

VOIP

La VOIP « Voice Over IP » est une technologie qui permet de transmettre la voix et d'autres données multimédias sur des réseaux IP, tel qu'internet, au lieu des réseaux traditionnels de téléphonie fixe. Cette technologie convertit la voix en données numériques et les transmet sous forme de paquets de données, offrant ainsi une alternative plus flexible et souvent plus économique aux systèmes de téléphonie traditionnels.

La VoIP utilise le protocole IP qui sert à faire transiter les informations sur le web. La voix est convertie en données, en code binaires, qui sont transmises et restituées lors d'un appel téléphonique, le code est transmis sous forme vocale. La VoIP est utilisable grâce à une connexion internet.

Cette téléphonie VoIP fonctionne grâce à une solution cloud, et est rendue possible par l'utilisation d'un logiciel spécifique. Elle peut être installée en quelques clics sur les appareils de l'entreprise, PC, Mac ou smartphones. Aucun besoin d'installation de matériel physique. Une fois le logiciel installé, il suffit de le configurer.

Les avantages de cette solution téléphonique en entreprise :

- La simplicité d'installation et d'utilisation du système. Plus besoin de standard physique qu'il faut installer et paramétrer, et plus besoin de formation spécifique à l'utilisation d'un standard téléphonique.
- Les économies réalisées sont aussi un atout non négligeable pour l'entreprise, et notamment pour les appels longue distance.
- La centralisation des relations clients et un meilleur suivi commercial. Avec la VOIP, on évite les recherches annexes et on peut partager les informations en temps réel.
- La VOIP s'adapte aussi parfaitement au travail à distance. Avec un appareil connecté, on peut passer des appels et accéder aux données, avec un partage d'informations en temps réel.
- SVI : Service Vocal Interactif, pour un accueil téléphonique professionnel. Un standard automatique aide à diriger les appelants vers la bonne personne ou le bon service, renvoyer les appels vers une boîte vocale ou ailleurs en dehors des heures de bureau.
- Cryptage des appels et sécurité : pour les informations et conversations confidentielles. Les technologies VoIP comme TLS et SRTP brouillent les données des appels, ce qui rend l'écoute clandestine presque impossible.
- L'entreprise peut conserver ses combinés téléphoniques traditionnels en utilisant un adaptateur spécial pour modifier les téléphones classiques existants. Elle peut aussi acheter des téléphones de bureau IP puisqu'ils ne sont pas chers.
- Les appels VoIP ont un son plus clair et plus net que ceux d'un téléphone fixe, selon la bande passante de la connexion fibre optique.

Installation et mise en service d'Asterisk

1. On installe le logiciel serveur **Asterisk** sur une VM, Debian 12, pour avoir moins d'instabilités.

Ensuite, on installe les paquets nécessaires avec la commande :

apt install git curl build-essential libedit-dev git curl wget libnewt-dev libssl-dev libncurses5-dev subversion libsqlite3-dev build-essential libjansson-dev libxml2-dev uuid-dev avahi-daemon libsrtplib2-dev

2. On change de répertoire pour installer asterisk : **cd /usr/src**

Puis, on télécharge asterisk sur le site officiel : **wget** (fichier compressé)

<http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-21-current.tar.gz>

Ensuite, on extrait les fichiers de l'archive avec la commande **tar xvf asterisk-21-current.tar.gz**

On supprime le fichier **rm asterisk-21-current.tar.gz**, et on change de répertoire **cd asterisk-21***

3. La compilation d'asterisk, on utilise la commande **./configure --with-srtp**, et pour télécharger des fichiers MP3 on utilise la commande **contrib/scripts/get_mp3_source.sh**

