

VoIP

La VoIP (Voice over Internet Protocol) est une technologie qui permet de transmettre la voix et d'autres données multimédias sur Internet au lieu des réseaux de téléphonie traditionnels.

Avantages :

Coût Réduit

L'un des avantages les plus évidents de la VoIP est son coût réduit par rapport aux lignes téléphoniques traditionnelles. Les appels internationaux et longue distance sont considérablement moins chers, ce qui peut permettre des économies substantielles, surtout pour les entreprises. Cela concerne aussi les coûts opérationnels et de maintenance cependant, il est important de noter que les coûts opérationnels et de maintenance peuvent varier en fonction de divers facteurs, notamment la taille de l'entreprise, la complexité du système VoIP mis en place et les besoins spécifiques en matière de communication.

Flexibilité

La VoIP offre une grande flexibilité. Vous pouvez effectuer des appels depuis n'importe quel endroit avec une connexion Internet, ce qui est idéal pour les travailleurs à distance et les voyageurs d'affaires. De plus, la plupart des systèmes VoIP offrent des fonctionnalités avancées telles que la messagerie vocale, la vidéoconférence et la messagerie instantanée.

Fonctionnalités avancées

La VoIP propose une gamme de fonctionnalités avancées, comme la redirection d'appels, le suivi des appels, la gestion des appels en attente, la conférence téléphonique et plus encore. Ces fonctionnalités peuvent améliorer la productivité et l'efficacité des entreprises.

Évolutivité

La VoIP est facilement évolutive. Vous pouvez ajouter ou supprimer des lignes téléphoniques en fonction des besoins de votre entreprise, ce qui en fait une solution idéale pour les petites entreprises en croissance.

Intégration avec d'autres applications

La VoIP peut être intégrée à d'autres applications de communication et de gestion d'entreprise, telles que les systèmes de CRM (Customer Relationship Management), les outils de collaboration et les applications de bureau. Cela permet une communication plus fluide et une meilleure gestion des données clients (exemple : Amazon, AirBnB, Disney +)

Voici quelques techniques de chiffrement souvent utilisées pour sécuriser les informations d'authentification et les appels vocaux dans les systèmes VoIP :

Le TLS, Transport Layer Security, permet de sécuriser le trafic entrant et sortant entre les personnes appelées.

Le SRTP, Secure Real-Time Transport Protocol, crypte les paquets de données transmis pendant l'appel pour que les fraudeurs ne puissent pas les décrypter.

Enfin, le VPN, Virtual Private Network, fournit un tunnel sécurisé et crypté, qui permet d'envoyer et recevoir de la donnée en toute sécurité.

Encryption by SDES (Session Description Protocol Security Descriptions) : SDES est une méthode de chiffrement utilisée dans le protocole SIP (Session Initiation Protocol) pour sécuriser les communications VoIP. Il utilise des clés de chiffrement pour protéger les flux de données audio et vidéo entre les participants à un appel.

Inconvénients :

Dépendance à l'Internet

L'un des principaux inconvénients de la VoIP est sa dépendance à une connexion Internet stable. Si votre connexion est lente ou instable, la qualité des appels peut en souffrir, avec des interruptions et une mauvaise qualité audio.

Alimentation électrique

Contrairement aux téléphones traditionnels qui fonctionnent même en cas de panne de courant, la VoIP nécessite une alimentation électrique constante. En cas de panne de courant ou de défaillance électrique, vous risquez de perdre la communication.

Qualité de service

La qualité des appels VoIP peut varier en fonction de divers facteurs, notamment la bande passante, la congestion du réseau et la qualité des équipements. Il est essentiel de s'assurer que votre réseau est configuré pour fournir une qualité de service optimale.

Interopérabilité

La compatibilité entre les différents systèmes VoIP peut être un défi, ce qui peut entraîner des problèmes de communication lors de l'appel de personnes utilisant des services VoIP différents.

Service d'urgence :

Les services VoIP peuvent avoir des limitations en matière d'appels d'urgence, car ils ne sont pas toujours associés à une adresse physique précise.

Voici quelques-uns des principaux concepts et protocoles associés aux serveurs VoIP :

Protocole SIP (Session Initiation Protocol) : SIP est un protocole de signalisation largement utilisé pour établir, modifier et résilier des sessions multimédias, notamment des appels vocaux, des conférences audio et des sessions de messagerie instantanée. Il est utilisé pour l'initiation des appels, la gestion des fonctions d'appel (transfert, mise en attente, renvoi d'appel, etc.) et la négociation des paramètres de communication entre les clients VoIP et les serveurs.

