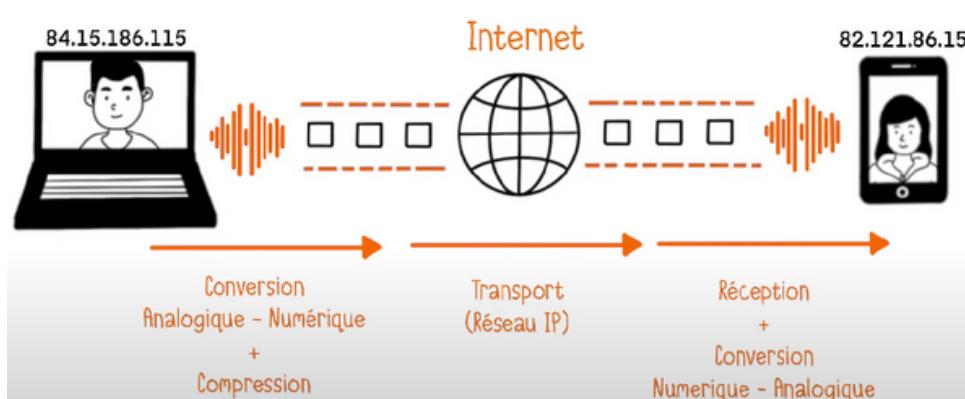


Voice Over Internet Protocol

Technologie qui permet de transmettre des signaux audio analogiques (voix) via un réseau informatique.
TOIP, seulement sur un réseau local.



Avantages



Économie
de coûts



Simplicité
d'implémentation



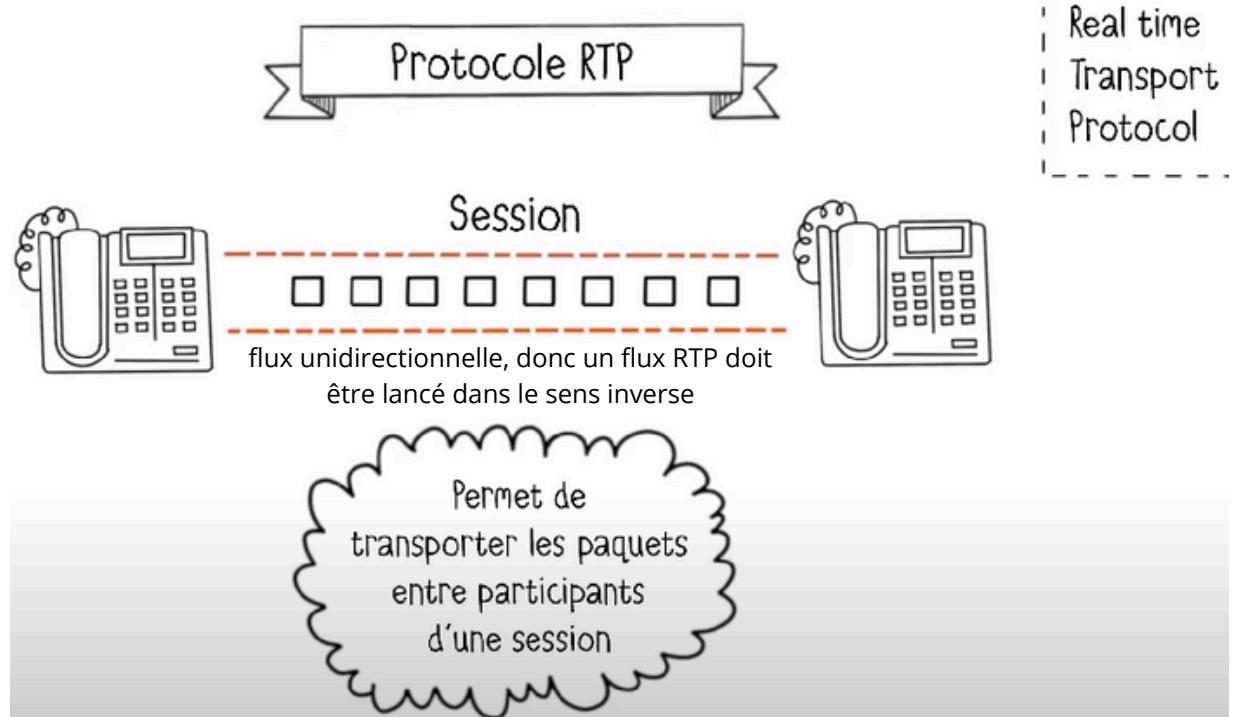
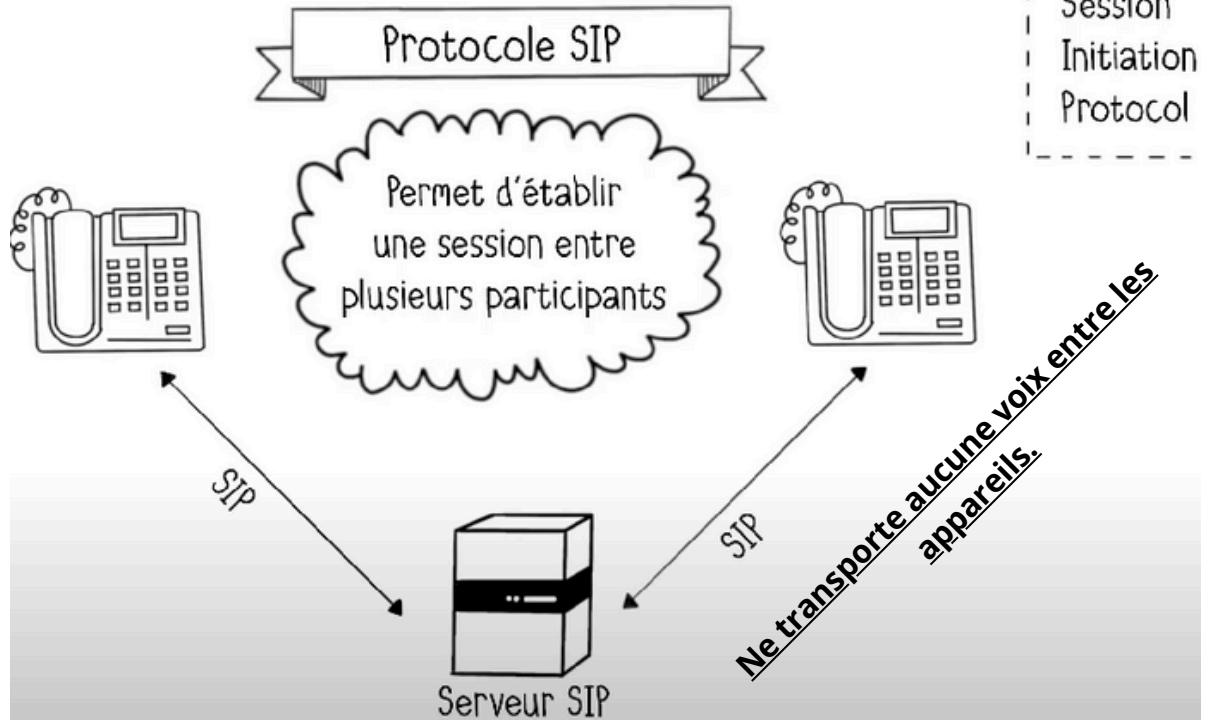
Nombreuses
fonctionnalités

