

Philippe Sultan

**Cahiers
de
l'Admin**

Asterisk

La téléphonie d'entreprise libre

Collection dirigée par **Nat Makarévitch**

Préface de **Mark Spencer**
(Digium, Inc.)

© Groupe Gyrolles, 2010, ISBN : 978-2-212-12434-7

EYROLLES



Préface

Nombreux sont ceux qui attribuent mon invention d'Asterisk à une clairvoyance extraordinaire quant à l'importance qu'il prendrait dans l'univers des télécommunications. En réalité, j'ai commencé à développer Asterisk parce que j'avais besoin d'un système de téléphonie dont l'acquisition aurait été trop coûteuse.

Dès le début cependant, j'ai eu le sentiment que si je parvenais à recevoir un appel téléphonique sur mon PC, les possibilités seraient ensuite infinies... C'est cette idée simple et puissante qui a mené Asterisk au point où il en est aujourd'hui.

Qui aurait pu prévoir, en 1999, que ce petit morceau de logiciel que j'avais commencé à développer dans le fond d'un hangar, se diffuserait non seulement dans tout mon pays natal, les États-Unis, mais aussi tout autour du globe, avec la constitution d'une communauté mondiale d'utilisateurs et de développeurs – tout en donnant naissance à des centaines de petites entreprises ?

Quand j'ai mis au point Asterisk pour la première fois, je le considérais comme un PABX, c'est-à-dire un système de téléphonie. Je croyais alors ne développer qu'une application indépendante... Aujourd'hui, Asterisk est bien plus que cela ; il est devenu le moteur sur lequel s'adosent nombre d'autres développeurs pour construire d'imposants systèmes de télécommunications.

Ce livre sera très utile à de nombreux égards. Premièrement, il fournit les prérequis fondamentaux en matière de télécommunications pour comprendre ce que fait Asterisk. Deuxièmement, il détaille comment l'installer pour fournir une large palette de services via toute une variété d'interfaces, traditionnelles aussi bien que de voix sur IP – et il explique bien sûr comment sécuriser ces services contre les usages non autorisés. Troisièmement, il donne les clés pour diagnostiquer les problèmes lorsque tout ne fonctionne pas comme prévu (un

indice en passant : lorsque l'utilisateur déclare « ne rien avoir changé », ne le croyez pas !). Enfin, vous verrez comment tirer parti des interfaces fournies par Asterisk pour mettre au point des produits encore meilleurs.

S'il est vrai que c'est moi qui ai initié ce projet, et que j'ai écrit beaucoup, beaucoup de lignes de code pour lui, Asterisk tel qu'il existe aujourd'hui est en vérité le fruit de l'effort de centaines de développeurs à travers le monde, dont l'impact est décuplé par les centaines d'autres entreprises et développeurs qui l'utilisent pour construire de nouvelles applications. J'espère donc qu'au-delà de vous guider dans l'utilisation d'Asterisk, ce livre vous encouragera à participer à la communauté qui l'a rendu si fort.

Mark Spencer,
Digium, Inc.,
le 24 août 2009

Avant-propos

Le monde de la téléphonie a considérablement évolué depuis une dizaine d'années. La principale raison de cette évolution est technique et tient à l'émergence de la téléphonie sur IP (ToIP). En effet, de la même façon qu'il s'est imposé comme protocole de base des réseaux de télécommunications, IP (*Internet Protocol*) se situe désormais aussi au centre des services et architectures téléphoniques actuels.

Parallèlement, profitant notamment de l'interface IP vers la téléphonie comme canal de développement d'applications, le monde des logiciels libres de télécommunications connaît depuis la même période un essor important. Plusieurs logiciels libres ont constitué autour d'eux des communautés très actives.

Parmi eux, le plus célèbre et le plus riche est sans aucun doute Asterisk. Le succès d'Asterisk tient à mon sens en trois éléments fondamentaux :

- ses nombreuses implémentations de protocoles de télécommunications, tant standards que propriétaires (RNIS, SIP, H.323, SCCP, etc.) ;
- les services téléphoniques intégrés (IPBX, messagerie vocale, audioconférences) ;
- ses interfaces vers les applications informatiques, essentielles ou non (web, e-mail, messagerie instantanée, etc.).

Ces éléments font d'Asterisk l'outil idéal, non seulement pour gérer un ensemble de postes téléphoniques de taille quelconque (chez un particulier ou dans une entreprise de taille moyenne), mais aussi pour développer des

services avancés comme l'appel automatique en cliquant sur un lien d'une page web (*click-to-call*), la réception de messages téléphoniques dans une boîte e-mail, etc.

Aujourd'hui, pour une entreprise, les applications et services téléphoniques que l'on peut développer grâce à Asterisk surpassent ceux qui sont proposés dans le cadre de solutions téléphoniques propriétaires. La comparaison ne se mesure pas ici en considérant les richesses fonctionnelles respectives, qui sont équivalentes, mais se fonde sur le coût et la faculté d'intégration dans un environnement informatique existant.

En effet, outre le fait d'être infiniment plus chères qu'un logiciel libre, les solutions propriétaires n'offrent bien souvent pas la souplesse du libre en termes d'intégration dans le système d'information. En d'autres mots, elles ne s'intègrent qu'à des applications d'une marque donnée, alors que les logiciels libres comptent sur le développement d'un protocole de communication commun, le plus souvent normalisé.

Pourquoi ce livre ?

Asterisk, et d'autres logiciels libres de téléphonie comme Kamailio (autrefois appelé OpenSER) permettent aux ingénieurs réseau et aux développeurs de s'approprier la téléphonie et de construire des services et applications innovantes et utiles. De par l'intérêt que je porte à la ToIP et aux logiciels libres, j'ai souhaité faire partager mon expérience dans ces domaines au sein de cet ouvrage.

Ce livre s'adresse donc à tous ceux qui s'intéressent à la ToIP et à Asterisk, qu'ils soient amateurs ou ingénieurs expérimentés.

Il vient compléter par des exemples concrets d'installation, de déploiement d'applications, de sections détaillées de configuration ou de code, les nombreuses sources de documentation existantes sur Asterisk. Archives de listes de diffusion, de forums, blogs et sites spécialisés constituent en effet une mine d'informations brutes dont le volume important peut paradoxalement les rendre peu accessibles, ou dont le contexte de présentation peut être inadapté – voire, tout simplement obsolète.

Que contient ce livre ?

Ce livre est construit autour de l'étude de cas d'une entreprise amenée à renouveler son système de téléphonie. Nous n'avons volontairement pas voulu faire un inventaire exhaustif des possibilités d'Asterisk ou un état de l'art de la ToIP, afin de nous concentrer sur une problématique concrète qui amène le lecteur à explorer progressivement le domaine de la ToIP et la façon dont Asterisk répond à des besoins exprimés, tout en lui présentant les perspectives qu'il ouvre.

L'ouvrage traite notamment :

- de la présentation des fonctionnalités de base et possibilités étendues d'Asterisk ;
- du développement d'applications à partir d'Asterisk ;
- de l'intégration d'Asterisk dans le système d'information (messagerie électronique, annuaire...) ;
- de l'intégration avec d'autres logiciels libres de télécommunications comme Kamailio (anciennement OpenSER) ;
- des protocoles de ToIP fondamentaux et émergents (RNIS, SIP, XMPP).

Le **chapitre 1** contient une présentation générale d'Asterisk et de ses possibilités d'utilisation, ainsi que de l'écosystème des logiciels libres de télécommunication.

Dans les **chapitres 2 et 3**, nous présentons l'étude de cas qui sert de trame au livre. Les raisons qui motivent le choix de la ToIP libre et d'Asterisk pour une entreprise seront traitées ici. Nous nous attardons en outre sur l'environnement informatique (réseau, applications, annuaire) et téléphonique de l'entreprise objet de l'étude de cas.

Le **chapitre 4** nous amène à décrire en détail l'installation d'un serveur Asterisk relié à un opérateur téléphonique traditionnel. Ce chapitre est le premier sentier de l'exploration technique menée tout au long du livre ; il traite en outre des outils compagnons fondamentaux que sont DAHDI et libpri, qui constituent l'interface d'Asterisk vers la téléphonie traditionnelle.

Trois points majeurs sont abordés dans le **chapitre 5** : l'architecture du réseau sur lequel s'appuiera le nouveau système de téléphonie, d'abord, véritable socle du service de téléphonie ; le protocole SIP, ensuite, et plus particulièrement ses interfaces dans Asterisk et dans les postes téléphoniques déployés ; enfin, le plan de numérotation d'Asterisk, aussi appelé *dialplan*, qui constitue le cœur d'Asterisk et de tout PABX.

Dans le **chapitre 6**, nous présentons une nouvelle installation d'Asterisk, destinée à répondre au besoin de notre entreprise de délivrer un service téléphonique à un site géographiquement éloigné.

Nous poursuivrons notre visite d'Asterisk par une exploration avancée des services fondamentaux qu'il peut rendre : la messagerie vocale (et son intégration avec la messagerie électronique), le service vocal interactif (SVI), la supervision d'appels. Par ailleurs, ce **chapitre 7** est le premier à détailler l'intégration d'Asterisk dans un environnement web, intégration illustrée par l'exemple d'un service d'envoi de fax via le Web.

L'accès à distance aux ressources téléphoniques, par exemple depuis l'ordinateur de son domicile, est l'une des conséquences heureuses de l'émergence de la ToIP. Asterisk est parfaitement adapté pour l'assurer, comme nous le montrons au **chapitre 8**. Ce chapitre est l'occasion d'évoquer des obstacles majeurs au déploiement de la ToIP, comme le NAT et le filtrage de ports, et de détailler les solutions apportées par Asterisk. En particulier, nous verrons pourquoi le protocole IAX peut être préféré au couple SIP/RTP en fonction de l'environnement réseau. Par ailleurs, nous y présentons un exemple simple d'installation d'Asterisk depuis le domicile d'un particulier connecté au réseau SIP de l'opérateur Free.

Les **chapitres 9 et 10** nous amènent au cœur de l'intégration d'Asterisk avec le Web... et le système d'information. Nous mesurerons toute la puissance d'Asterisk dans ce domaine et verrons en détail la façon dont le développeur et l'ingénieur peuvent se servir d'Asterisk comme une véritable boîte à outils pour construire de nouvelles applications. Les audioconférences, l'appel automatique depuis un lien web, l'appel par composition d'adresse e-mail sont autant de services qui mettent en évidence les facultés d'intégration d'Asterisk dans le système d'information, réalisant ainsi les promesses jamais tenues par les constructeurs téléphoniques traditionnels autour du CTI (couplage téléphonie informatique). Les interfaces AGI et AMI, qui servent de support aux fonctionnalités de CTI présentées dans ces chapitres sont abordées en détail.

Dans les **chapitres 11 et 12**, nous présentons les outils nécessaires à l'administration d'un serveur Asterisk dans le cadre d'une exploitation régulière, ainsi que les outils permettant de diagnostiquer d'éventuels dysfonctionnements.

Asterisk n'est pas le seul logiciel libre de ToIP, et le **chapitre 13** offre une illustration de la complémentarité d'Asterisk et du logiciel libre Kamailio (autrefois plus connu sous le nom d'OpenSER). Nous y décrivons également les moyens offerts par Kamailio pour interconnecter de multiples IPBX SIP.

Quelques axes de travail de la communauté Asterisk sont exposés dans le **chapitre 14** : IPv6, la vidéo et la messagerie instantanée par XMPP (Jabber). Ce dernier point fournit l'occasion d'aborder les possibilités d'Asterisk dans le domaine de la messagerie instantanée et de la gestion de présence, véritables socles d'applications nouvelles de travail collaboratif.

Enfin, le **chapitre 15** traite des moyens de communiquer avec les utilisateurs d'Asterisk et les contributeurs au projet, qui forment désormais une communauté internationale très importante. Nous présentons par ailleurs les différentes formes de contributions qui peuvent être utiles au développement d'Asterisk.