4. On configure les options, notamment pour choisir la langue française avec sa sélection dans le menu par les commandes **makemenuselect.makeopts**, **make menuselect**.

```
checking for ncurses.h... (cached) yes
checking for a sed that does not truncate output... /usr/bin/sed
checking for LIBXML2... yes
checking for GTK2... no
configure: creating ./config.status
config.status: creating makeopts
config.status: creating autoconfig.h
configure: Menuselect build configuration successfully completed
```

```

$$$$$$$$$$$$$$$$$=..
.7$7..      .7$7:..
.$$.      .7.7
.7.      7$$$
..$.      $$$$      .$$$7
..7$      .?      $$$$      .?      7$$$
$. $.      .$$$7. $$$$7 .7$$$      .$$$
.777.      .$$$$$77$$$77$$$$$7.      $$$
$$$~      .7$$$$$$$$$$$7.      .$$$
.7$7      .7$$$$$$$7:      ?$$$
$$$      ?7$$$$$$$$$I      .$$$7
$$$      .7$$$$$$$$$$$$$$$      :$$$
$$$      $$$$$$7$$$$$$$$$$$      .$$$
$$$      $$$ 7$$$7 .$$$      .$$$
$$$$      $$$7      .$$$
7$$$7      7$$$      7$$$
$$$$$      $$$
$$$7.      $ (TM)
$$$$$,      .7$$$$$ $$
$$$$$$$$$7$$$$$$$$$. $$$$
$$$$$$$$$.
```

```
configure: Package configured for:
configure: OS type : linux-gnu
configure: Host CPU : x86_64
configure: build-cpu:vendor:os: x86_64 : pc : linux-gnu :
configure: host-cpu:vendor:os: x86_64 : pc : linux-gnu :
root@debian:/usr/src/asterisk-21.2.0#
```

5. A la fin de la configuration, Après l'installation, on peut compiler Asterisk avec les commandes :

make

make install

make samples

make basic-pbx

make config

```
[LD] res_stasis_answer.o -> res_stasis_answer.so
[CC] res_stasis.c -> res_stasis.o
[CC] stasis/app.c -> stasis/app.o
[CC] stasis/command.c -> stasis/command.o
[CC] stasis/control.c -> stasis/control.o
[CC] stasis/messaging.c -> stasis/messaging.o
[CC] stasis/stasis_bridge.c -> stasis/stasis_bridge.o
[LD] res_stasis.o stasis/app.o stasis/command.o stasis/control.o stasis/messaging.o stasis/stasis
bridge.o -> res_stasis.so
[CC] res_stasis_device_state.c -> res_stasis_device_state.o
[LD] res_stasis_device_state.o -> res_stasis_device_state.so
[CC] res_stasis_playback.c -> res_stasis_playback.o
[LD] res_stasis_playback.o -> res_stasis_playback.so
[CC] res_stasis_recording.c -> res_stasis_recording.o
[CC] stasis_recording/stored.c -> stasis_recording/stored.o
[LD] res_stasis_recording.o stasis_recording/stored.o -> res_stasis_recording.so
[CC] res_stasis_snoop.c -> res_stasis_snoop.o
[LD] res_stasis_snoop.o -> res_stasis_snoop.so
[CC] res_statsd.c -> res_statsd.o
[LD] res_statsd.o -> res_statsd.so
[CC] res_stun_monitor.c -> res_stun_monitor.o
[LD] res_stun_monitor.o -> res_stun_monitor.so
[CC] res_timing_pthread.c -> res_timing_pthread.o
[LD] res_timing_pthread.o -> res_timing_pthread.so
[CC] res_timing_timerfd.c -> res_timing_timerfd.o
[LD] res_timing_timerfd.o -> res_timing_timerfd.so
[CC] res_tonedetect.c -> res_tonedetect.o
[LD] res_tonedetect.o -> res_tonedetect.so
Building Documentation For: channels pbx apps codecs formats cdr cel bridges funcs tests main res ac
tions
+----- Asterisk Build Complete -----+
+ Asterisk has successfully been built, and +
+ can be installed by running:           +
+                                     +
+               make install              +
+-----+
root@debian:/usr/src/asterisk-21.2.0# _
```

To direct input to this VM, click inside or press Ctrl+G

```

asterisk-moh-opsound-g72 100%[=====>] 7,56M 9,60MB/s ds 0,8s
2024-04-23 01:03:27 (9,60 MB/s) - « asterisk-moh-opsound-g722-2.03.tar.gz » sauvegardé [7924391/7924391]

make[1] : on quitte le répertoire « /usr/src/asterisk-21.2.0/sounds »
find rest-api -name "*.json" | while read x; do \
    /usr/bin/install -c -m 644 $x "/var/lib/asterisk/rest-api" ; \
done
+---- Asterisk Installation Complete -----+
+
+ YOU MUST READ THE SECURITY DOCUMENT +
+
+ Asterisk has successfully been installed. +
+ If you would like to install the sample +
+ configuration files (overwriting any +
+ existing config files), run: +
+
+ For generic reference documentation: +
+ make samples +
+
+ For a sample basic PBX: +
+ make basic-pbx +
+
+----- or -----+
+
+ You can go ahead and install the asterisk +
+ program documentation now or later run: +
+
+ make progdocs +
+
+ **Note** This requires that you have +
+ doxygen installed on your local system +
+-----+
root@debian:/usr/src/asterisk-21.2.0# make samples_

Installing file phoneprov/000000000000.cfg
Installing file phoneprov/000000000000-directory.xml
Installing file phoneprov/000000000000-phone.cfg
Installing file phoneprov/polycom_line.xml
Installing file phoneprov/polycom.xml
Installing file phoneprov/snom-mac.xml
root@debian:/usr/src/asterisk-21.2.0# make basic-pbx_