Fonctionnalités Avancées : messagerie vocale, messagerie instantanée la vidéoconférence, appels internationaux, efficacité opérationnelle
Adaptable aux besoins croissants des entreprises

Flexibilité et Mobilité : Les utilisateurs peuvent effectuer des appels depuis n'importe quel endroit avec une connexion Internet, idéal pour les travailleurs à distance⁵⁹.

Fonctionnalités Avancées : La VoIP propose des fonctionnalités telles que la messagerie vocale, la vidéoconférence, et la gestion des appels, améliorant l'efficacité opérationnelle⁵⁹.

Évolutivité : Facilement adaptable aux besoins croissants des entreprises, permettant d'ajouter ou de supprimer des lignes téléphoniques selon les besoins⁵⁶.



SRTP : extension du protocole RTP qui ajoute des fonctionnalités de chiffrement, d'authentification et d'intégrité pour les flux audio et vidéo en temps réel

Le fichier **pjsip.conf** permet de **définir divers objets de configuration** tels que les transports, les enregistrements, les points de terminaison (endpoints), les mécanismes d'authentification, et les listes de contrôle d'accès (ACL). **Chaque section du fichier est dédiée à un type spécifique de configuration.**

Asterisk **n'est pas un simple binaire prêt à être installé.**

Il faut le **compiler à partir de son code source.**

Pour cela, on a besoin d'un compilateur (gcc via build-essential) et des bibliothèques qu'Asterisk utilise (libjansson, libxml2, libsqlite3).
Sans ces paquets, la compilation échouerait.

sudo apt install wget build-essential libjansson

```
valdeb@vmvoip:/usr/src$ sudo wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz
--2025-02-11 12:19:50-- http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz
Résolution de downloads.asterisk.org (downloads.asterisk.org)... 165.22.184.19, 2604:a880:400:d0::14:9001
Connexion à downloads.asterisk.org (downloads.asterisk.org)|165.22.184.19|:80... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 28395184 (27M) [application/octet-stream]
Sauvegarde en : « asterisk-20-current.tar.gz »

asterisk-20-cur 100%[=====] 27,08M 1,92MB/s   ds 20s

2025-02-11 12:20:11 (1,34 MB/s) - « asterisk-20-current.tar.gz » sauvegardé [28395184/28395184]
```

Installer les librairie nécessaire pour le bon fonctionnement d'Asterisk

sudo apt install build-essential wget libncurses5 libssl-dev libsqlite3-dev uuid-dev -y

sudo tar -xvf asterisk-20-current.tar.gz

```
valdeb@vmvoip:/usr/src$ ls
asterisk-20.12.0  asterisk-20-current.tar.gz
```

On se **déplace dans le dossier extrait** et on va pouvoir **configurer**.