Nombre d'apartés viennent compléter le propos principal de ce cahier, qu'il s'agisse de définitions rappelant les concepts réseau de base, de renseignements sur le logiciel, la communauté d'Asterisk ou la culture de l'informatique libre, ou de tout type d'informations susceptible d'être utile aux lecteurs de cet ouvrage, quel que soit leur niveau.

CULTURE Le Jargon Français

Pour compléter utilement la lecture de cet ouvrage et renforcer vos connaissances, n'hésitez pas à consulter le dictionnaire d'informatique francophone en ligne, « Le Jargon Français ». Référence d'un bon nombre des définitions « B.A.-Ba » de cet ouvrage, vous y retrouverez également les définitions originales proposées par l'auteur de ce livre.

► <http://jargonf.org/wiki/Jargonf:Accueil>

Remerciements

Je remercie tous ceux qui m'ont permis d'arriver au bout de cette aventure passionnante et exigeante qu'est l'écriture d'un livre :

Karine Joly, Muriel Shan Sei Fan et Nat Makarevitch, ainsi que toute l'équipe des éditions Eyrolles, Sophie Hincelin, Eric Bernauer et Gaël Thomas, pour leur accompagnement et leurs conseils avisés.

Mark Spencer, bien sûr, pour avoir créé ce merveilleux logiciel qu'est Asterisk et pour avoir accepté de préfacier cet ouvrage. Russell Bryant, Kevin Fleming et John Todd pour leur activité dans la communauté Asterisk et leur soutien. Mon ami Olle Johansson, qui me guide toujours avec bienveillance dans mes contributions à Asterisk. Elena Ramona Modroiu et Daniel Constantin Mierla, du projet Kamailio, Bogdan Andrei Iancu, du projet OpenSIPS.

Asterisk

Philippe Hensel, de l'IUT de Colmar. Jacques Heitzmann, du Forum Atena. Jean-David Benichou, Frédéric Clément et Laurent Pierre, de la société Via-telecom. Pierre Lombard et Cyrille Judas de la société Hubiquity.

La communauté Asterisk-France, si indispensable ! Les sociétés PerenIP, QosmiQ Networks et Iviflo.

Mes collègues Denis Joiret, Laurent Le Pendeven, Nicolas Chevènement, Éric Gallula, Amine Hassim, pour leur savoir et leur dynamisme.

Mes amis qui ont contribué par des échanges ou par un éclairage particulier à l'élaboration de ce livre, Abdelkader Allam, Michaël Benharrosh, Eric Danan, Olivier Krief, Romain Laclaverie, Aaron Partouche, Luc Podrzycki, Philippe Quesson, Daniel Smadja, Nicolas Tiphaine.

Ma mère, mon père, mes frères, qui me sont tous si chers.

Je remercie naturellement ma femme, Anne, pour son soutien et sa patience, et surtout pour son amour et les merveilleux enfants qu'elle m'a donnés.

À ma grand-mère.

Table des matières

| | |
|---|-----------|
| Avant-propos | 1 |
| Pourquoi ce livre ? | 2 |
| Que contient ce livre ? | 3 |
| Remerciements | 5 |
| | |
| CHAPITRE 1 | |
| Asterisk et la ToIP libre | 7 |
| Qu'est-ce qu'Asterisk ? | 7 |
| Les fonctions assurées par Asterisk | 8 |
| Serveur d'audioconférences | 9 |
| Messagerie vocale | 9 |
| Serveur vocal interactif (SVI) | 10 |
| Passerelle vers le RTC, interconnexion de sites | 11 |
| Une plate-forme applicative pour la téléphonie : Asterisk et le CTI | 12 |
| Asterisk et les autres logiciels libres de ToIP | 13 |
| Kamailio (ex OpenSER) : un proxy SIP, un vrai | 14 |
| Les concurrents d'Asterisk | 16 |
| <i>CallWeaver : un fork d'Asterisk</i> | 16 |
| <i>FreeSWITCH : la promesse de la fiabilité</i> | 17 |
| <i>Yate : encore un autre IPBX logiciel !</i> | 17 |
| | |
| CHAPITRE 2 | |
| Présentation de l'étude de cas | 21 |
| Situation géographique | 21 |
| Les raisons du changement de système téléphonique | 22 |
| Un existant vieillissant et coûteux | 22 |
| La nécessité de nouveaux services | 22 |
| Les raisons du choix de la ToIP libre | 23 |
| Réduire les coûts | 23 |

Asterisk

| | |
|---|----|
| S'approprier réellement le système | 24 |
| Maintenance et évolution du logiciel libre en téléphonie | 25 |
| Pourquoi Asterisk ? | 25 |
| Les services offerts aux collaborateurs de l'entreprise | 26 |
| Reconduire l'existant : fax, transfert et notification d'appel | 26 |
| Ajouter des services : la mobilité et l'intégration dans le système d'information | 27 |
| L'accompagnement de la croissance de l'entreprise | 28 |
| Interconnexion de sites distants par SIP | 28 |
| Travail collaboratif et gestion de présence | 28 |
| Vidéo | 28 |

CHAPITRE 3

Description de l'existant..... 29

| | |
|---|----|
| Les autocommutateurs (PABX) | 30 |
| Un annuaire pour le PABX et un autre pour l'entreprise | 31 |
| Des capacités de CTI prometteuses, mais jamais mises en œuvre | 31 |
| Les postes téléphoniques | 33 |
| Plan de numérotation | 33 |
| Postes numériques, analogiques et fax | 33 |
| Les services utiles : annuaire et traçabilité des appels | 33 |
| L'annuaire des collaborateurs de l'entreprise | 34 |
| Le serveur RADIUS | 34 |
| Un réseau informatique prêt pour la ToIP | 35 |
| Contraintes pour la ToIP | 36 |
| Sécurisation des flux par séparation des réseaux | 37 |

CHAPITRE 4

Installation d'Asterisk..... 39

| | |
|--|----|
| Choix du système d'exploitation | 40 |
| Un logiciel pour systèmes Unix | 40 |
| GNU/Linux, le choix naturel | 40 |
| <i>Quelle distribution GNU/Linux choisir ?</i> | 41 |
| <i>Prérequis (avant compilation)</i> | 42 |
| DAHDI et libpri, les compléments d'Asterisk | 43 |
| Installation de DAHDI | 43 |
| <i>Téléchargement, compilation et installation des sources</i> | 44 |
| <i>Création du script de démarrage et installation des fichiers de configuration</i> | 46 |
| <i>Lancement de DAHDI et détection de la carte</i> | 50 |
| Installation de libpri | 51 |
| Adressage réseau | 53 |

| | |
|---|-----------|
| Quelle distribution d'Asterisk ? | 54 |
| Installation d'Asterisk standard | 54 |
| Choix des modules | 56 |
| Fin de l'installation et lancement d'Asterisk | 57 |
| À chaque module son attribution | 58 |
| Quelles différences entre les fonctions et les applications dans Asterisk ? | 59 |
| Pourquoi un codec plutôt qu'un autre ? | 61 |
| | |
| CHAPITRE 5 | |
| Intégration des postes téléphoniques..... | 63 |
| Architecture réseau | 63 |
| Choix de l'architecture | 63 |
| Filtrage sur les réseaux local et distant | 64 |
| Configuration des postes téléphoniques IP | 65 |
| SIP : le protocole pour les postes physiques | 65 |
| Configuration SIP sur Asterisk : le fichier /etc/asterisk/sip.conf | 66 |
| Postes IP physiques | 68 |
| <i>Configuration sur le serveur Asterisk principal (voiceserver)</i> | 68 |
| <i>Éléments de configuration des postes Thomson ST2030</i> | 72 |
| Téléphones logiciels (softphones) | 74 |
| Le plan de numérotation, au cœur d'Asterisk | 74 |
| Contextes | 76 |
| Extension, action ! | 77 |
| Les variables dans le dialplan | 83 |
| Différents moyens pour passer des appels | 84 |
| <i>Placer des fichiers dans une file d'attente</i> | 84 |
| <i>AMI (Asterisk Manager Interface)</i> | 85 |
| <i>La console</i> | 85 |
| <i>AGI (Asterisk Gateway Interface)</i> | 86 |
| | |
| CHAPITRE 6 | |
| Asterisk en point de terminaison télécom..... | 87 |
| Interface vers le réseau de téléphonie classique | 88 |
| Installation et configuration d'Asterisk | 89 |
| DAHDI et libpri | 89 |
| Asterisk | 91 |
| <i>Configuration SIP sur les serveurs secondaire et principal</i> | 94 |
| <i>Configuration du plan de numérotation sur les serveurs secondaire et principal</i> | 95 |

CHAPITRE 7

Services téléphoniques : messagerie vocale, notification d'appel, fax... 101

- Asterisk en tant que SVI : les fichiers vocaux 101
 - Le choix de la langue 102
 - Les applications jouant des fichiers vocaux 103
 - Record : enregistrer ses propres fichiers vocaux 104
- La messagerie vocale 105
 - Activation de la boîte vocale par le Web 107
 - Notification de message en attente 108
 - MWI, la notification sur le poste* 108
 - Notification par envoi de courriel* 111
 - Renvoi vers la messagerie vocale 112
 - Consultation des messages 113
- Le nouveau service de fax 115
 - Installation des modules nécessaires dans Asterisk 115
 - Envoi de fax par le Web (serveur Apache 2.2.3, PHP 5.1.6) 117
 - Formulaire d'envoi* 118
 - Conversion du PDF en image TIFF* 120
 - Envoi du fax* 121
 - Configuration d'Asterisk pour l'envoi de fax 123
- La notification d'appel : détail du cas d'un chef de service et de son assistant .. 124
 - Configuration des postes téléphoniques 125
 - Configuration d'Asterisk 126

CHAPITRE 8

Accès à distance aux ressources téléphoniques 129

- Contraintes particulières à l'accès distant 130
 - SIP, IAX : quel protocole pour l'accès distant ? 131
 - La sécurité 132
 - Le code d'Asterisk* 132
 - VPN IPSec pour la confidentialité* 132
- Gestion des comptes des utilisateurs 134
 - Impossible d'utiliser la base des comptes existante 134
 - Réutiliser les comptes associés aux boîtes vocales 135
 - Contrôle d'accès par le dialplan 136
- Installation d'Asterisk chez un particulier 137
 - NAT (Network Address Translation) : l'obstacle majeur 137

| | |
|--|-----|
| Installer et configurer Asterisk | 144 |
| <i>Un minimum de fichiers de configuration</i> | 145 |
| Enregistrer Asterisk | 148 |
| Configurer un softphone | 152 |
| <i>Traitement des appels entrants</i> | 152 |
| <i>Traitement des appels émis par le softphone</i> | 154 |

CHAPITRE 9

Réunions téléphoniques (audioconférences)..... 155

| | |
|--|-----|
| Des réunions téléphoniques plus sûres | 155 |
| Mise en place du serveur d'audioconférences : éléments indispensables | 157 |
| dahdi_dummy : DAHDI en version allégée | 157 |
| La connexion vers le serveur Asterisk principal | 161 |
| MeetMe : les audioconférences par Asterisk | 163 |
| Informations sur les conférences | 164 |
| Intégration dans une application web | 165 |
| <i>Avec rasterisk</i> | 165 |
| <i>Via le répertoire de file d'attente</i> | 166 |
| Accéder aux audioconférences par le Web (serveur Apache 2.2.3, PHP 5.1.6) | 167 |
| Configuration Apache/PHP | 167 |
| Configuration du dialplan sur le serveur d'audioconférences (conferenceserver) | 175 |