Protocole RTP (Real-time Transport Protocol) : RTP est un protocole utilisé pour transporter des données audio et vidéo en temps réel sur les réseaux IP. Il est souvent associé au protocole SIP pour acheminer les flux audio des appels vocaux entre les participants. RTP assure la synchronisation, le contrôle de la congestion, et la correction d'erreur pour garantir une transmission fluide des médias en temps réel.

Protocole RTCP (RTP Control Protocol) : RTCP est un protocole complémentaire à RTP, utilisé pour surveiller la qualité de service (QoS) et fournir des informations de contrôle pour les sessions RTP. RTCP permet aux participants d'un appel VoIP de collecter des statistiques sur les performances du réseau, telles que les pertes de paquets, les retards de transmission, et les goulots d'étranglement, afin d'optimiser la qualité des appels.

Protocole SDP (Session Description Protocol) : SDP est un protocole utilisé pour décrire les caractéristiques des sessions multimédias, telles que les types de médias pris en charge, les codecs utilisés, et les adresses IP des participants. Il est souvent utilisé en conjonction avec SIP pour échanger des informations de description de session lors de l'initiation des appels VoIP.

Protocole H.323 : H.323 est un ensemble de protocoles utilisés pour la communication multimédia sur les réseaux IP. Bien qu'il soit moins courant que SIP dans les nouvelles mises en œuvre VoIP, il est toujours utilisé dans certaines infrastructures existantes, notamment dans les systèmes de vidéoconférence et les réseaux d'entreprise.

Ensemble, ces concepts et protocoles fournissent les bases nécessaires pour établir des communications vocales fiables et efficaces via des serveurs VoIP, en assurant l'initiation, le transport, la surveillance et le contrôle des sessions multimédias sur les réseaux IP.

Un protocole couramment utilisé pour sécuriser et chiffrer les appels transitant par un serveur SBC (Session Border Controller) est le protocole SRTP (Secure Real-time Transport Protocol).

SRTP est une extension du protocole RTP (Real-time Transport Protocol) qui offre des fonctionnalités de chiffrement pour sécuriser les flux audio en temps réel. Il utilise des algorithmes de chiffrement robustes pour garantir la confidentialité des conversations vocales et protéger contre les écoutes clandestines.

Lorsque les appels VoIP passent par un SBC, celui-ci peut mettre en œuvre le chiffrement SRTP pour sécuriser les flux audio entre les différents participants à l'appel. SRTP assure ainsi que seules les parties autorisées peuvent accéder au contenu des appels, renforçant ainsi la confidentialité des communications.

En résumé, SRTP est le protocole couramment utilisé pour sécuriser et chiffrer les appels transitant par un serveur SBC, offrant une protection robuste contre les menaces potentielles telles que l'interception et l'écoute clandestine.

Mise en place du serveur VoIP :

1. Modification du fichier 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

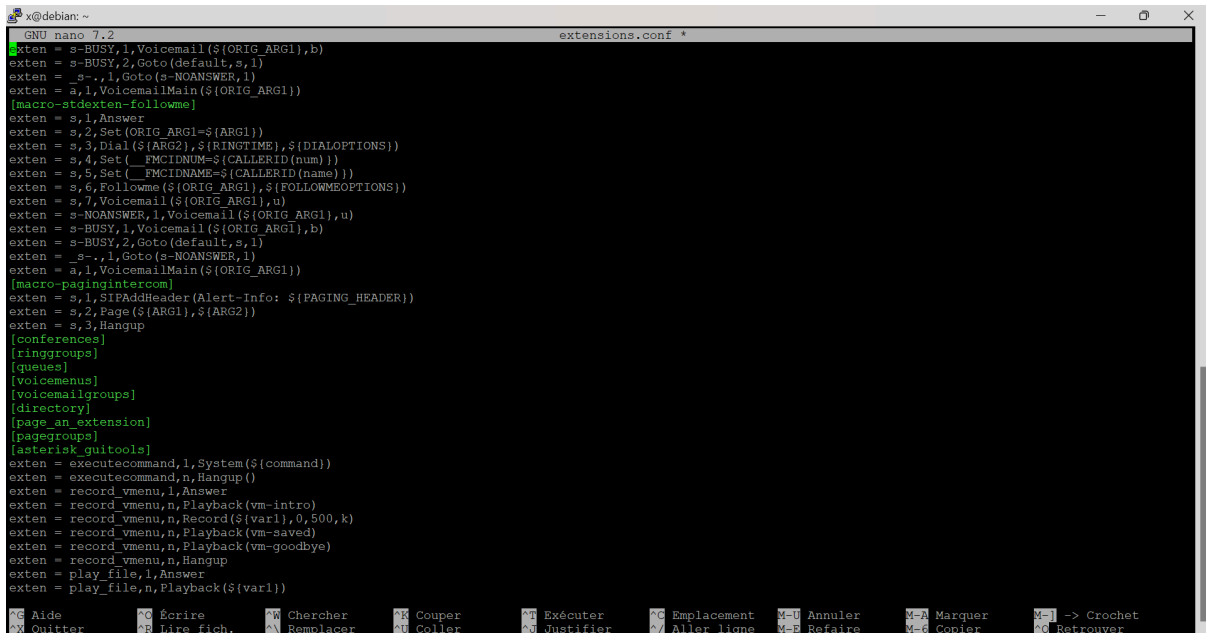
[aor_dynamic](!)
type=aor
max_contacts=1