./configure est un script de configuration présent dans le code source de nombreux logiciels sous Linux. Son but est de préparer la compilation en vérifiant que ton système a tout ce qu'il faut.

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo ./configure
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for pkg-config... no
configure: error: pkg-config not found
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ S
```

sudo ./configure nous indique que **pkg-config** est manquant, il faut le **télécharger** pour pouvoir continuer la configuration .

pkg-config est un outil qui permet aux logiciels de trouver les bibliothèques et les fichiers nécessaires à la compilation. Il aide **./configure** à détecter correctement les dépendances.

```
checking for history_init in -ledit... no
configure: error: *** Please install the 'libedit' development package.
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo apt install libedit-dev
```

sudo ./configure nous indique que **libedit-dev** est manquant, il faut le **télécharger pour pouvoir continuer la configuration ..**

libedit est une bibliothèque qui fournit des fonctionnalités d'édition de ligne de commande (comme l'historique des commandes et l'édition en ligne). Elle est utilisée par Asterisk pour gérer les interactions en ligne de commande.

```
checking for uid_generate_random... no
configure: error: *** uid support not found (this typically means the uid development package is missing)
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo apt install uuid-dev
```

sudo ./configure nous indique que **uuid-dev** est manquant, il faut le **télécharger pour pouvoir continuer la configuration ..**

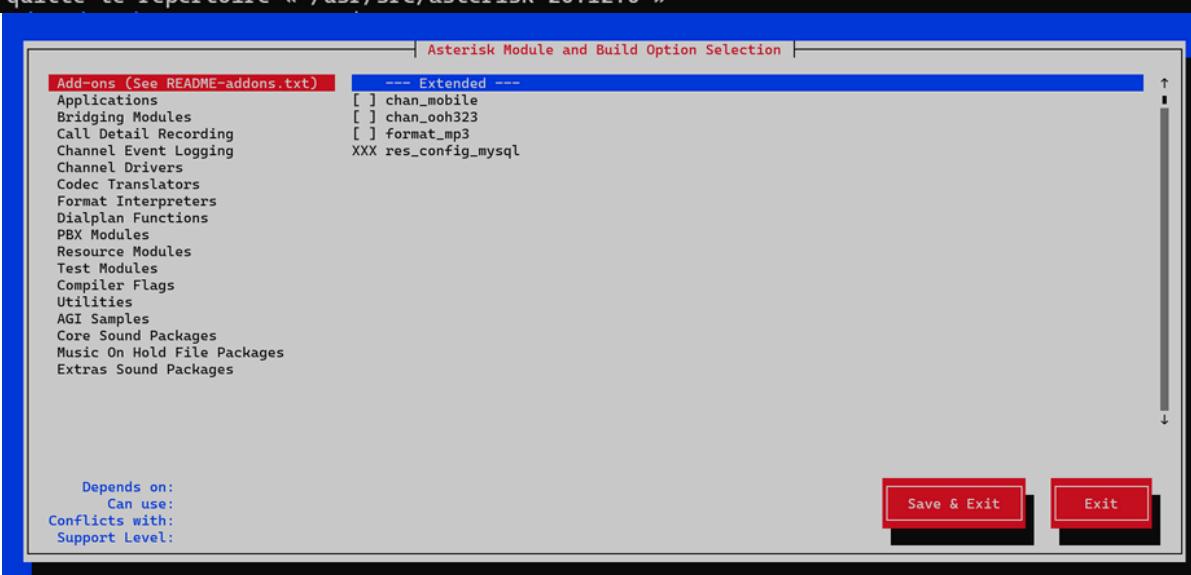
UUID (Universally Unique Identifier) est utilisé pour générer des identifiants uniques dans Asterisk.

```
ldeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo make menuselect
 $=\text{cc}$  CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" LDFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" nmenuselect
ke[1] : on entre dans le répertoire « /usr/src/asterisk-20.12.0/menuselect »
ke[1]: rien à faire pour « nmenuselect ».
ke[1] : on quitte le répertoire « /usr/src/asterisk-20.12.0/menuselect »
 $=\text{cc}$  CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" LDFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" gmenuselect
ke[1] : on entre dans le répertoire « /usr/src/asterisk-20.12.0/menuselect »
ke[1]: rien à faire pour « gmenuselect ».
ke[1] : on quitte le répertoire « /usr/src/asterisk-20.12.0/menuselect »
ke[1] : on entre dans le répertoire « /usr/src/asterisk-20.12.0 »
rminal must be at least 80 x 27.
nuselect changes NOT saved!
nuselect changes NOT saved!
ke[1] : on quitte le répertoire « /usr/src/asterisk-20.12.0 »
```

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo make install
Installing modules from channels...
Installing modules from pbx...
Installing modules from apps...
```

```
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
+-----
```