```

```

Installing file configs/samples/ss7.timers.sample
Installing file configs/samples/stasis.conf.sample
Installing file configs/samples/statsd.conf.sample
Installing file configs/samples/stir_shaken.conf.sample
Installing file configs/samples/test_sorcery.conf.sample
Installing file configs/samples/udptl.conf.sample
Installing file configs/samples/unistim.conf.sample
Installing file configs/samples/users.conf.sample
Installing file configs/samples/voicemail.conf.sample
Installing file configs/samples/xmpp.conf.sample
Updating asterisk.conf
/usr/bin/install -c -d "/var/spool/asterisk/voicemail/default/1234/INBOX"
build_tools/make_sample_voicemail "/var/lib/asterisk" "/var/spool/asterisk"
Installing file phoneprov/000000000000.cfg
Installing file phoneprov/000000000000-directory.xml
Installing file phoneprov/000000000000-phone.cfg
Installing file phoneprov/polycom_line.xml
Installing file phoneprov/polycom.xml
Installing file phoneprov/snom-mac.xml
root@debian:/usr/src/asterisk-21.2.0# make basic-pbx
Installing basic-pbx config files...
Installing file configs/basic-pbx/asterisk.conf
Installing file configs/basic-pbx/cdr.conf
Installing file configs/basic-pbx/cdr_custom.conf
Installing file configs/basic-pbx/confbbridge.conf
Installing file configs/basic-pbx/extensions.conf
Installing file configs/basic-pbx/indications.conf
Installing file configs/basic-pbx/logger.conf
Installing file configs/basic-pbx/modules.conf
Installing file configs/basic-pbx/musiconhold.conf
Installing file configs/basic-pbx/pjsip.conf
Config file configs/basic-pbx/pjsip_notify.conf is unchanged
Installing file configs/basic-pbx/queues.conf
Installing file configs/basic-pbx/README
Installing file configs/basic-pbx/voicemail.conf
Updating asterisk.conf
root@debian:/usr/src/asterisk-21.2.0# make config_

```

```

[LD] res_stasis_answer.o -> res_stasis_answer.so
[CC] res_stasis.c -> res_stasis.o
[CC] stasis/app.c -> stasis/app.o
[CC] stasis/command.c -> stasis/command.o
[CC] stasis/control.c -> stasis/control.o
[CC] stasis/messaging.c -> stasis/messaging.o
[CC] stasis/stasis_bridge.c -> stasis/stasis_bridge.o
[LD] res_stasis.o stasis/app.o stasis/command.o stasis/control.o stasis/messaging.o stasis/stasis
_stasis_bridge.o -> res_stasis.so
[CC] res_stasis_device_state.c -> res_stasis_device_state.o
[LD] res_stasis_device_state.o -> res_stasis_device_state.so
[CC] res_stasis_playback.c -> res_stasis_playback.o
[LD] res_stasis_playback.o -> res_stasis_playback.so
[CC] res_stasis_recording.c -> res_stasis_recording.o
[CC] stasis_recording/stored.c -> stasis_recording/stored.o
[LD] res_stasis_recording.o stasis_recording/stored.o -> res_stasis_recording.so
[CC] res_stasis_snoop.c -> res_stasis_snoop.o
[LD] res_stasis_snoop.o -> res_stasis_snoop.so
[CC] res_statsd.c -> res_statsd.o
[LD] res_statsd.o -> res_statsd.so
[CC] res_stun_monitor.c -> res_stun_monitor.o
[LD] res_stun_monitor.o -> res_stun_monitor.so
[CC] res_timing_pthread.c -> res_timing_pthread.o
[LD] res_timing_pthread.o -> res_timing_pthread.so
[CC] res_timing_timerfd.c -> res_timing_timerfd.o
[LD] res_timing_timerfd.o -> res_timing_timerfd.so
[CC] res_tonedetect.c -> res_tonedetect.o
[LD] res_tonedetect.o -> res_tonedetect.so
Building Documentation For: channels pbx apps codecs formats cdr cel bridges funcs tests main res ac
tions
+----- Asterisk Build Complete -----+
+ Asterisk has successfully been built, and +
+ can be installed by running:           +
+                                     +
+             make install               +
+-----+
root@debian:/usr/src/asterisk-21.2.0# _