;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
```

2. Modification du fichier extensions.conf



```
x@debian: ~
GNU nano 2.2.1 extensions.conf
exten = s-BUSY,1,VoiceMail($ORIG_ARG1),b
exten = s-BUSY,2,Goto(default,s,1)
exten = s-,1,Goto(s-NOANSWER,1)
exten = a,1,VoiceMailMain($ORIG_ARG1)
[macro-stdexten-followme]
exten = s,1,Answer
exten = s,2,Set(ORIG_ARG1=$ARG1)
exten = s,3,Dial($ARG2,$(RINGTIME),${DIALOPTIONS})
exten = s,4,Set(_FMCIDNUM=${CALLERID(num)})
exten = s,5,Set(_FMCIDNAME=${CALLERID(name)})
exten = s,6,Followme($ORIG_ARG1,${FOLLOWMEOPTIONS})
exten = s,7,VoiceMail($ORIG_ARG1),u
exten = s-NOANSWER,1,VoiceMail($ORIG_ARG1),u
exten = s-BUSY,1,VoiceMail($ORIG_ARG1),b
exten = s-BUSY,2,Goto(default,s,1)
exten = s-,1,Goto(s-NOANSWER,1)
exten = a,1,VoiceMailMain($ORIG_ARG1)
[macro-pagingintercom]
exten = s,1,SIPAddHeader(Alert-Info: ${PAGING_HEADER})
exten = s,2,Page($ARG1,${ARG2})
exten = s,3,Hangup
[conferences]
[ringgroups]
[queues]
[voicemailgroups]
[voicemailgroups]
[directory]
[page_an_extension]
[pagegroups]
[asterisk_guitools]
exten = executecommand,1,System(${command})
exten = executecommand,n,Hangup()
exten = record_vmenu,1,Answer
exten = record_vmenu,n,Playback(vm-intro)
exten = record_vmenu,n,Record(${var1},0,500,k)
exten = record_vmenu,n,Playback(vm-saved)
exten = record_vmenu,n,Playback(vm-goodbye)
exten = record_vmenu,n,Hangup
exten = play_file,1,Answer
exten = play_file,n,Playback(${var1})

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler  M-A Marquer  M-I -> Crochet
^X Quitter   ^R Lire fich.^N Remplacer ^U Collier   ^V Justifier ^Y Aller ligne M-E Refaire  M-E Copier   ^O Retrouver
```

3. Modification 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
```

4. Modification du fichier asterisk.conf

```
[options]
;verbose = 5
;debug = 2
;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.
;defaultlanguage = fr
;systemname = my_system_name ; Prefix uniqueid with a system name for
; Global uniqueness issues.
;autosystemname = yes ; Automatically set systemname to hostname,
; uses 'localhost' on failure, or systemname if
; set.
;mindtmfduration = 80 ; Set minimum DTMF duration in ms (default 80 ms)
```

Petit test :

```
root@debian:/# asterisk -rx "core restart now"
root@debian:/# ss -nlut | grep -E '5060|State'
Netid State Recv-Q Send-Q Local Address:Port Peer Address:PortProcess
udp UNCONN 0 0 0.0.0.0:5060 0.0.0.0:*
```

