

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

Configuration du serveur SIP Asterisk

I Présentation

I.1 PABX

Un PABX (Private Automatic Branch eXchange) sert principalement à relier les postes téléphoniques d'un établissement (lignes internes) avec le réseau téléphonique public (lignes externes). Il permet en plus la mise en œuvre d'un certain nombre de fonctions, notamment :

1. relier plus de lignes internes qu'il n'y a de lignes externes ;
2. permettre des appels entre postes internes sans passer par le réseau public ;
3. programmer des droits d'accès au réseau public pour chaque poste interne ;
4. proposer un ensemble de services téléphoniques (conférences, transferts d'appel, renvois, messagerie, appel par nom...) ;
5. gérer les SDA (sélection directe à l'arrivée) ;
6. gérer la ventilation par service de la facture téléphonique globale ;
7. apporter des services de couplage téléphonie-informatique (CTI) tels que la remontée de fiche essentiellement via le protocole CSTA ;
8. gérer les appels d'urgence dans les structures d'accueil hospitalières, maisons de retraite, etc. ;
9. gérer un portier interphone d'immeuble et commander une gâche électrique.

I.2 VoIP

La téléphonie sur IP (ou VoIP pour Voix sur IP) est un mode de téléphonie utilisant le protocole de télécommunications créé pour Internet (IP pour Internet Protocol).

Le principe est le suivant :

1. Le signal audio (capté à l'aide du microphone), est converti dans un format numérique par un programme appelé codec qui ensuite le compresse.
2. La trame numérique ainsi obtenue est véhiculée grâce au protocole IP à travers le réseau internet.
3. À l'autre bout de la chaîne (au niveau de l'interlocuteur), l'opération inverse est effectuée : décompression puis reconstitution du signal analogique audio.

I.3 Compte SIP

La VoIP passe entre autres par la prise en charge d'un protocole standard, ouvert et très largement utilisé, le SIP (Session Initiation Protocol). SIP est un protocole très proche d'HTTP qui n'est pas limité à la seule voix mais qui prend aussi en charge la vidéo et la messagerie instantanée. Il est actuellement le protocole le plus utilisé pour la téléphonie par internet ou Voix sur IP (VoIP).

Le principal avantage d'ouvrir un compte SIP gratuit ou trunk SIP (SIP trunking) chez un opérateur VoIP, est qu'il est possible d'appeler gratuitement tous les autres utilisateurs d'autres fournisseurs SIP dans le monde en composant simplement leurs adresses SIP, à condition que le réseau SIP de l'opérateur soit ouvert.

Le seul acteur majeur n'utilisant pas le protocole SIP est le service « Skype », ce qui le prive d'être compatible avec la plupart des matériels, logiciels et applications SIP du marché. Mieux vaut donc choisir un fournisseur VoIP compatible SIP afin de téléphoner par Internet depuis son ordinateur, un téléphone IP, standard IP (IPBX) ou un smartphone, ou tout autre moyen à venir. Il est vivement conseillé d'utiliser un service SIP autre que celui du FAI de la maison afin d'éviter les risques de fraudes sans limite et pour une plus grande souplesse d'utilisation et de services.

Le service SIP permet d'utiliser votre ligne téléphonique de votre fournisseur à partir de n'importe quel ordinateur connecté à Internet disposant d'un micro et de haut-parleurs.

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

I.4 Asterisk

Asterisk permet de transformer un ordinateur en commutateur téléphonique performant. Il se présente sous la forme d'un logiciel libre édité par la société américaine Digium. Asterisk est un IPBX, c'est -à-dire un PABX qui est juste connecté sur Internet. D'un point de vue fonctionnalité, Asterisk permet tout ce qu'on peut attendre d'un PABX moderne, et plus particulièrement dans notre contexte :

- x La prise en charge de tous vos comptes SIP (et pas seulement, Skype et Gtalk aussi) ;
- x La gestion des postes téléphoniques sur IP locaux, téléphones physiques ou logiciels (ou SoftPhone) comme Ekiga ou csipsimple ;
- x La téléphonie de poste à poste en local, comme le permettent les téléphones DECT ;
- x Un nombre illimité de boîtes vocales, et en français ;
- x La mise en attente et le transfert d'un appel, d'un poste local à l'autre.

