H.323, développé par l'ITU a longtemps été le plus populaire, par son large déploiement sur le « backbone » des grands opérateurs. SIP a gagné en popularité avec l'avancée de la Voix sur IP dans la « boucle locale »⁶. Récemment un nouvel arrivé, IAX, a fait son apparition. Largement basé sur un effort communautaire, IAX a su tirer les leçons des limitations de ses prédécesseurs et corriger beaucoup d'entre elles. Bien que n'ayant pas le statut de protocole officiel (RFC⁷), il a gagné un énorme respect par la communauté et a tous les requis pour se poser de facto en remplaçant de SIP.

L'un des principaux défauts des protocoles de VoIP traditionnels est le gâchis de bande passante généré par les en-têtes de paquets. Ce problème revêt une importance particulière dans les pays émergents, dans lesquels l'accès à la bande passante internationale est cruellement limité et les coûts d'accès à Internet sont au moins 100 fois supérieurs à ceux pratiqués en Europe ou en Amérique du Nord⁸.

Pour comprendre l'ampleur du surplus de bande-passante engendré, considérons le fait qu'un flux de 5,6 kbit/s de données audio compressées nécessitera 18 kbit/s de bande passante. La différence vient des en-têtes de paquet. Les en-têtes de paquet contiennent l'information nécessaire au transport des paquets de voix jusqu'au destinataire. IAX2 est parvenu à réduire ce surplus en limitant le nombre de bits supplémentaires par paquet. Il tire aussi avantage d'un concept consistant à regrouper le contenu des paquets ayant le même destinataire en un

2 Le protocole Skinny Client Control Protocol est un protocole propriétaire de contrôle de terminaux originellement développé par Selsius Corporation, et qui appartient aujourd'hui à Cisco Systems, Inc. L'un des clients Skinny les plus connus est la série de téléphones IP Cisco 7900.

3 Le Protocole d'Initiation de Session (Session Initiation Protocol, SIP) est un protocole de l'IETF qui décrit la création de session avec un ou plusieurs interlocuteurs.

4 H.323 regroupe un ensemble de recommandations de l'ITU-T qui définissent les protocoles nécessaires pour créer des sessions audio-visuelles sur tout réseau basés sur des paquets. Le célèbre Netmeeting se base sur ce protocole.

5 IAX2 (Inter-Asterisk) est le protocole utilisé par Asterisk, un serveur PBX opensource. Il permet de passer des appels de VoIP entre des serveurs Asterisk et des clients IAX2.

6 La boucle locale est le lien physique qui relie un client aux services d'un fournisseur de télécommunication.

7 Les « Requests for Comments » (RFC) sont une série numérotée de documents et normes concernant Internet.

8 1Mbit/s en Afrique de l'Est coûte plus de 1000 USD par mois, la même capacité coûtant en Suède moins de 10 USD par mois.

seul paquet.

Certains des tests effectués lors de la rédaction de ce précis (utilisant une connexion Internet par modem) ont mis en évidence qu'utiliser IAX2 plutôt que SIP rend une conversation téléphonique basée sur IP plus facilement réalisable⁹.

2.2 Standards ouverts et logiciel libre

Nous ne pouvons parler de la liberté de construire notre propre réseau téléphonique sans évoquer les standards ouverts et les logiciels libres. Les standards ouverts permettent à chacun de bâtir des systèmes de communication inter-opérables. Grâce à l'inter-opérabilité, nous pouvons mettre en place notre propre système de téléphonie, et dans un second temps le connecter au réseau mondial. Grâce au logiciel libre, nous bénéficions de l'expérience communautaire, pouvons intégrer nos propres solutions et au final partager nos résultats.

L'une des premières questions à laquelle nous devons répondre : pourquoi construire notre propre infrastructure de VoIP lorsque nous pourrions utiliser un service gratuit tel que Skype ?

La réponse est simple : viabilité et flexibilité. Les services gratuits peuvent constituer une solution à notre problème du moment, mais ne garantissent pas la propriété du service et donc ne peuvent garantir notre indépendance sur le long terme. La question n'est pas de déterminer quelle technologie est la meilleure, mais quelle technologie permet aux communautés d'être propriétaires de leur processus de développement et de l'adapter à leurs besoins.

Il est difficile d'imaginer un développement viable sans transfert de connaissance et sans être propriétaire de la technologie. Une solution basée sur les standards ouverts et les logiciels libres est non-seulement efficace techniquement, mais permet aussi une adaptation aisée aux besoins de chacun.

Pour comprendre l'importance des standards ouverts, nous commençons par donner une simple définition de ce que nous appelons un « standard ». Un standard est un ensemble de règles, conditions, ou exigences décrivant le matériel, produits, systèmes, services ou pratiques. En téléphonie, les standards techniques garantissent que les centraux téléphoniques au quatre coins du globe puissent inter-opérer. Sans standard le système téléphonique d'une région ne pourrait peut-être pas fonctionner avec un autre système distant de quelques kilomètres. Bien que beaucoup de standards téléphoniques soient disponibles publiquement, la téléphonie a toujours été sous le contrôle d'un nombre réduit de fabricants capables de négocier des contrats au niveau régional ou national. Cela explique probablement pourquoi l'on trouve souvent le même type d'équipement sur tout un pays. Un équipement téléphonique a toujours été un matériel conçu pour répondre à un besoin particulier, que l'on nomme matériel ou logiciel dédié. Si les règles (ou standards) qui régissent la téléphonie sont relativement ouvertes, celles qui régissent le matériel ont été tenues secrètes.

⁹ Une conversation téléphonique utilisant un algorithme de compression de la voix telle que G.729 (8 kbit/s) requiert 30 kbit/s lorsque SIP est utilisé, 24 kbit/s avec IAX2. Si nous regroupons (« trunking ») cinq appels, le débit par appel est de 13 kbit/s.

Les standards ouverts étaient l'une des conditions requises pour permettre à chacun de bâtir une infrastructure de téléphonie, mais le facteur déclencheur de cette « révolution » aura été la possibilité d'émuler ces systèmes coûteux (basés sur du matériel) avec un ordinateur et des logiciels. Nous avons à présent tous les ingrédients à notre disposition :

- le logiciel et matériel nécessaire pour établir des conversations téléphoniques
- le réseau ouvert et publique pour transporter ces conversations
- la possibilité d'adapter ces ingrédients à nos besoins

2.3 Asterisk

Asterisk est un logiciel libre qui implémente un central téléphonique. Le logiciel permet à un certain nombre de téléphones connectés d'échanger des appels, et de les relier au réseau téléphonique mondial. Le code fut originellement écrit par Marc Spencer (Digium), ce dernier se basant sur une idée et une première implémentation par Jim Dixon (Zapata Telephony Project). La communauté open source a contribué au logiciel et ses nombreuses nouvelles fonctionnalités (y compris par ses rapports de bogues). Si *Asterisk* fonctionne sur différentes plateformes, GNU/Linux est le système d'exploitation offrant le meilleur support.

Pour exécuter *Asterisk*, un ordinateur personnel (PC) suffit, mais du matériel supplémentaire sera nécessaire si vous souhaitez y connecter des téléphones ordinaires, ou le relier au réseau téléphonique.

3. La recette

Cette section résume les concepts fondamentaux de la VoIP. Comprendre chacun d'entre eux vous sera particulièrement utile lorsque vous serez amené à configurer tout logiciel ayant trait à la téléphonie IP. Si la VoIP est un vaste domaine de connaissance, nous avons sélectionné avec soin quelques concepts fondamentaux. Cette section vous fournira une connaissance basique mais solide de ce que vous devez savoir avant de vous lancer dans la construction de votre premier système téléphonique.

3.1 PBX

Un PBX ou PABX est un acronyme technique pour (Auto)commutateur Privé. En termes simplifiés, la plus courante utilisation d'un PBX est le partage d'une ou plusieurs lignes de téléphone entre plusieurs utilisateurs. Un PBX est placé entre les ligne de téléphones et plusieurs terminaux. Un PBX est capable de de redirigé les appels entrant vers un téléphone en particulier, ou de permettre aux téléphones de choisir une ligne en particulier pour passer un appel. Tel un routeur sur Internet est responsable de rediriger les paquets de données d'une source vers une ou plusieurs destinations, un PBX redirige les appels téléphoniques.

Le terme « Privé » dans l'acronyme PBX désigne le fait que le propriétaire du système gère l'équipement et décide de la manière dont les lignes externes sont partagées parmi ses utilisateurs.

Un PBX permet non-seulement de partager une ligne de téléphone parmi plusieurs utilisateurs, mais apporte aussi des services à valeur ajoutée tels que le transfert d'appel, la conférence à trois¹⁰, la messagerie vocale (ou messagerie vocale vers e-mail)¹¹, les serveurs interactifs vocaux (IVR)¹², etc. ...

Un PBX se révèle utile dans de nombreux scénarios. Par exemple dans les régions émergentes, l'accès au réseau téléphonique implique souvent la visite au point d'appel (cabine téléphonique privée) ou une longue marche jusqu'au télécentre. Une seule ligne de téléphone est disponible dans un immeuble de bureaux ou une région rurale est une situation récurrente, mais plusieurs organisations, situées dans le voisinage, aimeraient pouvoir placer et recevoir des appels. Un PBX permet aux organisations et individus dans un village rural de connecter et partager la seule ligne de téléphone disponible.

3.2 PSTN

PSTN est l'abréviation du réseau téléphonique public commuté, le « réseau des réseaux téléphoniques », ou plus couramment « réseau téléphonique ». Tel Internet qui inter-connecte tous les réseaux IP, le PSTN est le réseau résultant de l'interconnexion de tous les réseaux téléphoniques commutés du monde entier. Une différence importante avec Internet est cependant la définition du « flux d'information ». En Téléphonie (PSTN) un flux d'information est un « appel téléphonique » dans son ensemble, contrairement à Internet pour lequel chaque paquet constitue un flux à lui tout seul. Le PSTN et Internet représentent deux philosophies très différentes (techniquement et politiquement). Lorsqu'un appel est passé sur le PSTN, un circuit dédié de 64 kbit/s doit être réservé. Si l'appel est passé sur Internet, il coexiste avec de nombreuses autres services. Si cette différence semble dérisoire au premier abord, elle revêt une importance particulière dans le développement des technologies de l'information et de la communication (ITC) dans les pays développés ou émergents. Dans le modèle traditionnel, un « câble de cuivre » donne accès au PSTN et offre un service unique : un canal de transport de la voix analogique. Si ce même câble est utilisé pour se connecter à un réseau à commutation de paquets, tel qu'Internet, tout type de service basé sur IP peut être proposé.