Petits ajustements :

```
root@debian:/tmp# git clone https://github.com/thierry-rami/Asterisk_config
Clonage dans 'Asterisk_config'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 17 (delta 5), reused 15 (delta 3), pack-reused 0
Réception d'objets: 100% (17/17), 1.75 Mio | 1.79 Mio/s, fait.
Résolution des deltas: 100% (5/5), fait.
root@debian:/tmp# cd Asterisk_config/
root@debian:/tmp/Asterisk_config# service asterisk stop
root@debian:/tmp/Asterisk_config# cp *.conf /etc/asterisk/.
root@debian:/tmp/Asterisk_config# service asterisk start
root@debian:/tmp/Asterisk_config# asterisk -rvvvvvvvvv
Asterisk 20.7.0, 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 20.7.0 currently running on debian (pid = 1477)
debian*CLI> pjsip list auths

I/OAuth: <AuthId/UserName.....>=====

Auth: alice/alice
Auth: bob/bob
Auth: martin/martin

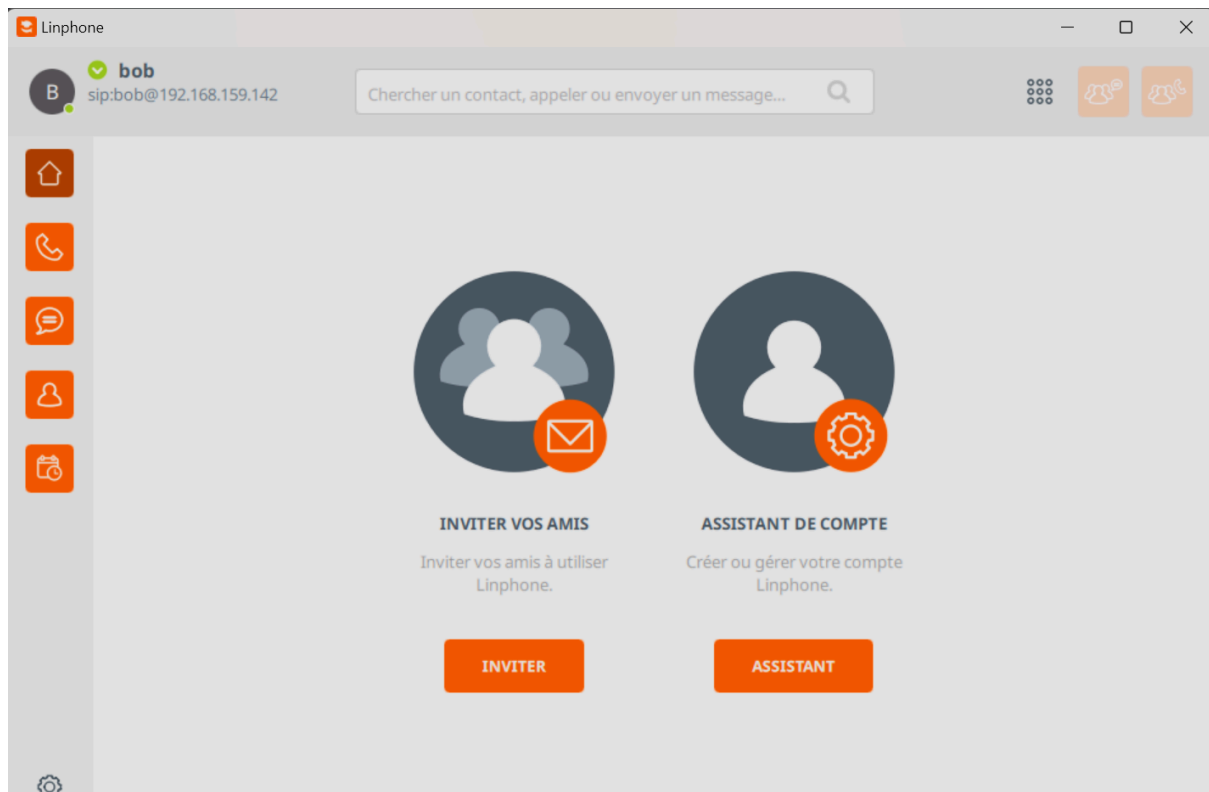
Objects found: 3
debian*CLI> pjsip list aors

Aor: <Aor.....> <MaxContact>=====

Aor: alice 1
Aor: bob 1
Aor: martin 1

Objects found: 3
```

Connexion avec des logiciels de téléphone IP :



Compte



Nom du compte

alice

Serveur SIP



Proxy SIP



Nom d'utilisateur *

alice



Domaine *

192.168.159.142



Login

alice



Mot de passe



Nom à afficher

alice



N° de la boîte vocale

6001



Préfixe d'appel



Plan de numérotation

☐ Hide Caller ID

Chiffrement

Désactivé



Transport

UDP



Adresse publique

Auto



Actualiser l'enregist...

300

Signalisation

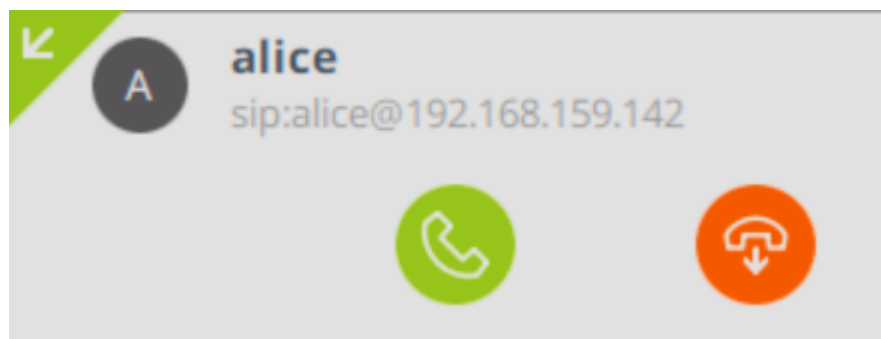
15

☐ Afficher ma présence☐ Autoriser la réécriture de l'IP☐ ICE☐ Désactiver les minuteurs de session

Sauvegarder

Annuler

Quelques appels pour tester les configurations :



Résultats : Les utilisateurs peuvent s'appeler. Résultat Ok, autrement dit les appels fonctionnent.