II Installation et configuration de base

II.1 Installation

Pour installer Asterisk, il suffit d'installer deux paquets : asterisk et asterisk-prompt-fr-armelle grâce à la commande :

```
root@debian:~# apt-get install asterisk asterisk-prompt-fr-armelle
```

Spécifier 33 pour le code pays si nécessaire.

Le paquet asterisk-prompt-fr-armelle permet d'avoir des voix en français et non en anglais pour la messagerie.

Pour vérifier que Asterisk a bien été installé, vous pouvez utiliser la commande suivante :

```
# asterisk -vrc
*CLI> quit
#
```

Suite à cette installation, les répertoires suivants ont été créés :

- x **/usr/sbin/asterisk/** contient les fichiers binaires d'Asterisk
- x **/etc/asterisk/** contient les fichiers de configuration d'Asterisk
- x **/usr/lib/asterisk/modules/** contient les modules d'application, codecs, formats, événement sonore ...
- x **/var/lib/asterisk/moh/** contient les musique d'attente
- x **/var/lib/asterisk/** contient la base de données d'Asterisk
- x **/var/spool/asterisk/** contient des informations sur la boîte vocale ou l'accueil des utilisateurs
- x **/var/log/asterisk/** contient les journaux d'Asterisk
- x **/var/log/asterisk/cdr-csv** contient les détails sur les appels enregistrés dans le format CSV.

II.2 Sauvegarde des fichiers de configuration

La configuration d'Asterisk s'articule sur les fichiers de configuration suivants :

- x **/etc/asterisk/sip.conf** : configuration globale d'Asterisk
- x **/etc/asterisk/users.conf** : configuration des utilisateurs
- x **/etc/asterisk/extensions.conf** : configuration du plan de numérotation (Dialplan)
- x **/etc/asterisk/features.conf** : configuration de fonctions spécifiques (parcage d'appel, transfert d'appel, ...)
- x **/etc/asterisk/voicemail.conf** : configuration de la messagerie vocale

Avant toute modification des fichiers, vous devez en faire une copie de sauvegarde :

```
root@debian:~# cp /etc/asterisk/sip.conf /etc/asterisk/sip.conf.orig
root@debian:~# cp /etc/asterisk/users.conf /etc/asterisk/users.conf.orig
root@debian:~# cp /etc/asterisk/extensions.conf /etc/asterisk/extensions.conf.orig
root@debian:~# cp /etc/asterisk/iax.conf /etc/asterisk/iax.conf.orig
root@debian:~# cp /etc/asterisk/features.conf /etc/asterisk/features.conf.orig
root@debian:~# cp /etc/asterisk/voicemail.conf /etc/asterisk/voicemail.conf.orig
```

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

Ces copies permettent de revenir en arrière en cas d'erreur.

II.3 Configuration de base

La première étape est d'éditer le fichier « **/etc/asterisk/sip.conf** », chercher la ligne :

```
language=en
```

Et la transformer en :

```
language=fr
```

C'est tout pour la configuration de base du logiciel, les valeurs entrées par défaut sont suffisantes. Le reste de la configuration se fera dans d'autres fichiers.

III La console Asterisk

Le serveur Asterisk permet d'interagir directement avec le système sans avoir à modifier les fichiers de configuration avec la CLI « Interface de ligne de commande ».

Le lancement d'Asterisk se fait grâce à la commande :

```
asterisk -[hcvrx]
```

Les options sont :

- x -h : pour avoir la commandes complète d'options
- x -c : permet de se connecter au CLI d'Asterisk
- x -v : verbosité, utiliser une ou plusieurs fois (-vvvv), elle permet de paramétrer la quantité de sortie.
- x -g : Core dump. Si Asterisk plantait subitement un fichier core serait alors créer. Ensuite avec gdb on pourrait voir par quoi il a été causé.
- x -r : pour se reconnecter à un processus d'Asterisk déjà lancé en local.
- x -x : utiliser en combinaison avec -r permet d'exécuter une commande sans avoir à se reconnecter à la CLI et la taper manuellement.