CHAPITRE 10

CTI : l'intégration dans le système d'information..... 179

| | |
|---|-----|
| AGI (Asterisk Gateway Interface) et l'intégration de scripts | 179 |
| Présentation du nom de l'appelant par interrogation de l'annuaire central | 181 |
| Réception de fax par Asterisk et envoi par courriel au destinataire | 185 |
| Appel par composition d'adresse de courriel depuis les softphones | 188 |
| AMI (Asterisk Manager Interface) : le CTI par Asterisk | 190 |
| Un service de click-to-call par le Web (serveur Apache 2.2.3, PHP 5.1.6) | 196 |
| Configuration Apache/PHP | 197 |
| Configuration du dialplan d'Asterisk | 202 |
| Dynamiser l'interface web du service d'audioconférences avec Ajam | 204 |
| AMI + Ajax = Ajam | 204 |
| Activer Ajam et l'intégrer dans un serveur web Apache | 205 |
| Publication des événements par Asterisk | 207 |
| Configuration de l'interface web via les scripts PHP | 208 |

CHAPITRE 11

Administration des serveurs Asterisk..... 213

- Considérations générales sur la sécurité 213
 - Configuration du routeur filtrant 213
 - Configuration des canaux et contrôle d'accès dans le dialplan 214
 - Restriction du nombre de modules installés 214
 - Choix d'une version stable du logiciel 214
- Les journaux d'activité (fichiers de log) 215
 - Configuration 215
 - Archivage 216
- Tickets de taxation (CDR) 217
 - Les CDR sur le serveur Asterisk principal (voiceserver) 219
 - Configuration d'Asterisk en tant que client RADIUS 222
 - Informations enregistrées pour les services auxquels accède le Web 225

CHAPITRE 12

Diagnostiquer un dysfonctionnement avec Asterisk 227

- Diagnostiquer un problème 228
 - La console et les fichiers de log pour le débogage 228
 - Les commandes accessibles depuis la console 229
 - L'application DumpChan 229
- Problèmes de connectivité SIP 231
 - Identification des terminaux SIP sur Asterisk 231
 - SIP, RTP et NAT 233
 - Les outils d'analyse réseau sont nos amis 237
- Problèmes d'accès RNIS primaire et de base 238

CHAPITRE 13

Interconnexion de sites 241

- Les systèmes de téléphonie sur les sites distants 242
- SIP : le protocole de connexion 243
- Une architecture construite autour de Kamailio 243
 - Installation de Kamailio 244
 - Configuration de Kamailio 246
 - Paramètres de configuration* 247
 - Script de routage des messages SIP* 247
 - Configuration du routage des messages SIP* 250
 - Autres fonctionnalités utiles de Kamailio 253

CHAPITRE 14

Domaines d'applications particuliers :**IPv6, SRTP, vidéo et XMPP..... 255**

| | |
|--|------------|
| Des domaines d'applications à conquérir | 255 |
| IPv6, le futur protocole de l'Internet | 256 |
| Confidentialité des communications SIP/RTP par SRTP | 257 |
| Traitement de la vidéo | 257 |
| Asterisk et XMPP (Jabber) | 258 |
| Implémentation XMPP dans Asterisk | 260 |
| GoogleTalk et Jingle : le multimédia sur XMPP | 262 |
| Le futur de XMPP dans Asterisk | 265 |
| <i>Jingle pour la ToIP par XMPP standard</i> | 265 |
| <i>JABBER_RECEIVE pour recevoir des messages instantanés</i> | 265 |
| <i>Notification d'état téléphonique par XMPP</i> | 267 |

CHAPITRE 15

Contribuer à Asterisk 269

| | |
|--|------------|
| Une communauté nombreuse et ouverte à tous | 269 |
| Différentes versions pour différents usages | 270 |
| Des versions stables pour un service téléphonique fiable | 270 |
| Une version de développement pour tester les dernières fonctionnalités | 272 |
| Communiquer avec les utilisateurs d'Asterisk | 273 |
| Listes de diffusion et autres moyens de communication | 273 |
| Le gestionnaire d'incidents (bug tracker) d'Asterisk | 275 |
| <i>Les rapports de bugs ou tickets d'incidents</i> | 275 |
| <i>Correctifs (patches)</i> | 276 |
| <i>Développement de nouvelles fonctionnalités</i> | 276 |
| Les utilisateurs français d'Asterisk | 277 |

Index..... 279

1

Asterisk et la ToIP libre

Asterisk est un logiciel de ToIP, certes, mais pas n'importe lequel ! Entrez dans l'univers d'Asterisk et de la ToIP, ses diverses fonctions et protocoles, sans oublier les autres logiciels de même type...

Qu'est-ce qu'Asterisk ?

Asterisk a pour principale fonction celle d'autocommutateur téléphonique. Cela consiste essentiellement à gérer les appels téléphoniques pour un ensemble de postes, tout comme pouvait le réaliser une opératrice voici de nombreuses années.

B.A.-Ba PABX, PBX, IPBX, IP-PBX...

Un PABX (*Private Automatic Branch eXchange*), ou PBX, est un équipement de gestion des appels téléphoniques. Généralement installé dans une entreprise, il permet de délivrer (ou de commuter) les appels téléphoniques vers les postes qui lui sont raccordés. Il sert en outre de relais entre les postes téléphoniques gérés et le réseau téléphonique commuté (RTC), assurant ainsi la communication avec l'extérieur.

Depuis l'émergence de la technologie ToIP (*Telephony over IP*, ou téléphonie sur IP), qui transfère le service de téléphonie sur le protocole IP donc sur un réseau informatique, les termes IPBX, PABX-IP, IP-PBX sont apparus. Ils traduisent le fait que le PABX est désormais livré sous la forme d'un logiciel destiné à un ordinateur générique, plutôt qu'un matériel spécialisé.

Le cadre classique d'installation d'Asterisk est celui de tout PABX : assurer la commutation des appels pour un ensemble donné de postes téléphoniques et relayer les appels de et vers l'extérieur au travers du RTC. Rien de moins.

Cependant, et contrairement à certaines solutions de ToIP propriétaires qui se contentent de reproduire ce schéma classique de la téléphonie d'entreprise, Asterisk concrétise les promesses de la technologie ToIP, en proposant de nouveaux services sans casser sa tirelire.

B.A.-Ba SIP, H.323, les protocoles pour la technologie ToIP

SIP (*Session Initiation Protocol*, protocole Internet de gestion d'une session de communication) et H.323 (son équivalent issu du monde des télécommunications) sont aujourd'hui les protocoles standards que l'on retrouve dans la plupart des équipements ToIP actuels. Vous trouverez en fin de chapitre un récapitulatif sur les protocoles utilisés en ToIP.

Les fonctions assurées par Asterisk

Asterisk satisfait de nombreux types de besoins. Il peut par exemple devenir un petit gestionnaire du téléphone de son domicile. Dans une telle configuration, on tire avantage de son double attachement au réseau téléphonique et à l'Internet, de façon simple et efficace. Profitant de la connexion Internet de son lieu de travail, qui permet de joindre le serveur Asterisk installé à son domicile, un particulier dispose d'un système de notification de tout appel arrivant durant les heures de travail sur le numéro de son domicile. Nous présentons plus en détail ce type de configuration au chapitre 8, traitant de l'accès distant aux ressources téléphoniques.

Asterisk est également utilisé dans un cadre professionnel, en tant que commutateur téléphonique d'entreprise, de passerelle vers le réseau téléphonique, de serveur de messagerie vocale ou de serveur d'audioconférence. Une des grandes forces d'Asterisk et des logiciels libres de télécommunication en général, tient dans ce caractère multifonctionnel. En effet, les fonctions mentionnées ci-dessus peuvent être assurées soit sur un système unique, par exemple pour réduire le coût d'installation, soit sur différents systèmes hébergeant chacun une instance d'Asterisk, afin d'améliorer la disponibilité d'ensemble des différents composants du service téléphonique.

COMMUNAUTÉ Naissance du projet

Asterisk est un logiciel libre et Open Source (license GPL ou alternative en accord avec la société Digium) apparu à la fin des années 1990. Sa première version, publiée par Mark Spencer, date exactement du 5 décembre 1999. Il s'inscrit dans la mouvance, apparue à la même époque, des logiciels libres de télécommunication développés autour de H.323 ou SIP, comme OpenH323 (aujourd'hui GnuGK) et SER (*SIP Express Router*).

À la différence de la plupart des logiciels libres relevant des télécommunications, Asterisk n'est pas le résultat des travaux d'un ou plusieurs développeurs réalisés à partir d'un document décrivant un protocole standard, comme une RFC (*Request For Comments*) ou une recommandation de l'UIT (Union internationale des télécommunications). Asterisk est né du besoin très pragmatique d'un jeune directeur de société de services d'assistance autour de Linux et des logiciels libres, nommé Mark Spencer, qui souhaitait améliorer l'efficacité du service d'assistance technique en offrant la possibilité aux clients de laisser des messages téléphoniques et en les dirigeant vers le technicien à même de les traiter.

Serveur d'audioconférences

La gestion des audioconférences est une fonction majeure d'Asterisk, qui ravira tout responsable de SI (système d'information) ayant eu la surprise de découvrir des sommes astronomiques sur ses factures, ou dont le service d'audioconférence est limité à cinq utilisateurs par session (limite tout à fait franchissable sous réserve d'acquérir la licence nécessaire).

Asterisk, par l'intermédiaire de l'application MeetMe, agit comme un véritable pont de conférence. Comme on peut l'attendre de tout logiciel libre, aucune licence ne restreint l'utilisation du service d'audioconférence ainsi déployé, les seules limites d'ordre quantitatif découlant des capacités du matériel.

Nous présentons en détail les possibilités d'Asterisk dans le domaine de l'audioconférence au chapitre 9 sur les réunions téléphoniques.

Messagerie vocale

La messagerie vocale est le complément indispensable à tout système de téléphonie digne de ce nom.

Là encore, Asterisk brille par sa souplesse ainsi que par la richesse de ses fonctionnalités. Il est ainsi tout à fait à même de réaliser ce que certains éditeurs ou constructeurs de système de téléphonie appellent parfois un peu

pompeusement la messagerie unifiée. L'envoi de message électronique (protocole SMTP) dans une boîte e-mail sur réception d'un message téléphonique est naturellement possible, de même que l'envoi d'un message instantané par Jabber (protocole XMPP).

B.A.-BA Protocoles de messagerie

Le protocole SMTP (*Simple Mail Transfer Protocol*) est un protocole de type IP utilisé pour le transfert de courrier électronique.

Quant à XMPP (*eXtensible Messaging and Presence Protocol*), c'est un ensemble de protocoles de messagerie instantanée et de présence basés sur le format ouvert XML. XMPP est aussi connu sous le nom de Jabber.

Voici quelques exemples de configuration d'un service de messagerie vocale sous Asterisk :

- Asterisk en tant qu'IPBX incluant la messagerie vocale ;
- Asterisk en tant que serveur de messagerie vocale connecté à un autocommutateur existant, par exemple grâce à SIP ;
- notification sur une boîte e-mail ou par la messagerie instantanée lors de la réception d'un message téléphonique.

Il n'est pas rare de voir un serveur Asterisk assigné à la seule tâche de la messagerie vocale. En séparant les fonctions sur différents serveurs, on améliore parfois la disponibilité des éléments qui constituent le service téléphonique fourni aux utilisateurs.

Serveur vocal interactif (SVI)

Un SVI (en anglais IVR, pour *Interactive Voice Response*) est une application téléphonique d'aide à la navigation. L'utilisateur appelant le système est guidé par une voix parmi différents menus qu'il explore ou sélectionne en appuyant sur des touches de son terminal (par exemple un poste téléphonique), chacune émettant alors un code sonore (DTMF) qui sera interprété par le SVI comme un chiffre.

B.A.-BA Codes DTMF (Dual Tone Multi Frequency)

Les touches pressées par l'utilisateur appelant le SVI sont interprétées par ce dernier à partir des codes DTMF. Ces codes correspondent à des combinaisons de deux fréquences dans la bande passante téléphonique [300 Hz, 3 400 Hz] impossibles à émettre par une voix humaine.