Si le PSTN a historiquement été régi par des standards techniques créés par l'ITU¹³, l'Internet est lui régulé par les standards de l'IETF¹⁴. Les deux réseaux, le PSTN et Internet, utilisent

10 Possibilité de converser avec plus de 2 interlocuteurs simultanément.

11 Le service de messagerie vocale par e-mail permet d'enregistrer un message (comme sur un répondeur), ce message étant par la suite envoyé à une adresse e-mail.

12 Un service interactif vocal (Interactive Voice Response, IVR) permet à l'appelant de sélectionner des options proposés dans un menu vocal en utilisant les touches de son téléphone.

13 L'ITU (International Telecommunications Union) est l'organisation responsable de la standardisation, de l'allocation du spectre radio et de l'organisation de l'interconnexion de différents pays pour permettre les appels internationaux. Elle fait parties du système des Nations Unies, et possède un système formel de membres.

14 L'IETF (Internet Engineering Taskforce) est l'organisation responsable du développement et de la promotion des standards d'Internet. L'organisation est ouverte et basée sur le volontariat, sans système de membres.

des adresses pour diriger leur flux d'information. Le PSTN utilise des numéros de téléphone pour commuter les appels entre centraux téléphoniques. Internet utilise des adresses IP pour diriger les paquets entre ses routeurs.

3.3 Signalisation en téléphonie traditionnelle

Les centraux téléphoniques sont les « routeurs » du PSTN. Un FXO (« Foreign Exchange Office ») représente un équipement qui du point de vue du central téléphonique se comporte comme un téléphone ordinaire. Un FXO doit pouvoir recevoir un signal d'appel entrant, être décroché et raccroché, et recevoir et émettre des signaux vocaux. Représentez-vous un FXO comme un « téléphone » ou n'importe quel appareil qui « sonne » (fax, modem).

Un FXS (Foreign Exchange Station) représente ce qui se trouve à l'autre extrémité de la ligne de téléphone. Un FXS délivre la tonalité aux téléphones. Sur les lignes analogiques, un FXS pourvoit la tonalité et le courant électrique nécessaires au fonctionnement des téléphones. Pour vous donner une idée, un FXS (ligne de téléphone) délivre environ 48 Volts de courant continu (DC) pendant une conversation téléphonique, et jusqu'à 80 Volts de courant alternatif (AC) lorsqu'il génère une sonnerie.

Un PBX qui intègre des interfaces FXO et FXS peut se connecter au PSTN et accueillir des téléphones. Les lignes de téléphones provenant de l'opérateur doivent être connectées à une interface FXO. Le(s) téléphone(s) dans votre bureau doive(nt) être connecté(s) à une (des) interface(s) FXS du PBX.

Deux règles simples que vous devriez mémoriser :

1. Un FXS doit être connecté à un FXO (telle qu'une ligne de téléphone doit être connectée à un téléphone et réciproquement).
2. Un FXS pourvoit en courant (actif) le téléphone FXO (passif).

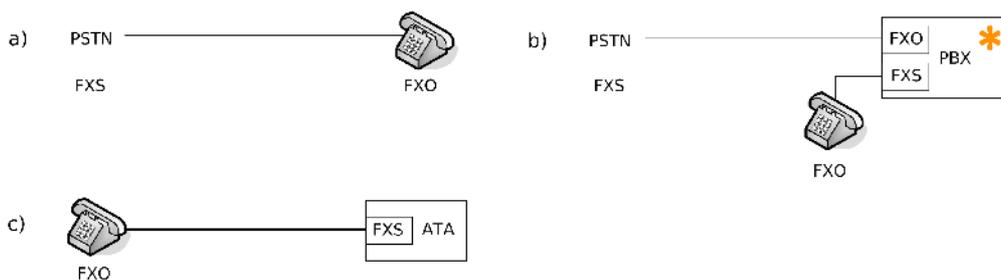


Image 1: a) Un téléphone peut être considéré comme un FXO connecté à un FXS (le PSTN). b) un PBX peut accueillir des FXO comme des FXS. c) Un ATA se comporte comme un FXS lorsqu'il connecte un téléphone (FXO) à un ordinateur ou PBX.

3.3.1 Signalisation analogique

Un ensemble de « signaux » est émis chaque fois que vous utilisez une ligne de téléphone pour présenter à l'utilisateur des informations sur le statut et le progrès de l'appel. Parmi ses signaux, on trouve la tonalité, le signal d'une ligne occupée, la sonnerie, et le statut du combiné (décroché ou raccroché). Ces signaux sont transmis entre le FXO et le FXS grâce à un protocole de signalisation.

Malheureusement ces signaux peuvent être générés de nombreuses manières différentes. Chaque mécanisme est une « méthode de signalisation ». Les méthodes de signalisation varient d'un endroit à un autre, c'est pourquoi il vous faut savoir quelle méthode est utilisée sur vos lignes. Deux méthodes de signalisation sont courantes, « loop start » et « ground start ». Si vous ne savez pas quelle méthode de signalisation est utilisée entre vos téléphones et votre ligne, vous devriez d'abord essayer « loop start ». Un symptôme courant d'une mauvaise sélection de la méthode de signalisation est l'interruption impromptue de la ligne.

3.3.2 Signalisation entre centraux téléphoniques

SS7 est un ensemble de standards développés par AT&T et l'ITU qui entre autres tâches gèrent l'établissement et l'acheminement des appels entre centraux téléphoniques du PSTN. L'important est de retenir que dans les réseaux téléphoniques traditionnels la voix et la signalisation sont séparées. Cela signifie qu'un « circuit » est utilisé pour la voix (conversation), et un autre circuit transporte des données additionnelles (supervision) nécessaires à l'établissement de l'appel. Ces données additionnelles (mais non indispensables) sont échangées grâce au protocole SS7.

Le fait que la « signalisation » et la « voix » sont séparées signifie qu'elles ne suivent pas forcément le même chemin physique lors de leur transmission. Les « conversations téléphoniques » sont susceptibles d'emprunter un câble lorsque les numéros de l'appelant et de l'appelé en empruntent un autre. Il est important de comprendre ce concept avant d'aborder la signalisation en téléphonie IP.

3.4 Signalisation en téléphonie IP

La signalisation de la Voix sur IP suit une philosophie similaire à celle du PSTN. Signalisation et conversation sont clairement différenciées. Dans cette section nous présentons deux protocoles de VoIP que nous allons intégrer à notre PBX : SIP et IAX2.

3.4.1 Protocole d'Initiation de Session (Session Initiation Protocol, SIP)

Le Protocole d'Initiation de Session (SIP) est un protocole Internet développé par l'IETF destiné à accomplir certaines des fonctions de SS7, mais sur des réseaux basés sur IP tels qu'Internet. Le protocole SIP est responsable d'établir les appels et d'assurer leur signalisation. Souvenez-vous que lorsque nous parlons de signalisation dans le contexte d'appels vocaux, nous désignons l'indication d'une ligne occupée, d'un appel entrant, ou que quelqu'un a répondu à l'autre extrémité de la ligne.

Les trois fonctions principales du protocole SIP sont :

- Gérer l'authentification
- Négocier la qualité¹⁵ d'un appel
- Gérer les adresses IP et les ports à utilisés lors de la transmission de la « conversation vocale »

3.4.2 Serveurs Proxy

Bien que deux équipements SIP (téléphones IP) puissent communiquer directement, SIP utilise des éléments additionnels appelés serveurs proxy afin de faciliter l'établissement d'un appel. Avec la téléphonie IP il est possible de déplacer votre « numéro de téléphone » partout dans le monde. Les « numéros de téléphone » ne sont pas liés à un emplacement physique déterminé. L'une des fonctions d'un serveur proxy SIP est d'agir en intermédiaire savant où trouver un numéro de téléphone sur le réseau. Un serveur proxy SIP est informé de la location actuelle d'un utilisateur par un processus appelé « enregistrement » (registration).

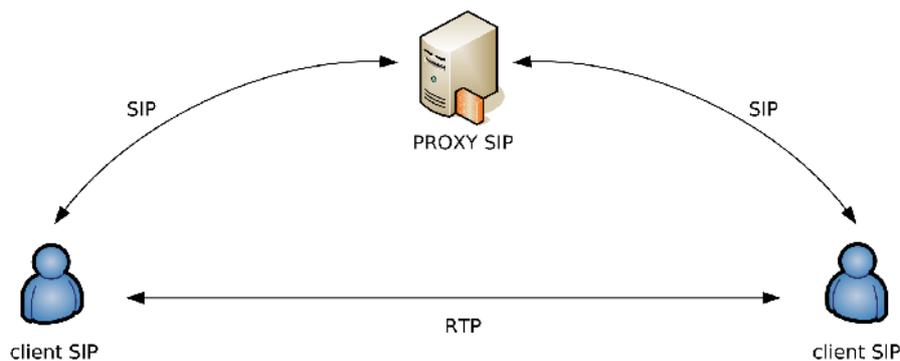


Image 2: Le processus d'enregistrement (register) d'un client au serveur proxy SIP.

¹⁵ Une différence importante en téléphonie basée sur IP est que la qualité d'un appel peut être négociée entre équipements.

3.4.3 Protocol de transport temps-réel (Real-time Transport Protocol, RTP) et NAT

Sur Internet, les conversations téléphoniques basées sur SIP sont constituées de flux de petits paquets transportés au moyen d'un autre protocole, RTP. Le protocole de transport temps-réel (Real-time Transport Protocol, RTP) est le véritable conteneur de la voix. RTP définit un format de paquet standard pour le transport de l'audio et de la vidéo sur Internet. Une conversation normale incluant 2 interlocuteurs, une conversation basée sur IP inclut de même deux flux RTP différents et séparés.

Les dispositifs de Traduction d'Adresse de Réseau (Network Address Translation, NAT) sont les grands ennemis de RTP. Un réseau NAT consiste en un ensemble d'ordinateurs partageant une seule adresse IP publique pour communiquer avec l'extérieur. Les ordinateurs à l'intérieur du réseau NAT se voient attribuer une adresse « privée ». Les NAT sont très utiles pour connecter facilement un grand nombre d'ordinateurs à Internet, mais les ordinateurs ainsi connectés ne sont pas complètement accessibles (« routable ») de l'extérieur du réseau.

Les réseaux NAT présentent de nombreux problèmes à la VoIP. Le problème le plus commun se nomme « audio à sens unique ». Souvenez-vous qu'une conversation normale inclut 2 flux RTP séparés. En présence d'un NAT, un seul des flux (de l'extérieur vers l'intérieur) est bloqué. En conséquence l'appelant ne peut entendre l'appelé.

Malheureusement les adresses privées sont presque omniprésentes dans les régions émergentes. C'est pourquoi les problèmes liés aux réseaux NAT sont fréquents lors d'implémentations de VoIP.

Installer un service de VoIP basé sur SIP en présence d'un réseau NAT n'est pas évident. Quelques règles générales sont incluses dans les scénarios décrits dans ce précis.