Exemples :

- x Connexion au CLI en mode un peu verbeux

```
root@debian:~# asterisk -vvvc
```

- x Exécution d'une commande sans avoir besoin de passer par le CLI

```
root@debian:~# asterisk -rx "restart now"
```

- x Connexion à un processus d'Asterisk déjà lancé en local

```
root@debian:~# asterisk -r
```

Ou bien

```
root@debian:~# rasterisk
```

Attention : Lors de tout changement de configuration dans un des fichiers de configuration d'asterisk, ne pas oublier de recharger la configuration afin de prendre en compte les modifications :

```
root@debian:~# service asterisk restart
```

ou

```
root@debian:~# asterisk -vrc
```

```
*CLI> reload
```

```
*CLI> quit
```

Remarque : ne pas hésiter à utiliser la commande "sudo asterisk -rvvvvv" afin de déceler les éventuels problèmes de configuration dans les fichiers d'Asterisk.

IV Configuration standard d'Asterisk

La configuration d'Asterisk comprend :

- x La création des comptes SIP (utilisateurs/postes) ;
- x La configuration du traitement des appels ;

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

- x La configuration spécifique si le réseau a deux serveurs Asterisk afin de pouvoir transférer les appels d'un serveur vers l'autre ;
- x Des configurations spécifiques comme le transfert d'appel.

IV.1 Ajout des comptes SIP

Pour ajouter un téléphone sur votre réseau de téléphonie IP, il faut dans un premier temps ajouter un utilisateur sur votre PBX Asterisk. Chaque poste de téléphone sera associé à un utilisateur.

Pour cela, vous devez éditer le fichier « **/etc/asterisk/users.conf** ».

Ce fichier a le même format qu'un fichier ini. Les fichiers ini sont composés de sections reconnaissables par leur nom entre [] suivi des paramètres pour cette section au format « <nom du paramètre> = <valeur du paramètre> ».

À la fin de ce fichier, il faut ajouter chaque compte SIP selon le modèle ci-dessous :

```
[600]           Compte = numéro SIP (de téléphone) dans ce cas
type = friend   Type d'objet SIP (friend = utilisateur qui peut appeler et être
appelé)
host = dynamic  Vous pouvez vous connecter à ce compte SIP de n'importe quelle
adresse IP
callerid = "J.Doe"  identifiant à afficher sur le téléphone appelé
secret = mdpsecret Mot de passe SIP obligatoire
context = work    Contexte des appels que passe l'utilisateur (exploité par le fichier
extensions.conf)
```

Les textes en rouge seront à modifier en fonction du compte ajouté. Un compte peut être associé à une personne ou un téléphone situé dans un emplacement particulier.

Les « ; » sont à mettre pour ajouter des commentaires à la suite de la ligne.

Il faut ensuite demander au serveur Asterisk de prendre en compte ces modifications en exécutant la commande CLI :

```
*CLI> sip reload
```

Vous pouvez vérifier le résultat grâce à la commande CLI :

```
*CLI> sip show users
```

IV.2 Contexte des appels

L'objectif de ce contexte des appels est de définir les actions qui seront exécutées suite à un appel, par exemple raccrochage au bout de 30 secondes ou bien renvoi vers une messagerie.

Le fichier « **/etc/asterisk/extensions.conf** » définit le dialplan, c'est à dire le plan de la numérotation. On explicite l'action qui se déroulera si on compose un numéro, grâce à la description des contextes.

Chaque description de contexte est constitué d'une section dont le nom est celui du contexte et d'instructions exten. Les instructions exten sont utilisées comme suit :

	Numéro appelé	Ordre d'instruction	Action effectuée	Temps en secondes avant de passer à l'instruction suivante (en fonction de l'action)
exten =>	200,	1,	Dial(SIP/John,	10)

Dans cet exemple, les appels arrivant sur le serveur Asterisk à destination du numéro de téléphone 200 sont envoyés vers le téléphone de John pendant 10 secondes puis sont envoyés sur la messagerie de John.

Exemple 1 :

```
[local]
exten => 205, 1, Dial(SIP/John, 10)
exten => 205, 2, Dial(SIP/Fred, 10)
```

Les appels arrivant sur le serveur Asterisk à destination du numéro 205 sont envoyés vers le téléphone de John puis vers le téléphone de Fred.

Si vous souhaitez configurer tous les postes avec un même contexte, par exemple tous les postes de 600 à 699, vous pouvez mettre _6XX (XX indiquant n'importe quelle valeur) comme dans l'exemple qui suit.