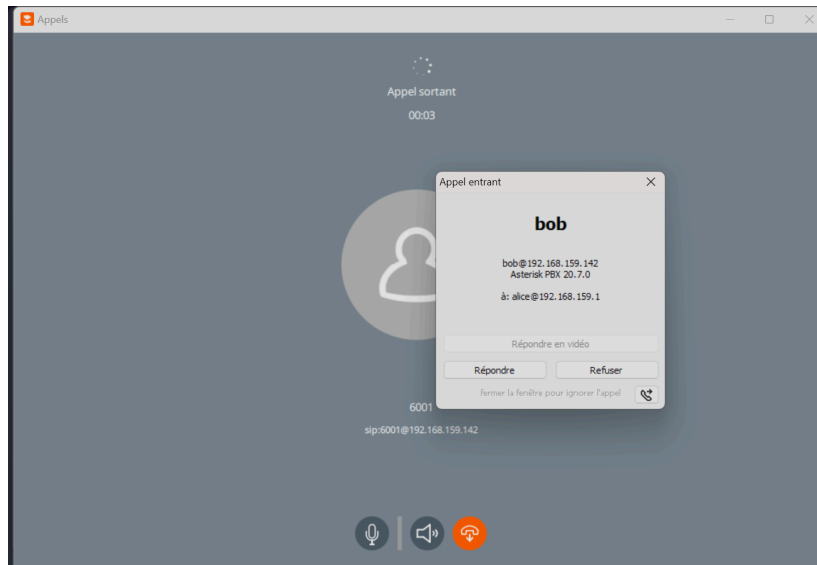
Nouveau test

Etat du système : Active

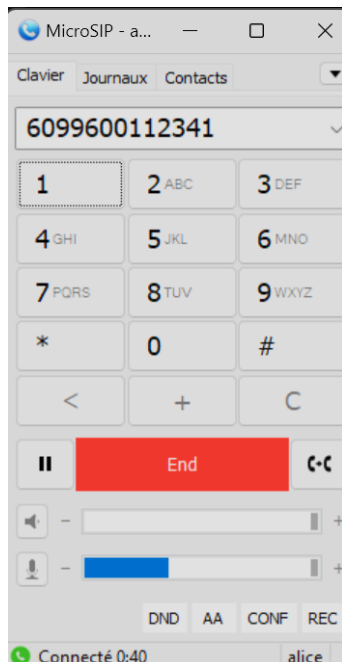
Fonction testée : Messagerie Vocale

On s'attend à ce que la messagerie vocale fonctionne.

Bob appelle Alice :



Alice appelle sa messagerie vocale :



Résultats : La messagerie vocale d'Alice a enregistré un message. Résultat Ok, autrement dit, la messagerie vocale fonctionne.

Menu principal :

- Taper 1 pour appeler Alice
- Taper 2 pour appeler Bob
- Taper 3 pour appeler Martin
- Taper 0 pour quitter

