



# VoIP

👤 Création par	Ⓕ Frederic Perry
🕒 Date de création	@19 février 2025 09:10
☰ Catégorie	
👥 Réviseurs	Ⓜ Michael Vandeveld Ⓔ Etienne Verschuere Ⓑ Bastien Llorca

## Introduction et contextualisation

**La VoIP (Voice over Internet Protocol) est une technologie qui permet de transmettre la voix sur des réseaux IP, offrant une alternative économique et flexible aux systèmes de téléphonie traditionnels. Ce projet vise à mettre en place un système de téléphonie VoIP pour gérer les communications vocales au sein d'une organisation. Dans un contexte où les communications sont cruciales, la VoIP offre des avantages significatifs en termes de coûts et de fonctionnalités.**

Solution existantes Gratuites, open Sources : Asterisk, Free PBX, Front Panel Operator

Exemples de solutions existantes payantes : Oracle Sibel, Microsoft Teams, Discord

## Présentation Fonctionnelle

**Un système VoIP utilise le protocole IP pour transmettre les communications vocales. Voici les principales fonctionnalités mises en place dans ce projet :**

- **Appels Voix** : Possibilité de passer et recevoir des appels via Internet.
- **Messagerie Vocale** : Configuration de boîtes vocales pour chaque utilisateur.
- **Conférences Téléphoniques** : Mise en place de salles de conférence pour des appels multi participants.
- **Intégration avec des Applications** : Compatibilité avec des logiciels de communication comme Skype, Microsoft Teams, etc.
- **Mobilité** : Accès aux fonctionnalités de téléphonie depuis n'importe quel appareil connecté à Internet.

# Avantages et Inconvénients.

## Avantages

### Réduction des Coûts

**Les appels via Internet sont généralement moins coûteux que les appels traditionnels.**

**Flexibilité : Possibilité d'intégrer diverses applications et services.**

**Évolutivité : Facilité d'ajouter de nouveaux utilisateurs et fonctionnalités.**

### Inconvénients :

Réactivité si problème type maj défaillantes, pas de vrai supports

**Dépendance à Internet : La qualité des appels dépend de la stabilité de la connexion Internet.**

**Sécurité : Nécessite des mesures de sécurité pour protéger les communications audio et vidéos.**

**Qualité de Service : Peut être affectée par la congestion du réseau, nécessite internet**

## Solutions Existantes sur le Marché

- Skype: Une solution populaire pour les appels vocaux et vidéo.
- Zoom: Connu pour ses fonctionnalités de visioconférence et d'appels vocaux.
- RingCentral: Une solution complète pour les communications d'entreprise.
- Linnphone: Une solution open source pour les appels VoIP.
- Teams Microsoft
- Discord

## Exemples d'Implémentations

### Télétravail

Une entreprise utilise un système VoIP pour permettre à ses employés de travailler à distance tout en restant connectés. Les appels peuvent être passés et reçus depuis n'importe quel endroit avec une connexion Internet.

## Centre d'Appels

Un centre d'appels utilise la VoIP pour gérer les appels entrants et sortants. Les agents peuvent travailler depuis différents lieux, offrant une flexibilité accrue.

Travail en déporter , Etranger par exemple

Customer Relationship Management : Traitement Multi Services

(exemple une entreprise avec un service technique, un service client , un support commercial)

## Conclusion

La VoIP représente une solution moderne et efficace pour les besoins de téléphonie. Bien qu'elle dépende de la qualité de la connexion Internet, ses avantages en termes de coûts et de flexibilité en font un choix judicieux pour de nombreuses organisations. Ce projet démontre la faisabilité et l'efficacité de l'utilisation de la VoIP pour les communications vocales.

En détail pour répondre avec précisions aux questions des consignes :

Son coût réduit. En effet, les appels VoIP sont généralement moins chers que les appels téléphoniques traditionnels.

**Peut-on dire la même chose de ses coûts opérationnels et de maintenance ?**

pas du tout car imaginons un central voip qui ne fonctionne pas, une entreprise qui a comme coeur de métier le support client, ne peut pas se permettre une coupure de service pour raisons x et y

sans rajout de coûts opérationnels, la maintenance est aussi liée aux produits on ne peut pas avoir les mêmes attentes d'un contrat utilisations basique que si on bénéficie d'un contrat pro en intervention a -4h comme orange le propose par exemple. Les attentes sont forcément différentes et de ce fait les coûts également.