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

Exemple 2 :

```
[work] ; nom du contexte
exten => _6XX,1,Dial(SIP/${EXTEN},20)    si on demande un numéro à trois chiffres
commençant par 6 (6XX), on l'appelle pendant 20 secondes, si personne ne décroche, on
passe à la ligne suivante
exten => _6XX,2,Hangup()                on arrive ici de la ligne précédente, on raccroche
```

Explications sur Dial(SIP/\${EXTEN},20)

- x **Dial** : application qui va être utilisé
- x **SIP** : Protocol qui va être utilisé
- x **\${EXTEN}** : variable de l'extension composé, si on appelle le 601, la variable \${EXTEN} prendra comme valeur **601**
- x **20** : temps d'attente avant de passer à l'étape suivante.

Donc la ligne **exten => _6XX,1,Dial(SIP/\${EXTEN},20)** se traduit par : quand on compose le numéro (par exemple) 601, on appelle le numéro 601 et si au bout de 20 secondes il n'y a pas de réponse, on passe à la ligne du dessous.

Dans le cas du numéro 601 la ligne devient comme ceci : **exten => 601,1,Dial(SIP/601,20)**, mais l'avantage de la ligne précédente est qu'elle permet d'appeler les numéros de 600 à 699.

La seconde ligne : **exten => _6XX,2,Hangup()** permet de raccrocher si il n'y a pas de réponse au bout des 20 secondes.

Il faut ensuite demander au serveur Asterisk de prendre en compte ces modifications en exécutant la commande CLI :

```
*CLI> dialplan reload
```

Vous pouvez vérifier le résultat grâce à la commande CLI :

```
*CLI> dialplan show
```

Si vous ne voulez visualiser qu'un contexte, il suffit de l'ajouter à la fin de la commande, par exemple pour le contexte travail :

```
*CLI> dialplan show travail
```

IV.3 Configuration des téléphones

L'étape suivante est de connecter au réseau et de configurer les postes téléphoniques. Après les avoir connectés au même réseau que le serveur, grâce à la documentation du téléphone, vous pouvez effectuer les étapes suivantes pour les configurer :

1. Configuration de l'adresse IP du téléphone ;
2. Connexion au téléphone à partir d'un ordinateur grâce à un navigateur WEB ;
3. Configuration des autres paramètres réseau ;
4. Enregistrement du téléphone par le serveur.

Les explications sur la configuration des téléphones se trouvent dans les documentations sur ces téléphones.

Après l'enregistrement, vous pouvez vérifier qu'il a bien été pris en compte grâce à la commande CLI suivante :

```
*CLI> sip show peers
```

Name/username	Host	Dyn	Forcerport	Comedia	ACL	Port	Status	Description
1001/s	Unspecified	D	Yes	Yes	11475	UNKNOW		
1002/s	192.168.11.1	D	Yes	No	A	1024	OK (29 ms)	
1003/s	192.168.11.2	D	Yes	No	5060	OK (14 ms)		

3 sip peers [Monitored: 2 online, 1 offline Unmonitored: 0 online, 0 offline]

Cet exemple nous donne les informations suivantes :

- x Le poste « 1001 » n'est pas enregistré sur le serveur Asterisk :
 - ✓ Son adresse IP n'est pas connue : elle vaut « Unspecified »
 - ✓ Son état est à « UNKNOW »
- x Le poste « 1002 » est bien enregistré sur le serveur Asterisk :

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

- ✓ Son état est à « OK »
- ✓ Son adresse IP est 192.168.11.1
- ✓ Sa latence SIP est de « 29ms »
- x Le poste « 1003 » est bien enregistré sur le serveur Asterisk :
 - ✓ Son état est à « OK »
 - ✓ Son adresse IP est 192.168.11.2
 - ✓ Sa latence SIP est de « 14ms »

La commande suivante permet de lister les paramètres détaillés d'un compte SIP :

```
*CLI> sip show peer 1001
```

Maintenant que les postes téléphoniques sont enregistrés, vous pouvez passer des coups de fil entre les différents postes et visualiser les communications en cours grâce à la commande suivante :

```
*CLI> core show channels
```

```
ChannelLocation State Application(Data)
SIP/1001-00009867 s@user:36 Up Dial(SIP/1002,15,hHTT)
SIP/1002-00009876 (None) Up AppDial((Outgoing Line))
2 active channels
1 active call
4012 calls processed
```

La lecture du résultat est la suivante :

- x Les postes « 1001 » et « 1002 » sont en communication (l'un avec l'autre)
- x C'est le poste « 1001 » qui appelle l'autre poste (la dernière application lancée étant « Dial(SIP/1002,15,hHTT) » qui se traduit par « appeler le poste SIP 1002, laisser sonner 15 secondes avant de passer à l'étape suivante du dialplan s'il ne décroche pas »).
- x On observe qu'il y a 1 appel en cours (1001 vers 1002) et qu'il y a eu 4012 appels passés depuis que le service Asterisk est démarré.