Asterisk dispose des éléments constitutifs d'un SVI évolué :

- un mécanisme de routage interne des appels reposant sur des tests conditionnels ;
- la possibilité de jouer des fichiers préenregistrés dans différents langages ;
- une application permettant d'enregistrer ses propres fichiers depuis un téléphone.

Le routage des appels est configurable via le *dialplan*, l'équivalent du plan de numérotation d'un PABX classique. Il décrit les combinaisons que les utilisateurs peuvent composer et ce à quoi mène chaque série de touches frappées.

Les fichiers préenregistrés sont inclus dans les distributions d'Asterisk et disponibles en plusieurs langues. Ils contiennent les menus vocaux qui seront joués à l'utilisateur au cours d'un appel au SVI.

Asterisk permet également d'enregistrer des fichiers vocaux qui pourront servir à établir un menu vocal personnalisé, par le biais de l'application Record.

Dans les chapitres traitant du couplage d'Asterisk avec Apache/PHP (chapitres 9 et 10 sur les réunions téléphoniques et l'intégration dans le système d'information) ou de la messagerie vocale (chapitre 7), nous présentons les aptitudes d'Asterisk en tant que serveur vocal interactif.

Passerelle vers le RTC, interconnexion de sites

Le succès d'Asterisk tient en grande partie à la possibilité de le connecter au réseau téléphonique commuté (RTC). Rappelons que le créateur du projet, Mark Spencer, avait besoin d'un PABX pour orienter les appels vers le service d'assistance technique de la société qu'il dirigeait. La fonction de passerelle vers le RTC est essentielle au service téléphonique et Asterisk ne fait là que reprendre une fonction de base attendue de tout PABX, pour un coût bien moindre.

Connecter un serveur Asterisk au RTC peut être effectué de différentes manières. Le plus souvent, une carte d'extension dédiée insérée dans un PC en fera une passerelle entre le réseau informatique et le RTC. Il est également possible de connecter Asterisk via un protocole de technologie ToIP à une passerelle existante. Par exemple, activer la pile SIP d'un vieux routeur Cisco 3620 équipé d'une carte d'interfaçage vers le RTC suffit pour obtenir une passerelle SIP/RTC. Asterisk et le routeur communiqueront simplement par SIP. En cas de problème d'interopérabilité, vous savez sur quel équipement il faudra intervenir, pour ajuster la configuration, ou éventuellement le code source... celui qui est libre !

Mais ce n'est pas tout. Asterisk est aussi conçu pour bâtir des architectures téléphoniques robustes, flexibles et peu coûteuses. Par exemple, dans le cas d'une entreprise étendue sur différents sites, disposer des serveurs Asterisk en entrée RTC d'un site et les connecter entre eux par un protocole ToIP (SIP, H.323, IAX, etc.) offre un service téléphonique d'entreprise avancé.

Asterisk disposant d'une multitude d'interfaces protocolaires, il est tout à fait possible de connecter plusieurs serveurs Asterisk entre eux ou à d'autres IPBX ou PABX.

Ce type de configuration est exactement celui de l'entreprise que nous suivons dans cet ouvrage ; nous y revenons en détail dans les chapitres 6 et 13, « Asterisk en point de terminaison télécom » et « Interconnexion de sites ».

BONNE PRATIQUE Utilisation d'Asterisk en complément d'une architecture téléphonique existante

Certes, Asterisk brille par l'exploitation des postes téléphoniques quand on l'utilise comme IPBX. Mais il est tout aussi puissant quand il vient compléter une architecture téléphonique existante, en prenant en charge les fonctions complémentaires, mais souvent très coûteuses, que sont la messagerie vocale ou les réunions téléphoniques (audioconférences). Aujourd'hui, la plupart des IPBX du marché disposent d'une interface SIP qui permet une interaction forte avec Asterisk. Dans le cas d'équipements plus anciens, insérer dans le PC animant Asterisk une carte d'extension (par exemple de type T0, T2 ou analogique) permettra d'intégrer Asterisk (voir la section sur DAHDI et libpri dans le chapitre 4).

Une plate-forme applicative pour la téléphonie : Asterisk et le CTI

Contrairement à un PABX classique, et du fait de son caractère libre qui lui confère une grande souplesse, on peut utiliser Asterisk dans d'autres cadres.

Par exemple, en équipant un serveur d'une deuxième carte de connexion vers le RTC, on peut le placer entre le RTC et le PABX existant afin de permettre l'ajout de nouvelles fonctionnalités téléphoniques en douceur, sans changer les composants du service téléphonique déjà en place. Interposer ainsi Asterisk entre le RTC et le PABX existant présente alors un intérêt immédiat : il supervise le trafic téléphonique, et l'étendue de ses possibilités de configuration va permettre de caractériser certains événements avant de déclencher des actions.

Sur réception d'un appel entrant, Asterisk pourra ainsi consulter une base de données de clients et lancer un programme de remontée de fiche détaillant ses caractéristiques au destinataire de l'appel. C'est un bel exemple de couplage réussi entre téléphonie et informatique (on appelle ce couplage CTI), avec un logiciel libre et pérenne.

Asterisk s'arrête là, mais il laisse la place à l'intégrateur ou au développeur qui exploitera ses interfaces applicatives pour mettre en place des services intégrés au système d'information :

- AGI (*Asterisk Gateway Interface*) permet d'exécuter tout type de programme externe indépendamment d'un langage de programmation.
- AMI (*Asterisk Manager Interface*) est une interface de pilotage d'Asterisk accessible par le réseau, par exemple grâce à HTTP.

Nous approfondirons l'utilisation d'AGI et AMI dans le chapitre 10 traitant de l'intégration au système d'information, puisque notre entreprise s'est résolument décidée à exploiter les capacités d'Asterisk dans le domaine du CTI, par exemple pour fournir un service de *click-to-call* accessible depuis le Web.

Asterisk et les autres logiciels libres de ToIP

S'il est certainement le plus populaire des logiciels serveurs de technologie ToIP, Asterisk n'est pas le seul. Quant à être le meilleur, c'est une question qui ne peut réellement être posée si on le compare à des logiciels qui lui sont fonctionnellement différents, comme SER ou OpenSER qui sont des proxys SIP.

B.A.-BA Proxy

Un *proxy* (mandataire, en français) est un logiciel servant d'intermédiaire entre un réseau rapide et sûr (souvent un réseau local) et un réseau plus lent (l'Internet) ou hostile. Il s'interpose entre ces réseaux pour protéger le premier, par exemple, en bloquant ou filtrant les accès, gérer un cache pour améliorer les performances ou garder une trace des éléments du trafic (journalisation).

Les logiciels présentés ci-après possèdent un certain nombre d'avantages par rapport à Asterisk ; ils sont mentionnés. Quant à ceux qui se distinguent fonctionnellement, leurs différences sont soulignées.

Kamailio (ex OpenSER) : un proxy SIP, un vrai

Kamailio est un *fork* de SER (*SIP Express Router*). Ses auteurs sont une équipe d'ingénieurs roumains qui, à l'époque des premiers développements, préparaient leur thèse à l'institut Fraunhofer de Berlin et travaillaient de façon intensive sur SER, la première implémentation libre d'un proxy SIP.

CULTURE Fork : la fourchette logicielle

Le monde du logiciel libre autorise et même encourage les modifications de code dans le but d'améliorer le logiciel qui en fait l'objet, et ce d'où qu'elles viennent. Cependant, tout logiciel libre est dirigé par une équipe de développeurs qui en modifient directement le code et gèrent les contributions externes à l'équipe. Parfois, des désaccords portant sur l'évolution du logiciel mènent un groupe de développeurs à « forker » le code d'un logiciel existant. Cela signifie qu'ils en prennent le contenu à un instant donné pour donner naissance à un nouveau logiciel qu'ils feront évoluer selon leur idée. Le modèle du logiciel libre les y autorise et, selon la licence applicable au logiciel d'origine, il sera éventuellement fait mention de l'héritage dans le code du nouveau logiciel obtenu. Kamailio (ex OpenSER) est un fork du logiciel SER (*SIP Express Router*), CallWeaver est un fork d'Asterisk.

CULTURE Kamailio, OpenSIPS, SER

Kamailio et OpenSIPS sont issus du logiciel OpenSER, qui fut lui-même créé à partir de SER. Kamailio et OpenSIPS se revendiquent tous deux comme descendants d'OpenSER, à juste titre, et leurs bases de code sont encore très similaires. Depuis novembre 2008, Les développeurs de Kamailio et de SER travaillent conjointement sur un nouveau projet nommé *SIP Router Project*. Cette multiplication des projets descendants de SER prête à confusion, même si les logiciels nés des efforts des contributeurs sont de grande qualité.

On a parfois tendance à opposer Asterisk à Kamailio alors qu'ils sont fondamentalement différents, et en fait complémentaires.

Les domaines dans lesquels Kamailio se révèle particulièrement remarquable sont le traitement des connexions SIP NAT, la montée en charge et la haute disponibilité. On ne peut que saluer l'excellence des contributeurs de ce logiciel, en particulier Daniel Constantin Mierla et Elena Ramona Modroiu. Ce logiciel constitue la clef de voûte de l'architecture ToIP de nombreux opérateurs.

B.A-BA NAT ou la traduction d'adresses

La traduction d'adresses réseau (NAT ou *Network Address Translation*) consiste à traduire les adresses IP non routables (dans un réseau interne, le plus souvent) en adresses IP routables (c'est-à-dire ayant accès à un autre réseau, souvent l'Internet).

APPROFONDIR Asterisk versus Kamailio : proxy, registrar ou B2BUA SIP ?

Kamailio est un *proxy* SIP au sens strict de la RFC 3261, qui décrit le protocole SIP. Cela signifie d'abord qu'il implémente un unique protocole de ToIP : SIP. Asterisk, quant à lui, abrite une véritable foire aux protocoles de ToIP standards (SIP, IAX, H.323, MGCP) et propriétaires (SCCP de Cisco, Unistim de Nortel) – voir le récapitulatif sur les protocoles de ToIP en fin de chapitre.

Kamailio, comme tout *proxy* SIP, relaie des transactions entre différentes entités SIP. Par conséquent, il ne répondra pas de lui-même à une requête INVITE d'établissement de connexion SIP, qu'il se contentera de relayer vers un terminal ou un autre *proxy*. À l'inverse, Asterisk peut être programmé pour répondre à une telle requête, par exemple pour jouer un fichier pré-enregistré.

Dans une architecture SIP, Asterisk est en fait un B2BUA (*Back to Back User Agent*). À la différence d'un *proxy* SIP, un B2BUA peut tout à fait émettre et recevoir des appels SIP et les relayer en maintenant des canaux distincts et indépendants d'un point de vue SIP.

De plus, Asterisk et Kamailio implémentent tous les deux la fonction de serveur d'enregistrement SIP (*registrar*). Un *registrar* accepte des requêtes d'enregistrement de la part de terminaux SIP. Une fois enregistré, un terminal est accessible via l'adresse `username@X.X.X.X`, où `username` représente le nom du terminal soumis dans la requête REGISTER et `X.X.X.X` l'adresse IP du *registrar*. En général, les fonctions de *registrar* et *proxy* SIP sont incluses dans un même serveur (par exemple, Kamailio).

Là encore, on constate des différences dans le traitement des enregistrements entre Asterisk et Kamailio. Enregistrer un même identifiant (URI – *Universal Resource Identifier*) depuis différents terminaux est possible avec Kamailio, mais pas avec Asterisk. Cette fonctionnalité permet de faire sonner simultanément les terminaux s'étant enregistrés avec le même identifiant (technique dite de *forking* SIP, qui n'a rien à voir avec le fork d'un logiciel libre !). Pour enregistrer un même identifiant depuis différents terminaux sur Asterisk, il faut utiliser des protocoles différents sur chaque terminal, par exemple SIP et IAX (*Inter-Asterisk eXchange*), comme nous le verrons dans le chapitre 8 traitant des *softphones*.