**Pourriez-vous dire en quoi la configuration VoIP d'un call center serait différente de la configuration VoIP d'un standard téléphonique d'une entreprise ?**

les différences se situent en qualité de service, un standard téléphonique de call center est régi par des horaires fixes, un fonctionnement qui prends différents facteurs en charges tel que affectation de files, priorisations des appels par secteurs (service commercial, support

technique) différentes équipes et historisations complète des appels ce que ne permet pas forcément une configuration voip malgré la présence de commutateur logiciels.

Identifiez des sites marchands ou de service dont customer service implique des services VoIP, donnez quelques exemples et décrivez une architecture possible de leur système.

darty , la fnac, france travail, la poste (service bancaire, service livraison, service courrier)  
l'architecture est basé sur une authentification , une localisation et ensuite une orientation par service

Effectuez quelques recherches sur les chiffrements les mieux adaptés à la VoIP.

protocole chiffrements : SRTP , DTLS, TLS (voir notre présentation)

SSL, AES 256 par certificat, clés de chiffrements

en usage : Utilisation VPN type Proton VPN , Tunnel Bear version pro , Windscribe Version Pro  
compléments :

<https://datatracker.ietf.org/doc/html/rfc3711>

---

**La partie suivante a été faite par Étienne, Bastien , Michael et Fred. La Plateforme , Février 2025.**

## **Installation et Configuration**

### **1: Installation de Debian 12 sur une machine virtuelle VMware.**

image ISO dvd full

Téléchargement depuis le miroir proxad (free) de la distribution :

<http://debian.proxad.net/debian/dists/Debian12.9/>

### **2.Mise à jour du Système bash:**

commandes :

**sudo apt update**

**sudo apt upgrade**

### **3.Installation de SUDO:**

**sudo apt install sudo**

### **4.Téléchargement et Installation du serveur VO IP Asterisk:**

nous changeons de répertoire de travail , nous fixons le workdir également.

commande : 'cd /usr/src' , avec également un MKDIR 'asterisk' en création de dossier

nous téléchargeons ensuite la dernière version stable du serveur depuis le site éditeur (asterisk.org)

**sudo wget <https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-22.2.0.tar.gz>**

le repository github se trouvant également ici :

<https://github.com/asterisk/asterisk>

ensuite nous décompressons l'archive tar (gzip)

**sudo tar xvf asterisk-22.2.0.tar.gz**

**cd asterisk-22.2.0**

**sudo ./configure**

**sudo make**

**sudo make install**

**sudo make config**

### **5.Installation d'OpenSSH:**

nous installons ensuite ssh de manière à sécuriser l'environnement

et également nous faciliter la saisie de commandes.

**sudo apt install openssh-server**

**sudo systemctl enable ssh**

**sudo systemctl start ssh**

### **6.Installation des Codecs Audio :**

**sudo apt install libsox-fmt-mp3**

nous n'avons pas sélectionné les fichiers wav , nous avons privilégié la consommation de bande passante :

codes , débit :

- ADPCM, 32kbit/s
- G.711 A-law, 64kbit/s
- G.711  $\mu$ -law, 64kbit/s
- G.722, 64kbit/s
- G.726, 32kbit/s
- GSM, 13kbit/s
- LPC-10, 2.4kbit/s

## 7.Configuration d'Asterisk:

- Éditez `/etc/asterisk/pjsip.conf` pour configurer les comptes SIP et créer les utilisateurs
- a noter que dans les anciennes versions d'Asterisk , pjsip est en fait le fichier sip.conf
- 
- Éditez `/etc/asterisk/extensions.conf` pour configurer les extensions et les plans de numérotation en anglais le Dial Plan.