3.4.4 Protocole d'Échange Inter-Asterisk (Inter-Asterisk eXchange, IAX)

Le protocole d'Échange Inter-Asterisk (Inter-Asterisk eXchange) version 2 (IAX2)¹⁶ propose une alternative aux protocoles de signalisation tels que SIP. IAX2 a été créé dans le cadre du projet de PBX open-source *Asterisk*. Contrairement à SIP qui utilise 2 paires de flux (l'une pour la signalisation, l'autre pour la voix), IAX utilise une seule paire de flux pour communiquer entre les extrémités de la ligne (téléphone ou central téléphonique). La signalisation comme les données (la conversation vocale) sont transmises sur le même canal, par opposition à SIP qui utilise un second canal (« out-of-band ») pour les flux de données (RTP)¹⁷ transportant la voix.

¹⁶ IAX2 est un protocole de VoIP économe en bande passante, conçu pour permettre à plusieurs PBX basés sur Asterisk de communiquer entre eux. Les données sont envoyées avec un entête de 4 octets seulement. Un entête plus complexe de 12 octets pour des fonctions de contrôle et dans certains paquets de données (un par minute environ).

¹⁷ Transmettre la signalisation dans le même flux (« in-band ») nécessite que le PBX sépare la signalisation et la voix. Bien que ceci monopolise davantage de ressources, cela facilite l'interaction d'IAX2 avec les NAT et pare-feu.

Comme nous l'avons vu les réseaux NAT sont relativement courants dans les régions émergentes, aussi IAX2 est-il le bienvenu dans beaucoup de ces environnements. De plus, IAX2 permet à plusieurs appels d'être rassemblés dans un seul ensemble de paquets IP, puisque qu'un seul paquet peut transporter des informations concernant plusieurs appels en cours. Ce mécanisme se nomme « trunking ». Avec IAX2, le « trunking » permet des économies de bande passante.

Le concept de « trunking » peut être expliqué par la métaphore suivante : imaginez que vous ayez à envoyer cinq lettres à des destinataires vivant dans un autre pays. Vous pouvez utiliser une enveloppe par lettre, ou inclure les cinq lettres dans une seule enveloppe et inclure le nom du destinataire final en première ligne de chacune des lettres. Le « trunking » opère de façon similaire et permet d'envoyer plusieurs lettres (appels) dans une seule enveloppe (paquet IP).

En résumé, IAX2 se révèle un bon candidat à la VoIP en région émergente, pour trois raisons :

1. Minimise la bande passante par appel
2. Inclus le support des réseaux NAT (plus facile à utiliser derrière un pare-feu)
3. Réduit la consommation de bande passante pour un ensemble d'appels (par l'utilisation du « trunking »)

3.5 Matériel de VoIP

3.5.1 Téléphone VoIP

Un téléphone VoIP est un matériel dédié qui se connecte à un réseau de VoIP. Un téléphone VoIP peut supporter un ou plusieurs protocoles de VoIP¹⁸.

Quelques fonctions intéressantes à prendre en considération lors de l'achat d'un téléphone VoIP :

- bande passante réduite : support des codecs à haute compression (par exemple G.729, Speex)
- bonne interface d'administration : interface Web
- interface audio : sortie audio externe et support des kits main-libre (pour la formation à distance)

Plusieurs modèles dont les fonctionnalités vont au delà du nécessaire et fonctionnent sans problème avec Asterisk sont disponibles dans des prix autour de 100 – 120 USD¹⁹.

¹⁸ Depuis septembre 2006, plusieurs compagnies produisent des téléphones compatibles IAX2.

¹⁹ Les modèles suivantes ont été évalués lors de la rédaction de ce précis : SwissVoice IP10S (150 USD), Thomson ST2030 (100 USD), Gulfstream Budgetome (75 USD), Cisco 7940 (300 USD).

3.5.2 Logiciels de VoIP

Une alternative aux téléphones de VoIP est l'installation d'un logiciel de VoIP sur un PC. Un tel logiciel est souvent appelé logiciel de téléphonie ou « softphone ». Le seul pré-requis est une carte son fonctionnelle et l'assurance que votre pare-feu personnel ne bloque pas le logiciel²⁰.

Si vous souhaitez bénéficier d'économies de bande passante, considérez l'utilisation d'un logiciel supportant IAX2.

3.5.3 Cartes d'interface vers le PSTN

Si vous souhaitez acheminer des appels de vos terminaux de VoIP vers le réseau téléphonique traditionnel (PSTN), il vous faudra inclure un équipement dédié à cet effet dans votre PBX. La carte PCI *TDM400P wildcard* de Digium est une solution flexible pour *Asterisk*, vous permettant de connecter des lignes et des terminaux analogiques.

Le terme « wildcard » (joker) vient de la capacité de cette carte à accueillir diverses combinaisons de modules FXO et FXS dans 4 ports. Cela signifie que la même carte PCI peut être utilisée pour connecter 4 lignes entrantes (4 modules FXO), ou 2 lignes entrantes (2 FXO) et 2 téléphones analogiques (2 modules FXS) etc ...

Pour commencer, envisagez l'achat d'une carte TDM400P, d'un module FXO (pour connecter une ligne de téléphone) et d'un module FXS (pour connecter un terminal analogique). S'il vous faut vous élargir votre installation plus tard vous pourrez acheter d'autres modules.

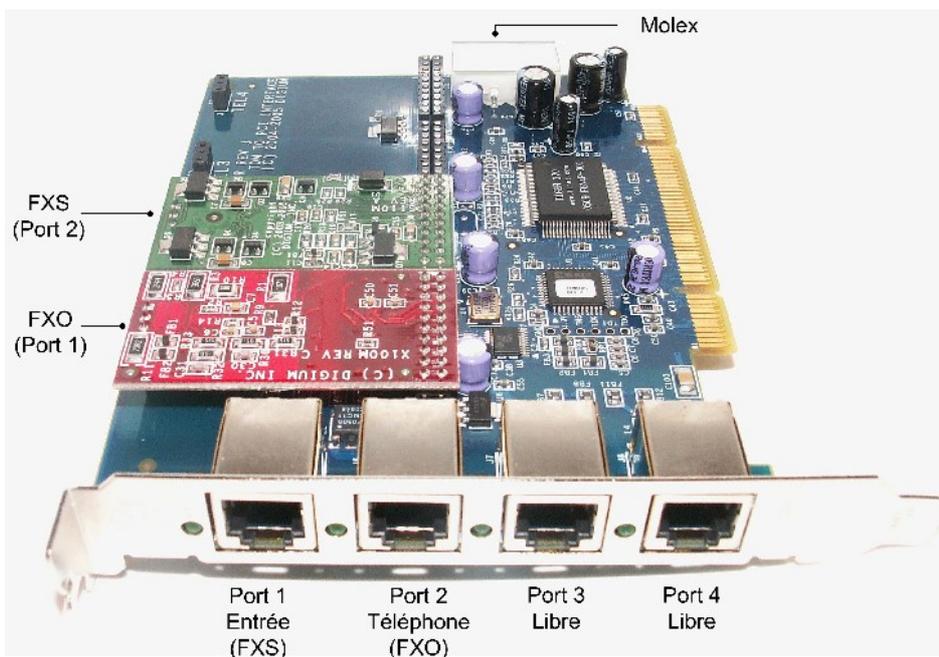


Image 3: Une carte TDM400P wildcard à 4 ports. Deux ports sont utilisés (un FXO et un FXS), les deux autres restant libres pour une utilisation ultérieure.

²⁰ Les logiciels de VoIP suivants ont été évalués lors de la rédaction de ce précis : IAXClient (IAX2), X-Lite (SIP)

3.5.4 Adaptateurs pour Téléphone Analogique

Un Adaptateur pour Téléphone Analogique (ATA) ou Adaptateur pour Téléphone (TA) connecte un téléphone analogique standard à un réseau de VoIP. Un ATA est pourvu d'un connecteur RJ11 (prise de téléphone) et d'un connecteur RJ45 (Ethernet). Un ATA fonctionne de façon analogue à un adaptateur FXS, parlant « analogique » au téléphone et « numérique » au réseau de VoIP. En cas de développement dans une région émergente, cette option est intéressante car généralement moins chère que des terminaux de VoIP. De plus, les ATA sont généralement plus petits et par conséquent moins susceptibles de vous poser problème auprès des douanes à l'importation. L'un des avantages des ATA est qu'il est possible d'y relier tout type de téléphone, y compris un téléphone payant, un fax ou un téléphone sans fil (DECT)²¹.

Le s101i de Digium est une solution ATA populaire, qui supporte IAX2. Cet ATA est aussi connu sous le nom IAXy²².

3.6 Codecs

Un algorithme de compression / décompression (codec) est un ensemble de transformations utilisées pour numériser la voix ou reconstituer le signal analogique à partir des données numérisées. Un codec prend en entrée un signal analogique et le convertit en un format binaire (des 0 et des 1). Il existe de nombreuses méthodes de numérisation, et en conséquence de nombreux types de codecs. Vous pouvez supposer qu'une plus grande compression engendrent une plus grande distorsion (moindre qualité). Un codec est meilleur s'il propose une restitution de meilleure qualité pour une consommation de bande passante équivalente.

Un circuit numérique de base sur le PSTN (votre téléphone de tous les jours) utilise typiquement un codec appelé « Pulse Code Modulation » (PCM). PCM est un codec de bonne qualité, mais requiert 64 kbit/s. Deux variantes du codec PCM sont utilisées, `μ-law` et `a-law`. Ces standards sont aussi connus sous les dénominations G.711u et G.711a. `μ-law` est courant en Amérique du Nord, `a-law` en Europe. Les codecs G.711 ne requièrent que peu de ressources et sont disponibles dans la plupart, si ce n'est la totalité, des équipements de VoIP.

Dans les régions émergentes, l'utilisation de G.711 consomme trop de bande passante pour un simple appel, et il vous faut considérer l'utilisation d'autres types de codecs.

GSM et Speex sont 2 codecs gratuits et bonnes qualité. Si G.729 donne également de bons résultats, il nécessite l'acquisition d'une licence pour un usage commercial²³.

21 Les modèles ATA suivants ont été évalués lors de la rédaction de ce précis : Sipura SPA-3000, GlobalTex IAD

22 L'IAXy est un ATA de petite taille, compatible IAX2. Il ne prend pas en charge les codecs à haute compression

23 G.729 est un codec au débit de 8 kbit/s (environ 30 kbit/s lorsque SIP est utilisé). Ce codec a été développé par un consortium d'organisations : France Telecom, Mitsubishi Electric Corporation, Nippon Telegraph and Telephone Corporation (NTT), et l'Université de Sherbrooke. Ce codec coûte environ 10 USD.