Les implémentations SIP d'Asterisk et de Kamailio diffèrent car les fonctions des deux logiciels sont différentes, comme indiqué ci-dessus. De plus, à la différence de Kamailio, Asterisk implémente bien d'autres protocoles de communication que SIP, comme H.323, MGCP (*Media Gateway Control Protocol*) et, surtout, des interfaces vers le RTC analogique et numérique.

Kamailio est un logiciel modulaire capable de traiter des milliers de messages SIP par seconde. Mieux, ses fonctions de répartition de charge autorisent une montée en charge simple par ajout de matériel. L'intérêt que peut susciter un tel logiciel (gratuit et libre !) chez les opérateurs est ainsi facilement compréhensible.

Finalement, on notera que Kamailio se trouve également dans l'architecture IMS de certains opérateurs de ToIP, en tant que SBC (*Session Border Controller*) ou pilote de SBC. Ses fonctions de traitement des en-têtes SIP et sa capacité à gérer les connexions NAT sont très appréciées des administrateurs de SBC pour filtrer les connexions SIP/RTP.

B.A.-Ba Architecture IMS, SBC

Une architecture IMS (*IP Multimedia Subsystem*) est une architecture NGN (*Next Generation Network*) standardisée pour la fourniture de services multimédia sur IP par les opérateurs téléphoniques.

Quant aux SBC (*Session Border Controller*), ou contrôleurs de session en périphérie, ce sont des équipements réseau utilisés dans le domaine voix sur IP pour contrôler les données transmises lors des appels.

Les concurrents d'Asterisk

D'autres logiciels libres de ToIP proposent des fonctionnalités similaires à celles d'Asterisk, en premier lieu plusieurs protocoles de communication (SIP, H.323, MGCP, RNIS, etc.). Certains des logiciels que nous présentons ici ont été forkés à partir d'Asterisk, souvent en raison de désaccords avec la société Digium, qui maintient et fait évoluer ce dernier.

L'un des reproches formulés sur Asterisk tient à sa diffusion sous double licence (GPL et propriétaire), qui interdit notamment l'inclusion de certaines bibliothèques dans le code d'Asterisk. Cela a parfois poussé certains contributeurs à quitter le projet.

Malgré tout, aussi efficaces, bien codés et fiables soient les logiciels présentés ci-après, aucun d'entre eux n'est appuyé par une communauté de développeurs, testeurs et utilisateurs de l'envergure de celle d'Asterisk.

CallWeaver : un fork d'Asterisk

CallWeaver, anciennement connu sous le nom d'OpenPBX.org, est un fork de la version 1.2 d'Asterisk. Il s'agissait d'abord de se démarquer du modèle

de développement d'Asterisk, jugé trop lié à la société Digium et n'exploitant pas suffisamment les bibliothèques libres existantes.

CallWeaver utilise par exemple les implémentations libres Sofia SIP (de la société Nokia) et SpanDSP. Cette dernière bibliothèque est utilisable pour la transmission et la réception de fax. On notera que désormais, Asterisk utilise aussi la bibliothèque SpanDSP pour le traitement des fax.

FreeSWITCH : la promesse de la fiabilité

Contrairement à CallWeaver, et bien qu'il fût lancé par des ex-développeurs d'Asterisk, FreeSWITCH est un projet bâti sur une base de code complètement nouvelle.

FreeSWITCH utilise, tout comme CallWeaver, la bibliothèque Sofia SIP. Celle-ci offre notamment SIP sur les protocoles TCP et SCTP, ainsi que la sécurisation par TLS. Notons que depuis la version 1.6, Asterisk gère SIP sur TCP, ainsi que la sécurisation par TLS.

B.A.-Ba TCP, SCTP et TLS

TCP (*Transmission Control Protocol*) et SCTP (*Stream Control Transmission Protocol*) sont des protocoles de transport de données de la famille IP.

TLS (*Transport Layer Security*) est un protocole de sécurisation des échanges, qui a pris la suite du protocole SSL (*Secure Sockets Layer*).

D'une manière générale, FreeSWITCH est présenté comme plus fiable qu'Asterisk. Le développeur principal du projet FreeSWITCH, à l'époque où il travaillait sur Asterisk, en était l'un des contributeurs les plus actifs ; le grief majeur qu'il énonçait contre Asterisk était sa gestion des tâches simultanées (par *multithreading*), qu'il jugeait (et juge encore si l'on en croit ses nombreux blogspots sur le sujet) bâclée.

Que dire sinon qu'Asterisk, comme tout logiciel, n'est pas parfait, et doit sans cesse être amélioré. Nul doute que si FreeSWITCH connaît à l'avenir le même succès qu'Asterisk, de nombreux bugs seront découverts.

Yate : encore un autre IPBX logiciel !

Yate, littéralement *Yet Another Telephony Engine*, est un logiciel identique à Asterisk d'un point de vue fonctionnel. Il est poussé par une équipe très talentueuse de développeurs roumains. Il est intéressant de constater la pro-

fonde implication des ingénieurs roumains dans des projets de ToIP libres majeurs comme Yate ou Kamailio.

Yate a une réputation de grande fiabilité et gère des protocoles complexes. Il s'appuie notamment sur le projet Linux-HA (*High Availability*) et son composant Heartbeat pour offrir un haut niveau de disponibilité.

Tout comme les logiciels que nous avons exposés, auxquels on pourrait ajouter SIPX et Bayonne, Yate s'appuie sur une communauté d'utilisateurs moins étoffée que celle d'Asterisk. Dans le monde des logiciels libres, la communauté des utilisateurs, avec le talent des développeurs, est d'une importance primordiale quant à la qualité du logiciel développé. Elle agit un peu comme un service d'assistance, incluant utilisateurs expérimentés et développeurs aguerris.

PROTOCOLES Vue d'ensemble des standards de ToIP

H.323, le précurseur, et SIP, le successeur

Depuis le milieu des années 1990 et la publication des premiers standards jusqu'au début des années 2000, une bataille s'est déroulée entre les organismes de standardisation des protocoles de ToIP, opposant principalement H.323 (de la recommandation éponyme de l'UIT) à SIP (*Session Initiation Protocol*), qui est issu du monde de l'Internet (dit monde IP). Les deux protocoles ont des fonctions identiques. Ils ont été pensés à l'origine pour transporter la téléphonie sur le réseau IP (et ont tous deux évolué pour ajouter le transport de la vidéo). Aujourd'hui, de nombreux équipementiers implémentent SIP ou H.323 dans leurs postes téléphoniques.

La fin des années 1990 fut sans conteste une époque faste pour l'implémentation de H.323, puisque de nombreux réseaux d'opérateurs renouvelèrent alors leurs architectures à partir de ce protocole. On notera en outre que le monde du logiciel libre proposa alors une première implémentation de H.323 : OpenH323.

Aujourd'hui, H.323 est certes très présent dans les systèmes de vidéoconférence, mais force est de constater que les terminaux logiciels H.323 (les softphones), sont en voie d'extinction. Ainsi, Microsoft n'inclut plus dans les versions actuelles de son système d'exploitation le logiciel NetMeeting, et préfère ajouter une interface SIP à son client Windows Live Messenger. La popularité du protocole, si on la mesure grossièrement par le nombre de ses implémentations logicielles et en considérant ses domaines d'application (système d'exploitation grand public, téléphonie d'entreprise) semble avoir atteint son apogée, et se trouve de fait sur la voie du déclin. Cependant, l'activité dans le monde du logiciel libre autour d'H.323 ne stagne pas, loin de là. Des projets comme OpenH323, OPAL, Yate, Bayonne, Asterisk, FreeSWITCH, font régulièrement évoluer leurs piles logicielles H.323. Tout n'est peut-être pas perdu pour ce protocole, véritable précurseur de la ToIP.

En fait, depuis le début des années 2000, les choses ont nettement évolué en faveur d'une adoption massive de SIP. De nos jours, tous les constructeurs de téléphonie d'entreprise munissent leurs IPBX d'une interface de raccordement SIP, et certains vont même jusqu'à proposer une architecture de téléphonie d'entreprise complètement basée sur SIP ! On notera d'ailleurs avec un certain amusement que la tendance à l'option tout SIP dans la téléphonie d'entreprise existe depuis longtemps parmi les intégrateurs de logiciels libres de ToIP, même si elle ne porte que sur des installations de petite ou moyenne capacité. Par ailleurs, on ne compte plus les logiciels libres de ToIP SIP disponibles tant côté serveur (Asterisk, OpenSER, FreeSWITCH, Yate, Bayonne, etc.) que client (Ekiga, X-Lite, Zoiper, SIPhone, etc.).

Le coup d'accélérateur du processus d'adoption de SIP tient certainement dans le choix du 3GPP d'en faire le protocole de signalisation dans l'architecture IMS (*IP Multimedia Subsystem*, grâce à laquelle les opérateurs offrent du multimédia). Le 3GPP (*Third Generation Partnership Project*) est un organisme de standardisation international (intégrant l'ETSI, l'Institut européen des standards de télécommunication, pour l'Europe), qui travaille notamment à la normalisation des nouvelles architectures des réseaux d'opérateurs, dans le but de pouvoir proposer des services dits convergents, c'est-à-dire accessibles aussi bien depuis un terminal fixe que mobile. L'adoption de SIP comme élément de base de l'architecture IMS (avec IP) fut entérinée en 2004.

MGCP et IAX, les outsiders, et Jingle, le nouveau venu

En marge de la compétition entre les protocoles SIP et H.323, un candidat a fait son chemin dans la jungle des protocoles de téléphonie sur IP : MGCP (*Media Gateway Control Protocol*). Adapté à une architecture de type maître-esclave (à l'inverse de SIP et H.323, pour lesquels cette distinction n'existe pas), MGCP se retrouve aujourd'hui dans les services de téléphonie de type Centrex, dont le principe est d'héberger le serveur de téléphonie des entreprises clientes, chez qui aucun équipement de commutation téléphonique (PABX ou IP-PBX) n'est installé.

IAX (*Inter-Asterisk eXchange*), développé par la communauté Asterisk et pour les serveurs Asterisk, a finalement été standardisé (dans la catégorie informationnelle) en février 2009 dans la RFC 5456, ce qui devrait encourager les nouvelles implémentations. Côté serveur, il est implémenté dans Asterisk, FreeSWITCH et Yate. Côté client, on le trouve notamment dans Zoiper et SIPhone (qui implémentent également SIP).

Pour achever le survol des protocoles de ToIP, disons quelques mots d'un nouveau venu. Depuis 2006, sous l'impulsion de la XSF (*XMPP Standards Foundation*) et de Google, un protocole de ToIP nommé Jingle est apparu. Depuis lors, ce protocole a constamment évolué pour aboutir à une version stable dans le courant de cette année (juin 2009). Il permet d'établir des communications voix et vidéo entre les clients d'un réseau XMPP (aussi connu sous le nom Jabber). Le client de messagerie instantanée et ToIP GoogleTalk en fut la première implémentation.

Asterisk

Jingle vient compléter les services avancés accessibles aux utilisateurs des architectures XMPP, tels que la gestion de présence et la messagerie instantanée. Outre GoogleTalk, les clients libres XMPP commencent à se lancer dans l'implémentation de Jingle. Ainsi le client Empathy (disponible sur Linux en environnement Gnome) s'est vu ajouter une version stable de Jingle dans le cadre du projet OLPC (*One Laptop Per Child*), et d'autres ont suivi depuis (Psi et Pidgin). Asterisk implémente quant à lui le protocole depuis la version 1.4, dont la compatibilité avec GoogleTalk et Empathy a été validée. La dernière version du standard Jingle est en cours de développement dans Asterisk et permettra ainsi de faire se rejoindre les mondes de la téléphonie d'entreprise et de la messagerie instantanée.