Pour connaître le détail d'une communication en cours, la commande est la suivante :

```
*CLI> core show channel SIP/1001-00009867
```

V Configurations supplémentaires

V.1 Transfert d'appel

Le transfert d'appel permet de rediriger un appel vers un autre poste. Il existe deux types de transfert d'appel : celui dit « aveugle » et le « supervisé ». Le transfert d'appel dit aveugle est le fait de transférer un appel directement vers un autre poste. Le transfert d'appel dit supervisé est le fait d'appeler le destinataire du transfert avant de lui transférer l'appel.

Aveugle	Supervisé
1. 1 appelle 2 2. 1 et 2 sont en communication 3. 2 transfère à 3 (transfert aveugle) 4. 1 et 3 sont en communication 5. 2 est raccroché	1. 1 appelle 2 2. 1 et 2 sont en communication 3. 2 appelle 3 (transfert supervisé) 4. 1 est en attente 5. 2 et 3 sont en communication 6. 2 raccroche 7. 1 et 3 sont en communication

Le fichier à modifier pour activer les transferts d'appels est « **/etc/asterisk/features.conf** ». Dans ce fichier, il faut trouver les options suivantes et les modifier :

```
blindxfer => ## la combinaison de touches pour le transfert aveugle est ##
atxfer => ** la combinaison de touches pour le transfert supervisé est **
```

Il faut ensuite aller modifier la ligne correspondant au Dial dans le fichier « **/etc/asterisk/extensions.conf** » et ajouter l'option tT à la fin :

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

```
[work]
exten => _6XX,1,Dial(SIP/${EXTEN},20,tT)
exten => _6XX,2,Hangup()
```

Il faut recharger les fichiers de configuration par la commande :

```
*CLI> reload
```

Puis il est possible d'effectuer un transfert d'appel aveugle en appuyant sur les touches '##' et un transfert d'appel supervisé en appuyant sur les touches '**'.

V.2 Parking d'appel

Le Parking Call est la possibilité de mettre en attente une personne sur un numéro de parking, c'est à dire que vous transférez cette personne sur un numéro qui lui est attribué, le temps d'effectuer une recherche par exemple. Il vous suffit de rappeler ce numéro à partir de n'importe quel poste pour reprendre l'appel.

Le fichier à modifier pour activer le parking d'appels est « **/etc/asterisk/features.conf** ». Dans ce fichier, il faut trouver les options suivantes et les modifier :

```
parkext => 700          Numéro à appeler pour mettre une personne dans le parking : 700
parkpos => 701-720      Numéros de places de parking disponibles
context => parkedcalls  Contexte à appliquer pour activer le parking d'appel
```

Il faut ensuite aller ajouter le contexte parkedcalls dans la section work du fichier « **/etc/asterisk/extensions.conf** » :

```
[work]
include => parkedcalls
exten => _6XX,1,Dial(SIP/${EXTEN},20,tT)
exten => _6XX,2,Hangup()
```

Il faut recharger les fichiers de configuration par la commande :

```
*CLI> reload
```

Il est possible d'effectuer un transfert d'appel aveugle vers le numéro 700, une voix nous indiquera alors le numéro dans lequel l'appel a été parqué. Pour récupérer la personne, il suffira d'appeler ce numéro.

V.3 Configuration des boîtes vocales

L'objectif de cette partie est de permettre à l'appelant, en cas d'absence du destinataire du coup de fil, de laisser un message.