3.7 Qualité de Service

La Qualité de Service (QoS) est la capacité pour un réseau à accroître ses performances pour une portion spécifique de son trafic. L'une des plus grandes difficultés lors de l'implémentation de la VoIP, en particulier dans les régions émergentes, est de garantir que la bande passante requise aux appels téléphoniques soient toujours disponible, quelle que soit le ratio d'utilisation de la connexion à Internet. Lors de l'élaboration d'un réseau de VoIP, vous devriez optimiser l'utilisation de la bande passante, contrôler la gigue et minimiser les délais de latence.

3.7.1 Latence

La latence (ou délai) mesure le temps nécessaire à l'acheminement d'un paquet de données d'un point à un autre. Afin d'améliorer la qualité de vos conversations par VoIP, essayez de minimiser les délais en donnant priorité au trafic vocal. En d'autres termes, vous permettez aux paquets de VoIP d'être placés en meilleure position dans la queue des messages en attente de transmission²⁴.

Si la communication requiert l'utilisation d'un satellite, des délais de l'ordre de 300 ms sont inévitables. Afin de minimiser le délai, vous devez porter une attention particulière aux switchs et routeurs traversés. Les conversations sont possibles si la connexion passe par plusieurs relais satellite, mais prévoyez d'attendre la réponse de votre correspondant au moins une seconde sans parler. Une règle simple, placez votre PBX à l'endroit le moins saturé de votre réseau.

3.7.2 Gigue

En VoIP, la gigue (« jitter ») est la variation dans le temps du délais de transmission, due à la congestion du réseau, une dérive temporelle ou des changements de route. Un « tampon de gigue » (« jitter buffer ») permet de gérer la gigue et d'amoindrir ses effets néfastes. L'idée de base d'un tampon de gigue est d'améliorer la qualité d'un appel en retardant volontairement la lecture des données vocales pour permettre à tous les paquets retardataires d'arriver à temps.

Les équipements de VoIP vous permettent de définir un « tampon de gigue ». Un tampon de gigue est une zone de données partagée où les paquets de données vocales sont collectées, stockées, puis envoyées au processeur de voix à intervalles réguliers. Le « tampon de gigue », situé dans le terminal du destinataire de flux vocal, délaie intentionnellement les paquets arrivants de telle sorte que l'utilisateur bénéficie d'une connexion fluide et de peu de distorsions sonores.

Il existe deux types de tampons de gigue, statique et dynamique. Un tampon statique est

²⁴ Une attention particulière doit être portée lorsque vous utilisé un réseau sans-fil tels que ceux basés sur IEEE 802.11 b/g/a, afin de vous assurer de la priorité du trafic est accordée à la VoIP.

basée sur le matériel et se voit configuré par le fabricant. Un tampon dynamique est basé sur du logiciel, et peut être configuré par l'utilisateur. 100 ms est une valeur courante pour le tampon de gigue. Vous pouvez améliorer la qualité de la conversation en augmentant le tampon, mais au détriment d'un plus grand délai.

PARTIE 2 – En pratique

4. En pratique – Construire votre PBX

4.1 De quoi ai-je besoin ?

La première chose dont vous aurez besoin est un ordinateur personnel. Tout ordinateur produit après l'an 2000 devrait être assez puissant pour faire tourner *Asterisk* et vous permettre de débiter. Une variante du système d'exploitation Linux doit être installée²⁵.

Les téléphones logiciels constituent l'option la moins coûteuse et la plus simple pour commencer votre apprentissage. Votre premier exercice devrait être de passer des appels entre deux ordinateurs par l'intermédiaire de votre propre PBX. Deux ordinateurs équipés de cartes son, sur lesquels vous installerez deux logiciels de VoIP, et un troisième ordinateur sur lequel *Asterisk* sera installé, constitueront votre premier banc d'essai le plus simple.

Si vous souhaitez utiliser des terminaux de VoIP (téléphones VoIP ou ATA), ou vous connecter au PSTN, vous aurez besoin de quelques équipements supplémentaires décrits en section 3.5²⁶.

Si vous prévoyez de construire un PBX transportable, nécessitant peu de puissance, jetez un oeil aux cartes mini-ITX. La carte EPIA M10000 (aussi appelée Nehemiah) a deux ports PCI dans lesquels vous pouvez connecter une carte Digium TDM400P²⁷.

4.1.1 Conseils d'installation

Nous vous proposons ci-dessous quelques conseils d'installation :

Si vous avez besoin d'une carte TDM400P :

- Assurez-vous que votre PC a un port PCI 2.2 disponible
- Si vous prévoyez de connecter un module FXS à la carte TDM400P, il vous faudra un connecteur molex disponible²⁸
- Si vous utilisez une carte mère à taille réduite telle qu'une carte mini-ITX, il vous faudra une carte PCI « raiser » capable d'accueillir la carte TDM400P inclinée à 90 degrés.
- Ne connecter pas une ligne de téléphone (FXS) dans un port FXS, cela

²⁵ Si vous débutez dans l'utilisation de Linux, un bon choix serait de télécharger ou de commander un CD d'Ubuntu <http://www.ubuntu.com/>.

²⁶ Un bon fournisseur de matériel de VoIP, qui propose des rabais aux organisations à but non-lucratif <http://www.avandaza7.com/>.

²⁷ Un « PBX portable » vous permettra de démontrer votre technologie avec l'énorme avantage de pouvoir transporter votre central téléphonique en bagage à main.

²⁸ Une prise Molex est un connecteur d'alimentation électrique à 4 pins utilisés dans les PC. Les fils jaunes et rouges fournissent une tension de 12V et 5V respectivement, les fils blancs fournissant la masse.

endommagerait le port



Image 4: Un PBX basé sur une carte EPIA M10000 équipée d'une carte Digium TDM400P.

5. Installer Asterisk

Asterisk est un imposant logiciel. Sa puissante architecture et sa flexibilité engendrent une complexité due aux nombreuses options et fichiers de configuration. *Asterisk* vous permet d'accomplir quasiment tout ce que vous pouvez imaginer, aussi la première approche peut-elle sembler difficile. L'apprentissage de la configuration d'*Asterisk* m'a rappelé les innombrables nuits passées à dompter d'autres puissants logiciels tel que le serveur web *Apache* ou *Sendmail* il y a dix ans. L'immense étendue de leurs fonctionnalités rend parfois difficiles les plus simples accomplissements.

Dans ce petit guide, nous avons choisi de ne pas énumérer l'ensemble des commandes, mais de nous limiter à trois scénarios concrets d'utilisation d'*Asterisk*. Nous nous efforçons de conserver l'exercice aussi simple que possible. Nous savons que les solutions proposées ne sont pas exhaustives, aussi n'hésitez pas à expérimenter vos propres configurations.

5.1 Compiler Asterisk

Comme pour beaucoup d'autres logiciels libres, deux méthodes vous permettent d'installer *Asterisk* sur votre ordinateur. La première méthode consiste à télécharger et compiler le code source d'*Asterisk*. La seconde est de télécharger des paquets pré-compilés.

Si vous décidez de compiler le code source d'*Asterisk*, les conseils suivants devraient vous faciliter la tâche :

- Télécharger le code source d'Asterisk depuis le site <http://www.asterisk.org/>²⁹
- Pour les configurations de base, vous n'avez pas besoin des paquets « add-ons » et « music ».
- Lors de sa compilation, le code source d'Asterisk requiert d'autres logiciels sur votre système. Assurez-vous que les paquets suivants sont installés sur votre système :
 - bison (un générateur d'analyseur syntaxique)
 - zlib et zlib-devel (bibliothèque de compression – développement)
 - openssl et openssl-dev (libssl-dev) (bibliothèques de développement SSL)
 - libc6-dev (bibliothèque de développement et entêtes GNU C)
 - gcc et make (le compilateur C et l'utilitaire make)

La compilation d'Asterisk ne diffère pas de celles d'autres logiciels libres sous Linux :

Pour compiler :

```
# make
```

Pour installer le logiciel :

```
# make install
```

Pour installer les scripts de démarrage :

```
# make config
```

Pour installer les fichiers de configuration par défaut (exemples) :

```
# make samples
```

Si vous prévoyez d'utiliser une interface Digium Wildcard (tm) avec *Asterisk*, il vous faudra compiler et installer le module kernel zaptel pour votre carte.

- Télécharger le code source de Zaptel à partir du site <http://www.asterisk.org/>. Malheureusement le module zaptel n'est pas inclus au noyau Linux, c'est pourquoi il vous faut le compiler vous-même.
- Assurez-vous que le paquet contenant les entêtes de développement pour votre noyau est installé sur votre système³⁰.

²⁹ À ce jour, septembre 2006, la dernière version d'Asterisk est 1.2.12. La dernière version des drivers pour les cartes Zapata est 1.2.9

³⁰ Vous pouvez obtenir la version du noyau de votre système en utilisant la commande `# uname -a`. Par exemple pour un système mini-ITX sous Ubuntu Dapper, installer les entêtes suivants : `linux-headers-2.6.15-254386`

5.2 Se procurer les paquets Asterisk

Il est aussi possible de vous procurer une version pré-compilée d'*Asterisk*. Les logiciels pré-compilés se présentent sous la forme de « paquets ». Les formats courants de paquets sous Linux (suivant votre distribution) sont rpm, deb ou tgz. Si vous utilisez une distribution basée sur Debian, téléchargez et installez (apt-get install) les paquets suivants :

Paquet	Description
asterisk-lassic (obligatoire)	Private Branch Exchange (PBX) opensource – version originale par Digium
asterisk-config (suggéré)	Fichiers de configuration pour <i>Asterisk</i>
asterisk-dev (optionnel)	Fichiers de développement pour <i>Asterisk</i>
asterisk-doc (suggéré)	Documentation pour <i>Asterisk</i>
asterisk-sounds-extra (optionnel)	Fichiers de sons additionnels pour <i>Asterisk</i>
asterisk-sounds-main (optionnel)	Fichiers de sons pour <i>Asterisk</i>

À ce jour, il n'existe pas encore de version pré-compilée des modules de noyau zaptel. Vous n'avez pas d'autres alternatives que de suivre la méthode décrite dans la section précédente. Téléchargez le code source du module de noyau zaptel et compilez-le. Avant de procéder à la compilation du module zaptel, assurez-vous d'avoir installé les entêtes de développement correspondant à la version de votre noyau.

Paquet	Description
zaptel (obligatoire)	Utilitaires zaptel
zaptel-source (obligatoire)	Code source du module noyau zaptel
linux-headers-2.6.15-25-386 (spécifique à votre distribution)	Entêtes de développement du noyau Linux pour le noyau x386 de Ubuntu Dapper

Paquet	Description
zaptel (obligatoire)	Utilitaire zaptel
zaptel-source (obligatoire)	Code source du module noyau zaptel
linux-headers-2.6.15-25-386 (spécifique à votre distribution)	Entêtes de développement du noyau Linux pour le noyau x386 de Ubuntu Dapper

5.3 Commandes Asterisk de base

Asterisk inclus deux modules : un serveur, qui fonctionne en général en processus d'arrière-plan, et un client (CLI) qui permet de contrôler ce dernier. Le serveur et le client sont exécutés par la même commande « asterisk » mais suivie de différents paramètres.