2

Présentation de l'étude de cas

Une étude de cas nous permet de découvrir tout au long de ce livre différents cadres d'utilisation du formidable outil qu'est Asterisk. En voici la présentation...

L'objet de l'étude est le remplacement du système téléphonique d'une entreprise regroupant une centaine de collaborateurs, répartis sur deux sites disposant chacun des leurs propres équipements de téléphonie.

Situation géographique

Le site principal de notre entreprise est localisé en région parisienne. Environ 90 personnes y sont hébergées, chacune disposant de son poste téléphonique. Sur le site secondaire, situé dans le Sud-Est de la France, se trouve une dizaine de personnes, disposant là encore chacune de son propre poste téléphonique. Tous les membres de l'entreprise se sont vu attribuer un numéro de téléphone professionnel SDA, accessible directement depuis le RTC.

CULTURE SDA : sélection directe à l'arrivée

Un numéro SDA (sélection directe à l'arrivée) est un numéro directement accessible depuis le réseau téléphonique commuté (RTC), sans avoir besoin de passer par un standard de commutation. On peut faire une analogie avec le monde IP en considérant les numéros non SDA (ou privés) comme équivalents aux adresses IP non routables (type RFC 1918, par exemple celles de la forme 192.168.X.X ou 10.X.X.X), et le standard de commutation comme le pendant la passerelle NAT.

Asterisk

Sur chacun des deux sites, un serveur Asterisk sera installé en remplacement des anciens PABX. Les serveurs Asterisk seront ensuite reliés entre eux par le protocole SIP.

Les raisons du changement de système téléphonique

Si la raison principale du remplacement du système téléphonique est le vieillissement du matériel et du logiciel existant, le choix d'une solution de technologie ToIP ne dépend pas que de ce point. Il découle également du coût global, des attentes en matière de nouveaux services téléphoniques et de l'intégration souhaitée avec le système d'information, domaines dans lesquels Asterisk se révèle efficace.

Un existant vieillissant et coûteux

Le système téléphonique actuel date d'une dizaine d'années, c'est-à-dire d'une époque où la téléphonie sur IP permettait d'échanger des syllabes entre deux PC identifiés par leur adresse IP. L'expérience était certes enthousiasmante, mais la technique se révélait inadaptée aux entreprises.

Par ailleurs, la maintenance est devenue très onéreuse et, pis, le constructeur du système existant est sur le point d'annoncer qu'il abandonne sa fabrication.

La nécessité de nouveaux services

La téléphonie sur IP conduit à envisager différemment le futur système téléphonique de notre entreprise. Les utilisateurs poussent au changement et à l'innovation, à partir de leur expérience personnelle (par exemple, à partir des services téléphoniques offerts par des fournisseurs d'accès comme Free), par des questions telles que :

- Puis-je passer et recevoir des appels depuis mon ordinateur portable ?
- Si oui, puis-je le faire depuis chez moi ? Quel en est le coût ?
- L'annuaire de l'entreprise est-il accessible depuis mon téléphone ?
- Puis-je passer des appels en cliquant sur un bouton de page web ?
- Puis-je recevoir des messages vocaux par courriel ?
- Puis-je accéder à des ressources comme le fax ou les audioconférences via mon navigateur web ?

Par ailleurs, du point de vue de l'administrateur, la technologie ToIP présente des avantages certains, comme la gestion d'un seul réseau commun à l'informatique et à la téléphonie, ou la possibilité de développer simplement des applications avancées.

Les raisons du choix de la ToIP libre

Réduire les coûts

L'étude préalable à la sélection du futur système de téléphonie a mis en évidence une politique commune à l'ensemble des constructeurs/éditeurs de solutions de téléphonie sur IP : la multiplication des licences. La liste des objets potentiellement sujets à l'acquisition d'une licence est longue :

- les postes téléphoniques IP physiques ;
- les postes téléphoniques IP logiciels (softphones) ;
- les comptes à créer dans le système de messagerie vocale ;
- les équipements externes reliés par *trunking* standard (SIP, H.323, etc.) ;
- les points de connexion CTI (couplage téléphonie informatique).

Dans le pire des cas, toute fonctionnalité souhaitée devra faire l'objet de l'achat d'une licence. Dans le cas d'un système de téléphonie libre, la seule licence qui nous intéresse ne se prête pas à une quelconque procédure d'achat ; c'est celle appliquée au logiciel considéré : GPL, LGPL, etc.

CULTURE Licences libres

Pour en savoir plus sur les licences du logiciel libre comme la GNU GPL (*GNU General Public License*) ou la LGPL (*Lesser GPL*), rendez-vous sur les pages web suivantes :

▶ <http://jargonf.org/wiki/LGPL>

▶ <http://jargonf.org/wiki/GPL>

À venir également pour ceux qui s'y intéressent :

📖 Richard Stallman, Sam Williams, Christophe Masutti, *Richard Stallman et la révolution du logiciel libre - Une biographie autorisée*, Eyrolles, janvier 2010

S'approprier réellement le système

Deux questions importantes ont été incluses dans l'étude menée par la direction informatique : l'interopérabilité entre systèmes de téléphonie et l'intégration avec le système d'information, désignée par le sigle CTI.

CULTURE Interopérabilité et CTI

La fin des années 1990 a vu naître une volonté affichée des constructeurs de systèmes de téléphonie d'entreprise d'aller vers l'interopérabilité et le CTI (couplage téléphonie informatique). La téléphonie d'entreprise suivait ainsi le mouvement vertueux de la téléphonie d'opérateur qui tendait vers l'ajout de services sur ses réseaux par la promotion des NGN (*Next Generation Networks*). On espérait alors une nouvelle génération de logiciels applicatifs bien interconnectés avec les services de téléphonie.

Dix ans plus tard, on peut constater que l'interopérabilité s'imposa non pas grâce à une volonté commune des constructeurs, mais par l'apparition de la technologie ToIP dans l'entreprise. Quant au CTI généralisé, on l'attend encore.

Si SIP est aujourd'hui le protocole qui permet d'interconnecter des équipements de différentes marques, concrétisant ainsi l'idée d'interopérabilité, les systèmes de téléphonie d'entreprise (IP ou non) restent encore peu intégrés au système d'information. Le CTI n'a donc pas tenu ses promesses, et ce malgré l'apparition d'API propriétaires permettant de développer des applications.

Les logiciels de ToIP libres sont intégrables au système d'information de façon naturelle, pour plusieurs raisons.

- Les difficultés rencontrées sont facilement rapportées aux développeurs experts, qui sont accessibles.
- Les moyens de communication entre développeurs et utilisateurs permettent la réactivité (listes de diffusion, canaux IRC).
- Tout utilisateur peut ajouter une fonctionnalité, en la développant de lui-même ou par l'intermédiaire d'un prestataire.

Certes, on peut considérer que le dernier point relève d'une vision idéalisée du monde de l'Open Source, puisque tout utilisateur ne peut ou ne veut se plonger dans le code source d'un logiciel libre. Cependant, les logiciels libres apparaissent comme un moyen efficace de lier la téléphonie au système d'information, au moins pour une raison : les utilisateurs ne sont plus seuls devant un système plus ou moins fermé.

Maintenance et évolution du logiciel libre en téléphonie

En matière de téléphonie d'entreprise et de logiciel libre, un changement majeur dans la gestion du système intervient : celui de sa maintenance et de son évolution.

Dans le cas d'un système de téléphonie d'entreprise classique, les opérations de maintenance sont en grande partie réalisées par un prestataire, généralement un intégrateur, et plus rarement le constructeur du PABX. La gestion des tickets d'incidents passe par plusieurs niveaux d'assistance, et en cas de problème difficile nécessitant une modification, un nombre limité d'ingénieurs seront à même de pouvoir apporter un correctif.

De plus, les évolutions du système telles que l'ajout de nouvelles fonctionnalités, si mineures soient-elles, font l'objet d'une prestation dont le coût peut parfois sidérer le responsable technique le plus endurci.

Le PABX et les éventuels serveurs associés (taxation, messagerie vocale, etc.) forment ainsi un ensemble hermétique, précieux parce que coûteux et à ne surtout pas toucher.

L'apparition de logiciels libres de technologie ToIP comme Asterisk renverse complètement cette vision. Ces logiciels permettent de réellement s'approprier le système, mais également de pouvoir compter sur une communauté d'utilisateurs, de testeurs et de développeurs en nombre très important (au moins pour Asterisk). Bien qu'aucun temps limite de réaction ne puisse être exigé de la communauté, celle-ci constitue de fait un service d'assistance technique extrêmement efficace et gratuit. De surcroît, des prestataires proposent des services ou garanties similaires à celles de leurs concurrents du monde non libre (propriétaire).

Pourquoi Asterisk ?

Afin d'assurer dans les meilleures conditions la migration du système téléphonique, la direction informatique a été sollicitée pour étudier les offres disponibles et a retenu un certain nombre de critères conduisant à sélectionner Asterisk :

- continuer d'assurer les services téléphoniques couramment utilisés par les collaborateurs de l'entreprise ;
- intégrer le nouveau système téléphonique au système d'information (messagerie vocale, annuaire d'entreprise) ;

Asterisk

- apporter de nouveaux services aux collaborateurs de l'entreprise comme à ses prospects et clients ;
- ne pas adopter de solution peu diffusée, potentiellement peu pérenne ;
- payer tout cela aussi peu que possible !

L'idée d'un système de technologie ToIP Open Source a rapidement convaincu, et le choix d'Asterisk comme base du nouveau système est apparu évident, compte tenu de la liste des protocoles implémentés, qui va des standards pilotant les terminaux utilisateurs (SIP, H.323, MGCP, PSTN) aux équivalents propriétaires (SCCP, Unistim), en passant par de nombreux protocoles de connexion avec des équipements de cœur de réseau (SS7, ISDN, ENUM, SIP, H323).

RAPPEL Les protocoles de ToIP

Un aperçu récapitulatif des protocoles utilisée en ToIP est fourni en fin du chapitre 1.

Mais le principal avantage d'Asterisk sur les autres logiciels de ToIP Open Source est sa forte popularité qui, combinée aux fonctionnalités d'intégration dans le système d'information (LDAP, SQL, scripts AGI, pilote AMI) offre une infinité d'architectures et de configurations, et la possibilité d'échanger avec des experts et passionnés de plus en plus nombreux.

L'installation du nouveau système téléphonique se déroulera sur plusieurs étapes, en partant d'une intégration pilote du service informatique, pour terminer par la mise à disposition de services téléphoniques avancés accessibles par le Web.

Les services offerts aux collaborateurs de l'entreprise

Reconduire l'existant : fax, transfert et notification d'appel

Dans un premier temps, le service informatique validera un certain nombre de services téléphoniques de base, notamment ceux actuellement offerts par le système actuel ; citons parmi eux :

- le transfert d'appels ;
- la mise en attente accompagnée d'une musique d'attente ;

- la notification d'appel (par exemple entre un chef de service et son assistant).

Cette étape permettra de se familiariser avec Asterisk.

Ajouter des services : la mobilité et l'intégration dans le système d'information

Des services plus évolués seront testés et validés peu après, tels que :

- l'envoi et la réception de fax par le Web ;
- l'accès aux audioconférences gérées par un serveur web ;
- une messagerie vocale couplée avec la messagerie électronique ;
- la configuration de l'accès distant aux ressources téléphoniques ;
- une fonction de click-to-call accessible depuis l'intranet de l'entreprise.

Les machines fax ne feront plus partie la nouvelle architecture téléphonique, car un service web couplé avec Asterisk permettra d'en envoyer et d'en recevoir.

Les audioconférences seront créées dynamiquement par les utilisateurs à partir d'un accès web.