### Exemple de Fichier de Configuration

#### pjsip.conf :

[transport-udp] ; le protocole udp

type=transport

protocol=udp

bind= 0.0.0.0:5060 ; ici le binding, pour définir une adresse hôte par exemple et le port ici 5060 UDP , le port TLS étant 5061

## **extensions.conf :**

```
[default]
exten ⇒ 1001,1,Dial(SIP/1001)
exten ⇒ 1234,1,Answer()
same ⇒ n,Playback(welcome)
same ⇒ n,Background(demo-congrats)
same ⇒ n,Hangup()
```

```
[transport-udp]
```

```
type=transport
```

```
protocol=udp
```

```
bind=
```

```
0.0.0.0:5060
```

```
1000]
```

```
type=endpoint
```

```
context=from-internal ; de manière à permettre des appels internes
```

```
disallow=all ; codecs audio , on refuse tout codes sauf celui en allow ci dessous :
```

```
allow=ulaw
```

```
auth=1000 ; nom de l'utilisateur
```

```
aors=1000 ; mdp de l'utilisateur
```

## **8.Configurer un protocole pour sécuriser/chiffrer les appels transitant par le serveur.**

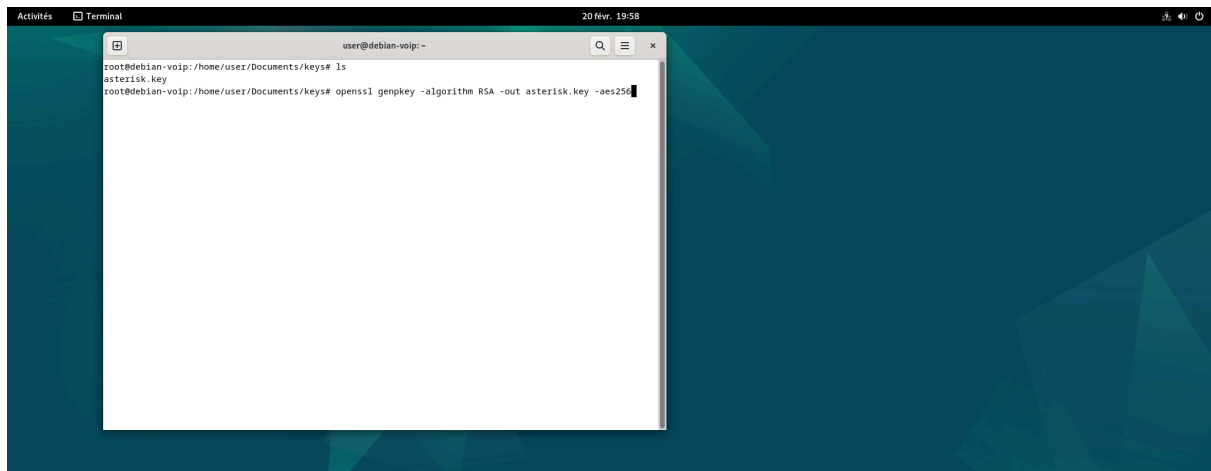
le protocole utilisé est TLS (Transport layer Security) , il existe 4 versions

TLS v1.0 créer en 1999

TLS v1.1 créer en 2006

TLS v1.2 créer en 2008

TLS v1.3 créer en 2018



```

        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:2d:27:08 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.1.131/24 brd 192.168.1.255 scope global dynamic ens33
        valid_lft 1428sec preferred_lft 1428sec
    inet6 fe80::20c:29ff:fe2d:2708/64 scope link
        valid_lft forever preferred_lft forever
root@debian-voip:/home/user/Documents/keys# openssl req -new -x509 -key asterisk.key -out asterisk.crt -c
ays 10
Enter pass phrase for asterisk.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:Marseille
Locality Name (eg, city) []:13
Organization Name (eg, company) [Internet Widgits Pty Ltd]:datatrust.com
Organizational Unit Name (eg, section) []:datatrust cybersecurit 
Common Name (e.g. server FQDN or YOUR name) []:fred
Email Address []:fredlaplateforme.io
root@debian-voip:/home/user/Documents/keys# ls
asterisk.crt  asterisk.key
root@debian-voip:/home/user/Documents/keys#

```

pour le test cr ation nous avons cr er un certificat d'une dur e de 10 jours

openssl genkey -algorithm RSA -out /etc/asterisk/asterisk.key -aes256

openssl req -new -x509 -key /etc/asterisk/asterisk.key -out /etc/asterisk/asterisk.crt -days 10

pour l'int gration dans le fichier de conf :



```

[global]
tls_certificate = /etc/asterisk/asterisk.crt
tls_private_key = /etc/asterisk/asterisk.key
tls_versions = tlsv1.2

[general]
tlsenable=yes
tlscertfile=/etc/asterisk/asterisk.crt
tlsprivatekey=/etc/asterisk/asterisk.key

type=transport
protocol=tls
bind=0.0.0.0:5061 ; le port 5061
cert_file=/etc/asterisk/keys/ ; la localisation des clefs et certificats
asterisk.pem
priv_key_file=/etc/asterisk/keys/asterisk.key
ca_list_file=/etc/asterisk/keys/ca.crt
method=tlsv1_2
verify_client=no ; les verifications coté client side
verify_server=yes ; les verifications coté server side