```

To direct input to this VM, click inside or press Ctrl+G

6. On peut maintenant éditer le fichier **/etc/asterisk/modules.conf**

Puis ajouter : **load = res_srtp.so**

7. Lancer Asterisk et vérifier son status par les commandes :

service asterisk start

service asterisk status

```

load = app_directory.so
load = app_confbridge.so
load = app_queue.so

; Bridging

load = bridge_builtin_features.so
load = bridge_builtin_interval_features.so
load = bridge_holding.so
load = bridge_native_rtp.so
load = bridge_simple.so
load = bridge_softmix.so

; Call Detail Records

load = cdr_custom.so

; Channel Drivers

root@debian:/usr/src/asterisk-21.2.0# service asterisk start
root@debian:/usr/src/asterisk-21.2.0# service asterisk status
• asterisk.service - LSB: Asterisk PBX
   Loaded: loaded (/etc/init.d/asterisk; generated)
   Active: active (running) since Tue 2024-04-23 01:40:55 CEST; 23s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 39234 ExecStart=/etc/init.d/asterisk start (code=exited, status=0/SUCCESS)
    Tasks: 41 (limit: 2265)
   Memory: 32.6M
      CPU: 811ms
   CGroup: /system.slice/asterisk.service
           └─39246 /usr/sbin/asterisk

avril 23 01:40:54 debian systemd[1]: Starting asterisk.service - LSB: Asterisk PBX...
avril 23 01:40:55 debian asterisk[39234]: Starting Asterisk PBX: asterisk.
avril 23 01:40:55 debian systemd[1]: Started asterisk.service - LSB: Asterisk PBX.
root@debian:/usr/src/asterisk-21.2.0#

```

8. On fait la commande **ls** afin d'afficher les fichiers de configuration que nous aurons à modifier.


```

amd.conf
app_skel.conf
ari.conf
ast_debug_tools.conf
asterisk.adsi
asterisk.conf
asterisk.conf.old
asterisk.conf.origin
calendar.conf
ccss.conf
cdr_adaptive_odbc.conf
cdr_beanstalkd.conf
cdr.conf
cdr.conf.old
cdr_custom.conf
cdr_custom.conf.old
cdr_manager.conf
cdr_odbc.conf
cdr_pgsql.conf
cdr_sqlite3_custom.conf
cdr_tds.conf
cel_beanstalkd.conf
cel.conf
cel_custom.conf
cel_odbc.conf
cel_pgsql.conf
cel_sqlite3_custom.conf
cel_tds.conf
chan_dahdi.conf
chan_mobile.conf
cli_aliases.conf
cli.conf
cli_permissions.conf
codecs.conf
confbridge.conf
confbridge.conf.old
root@debian:/etc/asterisk#

dundi.conf
enum.conf
extconfig.conf
extensions.ael
extensions.conf
extensions.conf.old
extensions.lua
extensions_minivm.conf
features.conf
festival.conf
fichier.txt
followme.conf
func_odbc.conf
geolocation.conf
hep.conf
http.conf
iax.conf
iaxprov.conf
indications.conf
indications.conf.old
logger.conf
logger.conf.old
manager.conf
meetme.conf
minivm.conf
modules.conf
modules.conf.old
motif.conf
musiconhold.conf
musiconhold.conf.old
ooh323.conf
phoneprov.conf
pjproject.conf
pjsip.conf
pjsip.conf.old
pjsip.conf.origin

queues.conf.old
README
res_config_mysql.conf
res_config_odbc.conf
res_config_sqlite3.conf
res_corosync.conf
res_curl.conf
res_fax.conf
res_http_media_cache.conf
res_ldap.conf
res_odbc.conf
resolver_unbound.conf
res_parking.conf
res_pgsql.conf
res_snmp.conf
res_stun_monitor.conf
rtp.conf
say.conf
sla.conf
smdi.conf
sorcery.conf
ss7.timers
stasis.conf
statsd.conf
stir_shaken.conf
telcordia-1.adsi
test_sorcery.conf
udptl.conf
unistim.conf
users.conf
voicemail.conf
voicemail.conf.old
voicemail.conf.origin
xmpp.conf