Une fois *Asterisk* installé correctement, vous devriez vous familiariser avec quelques unes des commandes de base :

Démarrer / Arrêter *Asterisk* à partir d'un « run-level »

Exécuter *Asterisk* (qui se placera en arrière plan) :

```
# /etc/init.d/asterisk (start|stop)
```

Démarrer *Asterisk* depuis la ligne de commande :

Une alternative est de démarrer *Asterisk* depuis la ligne de commande (en tant que daemon)

```
# asterisk
```

Exécuter le serveur *Asterisk* en mode « bavard » (-vvv) , et ouvrir une « console » cliente (-c) (la console cliente, ou CLI, vous permet de contrôler ce qu'il se passe dans le serveur *Asterisk*)

```
# asterisk -vvvc
```

Si le serveur est déjà lancé, ouvrir un terminal client et se connecter au serveur pour contrôler son statut (-r) :

```
# asterisk -r
```

Commandes de base de la CLI

Recharger tous les fichiers de configuration ;

```
#CLI> reload
```

Activer les informations de diagnostique pour SIP ou IAX2

```
#CLI> IAX2 debug  
#CLI> SIP debug
```

Désactiver les informations de diagnostique pour SIP ou IAX2

```
#CLI> IAX2 nodebug  
#CLI> SIP nodebug
```

Afficher le statut des utilisateurs, pairs et canaux pour SIP ou IAX2

```
#CLI> sip show users  
#CLI> sip show peers  
#CLI> sip show channels  
#CLI> iax2 show peers  
#CLI> iax2 show users  
#CLI> iax2 show channels
```

5.4 Fichiers de configuration

Le nombre de fichiers de configuration que vous aurez à modifier afin d'exécuter *Asterisk* dépend de quelles technologies de VoIP vous souhaitez combiner dans votre installation. La logique fondamentale derrière la configuration d'*Asterisk* peut être résumée en deux étapes :

Première étape : définir et configurer les canaux de communication

Premièrement, il vous faut définir et configurer les canaux de communication que vous souhaitez utiliser. Un moyen simple de se représenter un canal de communication est d'imaginer une ligne de téléphone. Les canaux sont les « lignes virtuelles » de votre PBX. Internet vous permettant d'avoir plusieurs conversations simultanées partageant la même ligne, il vous faut définir à quelle connexion physique vous associez ces lignes virtuelles.

Gardez à l'esprit qu'*Asterisk* vous permet de relier des réseaux de VoIP utilisant différents protocoles. En pratique, cela signifie que vous pouvez interconnecter plusieurs types d'équipements de téléphonie IP (téléphones VoIP, ATA, softphones) et non-IP. Les fichiers de

configuration que vous devrez rédiger dépendent du type de VoIP que vous souhaitez utiliser. N'oubliez pas d'installer les fichiers d'exemple dont vous pourrez vous inspirer.

Seconde étape : créer les règles de vos extensions (Définir votre plan de numérotation).

Dans un second temps, il vous faut décrire les règles définissant l'interaction des ces canaux. Les appels arrivent et repartent de votre PBX par les canaux définis précédemment, mais ces canaux peuvent interagir de nombreuses manières. Par exemple, vous pourriez décider qu'un appel arrivant du PSTN soit relayé automatiquement vers un terminal de VoIP, ou interconnecter deux téléphones IP séparés de 20 kilomètres par l'intermédiaire d'un réseau sans fil. Ce type de « logique » inter-canaux est définie dans un fichier de configuration appelés le *fichier d'extensions*. Le fichier d'extensions contient toutes les règles de votre plan de numérotation.

Afin de vous représenter ces concepts de VoIP, imaginez les temps anciens de la téléphonie, lorsqu'une personne (l'opérateur) gérait la connexion manuelle des câbles de téléphone. Afin de passer un appel entre deux lignes de téléphone (les canaux de communication), il était nécessaire de contacter d'abord l'opérateur (le PBX) puis d'indiquer quel numéro vous souhaitiez joindre (fichier d'extension). Les fichiers de configuration des canaux décrivent les différents types de ligne utilisés, et le fichier d'extensions joue le rôle de l'opérateur téléphonique.

Nous allons utiliser ces 5 fichiers de configuration dans trois scénarios :

Fichier de configuration	Description
/etc/asterisk/extensions.conf (toujours obligatoire)	Établit le plan de numérotation et relie les canaux de communication
/etc/asterisk/sip.conf	Permet de configurer les canaux basés sur SIP (terminaux de VoIP basés sur SIP et fournisseurs de services SIP)
/etc/asterisk/iax.conf	Permet de configurer les canaux basés sur IAX2
/etc/asterisk/zapata.conf	Contient la configuration du matériel d'interfaçage avec le PSTN. Utilisé par <i>Asterisk</i> au démarrage
/etc/zaptel.conf	Configuration de bas-niveau des interfaces zaptel. Utilisé par l'utilitaire de configuration Zaptel <i>ztcfg</i> avant de démarrer <i>Asterisk</i> .

5.5 Pairs, utilisateurs et amis

L'un des concepts les plus déstabilisant pour un débutant utilisant *Asterisk* (tout au moins il l'a été pour moi pendant longtemps) est celui de pairs (*peers*) et utilisateurs (*users*) utilisés par les fichiers de configuration *iax.conf* et *sip.conf*.

Les termes « *peer* » (pair), « *user* » (utilisateur) et « *friend* » (ami) sont utilisés pour classer les appels entrants et sortants. Si une connexion spécifiée comme étant « *user* » s'authentifie

auprès de notre PBX (c'est à dire effectue un appel entrant), une connexion dite « *peer* » représente au contraire un appel sortant. Les utilisateurs désignés comme « *users* » nous appellent et nous appelons les terminaux définis comme « *peers* »³¹. « *friend* » désigne une connexion qui peut être aussi bien entrante que sortante.

Lorsque nous recevons un appel d'un terminal dit « *user* » ou « *friend* », nous devons définir ce qu'il en adviendra. Le terme « *context* » (contexte) désigne la règle ou l'ensemble de règles du plan de numérotation (*extensions.conf*) qui seront appliquées à chaque appel. Un « *context* » associé à un appel entrant est le point d'entrée de cet appel dans le plan de numérotation.

Le fichier *extensions.conf* contient tous les numéros accessibles depuis le PBX, séparés en différentes sections (« *contexts* »). Chaque « *user* » (appels entrants) définis dans chacun des canaux de communication se voit associé à une certaine section du plan de numérotation (*extensions.conf*)

³¹ Notez une exception à cette définition simpliste. Lorsque nous jouons le rôle d'un « proxy », un appel entrant provenant d'un de nos « peers » sera traité dans la section des « peers » (et non pas celles des « users »). Autrement dit, un pair se comportant comme un proxy ne peut s'authentifier en tant qu'un autre utilisateur, et la seule information disponible pour l'authentification est l'adresse IP du pair. En résumé, une connexion sortante est toujours un « peer », une connexion entrante un « user » sauf lorsqu'ils se comporte en proxy.

PARTIE 3: Scénarios

6. SCÉNARIO A

- Réseau téléphonique privé dans une communauté rurale

6.1 Contexte

Dans notre premier scénario, nous souhaitons installer un PBX dans un télécentre rural et fournir un service de téléphonie IP à quatre organisations du voisinage. En cas de succès, chaque organisation pourra passer des appels gratuitement³², vers le télécentre et entre elles. Le tableau suivant résume les informations relatives à ces quatre organisations et les différentes technologies pouvant être utilisées lors de leur connexion à notre PBX. Afin de rendre cet exemple plus didactique, nous avons choisi différentes technologies pour connecter chacune des organisations. Lors d'une véritable installation, il est préférable de réduire le nombre des technologies afin de faciliter le support et la maintenance.

Organisation	Technologie	Extension
Bibliothèque communautaire	téléphone VoIP basé sur le protocole SIP	462
Hôpital régional	ATA basé sur le protocole SIP	463
École primaire	ATA basé sur le protocole IAX2	464
Collective agricole	Deux softphones utilisant SIP et IAX2	465, 466

³² Lorsque l'infrastructure est en place, les tarifs des appels internationaux couvriront les frais de maintenance. Un modèle communautaire dans lequel les appels internes sont gratuits peut être exploré.

6.2 Configurer les clients de VoIP

Avant de décrire la configuration du PBX, intéressons-nous à celle de chacun des clients de VoIP.

6.2.1 Bibliothèque communautaire

Le premier client se trouve dans une bibliothèque publique à 1 km du télécentre. Le téléphone SIP est relié directement à notre PBX au moyen d'une connexion sans fil dédiée (connexion point-à-point « bridée »). L'adresse IP du téléphone (192.168.46.2) est dans le même réseau que notre PBX (192.168.46.1), ainsi nous n'avons pas à nous soucier d'un quelconque problème de NAT.

Lors de la configuration d'un téléphone de VoIP, commencez par lire le manuel d'utilisation afin de déterminer comment se connecter à l'interface web³³. Une fois connecté, cherchez dans l'interface web les paramètres de base suivants.

Paramètre	Valeur
Adresse IP du téléphone de VoIP	192.168.46.2
Adresse IP du proxy SIP (notre PBX) « Registration » (enregistrement)	192.168.46.1 OUI
nom d'utilisateur/ identifiant d'authentification	462
Identifiant de l'appelant	462
Mot de passe	462pass
Codec	G.711 (u-law)

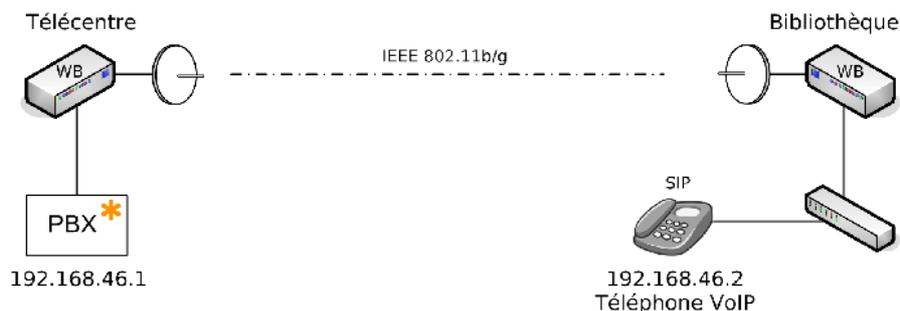


Image 5: La bibliothèque est connectée au PBX via une passerelle point-à-point sans fil. Un terminal de VoIP est utilisé dans la bibliothèque pour passer et recevoir des appels.