```

## 9. Configuration du Serveur Vocal Interactif (IVR)

Pour mettre en place un système d'automatisation avec Asterisk, nous avons configuré un serveur vocal interactif (IVR) et c'est un automate d'appel pour la prospection.

Voici comment on peut structurer cette partie du projet et configurer votre serveur Asterisk.

Pour configurer l'IVR, nous avons modifié le fichier `extensions.conf` d'Asterisk. Voici un exemple de configuration :

```

[default]
exten => s,1,Answer()
same => n,Background(welcome) ; Joue un message de bienvenue
same => n,WaitExten(5) ; Attend 5 secondes pour une entrée

```

exten ⇒ 1,1,Playback(service-compte) ; Message pour le service comptes

same ⇒ n,WaitExten(5)

exten ⇒ 2,1,Playback(service-rh) ; Message pour le service RH

same ⇒ n,WaitExten(5)

exten ⇒ 3,1,Queue(operators) ; Met en file d'attente pour parler à un opérateur

exten ⇒ 4,1,Playback(info-generales) ; Message pour les informations générales

same ⇒ n,Hangup()

exten ⇒ i,1,Playback(invalid) ; Message pour une entrée invalide

same ⇒ n,Goto(s,1)

exten ⇒ t,1,Playback(vm-goodbye) ; Message d'au revoir

same ⇒ n,Hangup()

Plan d'appel

poste 1000 Chuck Norris

poste 1001 Bruce Lee

poste 1002 Shewbacca

\*99 activation DnD

\*100 désactivation DnD

N°IVR 900

Choix 1 poste Chuck Norris

Choix 2 poste Bruce Lee

Choix 3 messagerie Chuck Norris

Choix 4 messagerie Bruce Lee

Choix 5 Shewbacca

Choix 9 Jul (passe un message et raccroche)

## 10. Automate d'Appel pour la Prospection

Pour mettre en place un automate d'appel qui randomise les appels à partir d'une liste de contacts, vous pouvez utiliser un script en combinaison avec Asterisk. Voici comment procéder :

1. **Créer un Fichier CSV** : Créez un fichier CSV contenant les contacts à appeler. Par exemple,

**contacts.csv** :

Nom,Numéro

ChuckNorris,123456789

BruceLee,987654321

Chewbacca,456789123

2. **Script de Prospection** : Créez un script qui lit le fichier CSV et passe les appels. Voici un exemple de script en Bash :

```
#!/bin/bash
```

Chemin vers le fichier CSV

```
CSV_FILE="contacts.csv"
```

Lire le fichier CSV et passer les appels

```
while IFS=, read -r nom numero
do
# Utiliser Asterisk pour passer l'appel
asterisk -rx "channel originate SIP/${numero} extension 1001@default"
sleep 5 # Attendre 5 secondes entre les appels
done < <(tail -n +2 $CSV_FILE) # Ignorer l'en-tête
```

3. **Configuration d'Asterisk pour l'Automate d'Appel** : Assurez-vous que votre fichier `extensions.conf` est configuré pour gérer les appels entrants sur l'extension 1001.