```

9. On récupère les fichiers de configuration sur :

https://github.com/thierry-rami/Asterisk_config

mv /etc/asterisk/pjsip.conf /etc/asterisk/pjsip.conf.origin

mv /etc/asterisk/voicemail.conf /etc/asterisk/voicemail.conf.origin

cp /etc/asterisk/asterisk.conf /etc/asterisk/asterisk.conf.origin

fichier pjsip avec 3 utilisateurs

alice, numéro : 6001, mot de passe : bonjour

bob, numéro : 6002, mot de passe : bonjour

martin, numéro : 6003, mot de passe : bonjour

```
GNU nano 7.2 pjsip.conf
type=auth
auth_type=userpass

[aor_dynamic](!)
type=aor
max_contacts=1

;Definitions for our phones, using the templates above

[alice](endpoint_internal)
auth=alice
aors=alice
[alice](auth_userpass)
password=bonjour; put a strong, unique password here instead
username=alice
[alice](aor_dynamic)

[bob](endpoint_internal)
auth=bob
aors=bob
[bob](auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=bob
[bob](aor_dynamic)

[martin](endpoint_internal)
auth=martin
aors=martin
[martin](auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=martin
[martin](aor_dynamic)
-
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^- Aller ligne M-E Refaire
```

On retrouve sur le fichier pjsip.conf les trois utilisateurs que l'on a créé pour le test ;

alice, numéro : 6001, mot de passe : bonjour

bob, numéro : 6002, mot de passe : bonjour

martin, numéro : 6003, mot de passe : bonjour

On enregistre tous les utilisateurs sur le fichier pjsip.conf :

```
GNU nano 7.2 pjsip.conf *
[ bob ] (aor_dynamic)

[ martin ] (endpoint_internal)
auth=martin
aors=martin
[ martin ] (auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=martin
[ martin ] (aor_dynamic)

[ marie ] (endpoint_internal)
auth=marie
aors=marie
[ marie ] (auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=marie
[ marie ] (aor_dynamic)

[ jean ] (endpoint_internal)
auth=jean
aors=jean
[ jean ] (auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=jean
[ jean ] (aor_dynamic)

[ camille ] (endpoint_internal)
auth=camille
aors=camille
[ camille ] (auth_userpass)
password=bonjour ; put a strong, unique password here instead
username=camille
[ camille ] (aor_dynamic)

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich. ^_ Remplacer ^U Coller    ^J Justifier ^_ Aller ligne M-E Refaire
```

On fait de même sur le fichier extensions.conf :

```
GNU nano 7.2 extensions.conf
;!  
;! Automatically generated configuration file  
;! Filename: extensions.conf (/etc/asterisk/extensions.conf)  
;! Generator: Manager  
;! Creation Date: Tue Jun 20 08:31:38 2023  
;!  
[from-internal]  
exten => 6001,1,Dial(PJSIP/alice,10)  
exten => 6002,1,Dial(PJSIP/bob,10)  
exten => 6003,1,Dial(PJSIP/martin,10) ; Après 10 secondes envoi vers la règle 2  
exten => 6004,1,Dial(PJSIP/marie, 10)  
exten => 6005,1,Dial(PJSIP/jean, 10)  
exten => 6006,1,Dial(PJSIP/camille, 10)  
exten => 6007,1,Dial(PJSIP/pierre, 10)  
exten => 6008,1,Dial(PJSIP/emilie, 10)  
exten => 6009,1,Dial(PJSIP/antoine, 10)  
exten => 6010,1,Dial(PJSIP/sophie, 10)  
exten => 6011,1,Dial(PJSIP/lucas, 10)  
exten => 6012,1,Dial(PJSIP/lea, 10)  
exten => 6013,1,Dial(PJSIP/thomas, 10)  
exten => 6099,1,VoiceMailMain() ; 6099 Numéro de téléphone du répondeur  
  
; Règles 2  
exten => 6001,2,VoiceMail(6001) ; Appel répondeur compte 6001  
exten => 6002,2,VoiceMail(6002) ; Appel répondeur compte 6002  
exten => 6003,2,VoiceMail(6003) ; Appel répondeur compte 6003  
exten => 6004,2,VoiceMail(6004) ; Appel répondeur compte 6004  
exten => 6005,2,VoiceMail(6005) ; Appel répondeur compte 6005  
exten => 6006,2,VoiceMail(6006) ; Appel répondeur compte 6006  
exten => 6007,2,VoiceMail(6007) ; Appel répondeur compte 6007  
exten => 6008,2,VoiceMail(6008) ; Appel répondeur compte 6008  
exten => 6009,2,VoiceMail(6009) ; Appel répondeur compte 6009  
  
^G Aide ^O Écrire ^W Chercher ^K Couper ^T Exécuter ^C Emplacement M-U Annuler  
^X Quitter ^R Lire fich. ^N Remplacer ^U Coller ^J Justifier ^_ Aller ligne M-E Refaire
```