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo make menuselect
CC="cc" CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" nmenuselect
make[1]: on entre dans le répertoire « /usr/src/asterisk-20.12.0/menuselect »
make[1]: rien à faire pour « nmenuselect ».
make[1]: on quitte le répertoire « /usr/src/asterisk-20.12.0/menuselect »
CC="cc" CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" gmenuselect
make[1]: on entre dans le répertoire « /usr/src/asterisk-20.12.0/menuselect »
make[1]: rien à faire pour « gmenuselect ».
make[1]: on quitte le répertoire « /usr/src/asterisk-20.12.0/menuselect »
make[1]: on entre dans le répertoire « /usr/src/asterisk-20.12.0 »
make[1]: rien à faire pour « menuselect ».
make[1]: on quitte le répertoire « /usr/src/asterisk-20.12.0 »
Terminal must be at least 80 x 27.
menuselect changes NOT saved!
menuselect changes NOT saved!
make[1]: on quitte le répertoire « /usr/src/asterisk-20.12.0 »
```



makes samples créé des exemples qui servent à :

Comprendre la structure des fichiers de configuration : En regardant les exemples, vous pouvez mieux comprendre comment organiser vos propres configurations.

Configurer rapidement un système de base : Les fichiers d'exemple peuvent servir de point de départ pour configurer un système téléphonique simple.

Apprendre les fonctionnalités avancées : Ils incluent souvent des exemples de fonctionnalités avancées comme la messagerie vocale, les files d'attente, etc.

On crée les exemples avec **makes sample** :

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo make samples
Installing adsi config files...
/usr/bin/install -c -d "/etc/asterisk"
Installing configs/samples/asterisk.adsi
Installing configs/samples/telcordia-1.adsi
Installing other config files...
Installing file configs/samples/acl.conf.sample
Installing file configs/samples/adsi.conf.sample
Installing file configs/samples/aeap.conf.sample
Installing file configs/samples/agents.conf.sample
```

On crée un dossier **samples** dans /etc/asterisk:

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo mkdir /etc/asterisk/samples
```

puis on transfer tous les samples générés dans ce dossier:

```
valdeb@vmvoip:/etc/asterisk$ mv /etc/asterisk/*.* /etc/asterisk/samples/
```

sudo make basix-pbx génère des fichiers de déamrage

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo 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/confbridge.conf
```

sudo make config pour finir la configuration

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo make config
```

```
valdeb@vmvoip:/usr/src/asterisk-22.2.0$ systemctl status asterisk.service
● asterisk.service - LSB: Asterisk PBX
  Loaded: loaded (/etc/init.d/asterisk; generated)
  Active: active (running) since Tue 2025-02-11 17:13:56 CET; 3s ▶
    Docs: man:systemd-sysv-generator(8)
  Process: 47870 ExecStart=/etc/init.d/asterisk start (code=exited)
  Tasks: 56 (limit: 6682)
  Memory: 39.3M
    CPU: 820ms
   CGroup: /system.slice/asterisk.service
           └─47882 /usr/sbin/asterisk
lines 1-10/10 (END)
valdeb@vmvoip:/usr/src/asterisk-22.2.0$ sudo systemctl enable asterisk.service
```

```
valdeb@vmvoip:/usr/src/asterisk-20.12.0$ sudo asterisk -rvvv
Asterisk 20.12.0, Copyright (C) 1999 - 2025, Sangoma Technologies Corporation
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 20.12.0 currently running on vmvoip (pid = 4108)
vmvoip*CLI>
```

Configuration asterisk

La configuration consiste en la modification de 4 fichiers :

psjip.conf
extensions.conf
voicemail.conf
asterisk.conf

On va recréer ces fichiers car ils contiennent énormément d'information.

On va repartir de 0 pour le configurer nous même:

```
valdeb@vmvoip:/etc/asterisk$ ls
asterisk.conf          confbridge.conf      logger.conf      pjsip.conf.config2  README
asterisk.conf.origin    extensions.conf      modules.conf    pjsip.conf.origine   samples
cdr.conf                extensions.conf.origine  musiconhold.conf pjsip_notify.conf voicemail.conf
cdr_custom.conf         indications.conf     pjsip.conf       queues.conf        voicemail.conf.origine
```

On vérifie le fonctionnement de asterisk avec statut, stop, restart, status

sudo service asterisk start

On recherche port 5060 mais il n'est pas présent car le fichier pjsip.conf que l'on a modifié est vide. Si il était présent cela voudrait dire que le protocole Pjsip est en action.

```
valdeb@vmvoip:/etc/asterisk$ ss -anup
State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port      Pro
ress
UNCONN      0           0           0.0.0.0:53701      0.0.0.0:*
UNCONN      0           0           0.0.0.0:68        0.0.0.0:*
UNCONN      0           0           [::]:59011        [::]:*
valdeb@vmvoip:/etc/asterisk$
```

Configuration du fichier **extension.conf**

```
[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=>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
```

Configuration du fichier **voicemail.conf**

```
[general]
format=wav49|gsm|wav|ulaw
[default]
; Numéro de messagerie => mot de passe d'accès à la messagerie, nom d'utilisateur
6001 => 1234, alice
6002 => 1234, bob
6003 => 1234, martin
```

Configuration des utilisateurs dans **pjsip.conf**