Il faut tout d'abord éditer le fichier « **/etc/asterisk/voicemail.conf** » dans la partie « general » comme suit :

```
[general]
...
maxsilence=10          silence avant fin de l'enregistrement
silencethreshold=128  niveau de détection du silence
sendvoicemail=no       pas d'envoi de mail de notification
maxmsg=100             Nombre max de messages sur la Voicemail
maxsecs=0              Durée max d'un message. "0" = pas de limite
minsecs=20             Durée minimum d'un message
maxlogins=3            Nombre max d'erreur de login
review=no              Permet à l'appelant de réécouter son message avant de le transmettre
à la Voicemail. Accessible en terminant le message par #
saycid=no              Dicte le numéro de l'appelant avant de jouer le message qu'il a
laissé
```

Ensuite, il faut ajouter un contexte dans lequel mettre les utilisateurs :

```
[work]
601 => 1234,John DOE
602 => 1234,Albert DUPONT
```

Décortiquons la seconde ligne :

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

- x 601 => : Numéro de l'utilisateur
- x 1234 : Mot de passe pour accéder à la boîte vocale
- x John DOE: Prénom et nom de l'utilisateur

Maintenant nous allons éditer le fichier « **/etc/asterisk/extensions.conf** » pour configurer deux choses :

- x Le fait que au bout d'un certain temps Asterisk bascule sur la boîte vocale de l'utilisateur si celui-ci ne répond pas.
- x Création d'une extension pour créer le numéro qui servira à consulter la boîte vocale.

Dans le contexte d'appels, il faut remplacer la ligne :

```
exten => _6XX,2, Hangup()
```

Par celle-ci:

```
exten => _6XX,2,VoiceMail(${EXTEN}@work)
```

Cette ligne permet donc de de basculer sur la boîte vocale de l'utilisateur appelé s'il n'y a pas de réponse :

- x VoiceMail: On lance l'application Voicemail
- x \${EXTEN}@work : On récupère la numéro mis en variable et on se connecte a la boîte vocale associé du contexte work (comme précisé dans le fichier voicemail.conf)

Ensuite il faut ajouter la ligne suivante :

```
exten => 400,1,VoiceMailMain(${CALLERID(num)}@work)
```

Cette ligne sert à mettre en place un numéro pour que nos utilisateurs puissent consulter leurs boîtes vocales.

- x exten => : On déclare l'extension
- x 400 : On déclare le numéro 400 comme numéro de messagerie
- x 1 : Priorité
- x VoiceMailMain : On appelle l'application VoiceMailMain
- x \${CALLERID(num)} : On récupère le numéro de l'utilisateur qui appelle pour se connecter à sa boîte vocale
- x @work : on est toujours dans le contexte work

A partir de maintenant, vous devez exécuter un « reload » de votre serveur Asterisk :

```
*CLI> reload
```

Vous pouvez déposer des messages vocaux entre vos utilisateurs et faire en sorte qu'ils consultent leur messagerie en appelant le 400.

Lorsque que vous appelez le 600 le robot vous demandera le mot de passe de la boîte vocale spécifié dans le fichier voicemail.conf (1234).

Si vous ne souhaitez pas que vos utilisateurs aient à taper un mot de passe vous avez deux solutions:

- x Soit vous ne spécifiez pas de mot de passe dans le fichier voicemail.conf

Exemple:

```
[work]
6001 => ,jdoe
6002 => ,adupont
```

- x Soit vous rajoutez l'option « s » dans la déclaration du numéro de boîte vocale:

```
exten => 600,1,VoiceMailMain(${CALLERID(num)}@work,s)
```

V.4 Configuration si deux serveurs Asterisk

Le protocole IAX « Inter-Asterisk eXchange » est le protocole propre à Asterisk servant pour les trunks, c'est-à-dire les connexions entre des serveurs Asterisk.

Quand deux serveurs Asterisk sont utilisés, il faut préciser les deux serveurs, par exemple pour le premier, Asterisk1 (son adresse ip est dans l'exemple : 192.168.0.101) et l'autre Asterisk2 (@IP 192.168.0.102). On supposera que le premier serveur possède des numéros de téléphone de 600 à 699 et le second de 800 à 899. Le serveur concerné par les exemples suivants est le serveur Asterisk1.