10. Une fois les fichiers de configuration installés, on fait :

asterix -rx « core restart now »

11. On lance la commande **asterisk -rvvvvvvvvvv**

Pour démarrer une session interactive avec asterisk. On accède à la console CLI pour entrer directement les commandes avec l'interface en ligne.

```

root@debian:/tmp/asterisk_config# root@pbx:~# ss -nlut | grep -E'5060|State'
grep : option invalide -- '|'
Usage : grep [OPTION]... MOTIFS [FICHIER]...
Exécutez « grep --help » pour obtenir des renseignements complémentaires.
-bash: root@pbx:~# : commande introuvable
root@debian:/tmp/asterisk_config# cd /etc/asterisk
root@debian:/etc/asterisk# asterisk -rx "core restart now"
root@debian:/etc/asterisk# root@pbx:~# ss -nlut|grep -E'5060|State'
-bash: root@pbx:~# : commande introuvable
grep : option invalide -- '|'
Usage : grep [OPTION]... MOTIFS [FICHIER]...
Exécutez « grep --help » pour obtenir des renseignements complémentaires.
root@debian:/etc/asterisk# ss -tnlp | grep ':5060' | awk '{print $1,$2,$3,$4,$5,$6,$7,$8,$9}'
root@debian:/etc/asterisk# ss -nlut | grep -E '5060|State'
Netid State Recv-Q Send-Q Local Address:Port Peer Address:PortProcess
root@debian:/etc/asterisk# udp UNCONN 0 0 0.0.0.0:5060 0.0.0.0:*
-bash: udp : commande introuvable
root@debian:/etc/asterisk# ss -nlut | grep -E '5060|State'
Netid State Recv-Q Send-Q Local Address:Port Peer Address:PortProcess
root@debian:/etc/asterisk# ss -nlut | grep -E '5060|State' > fichier.txt
root@debian:/etc/asterisk# cat fichier.txt
Netid State Recv-Q Send-Q Local Address:Port Peer Address:PortProcess
root@debian:/etc/asterisk# asterisk -rvvvvvvvvvv
Asterisk 21.2.0, Copyright (C) 1999 - 2022, Sangoma Technologies Corporation and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 21.2.0 currently running on debian (pid = 1367)
debian*CLI> pjsip list aors
No objects found.

debian*CLI> pjsip auths
No such command 'pjsip auths' (type 'core show help pjsip auths' for other possible commands)
debian*CLI>

```

On se trouve dans l'invite de commande de Asterisk debian*CLI.