```
[transport-udp]
type=transport
protocol=udp
bind=0.0.0.0

;Templates for the necessary config sections

[endpoint_internal](!)
type=endpoint
context=from-internal
disallow=all
allow=ulaw
language=fr

[auth_userpass](!)
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)
```

Configuration du fichier **asterisk.conf**

```
[directories](!)
astcachedir => /var/cache/asterisk
astetcdir => /etc/asterisk
astmoddir => /usr/lib/asterisk/modules
astvarlibdir => /var/lib/asterisk
astdbdir => /var/lib/asterisk
astkeydir => /var/lib/asterisk
astdatadir => /var/lib/asterisk
astagidir => /var/lib/asterisk/agi-bin
astspooldir => /var/spool/asterisk
astrundir => /var/run/asterisk
astlogdir => /var/log/asterisk
astsbindir => /usr/sbin

[options]
;verbose = 3
;debug = 3
;trace = 0           ; Set the trace level.
;refdebug = yes      ; Enable reference count debug logging.
;alwaysfork = yes    ; Same as -F at startup.
;nofork = yes        ; Same as -f at startup.
;quiet = yes         ; Same as -q at startup.
;timestamp = yes     ; Same as -T at startup.
;execincludes = yes   ; Support #exec in config files.
;console = yes       ; Run as console (same as -c at startup).
;highpriority = yes   ; Run realtime priority (same as -p at
;                      ; startup).
;initcrypto = yes     ; Initialize crypto keys (same as -i at
;                      ; startup).
;nocolor = yes        ; Disable console colors.
;dontwarn = yes       ; Disable some warnings.
;dumpcore = yes       ; Dump core on crash (same as -g at startup).
languageprefix = yes  ; Use the new sound prefix path syntax.
;systemname = my_system_name ; Prefix uniqueid with a system name for
```

Sécuriser les appels

Asterisk - Chiffrement des Appels
La sécurisation des appels peut être intéressante à mettre en place, afin de protéger nos appels téléphoniques. Nous pouvons chiffrer le flux de signalisation ainsi que le flux audio. De cette manière...

Networlab / Feb 28, 2014

Pour sécuriser les appels, il nous faut chiffrer deux choses :

- Le flux SIP (la signalisation)
- Le flux RTP (la voix)

Chiffer le flux SIP avec TLS

Pour sécuriser le flux SIP nous utilisons le protocole **TLS – Transport Layer Security**.

Etape 1

Créer le fichier keys dans /etc/asterisk , puis dans /usr/src/asterisk/contrib/scripts, lancer le script :

./ast_tls_cert -C 192.168.224.7 -O "valnet" -d /etc/asterisk/keys

```
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ sudo ./ast_tls_cert -C 192.168.224.7 -O "Valnet" -d /etc/asterisk/keys
[sudo] Mot de passe de valdeb :
No config file specified, creating '/etc/asterisk/keys/tmp.cfg'
You can use this config file to create additional certs without
re-entering the information for the fields in the certificate
Creating CA key /etc/asterisk/keys/ca.key
Enter PEM pass phrase:
```

Ce script permet de créer dans le dossier **keys** de fichiers :

un certificat self-signed pour l'autorité de certificat

un certificat pour le serveur Asterisk.

une clé privée pour l'autorité de certificat et une pour Asterisk.

C : permet de spécifier le nom d'hôte du serveur Asterisk

O : permet de définir le nom de l'organisation

d : permet de spécifier le dossier de sortie

Les fichiers PEM regroupent la clé privée et le certificat.

Les fichiers CSR sont des fichiers de requête de certificat (nous n'en avons pas besoin).

```
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ ls /etc/asterisk/keys/
asterisk.crt asterisk.csr asterisk.key asterisk.pem ca.cfg ca.crt ca.key tmp.cfg
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$
```

Etape 2

A présent, nous devons **créer les clés et certificats** pour **chaque clients** en lancant le script :

sudo ./ast_tls_cert -m client -c /etc/asterisk/keys/ca.crt -k /etc/asterisk/keys/ca.key -C 6001.192.168.224.7 -O "Valnet" -d /etc/asterisk/keys -o 6001

```
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ sudo ./ast_tls_cert -m client -c /etc/asterisk/keys/ca.crt -k /etc/asterisk/keys/ca.key -C 6001.192.168.224.7 -O "Valnet" -d /etc/asterisk/keys -o 6001
```

m : indique qu'il faut créer un certificat client

c : permet de spécifier le chemin vers le certificat de l'autorité de certificat

k : permet de spécifier le chemin vers la clé privée de l'autorité de certificat

C : permet de spécifier le nom d'hôte du poste du client. Il est possible de spécifier une IP.

O : permet de définir le nom de l'organisation

d : permet de spécifier le dossier de sortie

o : permet de choisir le nom de la clé à créer

```
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ ls /etc/asterisk/keys/
6001.crt 6001.key asterisk.crt asterisk.key ca.cfg ca.key
6001.csr 6001.pem asterisk.csr asterisk.pem ca.crt tmp.cfg
```

les fichiers ont bien été créé dans keys pour l'utilisateur 6001, maintenant on le fait pour les autres utilisateurs