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

Le premier fichier à modifier est « **/etc/asterisk/iax.conf** » :

x au début du fichier dans la section [general] :

```
register => NomServeur1:MotDePasseServeur1@AdresseIPServeur2
```

Exemple :

```
register => Asterisk1:PassAst1@192.168.0.102
```

Le serveur Asterisk1 va se connecter au serveur Asterisk2 (192.168.0.102) avec le nom d'utilisateur Asterisk1 et le mot de passe PassAst1.

x À la fin du fichier, vous allez autoriser au serveur distant de se connecter :

<p>Contenu à ajouter :</p> <p>[NomServeur2]</p> <p>username=NomServeur2</p> <p>secret=MotDePasseServeur2</p> <p>type=friend</p> <p>host=dynamic</p> <p>qualify=yes</p> <p>usercontext=from-NomServeur2</p> <p>context=from-NomServeur2</p> <p>deny=0.0.0.0/0.0.0.0</p> <p>permit=AdresseIPServeur2/255.255.255.255</p>	<p>Exemple :</p> <p>[Asterisk2]</p> <p>username=Asterisk2</p> <p>secret=PassAst2</p> <p>type=friend</p> <p>host=dynamic</p> <p>qualify=yes</p> <p>usercontext=from-Asterisk2</p> <p>context=from-Asterisk2</p> <p>deny=0.0.0.0/0.0.0.0</p> <p>permit=192.168.0.102/255.255.255.255</p>
---	---

Attention : il ne faut pas mettre d'espace dans les commandes.

La connexion du serveur Asterisk2 se fait avec le nom d'utilisateur Asterisk2 et le mot de passe PassAst2. La ligne « qualify » indique qu'il faut contrôler si l'utilisateur est disponible. Les lignes deny et permit indiquent qu'il accepte des appels entrants de l'adresse 192.168.0.102.

Pour que Asterisk prenne en compte ces modifications, il faut utiliser la commande CLI suivante :

```
*CLI> iax2 reload
```

Pour visualiser le résultat, il faut utiliser les commandes CLI suivantes :

```
*CLI> iax2 show registry
```

```
*CLI> iax2 show peers
```

Exemple :

```
Asterisk2*CLI> iax2 reload
Asterisk2*CLI> iax2 show registry
Host                dnsmgr Username      Perceived           Refresh  State
192.168.0.101:4569  N          Asterisk2  192.168.0.102:4569    60  Registe
red
1 IAX2 registrations.
Asterisk2*CLI> iax2 show peers
Name/Username      Host                Mask                Port                Status
Asterisk1/Aster  192.168.0.101      (D)  255.255.255.255  4569 (T)          OK (1 ms)
1 iax2 peers [1 online, 0 offline, 0 unmonitored]
```

Dans cet exemple, on constate que le serveur Asterisk1 (192.168.0.101) est bien enregistré et l'état de la connexion est « OK ».

Le second fichier est « **/etc/asterisk/extensions.conf** » :

```
[from-Asterisk2]
```

```
include => work ; pour le contexte from-Asterisk2, on utilise le contexte work
```

```
[work] ; idem au cas avec un seul serveur
```

```
exten => _6XX,1,Dial(SIP/${EXTEN},20)
```

```
exten => _6XX,2,Hangup()
```

```
exten => _8XX,1,Dial(IAX2/Asterisk1:PassAst1@Asterisk2/${EXTEN})
```

BTS S.N.I.R.	Document ressource
Lycée Jean Rostand Villepinte	Configuration du serveur SIP Asterisk

Pour appeler les lignes à 3 chiffres commençant par 8, il faut passer par le serveur nommé Asterisk2 et lui donner le username Asterisk1 et le mot secret PassAst1

Maintenant un téléphone IP configuré sur le premier serveur (numéro 600 à 699) pourra appeler un téléphone IP du second (numéro 800 à 899). Les numéros de téléphones sont à adapter en fonction de la carte des numéros associés à chaque serveur Asterisk. La description du contexte est la même que dans le cas précédent avec un seul serveur.

Il faut demander au serveur Asterisk de prendre en compte ces modifications en exécutant la commande Asterisk :

```
*CLI> dialplan reload
```

Très important : Configuration en cas de NAT

En cas d'utilisation de NAT entre les deux serveurs :

- x il faut autoriser le port UDP 4569 pour IAX lors de la configuration du routeur;
- x du côté réseau public, il faut remplacer dans le fichier « iax.conf », l'adresse de l'autre serveur Asterisk par l'adresse du routeur (adresse vue de l'extérieur).