Mise en pratique :

```
root@debian:/etc/asterisk# apt install python3
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
python3 est déjà la version la plus récente (3.11.2-1+b1).
python3 passé en « installé manuellement ».
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@debian:/etc/asterisk#
```

```
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(FXSIP/alice,10)
exten => 6002,1,Dial(FXSIP/bob,10)
exten => 6003,1,Dial(FXSIP/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

exten => 8000,1,Answer()
same => n,Playback(messagelocal)
same => n,Hangup()

exten => s,2,WaitExten(2)
exten => s,3,Goto(messagelocal,s,1)

exten => 1,1,SayNumber(1)
exten => 1,2,Goto(from-internal,6001,1)

exten => 2,1,SayNumber(2)
exten => 2,2,Goto(from-internal,6002,1)

exten => 3,1,SayNumber(3)
exten => 3,2,Goto(from-internal,6003,1)

exten => 0,1,SayNumber(0)
exten => 0,2,Hangup()
```

```
x@debian: ~
root@debian:/etc/asterisk# pipx install gtts
-bash: pipx : commande introuvable
root@debian:/etc/asterisk# apt install python3-gtts
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  python3-bs4 python3-click python3-colorama python3-gtts-token python3-html5lib python3-lxml python3-soupsieve python3-webencodings
Paquets suggérés :
  python3-genshi python-lxml-doc
Les NOUVEAUX paquets suivants seront installés :
  python3-bs4 python3-click python3-colorama python3-gtts python3-gtts-token python3-html5lib python3-lxml python3-soupsieve python3-webencodings
0 mis à jour, 9 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 10618 ko dans les archives.
Après cette opération, 60764 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] o
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 python3-soupsieve all 2.3.2-1 [38,8 kB]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 python3-bs4 all 4.11.2-2 [121 kB]
Réception de :3 http://deb.debian.org/debian bookworm/main amd64 python3-colorama all 0.4.6-2 [36,8 kB]
Réception de :4 http://deb.debian.org/debian bookworm/main amd64 python3-click all 8.1.3-2 [92,2 kB]
Réception de :5 http://deb.debian.org/debian bookworm/main amd64 python3-gtts-token all 1.1.3-3 [50280 B]
Réception de :6 http://deb.debian.org/debian bookworm/main amd64 python3-gtts all 2.0.3-4 [26,1 kB]
Réception de :7 http://deb.debian.org/debian bookworm/main amd64 python3-webencodings all 0.5.1-5 [11,1 kB]
Réception de :8 http://deb.debian.org/debian bookworm/main amd64 python3-html5lib all 1.1-3 [93,0 kB]
Réception de :9 http://deb.debian.org/debian bookworm/main amd64 python3-lxml amd64 4.9.2-1+b1 [10194 kB]
10618 ko réceptionnés en 1s (10719 ko/s)
Sélection du paquet python3-soupsieve précédemment désélectionné.
(Lecture de la base de données... 57989 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../0-python3-soupsieve_2.3.2-1_all.deb ...
Dépaquetage de python3-soupsieve (2.3.2-1) ...
Sélection du paquet python3-bs4 précédemment désélectionné.
Préparation du dépaquetage de .../1-python3-bs4_4.11.2-2_all.deb ...
Dépaquetage de python3-bs4 (4.11.2-2) ...
Sélection du paquet python3-colorama précédemment désélectionné.
Préparation du dépaquetage de .../2-python3-colorama_0.4.6-2_all.deb ...
Dépaquetage de python3-colorama (0.4.6-2) ...
Sélection du paquet python3-click précédemment désélectionné.
Préparation du dépaquetage de .../3-python3-click_8.1.3-2_all.deb ...
Dépaquetage de python3-click (8.1.3-2) ...
Sélection du paquet python3-gtts-token précédemment désélectionné.
Préparation du dépaquetage de .../4-python3-gtts-token_1.1.3-3_all.deb ...
Dépaquetage de python3-gtts-token (1.1.3-3) ...
Sélection du paquet python3-gtts précédemment désélectionné.
Préparation du dépaquetage de .../5-python3-gtts_2.0.3-4_all.deb ...
```

```
root@debian:/etc/asterisk# gtts-cli "asterisk parle en français, faite 1 pour Alice, 2 pour Bob, 3 pour Martin, 0 pour quitter" -l fr --output /tmp/messagegcal.mp3
root@debian:/etc/asterisk# sox -V /tmp/messagegcal.mp3 -r 8000 -c 1 -t ul /var/lib/asterisk/sounds/fr/messagegcal.ulaw
sox:      SoX v14.4.2

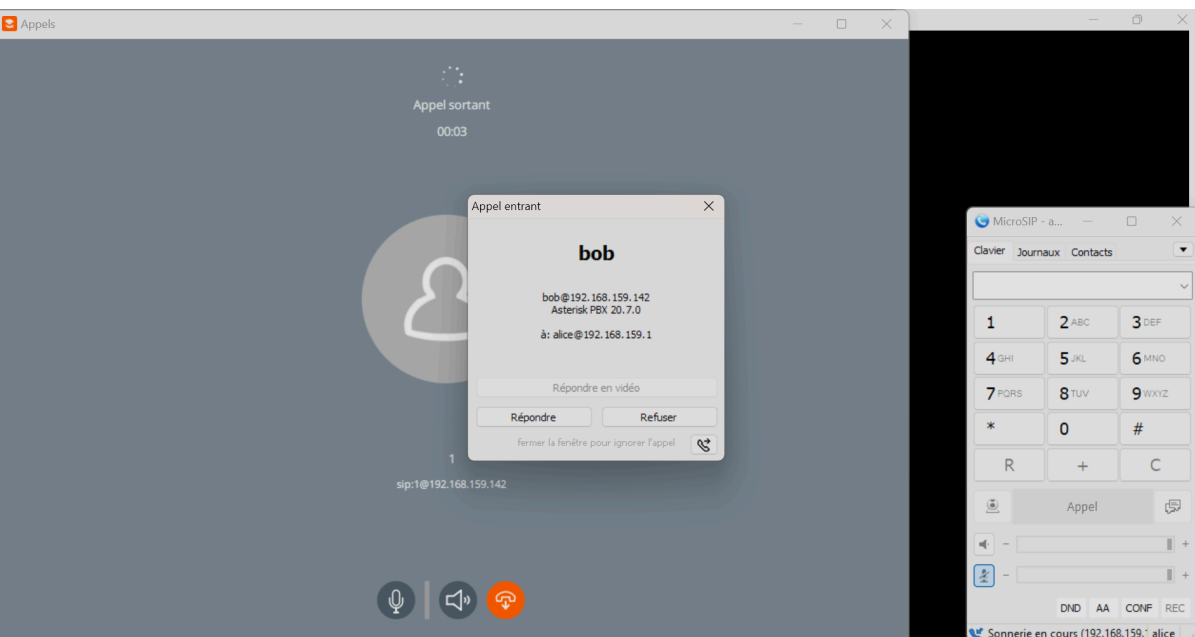
Input File      : '/tmp/messagegcal.mp3'
Channels        : 1
Sample Rate     : 24000
Precision       : 16-bit
Duration        : 00:00:08.11 = 194688 samples ~ 608.4 CDDA sectors
File Size       : 32.4k
Bit Rate        : 32.0k
Sample Encoding : MPEG audio (layer I, II or III)

sox INFO sox: Overwriting '/var/lib/asterisk/sounds/fr/messagegcal.ulaw'

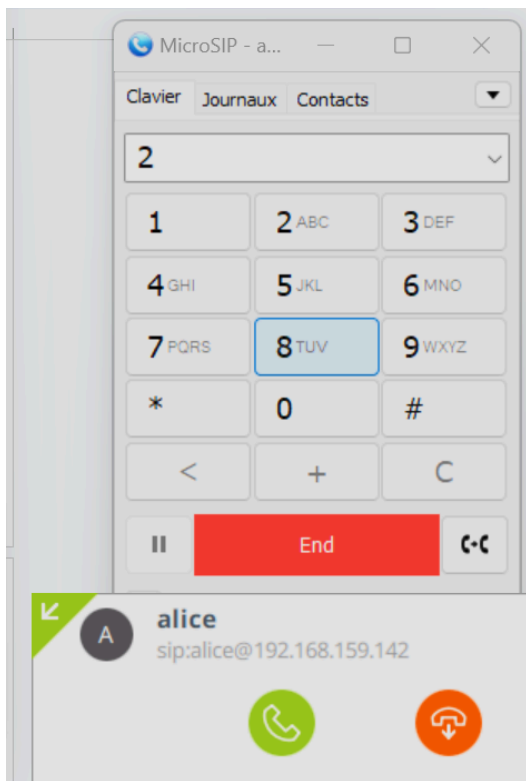
Output File     : '/var/lib/asterisk/sounds/fr/messagegcal.ulaw' (ul)
Channels        : 1
Sample Rate     : 8000
Precision       : 14-bit
Duration        : 00:00:08.11 = 64896 samples ~ 608.4 CDDA sectors
Sample Encoding : 8-bit u-law
Reverse Nibbles : no
Reverse Bits    : no
Comment        : 'Processed 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
```