```

valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ sudo ./ast_tls cert -m client -c /etc/asterisk/keys/ca.crt -k /etc/asterisk/keys/ca.key -C 6002.192.168.224.7 -o "Valnet" -d /etc/asterisk/keys -o 6002
[sudo] Mot de passe de valdeb :

No config file specified, creating '/etc/asterisk/keys/tmp.cfg'
You can use this config file to create additional certs without
re-entering the information for the fields in the certificate
Creating certificate /etc/asterisk/keys/6002.key
Creating signing request /etc/asterisk/keys/6002.csr
Creating certificate /etc/asterisk/keys/6002.crt
Certificate request self-signature ok
subject=CN = 6002.192.168.224.7, O = Valnet
Enter pass phrase for /etc/asterisk/keys/ca.key:
Combining key and crt into /etc/asterisk/keys/6002.pem
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ ls /etc/asterisk/keys/
6001.crt 6001.key 6002.crt 6002.key asterisk.crt asterisk.key ca.cfg ca.key
6001.csr 6001.pem 6002.csr 6002.pem asterisk.csr asterisk.pem ca.crt tmp.cfg
valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ sudo ./ast_tls cert -m client -c /etc/asterisk/keys/ca.crt -k /etc/asterisk/keys/ca.key -C 6003.192.168.224.7 -o "Valnet" -d /etc/asterisk/keys -o 6003
No config file specified, creating '/etc/asterisk/keys/tmp.cfg'
You can use this config file to create additional certs without
re-entering the information for the fields in the certificate
Creating certificate /etc/asterisk/keys/6003.key
Creating signing request /etc/asterisk/keys/6003.csr
Creating certificate /etc/asterisk/keys/6003.crt
Certificate request self-signature ok
subject=CN = 6003.192.168.224.7, O = Valnet
Enter pass phrase for /etc/asterisk/keys/ca.key:

valdeb@vmvoip:/usr/src/asterisk-22.2.0/contrib/scripts$ ls /etc/asterisk/keys/
6001.crt 6001.key 6002.crt 6002.key 6003.crt 6003.key asterisk.crt asterisk.key ca.cfg ca.key
6001.csr 6001.pem 6002.csr 6002.pem 6003.csr 6003.pem asterisk.csr asterisk.pem ca.crt tmp.cfg

```

Etape 3

A présent, nous devons configurer Asterisk pour autoriser l'utilisation de TLS en modifiant pjsip

```

GNU nano 7.2                                     pjsip.conf
[general]
tlsenable=yes
tlsbindaddr=0.0.0.0
tlscertfile=/etc/asterisk/keys/asterisk.pem
tlscafile=/etc/asterisk/keys/ca.crt
tlscipher=ALL
tlsclientmethod=tlsv1

```

```

GNU nano 7.2
[transport-tls]
type=transport
protocol=tls
bind=0.0.0.0:5061
cert_file=/etc/asterisk/keys/asterisk.pem
priv_key_file=/etc/asterisk/keys/asterisk.key
method=tlsv1_2

```

```

[endpoint_internal](!)
type=endpoint
context=from-internal
disallow=all
allow=ulaw
language=fr
transport=transport-tls

```

spécifier comment Asterisk doit gérer les connexions TLS.

spécifier que tous les endpoints utilisant ce modèle

```

vmvoip*CLI> pjsip show transports
Transport: <TransportId.....> <Type> <cos> <tos> <BindAddress>
=====
Transport: transport-tls          tls      0      0 0.0.0.0:5061
Transport: transport-udp         udp      0      0 0.0.0.0:5060
Objects found: 2

```

```
vmvoip*CLI> pjsip set logger on
```

pour avoir plus d'info sur les logs

Etape 4

Modifier dans l'application, le protocole UDP en TLS et modifier le port 5061



Capture en cours de Wi-Fi

Fichier Editer Yue Aller Capture Analyser Statistiques Telephonejé Wireless Qutils Aide

Appliquer un filtre d'affichage ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
2096	116.281711	2606:4700::6812:1b30	2a02:8440:c201:3fa6...	TLSv1.2	101	Application Data
2099	116.283448	2a02:8440:c201:3fa6...	2606:4700::6812:1b30	TLSv1.2	105	Application Data
2138	117.977566	2606:4700::6810:6670	2a02:8440:c201:3fa6...	TLSv1.2	112	Application Data
2139	117.978400	2a02:8440:c201:3fa6...	2606:4700::6810:6670	TLSv1.2	116	Application Data
2156	119.166729	192.168.224.11	192.168.224.7	TLSv1.2	481	Application Data
2158	119.171059	192.168.224.7	192.168.224.11	TLSv1.2	492	Application Data
2160	119.171971	192.168.224.7	192.168.224.11	TLSv1.2	625	Application Data
2164	119.172390	192.168.224.11	192.168.224.7	TLSv1.2	487	Application Data
1	0.000000	142.132.209.32	192.168.224.11	TLSv1.3	1474	Server Hello, Change Cipher Spec, Application Data
4	0.054348	142.132.209.32	192.168.224.11	TLSv1.3	1474	Application Data
5	0.054348	142.132.209.32	192.168.224.11	TLSv1.3	373	Application Data, Application Data
8	0.057799	192.168.224.11	142.132.209.32	TLSv1.3	134	Change Cipher Spec, Application Data
10	0.058209	192.168.224.11	142.132.209.32	TLSv1.3	146	Application Data
12	0.058444	192.168.224.11	142.132.209.32	TLSv1.3	641	Application Data

Chiffrer le flux Audio avec SRTP

Etape 1

Télécharger la librairie SRTP.