## Déploiement automatisé

Nous avons créer un fichier bash .sh

qui nous permet d'installer asterisk avec une configuration minimal fonctionnelle sur une machine propre

```
#!/bin/bash
```

```
apt-get update && apt-get -y dist-upgrade
```

```
apt install make -y
```

```
apt install build-essential -y
```

```
apt install git -y
```

```
apt install curl -y
```

```
apt install wget -y
```

```
apt install libjansson-dev -y
```

```
apt install libxml2-dev -y
```

```
apt install libsqlite3-dev -y
```

```
apt install uuid-dev -y
```

```
apt install pkg-config -y
```

```
apt install subversion -y
```

```
apt install libedit-dev -y
```

```

echo " On viens de mettre a jour et télécharger les paquets installés"
cd home/user/src
echo "On change de repertoire de travail le workdir"
wget https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-22.2.0.tar.gz
echo "On télécharge la derniere version stable de asterisk"
tar xzvf asterisk-22.2.0.tar.gz
echo "On décompresse le fichier tar"
asterisk-22.2.0/contrib/scripts/intall_prereq install
echo "On lance les install_prereq install"
asterisk-22.2.0/contrib/scripts/get_mp3_source.sh
echo "On lance get_mp3_source.sh pour les codecs audios"
cd asterisk-22.2.0
wget https://raw.githubusercontent.com/asterisk/third-party/master/pjproject/2.10/pjproject-2.10.tar.bz2
echo "On télécharge pjproject "
./configure./
echo "On lance la configuration"
make menuselect
echo "Choisir les extensions ,les codecs et on continue"
make
make -j2cdc
make install
make samples
echo "Ajout extensions"
mkdir /etc/asterisk/samples
mv /etc/asterisk/.
/etc/asterisk/samples/
echo "Déplacement du fichier"
make basic-pbx
make config
echo "Configuration finale"
systemctl enable asterisk.service
systemctl start asterisk.service
echo "Redémarrage asterisk"
systemctl status asterisk.service

```

```
echo "Vérification du statut"
```

```
asterisk -rvvvv
```

```
echo " lancement du service asterisk"
```

Fichier CSV :

1 )création fichier csv avec utilisateurs

```
Extension,Nom,MotDePasse  
1050,Jean-Bombeur,1234  
1051,Alex Terrieur,1234  
1052,Alain Terrieur,1234
```

2 )création fichier script

```
#!/bin/bash
CSV_FILE="/home/bastien/fichier_csv.csv"
PJSIP_CONF="/etc/asterisk/pjsip.conf"

if [ ! -f "$CSV_FILE" ]; then
    echo "Fichier CSV non trouvé !"
    exit 1
fi

if [ -f "$PJSIP_CONF" ]; then
    cp $PJSIP_CONF "$PJSIP_CONF.back"
else
    echo "Le fichier $PJSIP_CONF n'existe pas, création d'un nouveau."
    touch "$PJSIP_CONF"
fi

tail -n +2 "$CSV_FILE" | while IFS=',' read -r extension nom mdp
do
    echo ";===== " >> $PJSIP_CONF
    echo "[${extension}]" >> $PJSIP_CONF
    echo "type=endpoint" >> $PJSIP_CONF
    echo "context=lab" >> $PJSIP_CONF
    echo "disallow=all" >> $PJSIP_CONF
    echo "allow=ulaw,alaw" >> $PJSIP_CONF
    echo "auth=auth${extension}" >> $PJSIP_CONF
    echo "aors=${extension}" >> $PJSIP_CONF
    echo "callerid=\"${nom}\" <${extension}>" >> $PJSIP_CONF
    echo "" >> $PJSIP_CONF
    echo "[auth${extension}]" >> $PJSIP_CONF
    echo "type=auth" >> $PJSIP_CONF
    echo "auth_type=userpass" >> $PJSIP_CONF
    echo "username=${extension}" >> $PJSIP_CONF
    echo "password=${mdp}" >> $PJSIP_CONF
    echo "" >> $PJSIP_CONF
    echo "[${extension}]" >> $PJSIP_CONF
    echo "type=aors" >> $PJSIP_CONF
    echo "max_contacts=1" >> $PJSIP_CONF
    echo "remove_existing=yes" >> $PJSIP_CONF
    echo "" >> $PJSIP_CONF
    echo ";===== " >> $PJSIP_CONF
done

asterisk -rx "pjsip reload"

echo "Utilisateurs ajoutés avec succès !"
```

## LDAP

1) Installer les paquets nécessaire

- ldap-utils
- libldap2-dev
- asterisk-ldap

2) configuration du fichier res\_ldap.conf

```
[general]
enabled = yes
server=ldap://192.168.169.100/
basedn=ou=groups,dc=bastien,dc=com
binddn=cn=admin,dc=bastien,dc=com
bindauth=1234
filter=(objectClass=asteriskSIPUser)
extension_attr=asteriskExtension
callerid_attr=cn
secret_attr=userPassword
```