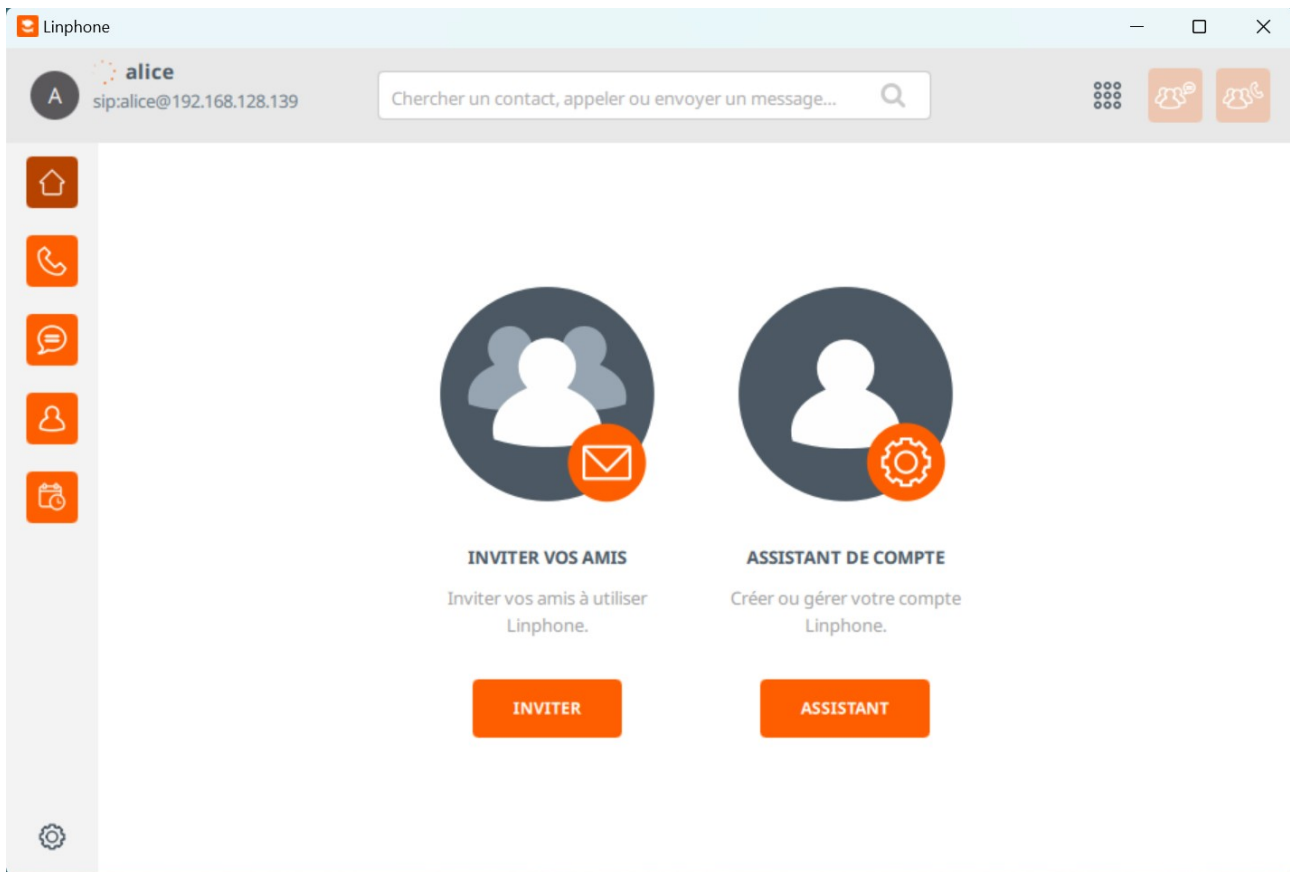
12. Maintenant, on installe un logiciel qui fourni les services VoIP. On choisit MicroSIP.

Communication entre Alice et Bob.



*

On prend un autre client Linphone pour assurer les échanges entre utilisateurs.



Protocole pour sécuriser/chiffrer les appels transitant par le serveur

On utilise SRTP (Secure Real-time Transport Protocol).

Il est utilisé pour la transmission de données audio et vidéo en temps réel sur les réseaux IP. Chiffrement, intégrité des données et authentification pour protéger les communications contre l'interception et la manipulation.

Les paramètres de SRTP se configurent dans le fichier de configuration pjsip.conf.

```
res_sorcery_memory.so      Sorcery In-Memory Object Wizard      2      Running
  core
res_sorcery_realtime.so    Sorcery Realtime Object Wizard       0      Running
  core
res_srtp.so                Secure RTP (SRTP)                    0      Running
  core
res_timing_timerfd.so      Timerfd Timing Interface             0      Running
  core
sounds                     Sounds Index                          1      Running
  core
udptl                      UDPTL                                2      Running
  core
96 modules loaded
```

AUTOMATISATION

Pour diffuser automatiquement des messages audio, on utilise les IVR, Interactive Voice Response, en fait, c'est un standard automatique, très pratique pour une entreprise. L'IVR fonctionne de la manière suivante :

- Quand on compose le numéro de l'entreprise
- L'appel est décroché automatiquement
- Un message est lu avec plusieurs options (tapez 1 pour joindre le service client, tapez 2 pour le service comptable etc.)
- Le choix dirige l'appel vers le bon correspondant.