```
valdeb@vmvoip:/usr/src$ cd /usr/src
wget https://github.com/cisco/libsrtp/releases/download/v2.5.0/libsrtp-2.5.0.tar.gz
--2025-02-17 17:47:38-- https://github.com/cisco/libsrtp/releases/download/v2.5.0/libsrtp-2.5.0.tar.g
Résolution de github.com (github.com)... 140.82.121.3
Connexion à github.com (github.com)|140.82.121.3|:443... connecté.
requête HTTP transmise, en attente de la réponse... 404 Not Found
2025-02-17 17:47:38 erreur 404 : Not Found.
```

Nous avons du clonner le dépôt git hub libsrtp apres avoir installer git

```
valdeb@vmvoip:/usr/src$ sudo apt install git
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront insta
git-man liberror-perl
```

```
valdeb@vmvoip:/usr/src$ sudo git clone https://github.com/cisco/libsrtp.git
Clonage dans 'libsrtp'...
remote: Enumerating objects: 12134, done.
remote: Counting objects: 100% (2543/2543), done.
remote: Compressing objects: 100% (402/402), done.
remote: Total 12134 (delta 2320), reused 2141 (delta 2141), pack-reused 9591
Réception d'objets: 100% (12134/12134), 7.11 Mio | 5.35 Mio/s, fait.
Résolution des deltas: 100% (6773/6773), fait.
```

On vérifie les version dispo avec git tag puis on instal avec git checkout version

```
valdeb@vmvoip:/usr/src/libsrtp$ git tag
cj-review
cvsImport27Mar2013
v1
v1.5
v1.5.0
v1.5.1
v1.5.2
v1.5.3
v1.5.4
v1.6
v1.6.0
v2
v2.0
v2.0.0
v2.1
v2.1.0
v2.2
v2.2.0
v2.3
v2.3.0
v2.4
v2.4.0
v2.4.1
v2.4.2
v2.5
v2.5.0
v2.6
v2.6.0
valdeb@vmvoip:/usr/src/libsrtp$ git checkout v2.6.0
```

Puis on Configure

```
valdeb@vmvoip:/usr/src/libsrtp$ ./configure
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name...
checking for suffix of executables...
configure: creating ./config.status
config.status: creating Makefile
config.status: creating crypto/Makefile
config.status: creating doc/Makefile
config.status: creating fuzzzer/Makefile
config.status: creating libsrtp2.pc
config.status: creating crypto/include/config.h
```

Puis make

```
valdeb@vmvoip:/usr/src/libsrtp$ make
gcc -DHAVE_CONFIG_H -Icrypto/include -I./
zations -funroll-loops -c srtp/srtp.c -o
gcc -DHAVE_CONFIG_H -Icrypto/include -I./
zations -funroll-loops -c crypto/cipher/c
gcc -DHAVE_CONFIG_H -Icrypto/include -I./
zations -funroll-loops -c crypto/auth/a
```

Puis make install

```
valdeb@vmvoip:/usr/src/libsrtp$ sudo make install
/usr/bin/install -c -d /usr/local/include/srtp2
/usr/bin/install -c -d /usr/local/lib
cp ./include/srtp.h /usr/local/include/srtp2
cp ./crypto/include/cipher.h /usr/local/include/srtp2
cp ./crypto/include/auth.h /usr/local/include/srtp2
cp ./crypto/include/crypto_types.h /usr/local/include
if [ -f libsrtp2.a ]; then cp libsrtp2.a /usr/local/l
```

Etape 2

Charger le module SRTP dans Asterisk :

```
vmvoip*CLI> module load res_srtp.so
Loaded res_srtp.so
vmvoip*CLI> |
```

Ajouter **media_encryption=sdes** dans la partie endpoint_internal de pjsip.conf