Le système de messagerie vocale sera avantageusement remplacé par Asterisk, qui permettra notamment d'envoyer les messages vocaux sous forme de pièce jointe à un courrier électronique.

L'accès distant aux ressources téléphoniques depuis un site distant quelconque connecté à l'Internet, par exemple depuis le domicile d'un collaborateur de l'entreprise équipé d'une connexion ADSL, sera rendu possible. Le cas d'une connexion via un téléphone logiciel (softphone) sera présenté, ainsi que celui d'un serveur Asterisk localisé au domicile du collaborateur pour assurer des tâches telles que la notification des appels reçus ou le renvoi vers d'autres numéros.

La fonction click-to-call permet aux collaborateurs de l'entreprise de cliquer sur un lien d'une page web, représentant par exemple un autre collaborateur ou un numéro de téléphone quelconque, dans le but d'établir une communication téléphonique. Les capacités en terme de CTI (couplage téléphonie informatique) d'Asterisk permettent de mettre en œuvre très simplement cette fonction.

L'accompagnement de la croissance de l'entreprise

Notre entreprise a des perspectives d'évolutions très favorables, ce qui l'amène à considérer l'acquisition d'une société concurrente basée à l'étranger.

Interconnexion de sites distants par SIP

Là encore, la direction informatique a été sollicitée pour anticiper la montée en charge du système de télécommunications et doit préparer un plan d'interconnexion de ce nouveau site, qui gardera tels quels ses équipements de téléphonie. Ceux-ci sont en effet trop récents pour être changés, et ils offrent une interface de technologie ToIP désormais commune à tous les équipements de téléphonie d'entreprise : le protocole SIP.

Nous verrons comment Asterisk, seul ou combiné à d'autres logiciels libres de ToIP, comme Kamailio (un fork d'OpenSER), permet de se connecter à tout type d'équipement SIP.

Travail collaboratif et gestion de présence

Dans le but d'améliorer l'accessibilité de ses collaborateurs, la direction considère la possibilité d'installer un système de gestion de présence offrant des services annexes comme la messagerie instantanée à deux ou en groupe.

Cette perspective d'évolution a bien été intégrée par la direction informatique, et l'un des points de l'étude du nouveau système téléphonique traite de l'intégration avec les services de gestion de présence et de messagerie instantanée. Ce point fera l'objet d'une description détaillée des possibilités d'intégration d'Asterisk dans le monde de la messagerie instantanée via le standard Jabber (XMPP).

Vidéo

La vidéo fait partie des fonctionnalités avancées jugées intéressantes par la direction, mais ne nécessitant pas de déploiement immédiat. Un état de l'art des possibilités pertinentes d'Asterisk a été demandé à la direction informatique.

Index

Numeriques

3GPP (Third Generation Partnership Project) 19
802.1X (protocole) 37

A

ABE (Asterisk Business Edition) 40
accès distant 27, 129
 sécurité 132
 voir aussi VPN
accessibilité 241
ACL (Access Control List) 192
administration 213
 sécurité 213
adressage réseau 53
adresse
 de courriel
 appel via ~ 188
 MAC 256
 réseau
 choix 53
 traduction *voir* NAT
AEL (Asterisk Extension Language) 75
 goto 177
 label 177
 priorité 177
AGI (Asterisk Gateway Interface) 13, 86, 180
 arguments 183
 script 182, 187, 189
 variables 183

Ajam (Asynchronous JavaScript Asterisk Manager) 204
Ajax (Asynchronous JavaScript language And XML) 199, 205
 Prototype 209
A-Law (codec) 61
ALG (Application Level Gateway) 144
AMI (Asterisk Manager Interface) 13, 85, 190
 Ajam 205
 Ajax 196
 Astmanproxy 196
 événements 207
 fichier de configuration 192
 liste des commandes 192
 PHP 196
 sécurisation 192
 WaitEvent (commande) 211
annuaire 31, 34, 181
 LDAP 34, 117, 135, 156
annulation d'écho 124
Answer (application) 185
Apache 118, 167, 197
 intégration d'Ajam 205
 reverse proxy 206
API (Application Programming Interfaces) 32
app_voicemail (module) 106
appel 78
 notification 27, 124
 renvoi 27
 SIP 232
architecture

Asterisk

- IMS 16, 19
 - logicielle 58
 - réseau 63
 - téléphonique 12, 241, 243
 - ARP (Address Resolution Protocol) 256
 - Asterisk
 - Business Edition 40, 54
 - communauté 269, 273
 - distribution 54
 - fichiers source 42
 - France 277
 - installation 39, 54
 - chez un particulier 137
 - licence 40
 - réinstallation 117
 - sous Linux 40
 - sous Windows 40
 - utilisateurs 273
 - versions 214, 270
 - Asterisk-France 277
 - AsteriskNOW 39, 54
 - attribut 69, 102
 - allowguest 67, 153, 233
 - allowmultiplelogins 195
 - auth 162
 - callcounter 127
 - call-limit 127
 - canreinvite 69
 - crypto 257
 - externhost 150, 233
 - externip 141, 149, 233
 - externrefresh 150
 - faxdetect 186
 - insecure 150
 - localnet 141, 149, 233
 - mailbox 108
 - nat 144, 234
 - realm 67
 - register 66, 150
 - secret 232
 - srvlookup 150, 153
 - State 150
 - stunaddr 150
 - subscribecontext 127
 - transfer 69
 - type 70
 - webenabled 195
 - audioconférence 9, 27, 155
 - application web 165
 - authentification 162
 - DAHDI 43
 - interface web dynamique 204
 - serveur 157
 - Authenticate (application) 104
 - authentification 222
 - AMI 190, 192
 - audioconférences 162
 - des utilisateurs 33, 117, 134
 - LDAP 106, 118, 197
 - RADIUS 253
 - autocommutateur 7, 30
 - autoconf 55, 245
 - automake 55, 245
 - autotools 55, 245
- ## B
- B2BUA (Back to Back User Agent) 15, 243, 258
 - Background (application) 103
 - Bayonne 18
 - bibliothèque 55
 - de programmation *voir* API
 - iksemel 260
 - libavcodec 258
 - libpri 40, 51
 - libsrtp 257
 - ncurses 42
 - Sofia SIP 17
 - spandsp 17, 115, 116
 - termcap 42
 - branche (SIP) 249
 - bridge 234
 - bug tracker 275
- ## C
- C (langage) 40
 - CallWeaver 14, 16, 131
 - canal 78, 202, 264

- configuration sécurisée 214
 - LOCAL 122, 174
 - carte
 - d'extension 11, 12
 - détection 51
 - Digium B410P 89
 - Digium TE120P 43
 - CDR (Call Detail Record) 122, 198, 202, 217
 - CGI (Common Gateway Interface) 180
 - chan_gtalk (module) 263
 - chan_jingle (module) 263
 - Channel *voir* canal
 - ChanSpy (application) 69
 - CLI (Command Line Interface) 39, 57
 - click-to-call 27, 191, 196
 - code
 - DTMF 10, 106
 - PIN 106, 155, 164
 - source *voir* source (fichiers)
 - codec 59, 61, 259
 - A-Law, Mu-Law 61
 - G711 A-Law 61
 - commande console *voir* CLI
 - communauté 269, 273
 - commutateur téléphonique 8
 - compilation
 - d'Asterisk 57, 91, 144
 - de DAHDI 45, 89, 158
 - de libpri 52, 91
 - outils 42
 - compte utilisateur 134
 - ConfBridge (application) 164
 - configure (script) 56
 - console 85
 - débogage 228, 229
 - contexte 66, 76
 - clicktocall 203
 - default 153, 233
 - from-pstn 136, 185
 - from-sipproxy 251
 - originate-clicktocall 202, 225
 - originate-webconferences 207
 - originate-webfax 123
 - phones 181
 - subscriptions 127
 - webconferences 207
 - webfax 122, 123
 - contrôle d'accès 37, 192, 222
 - AMI 195, 210
 - CDR 217
 - dialplan 214
 - correctif 276
 - crash 275
 - cron (programme) 217
 - CSV (Comma Separated Values) 217
 - CTI (couplage téléphonie informatique) 12, 24, 32, 190
- D**
- DAHDI (Digium Asterisk Hardware Device Interface) 40, 42, 43, 89, 157
 - compilation 45
 - fichiers de configuration 46
 - installation 43
 - supervision et diagnostic 238
 - dahdi_cfg (programme) 160
 - dahdi_dummy (module) 43, 157
 - dahdi_genconf (programme) 46, 159
 - dahdi_scan (programme) 50, 238
 - dahdi_voicebus (module) 50
 - dahdi-linux (paquetage) 45
 - dahdi-tools (paquetage) 45
 - débogage
 - activation 229
 - console 228
 - filtrage 237
 - gdb 228
 - Kamailio 247
 - outils d'analyse réseau 237
 - outils de supervision externes 240
 - RNIS 238
 - DEL (diode électroluminescente) 125, 238
 - DHCP 72
 - diagnostic 227
 - Dial (application) 78, 80, 112
 - dialogue (SIP) 232, 249
 - dialplan 76
 - applications 59

Asterisk

- contexte 76
- contrôle d'accès 214
- extensions 77
- fonctions 59
- variables 83
- voir aussi* plan de numérotation
- Digium 16, 40
- direct media (mode) 69, 234
- distribution
 - d'Asterisk 39, 54
 - de Linux 39, 41
- DNS (Domain Name System)
 - SRV (enregistrement) 150, 260
- driver 45
- DTLS (Datagram Transport Layer Security) 257
- DTMF (Dual Tone Multi Frequency) 10
- DumpChan (application) 229
- dysfonctionnement 227

E

- ejabberd 260
- enregistrement
 - DNS SRV 150
- Ethereal *voir* Wireshark
- Ethernet 35
 - CSMA-CD 35
- extension 175, 176, 186, 225
 - failed 124, 204, 208
 - h 123, 186, 207
 - i 189, 204, 208
 - prédéfinie 79
 - s 123, 154, 186
 - t 204, 208

F

- fax 27, 33, 115
 - accusé de réception 117
 - app_fax (module) 115
 - configuration d'Asterisk 123
 - détection 187
 - envoi par courriel 185
 - formulaire 118
 - Hylafax 115

- pré-décroché 185
- ReceiveFAX (application) 115
- réception 185
- SendFAX (application) 115
- SpanDSP 17
 - sur IP 115
- FFmpeg 258
- fichier
 - de configuration 93
 - de Kamailio 246
 - source *voir* source
 - vocal 101, 105
 - configuration 102
 - enregistrement 104
- file d'attente 166, 179
- filtrage
 - débogage 237
 - réseau 64
- Firebug 210
- fork 14
- forking SIP 15
- FreeBSD 40
- FreeRADIUS 221
- FreeSWITCH 17, 18, 131

G

- G711 A-Law (codec) 61
- generic bridge (mode) 69, 234
- gestion
 - de présence 28, 109, 258
 - des comptes utilisateurs 134
- gestionnaire d'incidents 275
- Ghostsript 120
- gigue 36
- Git 272
- GNU GPL (GNU General Public License) 23
- GnuGK 9
- GoogleTalk 259, 262
 - pilote 237