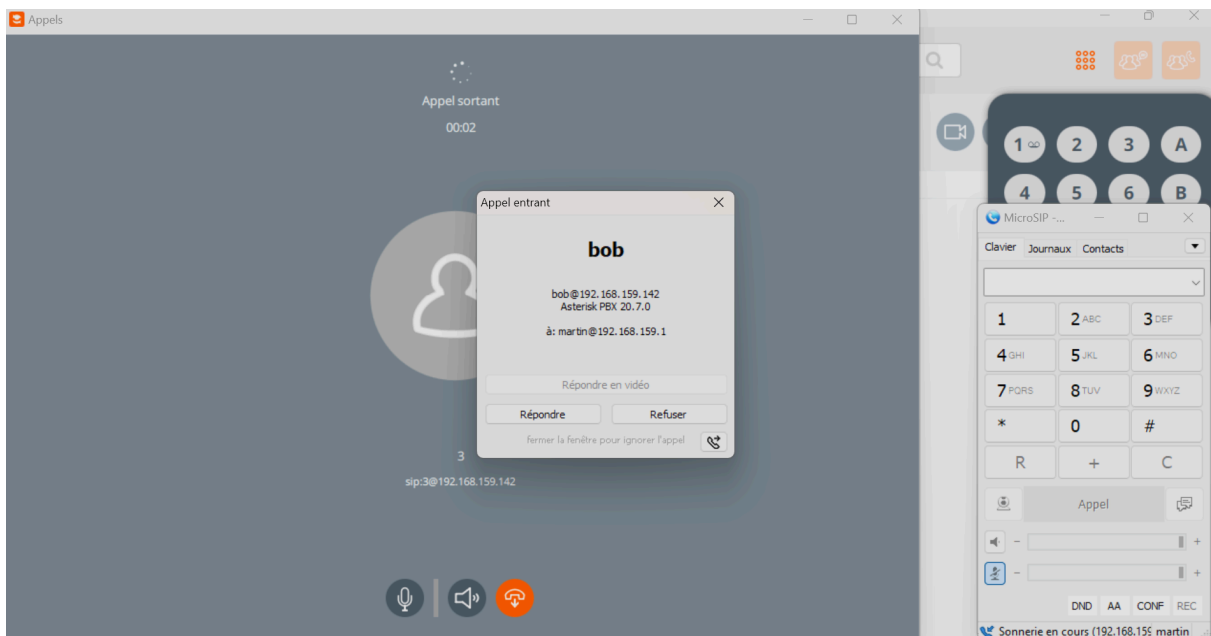
J'appelle Alice en tapant 1 :