3) Modification fichier ldap pour compatibilité asterisk

```
dn: uid=alice,ou=people,dc=bastien,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: asteriskSIPUser
cn: Alice
sn: users
uid: alice
asteriskExtension: 1053
uidNumber: 1001
gidNumber: 1001
homeDirectory: /home/alice
loginShell: /bin/bash
userPassword: 1234
```

```
dn: uid=bob,ou=people,dc=bastien,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: asteriskSIPUser
cn: Bob
sn: developers
uid: bob
asteriskExtension: 1054
uidNumber: 1002
gidNumber: 1002
homeDirectory: /home/bob
loginShell: /bin/bash
userPassword: 1234
```

```
dn: uid=claire,ou=people,dc=bastien,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: asteriskSIPUser
cn: Claire
sn: admins
uid: claire
asteriskExtension: 1055
uidNumber: 1003
gidNumber: 1003
homeDirectory: /home/claire
loginShell: /bin/bash
userPassword: 1234
```

#### 4)Création script bash pour synchroniser LDAP et Asterisk



```
#!/bin/bash
LDAP_SERVER="ldap://192.168.169.100"
BASE_DN="ou=groups,dc=bastien,dc=com"
BIND_DN="cn=admin,dc=bastien,dc=com"
BIND_PWD="1234"
PJSIP_CONF="/etc/asterisk/pjsip.conf"

cp "$PJSIP_CONF" "$PJSIP_CONF.bak"

ldapsearch -x -H "$LDAP_SERVER" -D "$BIND_DN" -w "$BIND_PWD" -b "$BASE_DN" "(objectClass=asteriskSIPUser)" cn asteriskExtension userPassword | while read -r line
do
    case "$line" in
        cn:*) NOM="${line#cn: }" ;;
        asteriskExtension:*) EXTENSION="${line#asteriskExtension: }" ;;
        userPassword:*) MDP="${line#userPassword: }" ;;
        *) ;;
    esac
    if [[ -n "$EXTENSION" && -n "$NOM" && -n "$MDP" ]]; then
        {
            echo "[${EXTENSION}]"
            echo "===== "
            echo "type=endpoint"
            echo "context=lab"
            echo "disallow=all"
            echo "allow=ulaw,alaw"
            echo "auth=auth${EXTENSION}"
            echo "aors=${EXTENSION}"
            echo "callerid=\"${NOM}\" <${EXTENSION}>"
            echo ""
            echo "[auth${EXTENSION}]"
            echo "type=auth"
            echo "auth type=userpass"
            echo "username=${EXTENSION}"
            echo "password=${MDP}"
            echo ""
            echo "[${EXTENSION}]"
            echo "type=aors"
            echo "max_contacts=1"
            echo "remove_existing=yes"
            echo ""
            echo "===== "
        } >> "$PJSIP_CONF"
        fi
        EXTENSION=""
        NOM=""
        MDP=""
    ;;
done
```

## 5)Vérification des fichiers conf dans asterisk

```
root@debianldap:/etc/asterisk# ls
asterisk.conf  cdr_custom.conf  extensions.conf  logger.conf  musiconhold.conf  pjsip.conf.bak  queues.conf  res_ldap.conf  voicemail.conf
cdr.conf      confbridge.conf  indications.conf  modules.conf  pjsip.conf       pjsip_notify.conf  README      samples
```

Pour nos tests nous avons choisi d'installer sur nos téléphone l'applications port sip pour les appels sécurisés nous avons configuré le tls dans le programme .

## Existe-t-il un moyen de standardiser l'installation complète sous forme de containers ? OUI

Pour répondre a cette partie , nous avons installer un module pour debian qui se nomme SNAPD

il est natif sur système ubuntu mais pas sur debian

il a la particularité d'installer Maria DB, Docker , les modules ssh, SQL Lite et PHP a manière d'un LAMP (Linux, apache, my sql , PHP) ce qui ensuite nous permet de standardiser la création de container sous la forme d'un container docker

de plus pour être plus efficace nous avons installer le module DIVE

qui permet de vérifier le contenu et les layers d'un container

apt install dive

```
19/02/2025 21:03.35 /home/mobaxterm ssh user@192.168.1.19
user@192.168.1.19's password:
Linux debian-voip 6.1.0-31-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.128-1 (2025-02-07) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Feb 19 21:04:29 2025
user@debian-voip:~$ su
Mot de passe :
root@debian-voip:/home/user# sudo snap install docker
docker 27.2.0 from Canonical✓ installed
root@debian-voip:/home/user#
```

donc :

apt install snapd, apt upgrade && update

apt install dive

touch dockerfile

le contenu paramétrable du fichier dockerfile :

FROM debian:latest

## Install dependencies

```
RUN apt-get update && apt-get install -y \
python3 \
python3-pip \
&& rm -rf /var/lib/apt/lists/*
```

## Set working directory

WORKDIR /app

## Copy your application files

COPY . .