H

- H.323 (protocole) 8, 18

I

- IAX (Inter-Asterisk eXchange) 15, 19, 131

- ICE (Interactive Connectivity Establishment) 149, 259, 265
 - ICMP (Internet Control Message Protocol) 237
 - IETF (Internet Engineering Task Force) 109
 - IKEv2 (Internet Key Exchange Protocol) 134
 - ImageMagick 120
 - IMS (IP Multimedia Subsystem) 16, 19
 - installation
 - d'Asterisk 39, 91, 144, 160
 - chez un particulier 137
 - de DAHDI 43, 89, 158
 - de Kamailio 244
 - de libpri 51, 91
 - interconnexion
 - de sites distants 28, 241
 - interopérabilité 24, 31, 32, 65, 241
 - interruption inattendue 275
 - IP (Internet Protocol) 1, 7
 - IPBX, IP-PBX (PBX sur IP) 7
 - IPSec (protocole) 133
 - tunnel 243
 - iptables 151
 - IPv6 256
 - IRC (Internet Relay Chat) 274
 - ISAKMP (Internet Security Association and Key Management Protocol) 134
 - ISDN (Integrated Services Digital Network)
 - voir* RNIS
 - BRI (Basic Rate Interface) 41
 - PRI (Primary Rate Interface) 41
 - IVR (Interactive Voice Response) *voir* SVI
- J**
- Jabber (protocole) 10, 19, 109, 258
 - JabberSend (application) 262
 - JID (Jabber Identifier) 259
 - Jingle (protocole) 19, 109, 259, 262
 - pilote 237
 - journal d'activité 215
 - archivage 216
 - configuration 215
 - débogage 228
 - diagnostic 228
- K**
- Kamailio 14, 243
 - configuration 246
 - débogage 247
 - installation 244
 - interconnexion de sites 241
 - interopérabilité 32
 - modes stateful et stateless 249
- L**
- langage
 - AEL 75
 - C 40
 - de Kamailio 247
 - LUA 75
 - Perl 42
 - PHP 118
 - langue
 - d'un terminal 102
 - des menus vocaux 101
 - latence 130
 - LDAP (Lightweight Directory Access Protocol) 31
 - authentification des utilisateurs 117
 - ldapsearch (programme) 182, 183, 187
 - LED *voir* DEL
 - LGPL (Lesser GPL) 23
 - libpri (bibliothèque) 40, 43, 91
 - installation 51
 - licence 16, 23
 - double~ 40
 - ligne de commande 39
 - Linux 39, 40
 - distribution 41
 - version du noyau 41
 - log *voir* journal d'activité
 - logiciel libre 25
 - logrotate (programme) 216
 - LUA (format) 75
- M**
- Mac OS X 40
 - Makefile 55
 - mandataire *voir* proxy

Asterisk

- matériel 29, 41
- MediaProxy 253
- MeetMe (application) 9, 103, 163
 - DAHDI 43
- menu vocal 101
- message
 - SIP 232
 - vocal
 - enregistrement 104
- messagerie
 - instantanée 10, 19, 28, 109, 258
 - protocoles 10
 - unifiée 10
 - vocale 9, 27, 105
 - activation de la boîte 107
 - consultation 113
 - consultation web 112
 - renvoi vers 112
- MGCP (Media Gateway Control Protocol) 15, 19
- mISDN 51, 88
- module
 - d'Asterisk 55, 58, 92
 - app_dial 207
 - app_dumpchan 229
 - app_meetme 207
 - app_userevent 207
 - cdr_csv 218
 - cdr_custom 218, 224
 - cdr_radius 218, 221
 - chan_gtalk 263
 - chan_jingle 263
 - chan_sip 237
 - choix 56
 - func_odbc 180
 - func_shell 180
 - installation 214
 - répertoires 59
 - res_jabber 260
 - de DAHDI 43, 50
 - dahdi_dummy 157
 - wcte12xp 47
 - de Kamailio 244
 - sl (stateless) 248
 - tm (Transaction Module) 249
 - driver 45
- Mu-Law (codec) 61
- multimédia 265
- MWI (Message Waiting Indicator) 73, 108
- N**
- Nagios 240
- NAT (Network Address Translation) 15, 53, 74, 137
 - dysfonctionnements 233
 - types de ~ 138
- native bridge (mode) 69, 234
- Netiquette 273
- NGN (Next Generation Network) 16, 24
- NoOp (application) 78, 80
- notification
 - d'appel 124, 261, 267
 - de message en attente 73, 108
 - par envoi de courriel 111
 - par MWI 108
 - des appels reçus 27
 - hint 127
- numéro SDA 21
- NVFXDetect (application) 187
- O**
- ODBC (Open DataBase Connectivity) 43
- OpenBSD 40
- Openfire 260, 267
- OpenH323 9, 18
- OpenPBX.org 16
- OpenSER 14, 244
- OpenSIPS 14, 32
- P**
- P2P bridge (mode) 69, 234
- PABX (Private Automatic Branch eXchange) 7, 30
- pare-feu 213
 - débogage 237
- passerelle
 - SIP/RTC 11
 - vers le RTC 11
 - voir aussi* NAT

- patch 276
 - PBX *voir* PABX
 - PDF (Portable Document Format) 118
 - Perl (langage) 42, 219
 - PHP (langage) 118, 167, 169, 197
 - fonctions d'audioconférence 170, 172
 - LDAP 119
 - scripts 202, 208
 - PIN (Personal Identification Number) 106, 155
 - plan de numérotation *voir aussi* dialplan 11, 33, 74
 - Playback (application) 81, 103
 - PoE (Power over Ethernet) 35
 - poste téléphonique 33
 - intégration 63
 - ST2030 68, 72, 108, 125
 - protocole 18, 263
 - ARP 256
 - de ToIP propriétaire 65
 - DHCP 72
 - DTLS 257
 - H.323 8, 18
 - HTTP 73
 - IAX 15, 19, 131
 - IAX2 74
 - ICE 149
 - ICMP 237
 - IKEv2 134
 - IP 7
 - IPSec 133
 - ISAKMP 134
 - Jabber 258
 - Jingle 19, 109, 259, 262
 - MGCP 15, 19
 - QSIG 31
 - RADIUS 34, 222
 - RTP 61
 - SCTP 17
 - SDP 141
 - SIP 8, 18, 65
 - SMTP 10
 - SRTP 133, 257
 - SSL 17
 - STUN 237
 - T.38 115
 - TCP 17
 - TFTP 72
 - TLS 17
 - UDP 53, 133, 237
 - XMPP 10, 109, 258
 - ZRTP 133
 - Prototype (API Ajax) 209
 - proxy 13
 - reverse ~ 206
 - RTP 247
 - SIP 14, 15, 243
- Q**
- QSIG (protocole) 31
- R**
- RADIUS
 - client 221, 222
 - dictionnaire 223
 - protocole 34
 - serveur 34
 - rasterisk (programme) 165, 179
 - ReceiveFAX (application) 187
 - Record (application) 11, 104
 - registrar 15
 - réinstallation (d'Asterisk) 117
 - renvoi d'appel 27
 - réseau
 - adressage 53
 - architecture 63
 - filtrage 64
 - local virtuel *voir* VLAN
 - numérique à intégration de services *voir* RNIS
 - téléphonique commuté *voir aussi* RTC 7
 - réunion téléphonique *voir aussi*
 - audioconférence 11
 - reverse proxy 206
 - RFC (Request For Comment) 9
 - 1855 273
 - 1918 141
 - 3261 15, 232, 237

Asterisk

- 3489 150
 - 3581 237
 - 3920 259
 - 3921 259
 - 4568 257
 - 5389 150
 - 5456 19, 131
 - Ringin (application) 186
 - RNIS (réseau numérique à intégration de services) 30, 41
 - ISDN PRI 88
 - S0/T0 41, 88
 - S2/T2 41
 - routage
 - des appels 11
 - des messages SIP 247
 - routeur filtrant 213
 - RTC (réseau téléphonique commuté) 7
 - connexion 11
 - RTP (Real Time Protocol) 61, 64
 - RTPProxy 247, 253
- S**
- S0/T0 *voir* ISDN BRI
 - S2/T2 *voir* ISDN PRI
 - SA (Security Associations) 134
 - safe_asterisk (programme) 228, 275
 - SBC (Session Border Controller) 16, 144
 - SCTP (Stream Control Transmission Protocol) 17
 - SDA (sélection directe à l'arrivée) 21
 - SDES (Security Descriptions) 257
 - SDP (Session Description Protocol) 141
 - SDSL (Symetric Digital Subscriber Line) 35
 - sécurité 213
 - accès distant 132
 - authentification 134
 - challenge-response 134
 - confidentialité 132
 - contrôle d'accès 136
 - dysfonctionnements 233
 - SendFAX (application) 185
 - SER (SIP Express Router) 9, 14
 - serveur
 - administration 213
 - Apache 117, 167, 197
 - architecture réseau 63
 - connexion 161
 - d'audioconférences 9, 157
 - d'enregistrement SIP *voir* registrar
 - ejabberd 260
 - Kamailio 251
 - Openfire 260, 267
 - principal 219
 - RADIUS 34
 - sécurité 213
 - STUN 259
 - vocal interactif *voir* SVI
 - XMPP 260
 - SFLPhone 132
 - SIP (module) 146
 - SIP (Session Initiation Protocol) 8, 18
 - authentification 134
 - B2BUA 15
 - branche 249
 - débogage 231
 - dialogue 232, 249
 - forking ~ 15
 - interconnexion 242
 - interopérabilité 242
 - NAT 233
 - pour les postes physiques 65
 - proxy 15
 - received et rport 237
 - requête 232
 - SDP 141
 - service de l'opérateur Free 146
 - transaction 15, 232, 249
 - trunk 98, 157, 231
 - URI 15
 - SIPp 240
 - Sipsak (SIP Swiss Army Knife) 240
 - SIPX 18
 - SMTP (Simple Mail Transfer Protocol) 10
 - softphone 18, 27, 132, 152, 188
 - appel par composition d'adresse de courriel 188
 - SFLPhone 132

- Zoiper 131, 132
- source (fichiers)
 - d'Asterisk 42, 54
 - de DAHDI 44
 - du noyau Linux 42, 45
 - sécurité 132
- spandsp (bibliothèque) 116
- Spencer, Mark 9
- SRTP (Secured RTP) 133, 257
- SSL (Secure Sockets Layer) 17
- stateful (mode) 249
- stateless (mode) 249
- STUN (Session Traversal Utilities for NAT) 150, 237, 259
- Subversion 272
- Sun Solaris 40
- supervision
 - DAHDI 238
 - de ligne 109, 125
 - outils externes 240
- SVI (serveur vocal interactif) 10, 101, 266
- syslog (programme) 215
- système
 - d'exploitation 40
 - d'information *voir aussi* CTI 24, 27, 179
- T**
- T.38 (protocole) 115
- TCP (Transmission Control Protocol) 17
- tcpdump (programme) 237
- téléphone logiciel *voir* softphone
- téléphonie
 - services 101
 - sur IP *voir* ToIP
- telnet (programme) 192
- TFTP (Trivial File Transfer Protocol) 72
- ticket de taxation 34, 84, 122, 217
 - contrôle d'accès 217
 - interfaces d'enregistrement 217
 - voir aussi* CDR
- TIFF (Tagged Image File Format) 120
- TLS (Transport Layer Security) 17
 - Kamailio 244
- ToIP (Telephony over IP) 1, 7
 - architectures 241
 - logiciels libres 24
 - protocoles 15, 18
 - propriétaires 65
 - traçabilité 222
 - traduction d'adresses réseau (*voir* NAT)
 - transaction SIP 249
 - travail collaboratif 28
 - trunk (*voir* version de développement) 272
 - trunk SIP 98, 157, 231
 - TShark (programme) 237
- U**
- UDP (User Datagram Protocol) 53, 133, 237
- Unix 40
- URI (Universal Resource Identifier) 15
- UserEvent (application) 207
- utilisateur
 - authentification 117
 - compte 134
- uuencode (programme) 186
- V**
- variable 103
 - CALLERID 83
 - CDR 226
 - dans le dialplan 83
 - d'environnement 83
 - de canal 80, 83
 - globale 83
 - partagée 83
 - EXTEN 80
 - LDAPNUM 189
 - UNIQUEID 187
- version 270
 - de développement 272
 - stable 270
- vidéo 28, 257
 - FFmpeg 258
- VLAN (Virtual Local Area Network) 37, 63
- Voicemail (application) 103, 106, 113
- VoicemailMain (application) 106, 113
- VPN (Virtual Private Network) 33, 133
 - IPSec 132, 243