³³ Il existe trois méthodes pour se connecter à l'interface web d'un téléphone de VoIP. Par clavier : composez l'adresse IP du téléphone en utilisant ses touches. Par DHCP : connectez le téléphone à un réseau muni de DHCP et observez quelle adresse est attribuée. Par l'adresse configurée en usine : déterminez l'adresse IP configurée par défaut.

6.2.2 Hôpital

Le second client de notre réseau téléphonique interne est un ATA se trouvant dans l'hôpital local. Ce dernier se trouve en face du télécentre et est connecté au moyen d'un câble Ethernet Cat5 long de 100 m. La configuration d'un ATA est similaire à celle d'un téléphone de VoIP. Comme précédemment nous allons utiliser l'interface web pour configurer l'ATA avec les paramètres suivants:

Paramètre	Valeur
Adresse IP de l'ATA	192.168.46.3
Adresse IP du proxy SIP (notre PBX)	192.168.46.1
« Registration » (enregistrement)	OUI
nom d'utilisateur/ identifiant d'authentification	463
Identifiant de l'appelant	463
Mot de passe	463pass
Codec	G.711 (u law)

En place d'un téléphone fixe traditionnel nous décidons de relier un récepteur sans fil DECT³⁴ au port RJ11 de l'ATA. Ainsi nous pouvons utiliser un téléphone sans fil partout dans l'hôpital. L'ATA inter-connecte le téléphone sans fil et le réseau de VoIP.

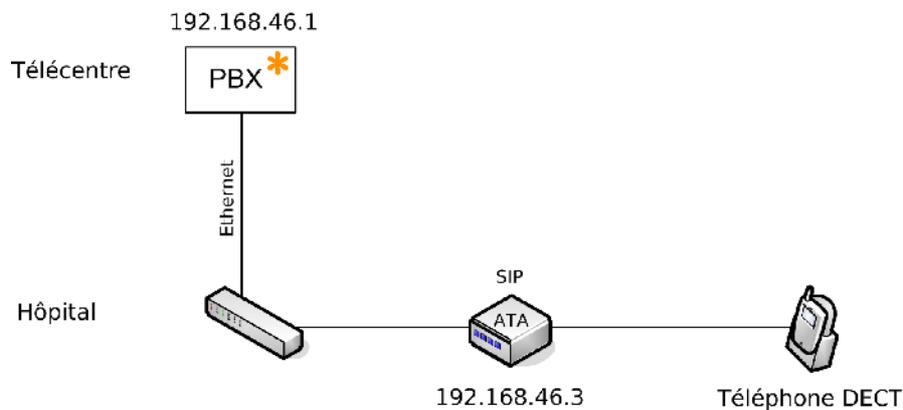


Image 6: L'hôpital est connecté au télécentre par un câble Ethernet CAT5 de 100 m. Un téléphone DECT est connecté à l'ATA (basé sur SIP) qui s'enregistre auprès du PBX du télécentre.

³⁴ Digital Enhanced Cordless Telecommunications (DECT) est un standard de téléphones sans fil numériques opérant dans la bande des 1.9 GHz

6.2.3 École primaire

Le troisième client de notre réseau, l'école primaire, est un bâtiment adjacent au télécentre, aussi pouvons-nous de nouveau utiliser un câble Ethernet.

Un autre ATA, celui-ci basé sur IAX2, est placé dans l'école primaire. Nous utilisons un simple ATA (interface FXS) appelé s10li ou IAXy. À cet ATA nous relierons un téléphone analogique quelconque.

L'IAXy ne propose pas d'interface web. Le plus simple pour configurer l'IAXy est d'utiliser *Asterisk*. Lorsque vous connecterez l'ATA pour la première fois, il se verra attribuer une adresse IP par DHCP, consultez les « logs » de votre serveur DHCP et notez l'adresse IP attribuée. L'étape suivante consiste à éditer le fichier `/etc/asterisk/iaxprov.conf` et y ajouter une section comme suit :

```
[iaxy_school]
ip: 192.168.46.4
netmask: 255.255.255.0
gateway: 192.168.46.1
codec: ulaw
server: 192.168.46.1.2
user: 464
pass: 464pass
register
```

Supposons que notre serveur DHCP fournisse l'adresse IP 192.168.46.100 à l'IAXy. Exécutez alors la commande suivante depuis la console *Asterisk* :

```
#asterisk -r <ENTER>
#CLI> iax2 provision 192.164.46.100 iaxy_school
```

Une alternative à *Asterisk* pour configurer l'IAXy est disponible pour les utilisateurs de Windows³⁵.

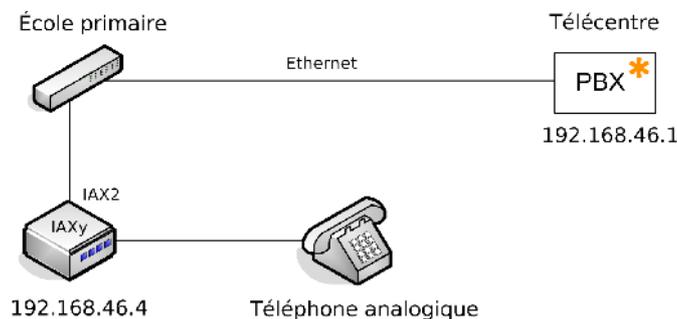


Image 7: L'École primaire est connectée au télécentre par un câble Ethernet. Un ATA (basé sur IAX2) permet de connecter un téléphone analogique au système.

³⁵ Le client Windows pour configurer l'IAXy est disponible sur le site <http://dacosta.dynip.com/asterisk/>.

6.2.4 Collective agricole

Le quatrième client du réseau est la collective agricole, qui est située à 20 km du télécentre (PBX). Ils possèdent deux ordinateurs déjà reliés au télécentre par un réseau sans fil NAT. La passerelle de ce dernier a l'adresse IP 192.168.46.5, et dessert aussi un réseau interne (10.10.46.0/24).

La configuration du softphone basé sur SIP constitue le plus grand défi. Afin de s'assurer que la voix soit transmise correctement dans les deux directions, nous devons procéder comme suit :

Dans le softphone :

- activer l'enregistrement (« registration »)
- activer les paquets « keep-alive »³⁶
- activer la possibilité de recevoir l'audio sur le même port que celui utilisé pour la transmission

Sur le PBX :

- informer *Asterisk* que le softphone se trouve derrière un NAT

Le softphone SIP X-Lite, par X-Ten, fonctionne convenablement derrière un NAT³⁷.

Les softphones basés sur IAX2 ne posent pas de problème majeur derrière un NAT. Assurez-vous que le port utilisé par IAX2, 4569, ne soit pas bloqué. Nous recommandons le softphone *iaxcomm*³⁸, basé sur IAX2.

Conceptuellement, la configuration d'un softphone ne diffère pas de celle d'autres terminaux de VoIP. Utilisez le nom d'utilisateur / mot de passe 465 / 465pass et 466 / 466pass respectivement. Assurez-vous que le codec G.711 (u-law) est activé et que l'adresse IP de votre PBX est correctement configurée (192.168.46.1).

³⁶ Les paquets « keep-alive » (garder en vie) sont des paquets « vides » qui permettent de garder le NAT ouvert à la réception d'appels.

³⁷ Une version gratuite peut être téléchargée sur le site <http://www.xten.com/index.php?menu=download>.

³⁸ *iaxComm* peut être téléchargé sur <http://iaxclient.sourceforge.net/>.

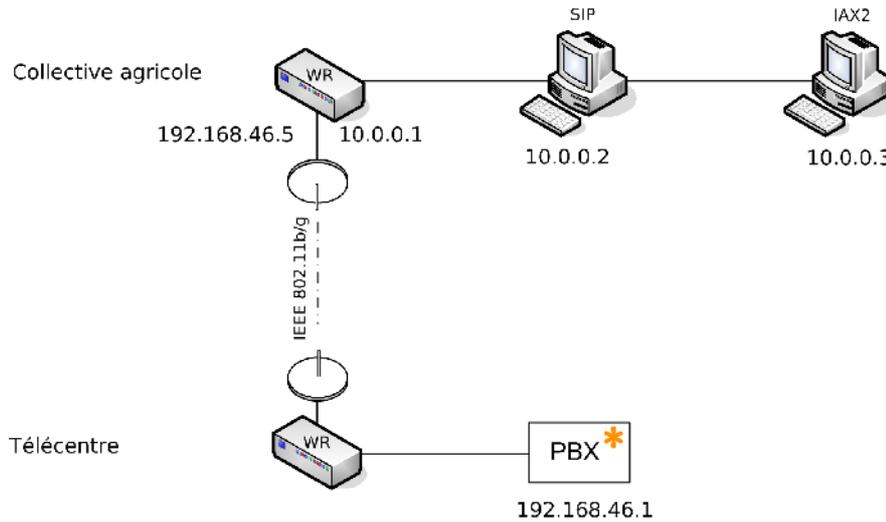


Image 8: La collective agricole est connectée au télécentre par un réseau NAT sans fil. Un softphone est installé sur chacun des 2 ordinateurs, l'un utilise SIP l'autre IAX2.

6.3 Configurer Asterisk

Première étape : définir et configurer les canaux de communication

Dans ce premier scénario, nous utilisons deux types de canaux de communication : SIP et IAX2. En conséquence il nous faut éditer deux fichiers, *sip.conf* et *iax2.conf*.

Notez que les commentaires dans les fichiers de configuration d'*Asterisk* sont précédés d'un point-virgule.

Dans le fichier *sip.conf*, vous devrez ajouter les lignes suivantes :

```
[462]
type=friend                ; Nous pouvons émettre et recevoir des appels
secret=462pass             ; Tous les « appels entrants » sont associés
context=internal_calls    ; au contexte internal_calls

host=192.168.46.2
callerid=Library
disallow=all               ; Nous commençons par interdire tous les codecs
allow=ulaw                 ; Puis nous listons ceux que nous acceptons

[463]
type=friend
secret=463pass
context=internal_calls
host=192.168.46.3
callerid=Hospital
disallow=all
allow=ulaw

[465]
type=friend
```

Un guide pour construire un infrastructure de télécommunication en région émergente - Réseau téléphonique privé dans une communauté rurale

```
secret=465pass
context=internal_calls
host=dynamic ; Nous ne connaissons pas l'adresse IP à l'avance
; Nous nous attendons à la recevoir par le processus register
callerid=Farmers1
disallow=all
allow=ulaw
; suivent les options spécifiques au NAT :
nat=yes ; les données sont envoyés à l'IP/port du NAT
qualify=yes ; Nous envoyons du pseudo-traffic pour garder le NAT ouvert
```

Dans le fichier *iax.conf*, vous devrez ajouter :

```
[464]
type=friend
secret=464pass
context=internal_calls
host=192.168.46.4
callerid=School
disallow=all
allow=ulaw

[466]
type=friend
secret=466pass
context=internal_calls
host=dynamic ; Afin d'obtenir l'adresse IP du client IAX2
callerid=Farmers2
disallow=all
allow=ulaw
```

Seconde étape : définir les règles de vos extensions (Créer votre plan de numérotation)

Dans ce premier scénario, nous avons placé tous les utilisateurs dans un même contexte (*internal_calls*). Aussi seul un « context » doit être défini dans le fichier *extensions.conf* :

```
[internal_calls]
exten => 462,1,Dial(SIP/462)
exten => 463,1,Dial(SIP/463)
exten => 465,1,Dial(SIP/465)
exten => 464,1,Dial(IAX2/464)
exten => 466,1,Dial(IAX2/466)
exten => t,1,Hangup() ; extension spéciale (Timeout)
exten => i,1,Hangup() ; extension spéciale (Invalid)
exten => s,1,Hangup() ; extension spéciale (No routing information)
```

La syntaxe du fichier *extensions.conf* est intuitive.