## Command to run when container starts

CMD ["python3", "app.py"] ; cette partie est entièrement paramétrable

sudo docker build -t mon\_image\_server\_voip

sudo docker images , en images :

docker images :

```
root@debian-voip:/home/my-docker-project# systemctl status asterisk.service
• asterisk.service - LSB: Asterisk PBX
   Loaded: loaded (/etc/init.d/asterisk; generated)
   Active: active (running) since Wed 2025-02-19 21:04:22 CET; 46min ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 35 (limit: 3704)
   Memory: 68.8M
      CPU: 26.123s
   CGroup: /system.slice/asterisk.service
           └─789 /usr/sbin/asterisk

févr. 19 21:04:22 debian-voip systemd[1]: Starting asterisk.service - LSB: Asterisk PBX...
févr. 19 21:04:22 debian-voip asterisk[746]: Starting Asterisk PBX: asterisk.
févr. 19 21:04:22 debian-voip systemd[1]: Started asterisk.service - LSB: Asterisk PBX.
root@debian-voip:/home/my-docker-project# docker images
REPOSITORY          TAG       IMAGE ID       CREATED        SIZE
mon_serveur_voip    latest    8aaee9663984   15 minutes ago 611MB
root@debian-voip:/home/my-docker-project#
```

le container mon serveur voip une fois la création faite avec le service asterisk en arrière plan

Dive en action avec notre container :

Layers				Current Layer Contents			
Comp	Size	Command		Permission	UID:GID	Size	Filetree
116 MB	FROM blobs			-rw-r--r--	0:0	6.0 kB	expat_external.h
494 MB	RUN /bin/sh -c apt-get update && apt-get install -y python3			-rw-r--r--	0:0	1.4 kB	fcntl.h
0 B	WORKDIR /app			-rw-r--r--	0:0	1.4 kB	features-time64.h
297 B	COPY . # buildkit			-rw-r--r--	0:0	18 kB	features.h
				-rw-r--r--	0:0	5.8 kB	fcntl.h
				-rw-r--r--	0:0	7.7 kB	fcntl.h
				-rw-r--r--	0:0	7.7 kB	fcntl.h
				-rw-r--r--	0:0	3.2 kB	fcntl.h
				-rw-r--r--	0:0	2.3 kB	fcntl.h
				-rw-r--r--	0:0	3.1 kB	fcntl.h
				-rw-r--r--	0:0	9.6 kB	fcntl.h
				-rw-r--r--	0:0	6.3 kB	fcntl.h
				-rw-r--r--	0:0	4.2 kB	fcntl.h
				-rw-r--r--	0:0	1.5 kB	fcntl.h
				-rw-r--r--	0:0	7.3 kB	fcntl.h
				-rw-r--r--	0:0	2.3 kB	fcntl.h
				-rw-r--r--	0:0	2.9 kB	fcntl.h
				-rw-r--r--	0:0	6.8 kB	fcntl.h
				-rw-r--r--	0:0	4.7 kB	fcntl.h
				-rw-r--r--	0:0	1.9 kB	fcntl.h
				-rw-r--r--	0:0	2.8 kB	fcntl.h
				-rw-r--r--	0:0	8.3 kB	fcntl.h
				-rw-r--r--	0:0	18 kB	fcntl.h
				-rw-r--r--	0:0	126 B	fcntl.h
				-rw-r--r--	0:0	1.4 kB	fcntl.h
				-rw-r--r--	0:0	4.6 kB	fcntl.h
				-rw-r--r--	0:0	5.7 kB	fcntl.h
				-rw-r--r--	0:0	7.8 kB	fcntl.h
				-rw-r--r--	0:0	4.7 MB	fcntl.h
				-rw-r--r--	0:0	6.9 kB	fcntl.h
				-rw-r--r--	0:0	3.9 kB	fcntl.h
				-rw-r--r--	0:0	10 kB	fcntl.h
				-rw-r--r--	0:0	1.1 kB	fcntl.h
				-rw-r--r--	0:0	993 B	fcntl.h
				-rw-r--r--	0:0	1.6 kB	fcntl.h
				-rw-r--r--	0:0	4.0 kB	fcntl.h
				-rw-r--r--	0:0	3.4 kB	fcntl.h
				-rw-r--r--	0:0	3.7 kB	fcntl.h
				-rw-r--r--	0:0	1.7 kB	fcntl.h
				-rw-r--r--	0:0	17 kB	fcntl.h
				-rw-r--r--	0:0	16 kB	fcntl.h
				-rw-r--r--	0:0	789 B	fcntl.h
				-rw-r--r--	0:0	3.7 kB	fcntl.h
				-rw-r--r--	0:0	213 B	fcntl.h