Pour la configuration, elle se fait dans le dialplan, dans le fichier **extensions.conf**.

Pour le message, on peut utiliser un fichier audio enregistré ou la synthèse vocale. Pour la synthèse vocale, il suffit de taper le message dans le fichier de configuration et Asterisk le transformera en message audio. Pour la synthèse vocale on utilise le module 'gTTS', il faut installer python au préalable

- `apt install python3-pip`
- `pip3 --version`
- `pip install gTTS`
- `gtts-cli 'Bonjour, vous êtes bien sur le répondeur de la Plateforme' --output repondeur.mp3`
- `ls`
- `apt-get install perl libwww-perl sox mpg123`
- `sox -V repondeur.mp3 -r 8000 -c 1 -t ul repondeur.ulaw`
- `ls`
- `mv repondeur.ulaw /var/lib/asterisk/sounds/fr`
- `systemctl restart asterisk`


```

Préparation du dépaquetage de .../libsox-fmt-mp3_14.4.2+git20190427-3.5_amd64.deb ...
Dépaquetage de libsox-fmt-mp3:amd64 (14.4.2+git20190427-3.5) ...
Paramétrage de libtwolame0:amd64 (0.4.0-2) ...
Paramétrage de libid3tag0:amd64 (0.15.1b-14) ...
Paramétrage de libmad0:amd64 (0.15.1b-10.1+b1) ...
Paramétrage de libsox-fmt-mp3:amd64 (14.4.2+git20190427-3.5) ...
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u4) ...
root@debian:/etc/asterisk# sox -V repondeur.mp3 -r 8000 -c 1 -t ul repondeur.ulaw
sox:      SoX v14.4.2

Input File      : 'repondeur.mp3'
Channels        : 1
Sample Rate     : 24000
Precision       : 16-bit
Duration        : 00:00:05.02 = 120384 samples ~ 376.2 CDDA sectors
File Size       : 20.1k
Bit Rate        : 32.0k
Sample Encoding: MPEG audio (layer I, II or III)

Output File     : 'repondeur.ulaw' (ul)
Channels        : 1
Sample Rate     : 8000
Precision       : 14-bit
Duration        : 00:00:05.02 = 40128 samples ~ 376.2 CDDA sectors
Sample Encoding: 8-bit u-law
Reverse Nibbles: no
Reverse Bits     : no
Comment         : 'Processed by SoX'

sox INFO sox: effects chain: input      24000Hz  1 channels
sox INFO sox: effects chain: rate      8000Hz   1 channels
sox INFO sox: effects chain: dither    8000Hz   1 channels
sox INFO sox: effects chain: output    8000Hz   1 channels
root@debian:/etc/asterisk# mv repondeur.ulaw /var/lib/asterisk/sounds/fr/
root@debian:/etc/asterisk# systemctl restart asterisk
root@debian:/etc/asterisk#

```

Les modules installés :

```
core
res_pjsip_registrar.so    PJSIP Registrar Support      0      Running
core
res_pjsip_rfc3326.so      PJSIP RFC3326 Support        0      Running
core
res_pjsip_sdp_rtp.so      PJSIP SDP RTP/AVP stream handler 0      Running
core
res_pjsip_send_to_voicemail.so PJSIP REFER Send to Voicemail Support 0      Running
core
res_pjsip_session.so      PJSIP Session resource       15     Running
core
res_pjsip_t38.so          PJSIP T.38 UDPTL Support     0      Running
core
res_pjsip_transport_websocket.so PJSIP WebSocket Transport Support 0      Running
core
res_pjsip_xpidsf_body_generator.so PJSIP Extension State PIDF Provider 0      Running
core
res_rtp_asterisk.so       Asterisk RTP Stack           0      Running
core
res_sorcery_astdb.so      Sorcery Astdb Object Wizard   3      Running
core
res_sorcery_config.so     Sorcery Configuration File Object Wizard 16     Running
core
res_sorcery_memory.so     Sorcery In-Memory Object Wizard 2      Running
core
res_sorcery_realtime.so   Sorcery Realtime Object Wizard 0      Running
core
res_srtp.so               Secure RTP (SRTP)            0      Running
core
res_timing_timerfd.so     Timerfd Timing Interface     0      Running
core
sounds                     Sounds Index                  1      Running
core
udptl                     UDPTL                        2      Running
core
96 modules loaded
debian*CLI> _
```