- Les crochets [nom_du_contexte] indique le début d'un « context » et le nom de celui-ci, tel qu'il est utilisé dans les fichiers *sip.conf* et *iax2.conf*
- Chaque ligne dans un « context » est une extension au format suivant :
exten => numéro,priorité,application-action

Dans ce fichier de configuration simple, nous déclarons que pour joindre chaque numéro d'extension 462-466, il faut exécuter la commande *Dial* et créer un *canal* SIP ou IAX2 vers le pair du même nom.

7. SCÉNARIO B

- Joindre le PSTN

Notre second étude de cas étend notre réseau de VoIP interne initial. Dans ce second scénario, nous souhaitons permettre à chacun des clients de VoIP de joindre le PSTN. À cette fin, nous aurons besoin d'un équipement spécial qui relie le PBX au PSTN.

Dans cet exemple, nous utilisons la carte PCI TDM400P de *Digium*, munie d'un port FXO. Souvenez-vous que la carte TDM400P est une carte versatile vous permettant de connecter un maximum de quatre modules FXO ou FXS, et qu'un module FXO permet au PBX de se connecter à *une* ligne de téléphone analogique.

7.1 Ajouter le support de la carte TDM400P

L'installation de la carte TDM400P requiert quatre étapes :

Première étape : Insérer la carte PCI

La première étape consiste à connecter la carte PCI dans un emplacement libre et brancher le connecteur molex (12/5 volts) à l'alimentation électrique de l'ordinateur. La carte utilise un supplément de puissance électrique qu'elle tire d'un connecteur molex femelle (il s'agit du même connecteur à 4 pins utilisés par les disques durs IDE standards). Si aucun connecteur molex mâle n'est disponible, il vous faudra vous procurer un connecteur Y.

Seconde étape : installer les drivers du matériel

En second lieu il vous faudra vous assurer que les drivers pour ce matériel ont été correctement compilés et chargés. En exécutant `#lsmod`, vous devriez contrôler que le module *wctdm* est chargé. Vous constaterez que le module *wctdm* dépend du module *zaptel*, lui-même dépendant du module *crc_ccitt*.

```
# lsmod
zaptel          191748  7 wctdm
crc_ccitt       2304    3 hisax,zaptel,irda
```

Troisième étape : configurer la carte TDM400P avec `ztcfg`

Il vous faut à présent configurer le matériel. Le module *wctdm* est un logiciel générique commun à toutes les versions de la carte TDM400P (gardez à l'esprit que la carte PCI peut accueillir jusqu'à quatre ports FXS/FXO). Nous commençons par indiquer que nous avons placé un module FXO dans le premier port de la carte PCI.

Le plus simple fichier de configuration */etc/zaptel.conf* est dans ce cas :

```
fxs1s=1
loadzone=us
defaultzone=us
```

La première ligne (*fxs1s=1*) signifie que nous utilisons une signalisation FXS *loopstart* dans le port 1 de la TDM400P. Souvenez-vous que le module FXO est signalé par FXS.

Les second et troisième paramètres du fichier de configuration *zaptel* indique le type de tonalité utilisé dans la région³⁹.

L'utilitaire de configuration *zaptel* */sbin/ztcfg*, qui fait partie de la distribution du code source d'*Asterisk* (ou du paquet *zaptel*), lit ce fichier de configuration.

Typiquement vous obtiendrez les messages suivants :

```
# ztcfg -vv
Zaptel Configuration
=====

Channel map:
Channel 01: FXS Loopstart (Default) (Slaves: 01)
1 channels configured.
```

Quatrième étape : Configurer Asterisk afin d'utiliser le matériel Zapata

La quatrième et dernière étape consiste à configurer *Asterisk* pour utiliser le matériel. Ceci se fait dans le fichier */etc/asterisk/zapata.conf*:

```
[channels]
usecallerid=yes
hidecallerid=no
callwaiting=no
threewaycalling=yes
transfer=yes
echocancel=yes
echotraining=yes

context=incoming_pstn
signalling=fxs_ls
channel => 1
```

Les trois dernières ligne du fichier de configuration sont les plus importantes dans une configuration standard. La ligne *context=incoming_pstn* indique que tous les appels entrant sont associés à ce contexte. Les deux lignes suivantes décrivent la signalisation (**fxs_ls**) pour ce canal zapata (**channel => 1**).

Une fois ce nouveau type de canal configuré (canal zapata), nous sommes proches de pouvoir recevoir et passer des appels vers et depuis le PSTN.

³⁹ Une liste des types de tonalité est disponible sur : <http://www.itu.int/ITU-T/inr/forms/files/tones0203.pdf>

7.2 Gérer les appels en provenance du PSTN

Les appels provenant du PSTN sont gérés de la façon suivante : lorsqu'un appel arrive sur la ligne analogique, nous souhaiterions qu'un serveur vocal (Interactive Voice Response, IVR) demande à l'appelant quelle extension il souhaite joindre. Puisque nous avons plusieurs utilisateurs dans notre réseau téléphonique interne, nous aimerions les rendre joignables à quiconque appelant sur la ligne analogique du PBX. La logique est implémentée dans le fichier *extensions.conf* en y ajoutant la section `[incoming_pstn]` comme ceci :

```
[incoming_pstn]
exten => s,1,Answer() ; Nous décrochons
exten => s,2,DigitTimeout(10) ; Mettre en place des timers
exten => s,3,ResponseTimeout(20)
exten => s,4,Background(vm-extension) ; Voix demandant l'extension
exten => i,1,Goto(incoming_pstn,s,1) ; Demander de nouveau si invalide
exten => t,1,Hangup() ; Raccroché après l'expiration du temps
include => internal_calls ; Rendre internal_calls disponible
```

Remarque : la version complète et finale du fichier *extensions.conf* est fournie en section 7.3 de ce document.

7.2.1 Rendre le PSTN accessible depuis le plan de numérotation

Afin de permettre à tous nos utilisateurs de VoIP de joindre le PSTN, nous devons tout d'abord ajouter un contexte à notre fichier *extensions.conf* :

```
[outgoing_calls]
exten => _0.,1,Dial(Zap/1/${EXTEN:1})
exten => t,1,Hangup()
```

Ceci signifie que pour joindre la ligne du PSTN, il vous faut inclure le préfixe « 0 ». La commande « *Dial()* » connectera l'appel au canal Zap 1. L'expression `${EXTEN:1}` indique que le premier chiffre (dans ce cas 0) sera omis lors de la numérotation vers l'extérieur.

Ajouter un contexte au plan de numérotation ne suffit pas. Il nous faut aussi le rendre disponible à nos utilisateurs de VoIP. Le plus simple pour ce faire est d'ajouter la ligne suivante à la fin de la section `[internal_calls]` :

```
include => outgoing_calls
```



Image 9. Le télécentre utilise une carte TDM400P wildcard pour se connecter le PBX au PSTN (module FXO) et pour relier un téléphone analogique au PBX (module FXS).

7.3 Connecter un téléphone analogique au PBX

Dans le premier scénario, nous avons configuré cinq clients de VoIP situés à quatre adresses différentes, mais le télécentre est toujours privé de téléphone. Un moyen simple pour y remédier est d'ajouter un second module (FXS) à la carte TDM400P existante. L'ajout d'un module FXS dans le second port de la carte TDM permet de connecter un téléphone analogique au PBX.

La procédure est simple : une fois le PBX éteint, nous connectons un module FXS dans le second port de la carte TDM. Après avoir rallumé le système, nous rajoutons une ligne au fichier de configuration `/etc/zaptel.conf`.

```
fxs1s=1
fxols=2
loadzone=us
defaultzone=us
```

Afin de s'assurer que le nouveau module est reconnu et configuré, nous exécutons l'outil de configuration `zaptel`, et obtenons le résultat suivant :

```
#ztcfg -vv
Zaptel Configuration
=====

Channel map:

Channel 01: FXS Loopstart (Default) (Slaves: 01)
Channel 02: FXO Loopstart (Default) (Slaves: 02)

2 channels configured.
```

Il nous faut aussi ajouter une nouvelle section au fichier `/etc/asterisk/zapata.conf`, dans laquelle nous associons les appels provenant du téléphone analogique (port 2 de la carte TDM) au contexte `[internal_calls]` :

```
[channels]
usecallerid=yes
hidecallerid=no
callwaiting=no
threewaycalling=yes
transfer=yes
echocancel=yes
echotraining=yes

context=incoming_pstn
signalling=fxs_ls
channel => 1

context=internal_calls
signalling=fxo_ls
channel => 2
```

7.4 Mettre à jour le plan de numérotation

Un nouveau plan de numérotation est requis pour :

1. Permettre les appels entrant et sortant via le canal zapata 1 (PSTN)
2. Inter-connecter le téléphone analogique relié au canal zapata 2, afin qu'il puisse passer et recevoir des appels.

Le fichier de configuration *extensions.conf* pour ce second scénario contient au final :

```
[incoming_pstn]
exten => s,1,Answer()
exten => s,2,DigitTimeout(10)
exten => s,3,ResponseTimeout(20)
exten => s,4,Background(vm-extension)
exten => i,1,Goto(incoming_pstn,s,1)
exten => t,1,Hangup()
include => internal_calls

[internal_calls]
exten => 461,1,Dial(Zap/2) ; Extension 461 apples via canal Zap 2
exten => 462,1,Dial(SIP/462)
exten => 463,1,Dial(SIP/463)
exten => 465,1,Dial(SIP/465)
exten => 464,1,Dial(IAX2/464)
exten => 466,1,Dial(IAX2/466)
exten => t,1,Hangup()
exten => s,1,Hangup()
exten => i,1,Hangup()
include => outgoing_calls ; PSTN disponible pour les clients VoIP

[outgoing_calls]
exten => _0.,1,Dial(Zap/1/${EXTEN:1}) ; Retirer le 0 avant de composer le
numéro externe
exten => t,1,Hangup()
```

8. SCÉNARIO C :

- Interconnecter des communautés par la VoIP

Dans notre troisième scénario, nous aimerions relier notre télécentre à un centre de formation situé dans un autre pays, en utilisant un lien Internet par satellite VSAT. Une fois la connexion établie, nous pouvons l'utiliser pour établir des conférences téléphoniques internationales, non seulement vers le pays du centre de formation mais aussi vers d'autres destinations.