```
[endpoint_internal](!)
type=endpoint
context=from-internal
disallow=all
allow=ulaw
language=fr
transport=transport-tls
media_encryption=sdes
```

Automatisation IVR



Asterisk : Mettre en place un menu interactif (IVR) avec Google TTS - DS1 e...
Un IVR (Interactive Voice Response) permet de guider les utilisateurs vers le service qu'ils souhaitent joindre, on peut s'en servir aussi pour simplement annoncer les horaires d'ouvertures etc...
toolin.fr Sep 14, 2016

Interactive Voice Response permet de guider les utilisateurs vers le service qu'ils souhaitent joindre, on peut s'en servir aussi pour simplement annoncer les horaires d'ouvertures etc...

Pour ce qui est du message à diffuser, il existe deux solutions,
Jouer un **fichier audio** que l'on a enregistré ou **Faire de la synthèse vocale**

La mise en place d'un IVR va se faire en deux étapes, la première il faut que nous apprenons à Asterisk à nous parler, la seconde est la création de l'IVR.

Etape 1

Pour **faire parler Asterisk**, nous allons **installer un moteur de synthèse vocale (Text To Speech)** et nous allons mettre en place celui de Google appelé **GoogleTTS**.

Tout d'abord GoogleTTS à besoins de quelques dépendances pour fonctionner.

Nous ajoutons donc le langage Perl, la librairie World Wide Web de Perl, Sox pour la conversion audio, et Mpg123 pour la lecture et la conversion audio.

```
valdeb@vmvoip:~$ sudo apt-get install perl libwww-perl sox mpg123
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
```

Il nous faut aussi récupérer le module TTS de Google pour Asterisk et le déplacer dans le bon dossier

```
valdeb@vmvoip:~/GoogleTTS$ ls          /var/lib/asterisk/agi-bin/ :
GoogleTTS.tar.gz

valdeb@vmvoip:~/GoogleTTS$ tar -xvzf GoogleTTS.tar.gz
zaf-asterisk-googletts-0592005/
zaf-asterisk-googletts-0592005/COPYING
zaf-asterisk-googletts-0592005/ChangeLog
zaf-asterisk-googletts-0592005/README
zaf-asterisk-googletts-0592005/authors
zaf-asterisk-googletts-0592005/cli/
zaf-asterisk-googletts-0592005/cli/googletts-cli-tiny.pl
zaf-asterisk-googletts-0592005/cli/googletts-cli.pl
zaf-asterisk-googletts-0592005/googletts-tiny.agi
zaf-asterisk-googletts-0592005/googletts.agi

valdeb@vmvoip:~/GoogleTTS$ ls
GoogleTTS.tar.gz  zaf-asterisk-googletts-0592005
valdeb@vmvoip:~/GoogleTTS$ cd zaf-asterisk-googletts-0592005/
valdeb@vmvoip:~/GoogleTTS/zaf-asterisk-googletts-0592005$ ls
authors  cli  googletts.agi  README
ChangeLog  COPYING  googletts-tiny.agi
valdeb@vmvoip:~/GoogleTTS/zaf-asterisk-googletts-0592005$ cp googletts.agi /var/lib/asterisk/agi-bin/
cp: impossible de créer le fichier standard '/var/lib/asterisk/agi-bin/googletts.agi': Permission non accordée
valdeb@vmvoip:~/GoogleTTS/zaf-asterisk-googletts-0592005$ sudo c
p googletts.agi /var/lib/asterisk/agi-bin/
r/ib/asterisk/agi-bin/
valdeb@vmvoip:~/GoogleTTS/zaf-asterisk-googletts-0592005$ cd /va
r/lib/asterisk/agi-bin/
valdeb@vmvoip:/var/lib/asterisk/agi-bin$ ls
googletts.agi
```

Etape 2

Modifier le fichier extensions.conf, pour y entrer les paramètres de l'IVR

```

[ivr_1]
exten => s,1,Answer()
exten => s,2,Set(TIMEOUT(response)=10)
exten => s,3,agi(googletts.agi,"Bonjour et bienvenue chez Valandco le sang"
exten => s,4,agi(googletts.agi,"Pour joindre le service RH, taper 1.",fr,ar
exten => s,5,agi(googletts.agi,"Pour joindre le directeur Rossi Valentin, t
exten => s,6,agi(googletts.agi,"Pour joindre le service Compta, taper 3.",f
exten => s,7,WaitExten()

exten =>1,1,Dial(PJSIP/alice,10)
|exten =>2,1,Dial(PJSIP/bob,10)
exten =>3,1,Dial(PJSIP/alice,10)

exten => _[04-9*#],1,agi(googletts.agi,"Entrée invalide, merci de taper un
exten => _[04-9*#],2,Goto(ivr_1,s,1)
exten => t,1,Goto(ivr_1,s,3)

```

A ce stade la configuration est sensée marcher mais ce n'était pas le cas, il a fallut que je modifie le fichier:

- 1.modules.conf pour mettre en autoload
2. rajouter la ligne load = re_agi.so

```

GNU nano 7.2                                     modules.conf
[modules]
autoload = yes

```

```

GNU nano 7.2                                     modules.conf *
load = func_devstate.so
load = func_strings.so

; Core/PBX

load = pbx_config.so

; Resources

load = res_http_websocket.so
load = res_musiconhold.so
load = res_piproject.so
load = res_agi.so
load = res_pjsip_acl.so
load = res_pjsip_authenticator_digest.so
load = res_pjsip_caller_id.so
load = res_pjsip_dialog_info_body_generator.so
load = res_pjsip_diversion.so
load = res_pjsip_dtmf_info.so
load = res_pjsip_endpoint_identifier_anonymous.so
load = res_pjsip_endpoint_identifier_in.so

```