Layer Details			
Tags:	(unavailable)		
Id:	blobs		
Digest:	sha256:573739a5feb76df410a11dae452afdf4afcb93a5c994d6267fb314a3c1fa765		
Command:	RUN /bin/sh -c apt-get update && apt-get install -y python3 python3-pip		
	&& rm -rf /var/lib/apt/lists/* # buildkit		

Image Details			
Image name:	mon_serveur_voip		
Total Image size:	611 MB		
Potential wasted space:	3.7 MB		
Image efficiency score:	99 %		

Count	Total Space	Path
2	1.6 MB	/var/cache/debconf/templates.dat
2	1.5 MB	/var/cache/debconf/templates.dat-old
2	290 kB	/var/lib/dpkg/status
2	289 kB	/var/lib/dpkg/status-old
2	21 kB	/var/cache/debconf/config.dat
2	20 kB	/etc/ld.so.cache

```

root@debian-voip:/home/my-docker-project# docker build -t mon_serveur_voip:latest .
[+] Building 6.2s (4/8)
=> [internal] load build definition from Dockerfile                                docker:default
=> => transferring dockerfile: 336B                                              0.0s
=> [internal] load metadata for docker.io/library/debian:latest                  1.3s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                    0.0s
=> [1/4] FROM docker.io/library/debian:latest@sha256:4abf773f2a570e6873259c4e3ba16de6c62 4.8s
=> => resolve docker.io/library/debian:latest@sha256:4abf773f2a570e6873259c4e3ba16de6c62 0.0s
=> => sha256:4abf773f2a570e6873259c4e3ba16de6c6268fb571fd46ec80be7c67822 8.52kB / 8.52kB 0.0s
=> => sha256:f6008b2d1c096556ec301a0c76fe0b40657594ffcf4a43aaf63ca9d40767 1.02kB / 1.02kB 0.0s
=> => sha256:18f9bd665a29a57601ba643beeb9471f549e3ccff439252551726cddeceb233 453B / 453B 0.0s
=> => sha256:a492eee5e55976c7d3feecce4c564aaf6f14fb07fdc5019d06f4154ed 19.92MB / 48.48MB 4.8s
=> [internal] load build context                                                 0.0s
=> => transferring context: 336B                                              0.0s

```

## Tests et validations :

Appel utilisateurs 1,2,3 en bi-directionnel , appel entrant - sortant = OK

Redirection en cas de non réponses xxx secondes = OK

utilisation LDAP : OK

Messagerie : OK

Déploiement par script SH sur nouvelle machine : OK testé fonctionnel

Déploiement par container Docker sur nouvelle machine : OK

## Sécurisation :

nous tester , installer , configurer Freepbx 17 et également Front Operator Panel 2 , avec LDAP et log pass simple qui sont des applications avec front end, interface graphique

pour cela nous avons installer gnome (apt install gnome) mais nous avons priorisé une approche par

programme , applications en ligne de commandes et non graphique d'ou le fait que nous ne développons pas plus sur cette partie , chapitre.

<http://forum.fop2.com>

<https://www.freepbx.org/>

## Conclusion

Malgré quelques difficultés nous avons mener ce projet a bien et incluant l'intégration LDAP, la conteneurisation par docker avec création de script et personnalisation de la messagerie mais également la création d'un script d'installation bash et un serveur asterisk fonctionnel en bi directionnel avec messagerie et renvoie d'appel, de plus l'importation depuis .csv et l'automatisation des appels est pleinement fonctionnelle.