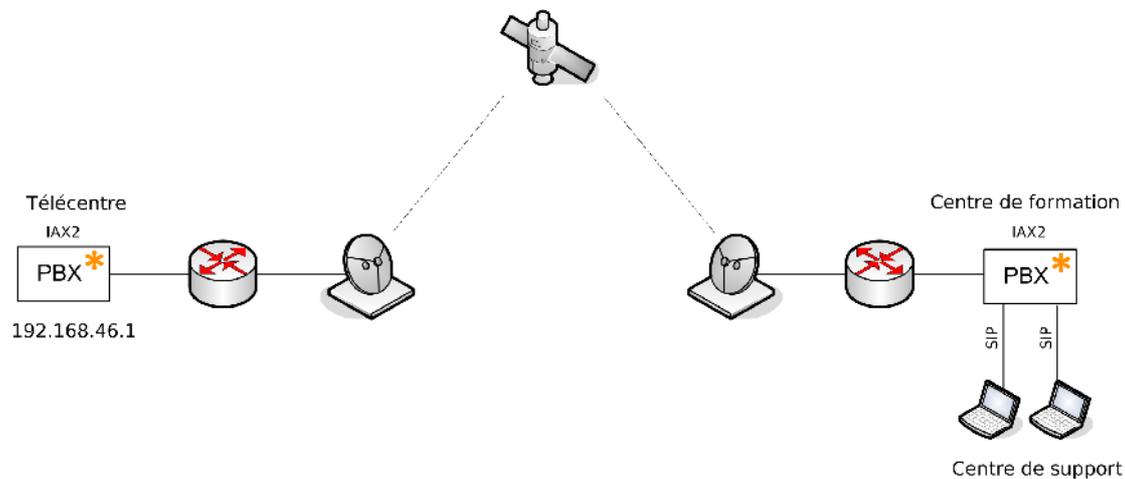


Image 10: Le télécentre et le centre de formation sont tous deux équipés d'un PBX. Les PBX sont connectés par un lien satellite.

8.1 Problèmes inhérents aux connexions par satellite

La connexion Internet du télécentre a une bande passante très limitée (128 / 64 Kbps), aussi notre solution doit-elle optimiser le débit. Dans l'exemple suivant, nous proposons de relier 2 PBX utilisant Asterisk par IAX2. Afin d'optimiser la consommation de bande passante, nous suggérons d'utiliser un codec à haute compression tel que G.729 ou Speex. Enfin, nous activerons le « trunking » afin que plusieurs appels simultanés puissent être regroupés afin de réduire le surplus de bande passante dû aux entêtes.

Très souvent avec une connexion Internet VSAT, le routeur de bordure est connecté à Internet avec un nombre réduit d'adresses IP publiques. Si aucune adresse publique n'est disponible pour votre PBX, assurez-vous que le port 4569 est correctement redirigé vers l'adresse interne de votre PBX. La configuration d'une telle redirection dépend du type d'équipement utilisé comme routeur de bordure.

Si vous avez un routeur CISCO configuré comme un NAT vous devrez exécuter une commande telle que

```
#ip nat inside source static udp 192.168.46.1 4569 interface fastEthernet 0/0
4569
```

Si votre routeur utilise Linux, il vous faut rediriger toutes les connections au port 4569 vers l'ordinateur interne (192.168.46.1) en utilisant iptables

```
#!/sbin/iptables -t nat -A PREROUTING -p udp --dport 4569 -i eth0 -j DNAT --to-
destination 192.168.46.1:4569
```

L'important est que le PBX soit joignable depuis Internet par UDP sur le port 4569⁴⁰.

8.2 Inter-connecter 2 serveurs Asterisk

8.2.1 Télécentre

Le fichier de configuration *iax.conf* au télécentre se présente comme suit :

```
[general]
bindaddr = 0.0.0.0
tos = lowdelay
disallow = all
allow = ulaw
allow = g729 ; Nous ajoutons le codec G.729

register => server2:server2pass@training_voip.org
; server2:server2pass est l'utilisateur
; et mot de passe utilisé par register

; Le compte utilise par l'autre serveur pour nous joindre
[server1]
type=friend
user=server1
secret=server1pass
host=dynamic ; Nous obtenons l'adresse IP lorsque l'autre PBX s'enregistre
context=incoming_training_centre_calls
auth=md5 ; Securiser l'authentification
disallow=all
allow=g729
trunk=yes ; Nous activons le trunking
```

Afin de joindre le centre de formation par Internet, nous ajoutons le contexte suivant au fichier *extensions.conf*. Lorsque le préfixe 99 est utilisé, l'appel est redirigé par Internet vers server1, qui se trouve dans le pays du centre de formation.

⁴⁰ Le port 4569 est le port de communication par défaut utilisé par IAX2.

```
[outgoing_training_centre_calls]
exten => _99.,1,Dial(IAX2/server2:server2pass@server1/${EXTEN:2})
exten => _99.,2,Congestion ; En cas d'echec une tonalite de congestion est
utilisee
```

À présent il nous faut décider ce qu'il advient des appels émis depuis le centre de formation.

```
[incoming_training_centre_calls]
exten => _X.,1,Dial(Zap/2) ; Appels provenant du centre de formation
; diriges vers le telephone du telecentre
```

8.2.2 Centre de formation

Le fichier de configuration *iax.conf* au centre de formation se présente comme suit :

```
[general]
bindaddr = 0.0.0.0
tos = lowdelay
disallow = all
allow = ulaw
allow = g729 ; Nous ajoutons le codec G.729
; Si nous n'avons pas de licences nous utilisons speex

register => server1:server1pass@rural.telecentres.org
; server1 est l'utilisateur pour register

[server2]
type=friend
user=server2
secret=server2pass
host=dynamic ; Obtenir l'adresse quand le serveur s'enregistre

context=incoming_telecentres_calls
auth=md5 ; securiser l'authentification
disallow=all
allow=g729
trunk=yes
```

Nous ajoutons ensuite le contexte suivant au fichier *extensions.conf*, afin que lorsque le préfixe 88 est utilisé, l'appel est redirigé vers le Télécentre (*peer [server2]*) dans le fichier *iax.conf*.

```
[outgoing_telecentres_calls]
exten => _88.,1,Dial(IAX2/server1:server1pass@server2/${EXTEN:2})
exten => _88.,2, Congestion
```

Au centre de formation, nous décidons que tous les appels en provenance du télécentre sont redirigés vers le centre de support.

```
[incoming_telecentres_calls]
exten => _X.,1,Dial(SIP/support-desk) ; Appels en provenance du telecentre.
; Redirige vers le centre de support41
```

41 La connexion SIP utilisée par la commande Dial() vers le terminal du centre de support doit être configurée dans le fichier sip.conf du centre de formation.

8.2.3 La commande « register »

Lorsque l'adresse IP du pair est inconnue, nous ne pouvons pas passer d'appel. Imaginez par exemple que l'un des PBX se voit attribué une adresse statique, mais pas le second. Pour remédier à cette situation, le PBX dont l'adresse est dynamique doit s'enregistrer auprès de l'autre au moyen de la commande « register ». Dans l'exemple précédent, nous utilisons 2 commandes « register », mais au sens strict ceci n'est nécessaire que lorsque l'un des PBX a une adresse dynamique. Lorsque l'adresse IP du pair est connue, nous pouvons remplacer `host=dynamic` par `host=<adresse_IP ou domaine>`.

9. Aller plus loin

- L'un des livres sur Asterisk se nomme : Asterisk, the Future of Telephony, Jim van Meggelen, Jared Smith, Leif Madsen, O'Reilly 2005. Disponible sous licence « Creative Commons ». <http://www.oreilly.com/catalog/asterisk/>
Téléchargement gratuit :
<http://www.asteriskdocs.org/modules/tinycontent/index.php?id=11>
- Pour garder un oeil sur l'actualité de la téléphonie émergente :
<http://www.oreillynet.com/etel/>
- Le site VoIP info est un énorme wiki regorgeant de conseils et d'astuces. Trouver ce dont vous avez besoin peut cependant s'avérer difficile.
<http://www.voip-info.org/wiki-Asterisk+tips+and+tricks>

10. Conclusion

Ce précis tente de présenter certains concepts essentiels à la téléphonie IP. En s'appuyant sur quelques exemples concrets, nous espérons vous avoir fait prendre conscience de l'immense potentiel que la VoIP apporte aux régions émergentes. La VoIP conjuguée aux technologies de réseaux sans fil à coût réduit permet d'apporter la voix et les données dans des régions privées de l'évolution numérique, tout en promouvant la création de réseaux téléphoniques opérés et administrés par la communauté.

En vous inspirant des exemples de fichier de configuration inclus, vous devriez pouvoir mettre en place votre premier système. Aucun document ne remplace l'expérience, aussi soyez patient. Votre persévérance est la clef du succès. Vous n'êtes pas seul, vous pouvez toujours compter sur l'aide fournie par de nombreux forums sur Internet, et partager votre expérience avec la communauté. Bienvenue dans un monde trépidant d'enthousiastes de la VoIP !

Nous aimerions avoir de vos nouvelles, contactez nous par VoIP !

11. Remerciements

Nous adressons nos remerciements à tous ceux qui ont rendu possible cet ouvrage :

Adel El Zaim (Édition en Arabe et Français), Anas Tawileh (Traduction en Arabe), Sylvia Cadena et Iñaki Cíviki (Édition en Espagnol), Johan Bilien (Traduction en Français) et Martin Benjamin (Édition en Anglais).



Tout le contenu de cette publication est distribué sous licence
Creative Commons Attribution NonCommercial-ShareAlike 2.5
<http://creativecommons.org/licenses/by-nc-sa/2.5/deed.fr>

Cet ouvrage a été conçu avec l'aide d'une subvention de l'Initiative
Acacia du Centre de Recherches pour le Développement International
(CRDI) du Canada.



Le Centre de Recherches pour le Développement International (CRDI) du Canada est l'une des institutions à la pointe de la création et l'application de nouvelles connaissances aux défis du développement international. Depuis plus de 30 ans, le CRDI a travaillé en collaboration rapprochée avec les chercheurs des régions émergentes afin d'élaborer les moyens nécessaires à la création des sociétés plus saines, équitables et prospères.

IDRC
International Development
Research Centre



CRDI
Centre de recherches pour le
développement international

it+46