J'appelle Bob en tapant 2 :



J'appelle Martin en tapant 3 :



Résultats : Ok, ce qui veut dire que le menu virtuel fonctionne.

Mise en place d'un

automate d'appel qui randomise les appels via un fichier CSV et un script :

1. Création du fichier CSV

```
GNU nano 7.2 utilisateurs.csv *
Nom,Mot de passe,Numéro
Alice,bonjour,1
Bob,bonjour,2
Martin,bonjour,3
test,bonjour,4
```

2. Création du script

```
GNU nano 7.2 generate_configurations.py
import csv

# Fonction pour générer la configuration de extensions.conf
def generate_extensions_conf(csv_file):
    extensions_conf = ""
    with open(csv_file, newline='') as csvfile:
        reader = csv.DictReader(csvfile)
        for row in reader:
            nom = row['Nom']
            numero = row['Numéro']
            extensions_conf += f"exten => {numero},1,NoOp(Call to extension {nom})\n"
            extensions_conf += f"same => n,Dial(PJSIP/{nom})\n"
            extensions_conf += f"same => n,Hangup()\n"
    return extensions_conf

# Fonction pour générer la configuration de pjsip.conf
def generate_pjsip_conf(csv_file):
    pjsip_conf = ""
    with open(csv_file, newline='') as csvfile:
        reader = csv.DictReader(csvfile)
        for row in reader:
            nom = row['Nom']
            password = row['Mot de passe']
            pjsip_conf += f"[{nom}]\n"
            pjsip_conf += f"type=endpoint\n"
            pjsip_conf += f"context=from-internal\n"
            pjsip_conf += f"disallow=all\n"
            pjsip_conf += f"allow=ulaw\n"
            pjsip_conf += f"auth={nom}\n"
            pjsip_conf += f"aors={nom}\n\n"

            pjsip_conf += f"[{nom}]\n"
            pjsip_conf += f"type=auth\n"
            pjsip_conf += f"auth_type=userpass\n"
            pjsip_conf += f"password={password}\n"
            pjsip_conf += f"username={nom}\n\n"

            pjsip_conf += f"[{nom}]\n"
            pjsip_conf += f"type=aor\n"
            pjsip_conf += f"max_contacts=1\n\n"

# Aide, Quitter, Écrire, Lire fich., Chercher, Remplacer, Couper, Coller, Exécuter, Justifier, Emplacement, Aller ligne, Annuler, Refaire, Marquer, Copier, -> Crochet, Retrouver
```

```
return pjsip_conf

# Chemin vers le fichier CSV contenant les utilisateurs
csv_file = "/etc/asterisk/samples/utilisateurs.csv"

# Générer les configurations pour extensions.conf et pjsip.conf
extensions_conf = generate_extensions_conf(csv_file)
pjsip_conf = generate_pjsip_conf(csv_file)

# print (pjsip_conf)

# Écrire les configurations dans les fichiers correspondants
with open('/etc/asterisk/extensions.conf', 'w') as file:
    file.write(extensions_conf)

with open('/etc/asterisk/pjsip.conf', 'w') as file:
    file.write(pjsip_conf)

print("Configurations générées avec succès.")
```

Petit test :

La partie “generate fichiers.conf” fonctionne mais les fichiers pjsip.conf et extensions.conf ne sont pas modifiés, ce qui est contraire aux attentes du test.

Résultat : Échec