Table des matières

Configuration du réseau. 2
Configuration du hostname 2
Configuration carte réseau 2
Ou Redirection NAT 3
Configuration dns 4
Mise à jour. 4
Installation d'asterisk. 5
Générer le certificat pour le chiffrement tls. 6
Configurer PJSIP 6
Backup du fichier pjsip.conf 6
Configuration du mode de transport 7
Configuration des endpoints 7
Configurer le fichier extension.conf 9
Backup extension.conf 9
Configuration DND 9
Configuration des appelles 9
Configuration du menu 11
Configuration automatique de prospection. 11
Configuration Voicemail 12
Backup de voicemail.conf 12
Configuration de mot de passe de la boîte vocale 12
Configurer modules.conf 12
Backup modules.conf 12
Configuration module.conf 13
Backup ari.conf 13
Vérifier la configuration des endpoints 13
Configuration Micro SIP. 14
Chiffrement transport layer security 15
Installation fail2ban 17
Backup fail2ban 17
Configuration fail2ban 17
Redémarrer le daemon fail2ban 17
AsteriskOnFly 18
Configuration Zabbix 19
Intégration Server LDAP 20
Documentation attendu 22

Installation serveur VoIP.

Configuration du réseau.

Configuration du hostname

Changer le nom de la machine.
adrien@VoIP:~$ sudo hostname VoIP
Redémarre le serveur.
adrien@VoIP:~$ sudo reboot

Configuration carte réseau

Permet d'éditer le fichier interfaces.
adrien@VoIP:~$ sudo nano /etc/network/interfaces

```
# source /etc/network/interfaces.d/*: Inclut les fichiers de configuration supplémentaires du répertoire /etc/network/interfaces.d/.

auto lo ; Active automatiquement l'interface de bouclage (loopback).
iface lo inet loopback ; Configure l'interface de bouclage pour utiliser l'adressage de bouclage.
allow-hotplug ens33 ; Autorise la connexion à chaud pour l'interface ens33.
iface ens33 inet static ; Configure l'interface ens33 avec une adresse IP statique.
 address 172.31.254.250/16 ; Adresse IP de l'interface avec le masque de sous-réseau /16.
 netmask 255.255.0.0 ; Masque de sous-réseau.
 gateway 172.31.254.254 ; Passerelle par défaut.
 broadcast 172.31.255.255 ; Adresse de diffusion.
 dns-nameservers 172.31.254.254 ; Serveur DNS.

allow-hotplug ens37 ; Autorise la connexion à chaud pour l'interface ens37.
iface ens37 inet dhcp ; Configure l'interface ens37 pour obtenir une adresse IP via DHCP.
```

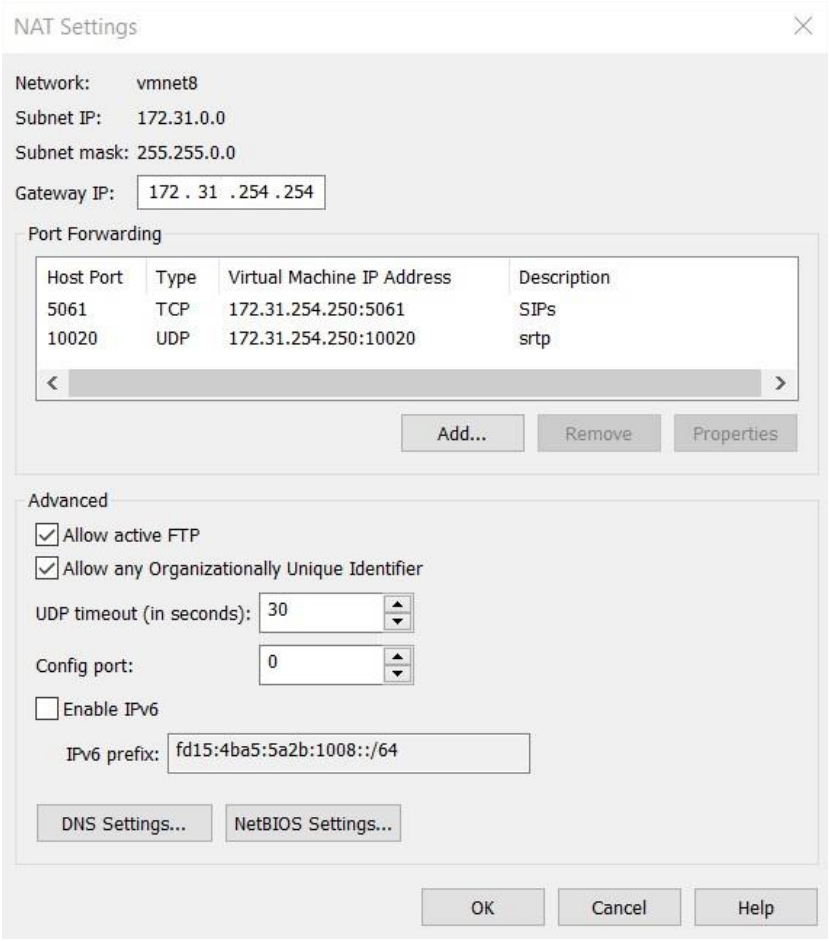
---

adrien@VoIP:~$ sudo systemctl restart networking[1]

Pour que la connexion au niveau de la deuxième interface il faut un service dhcp.

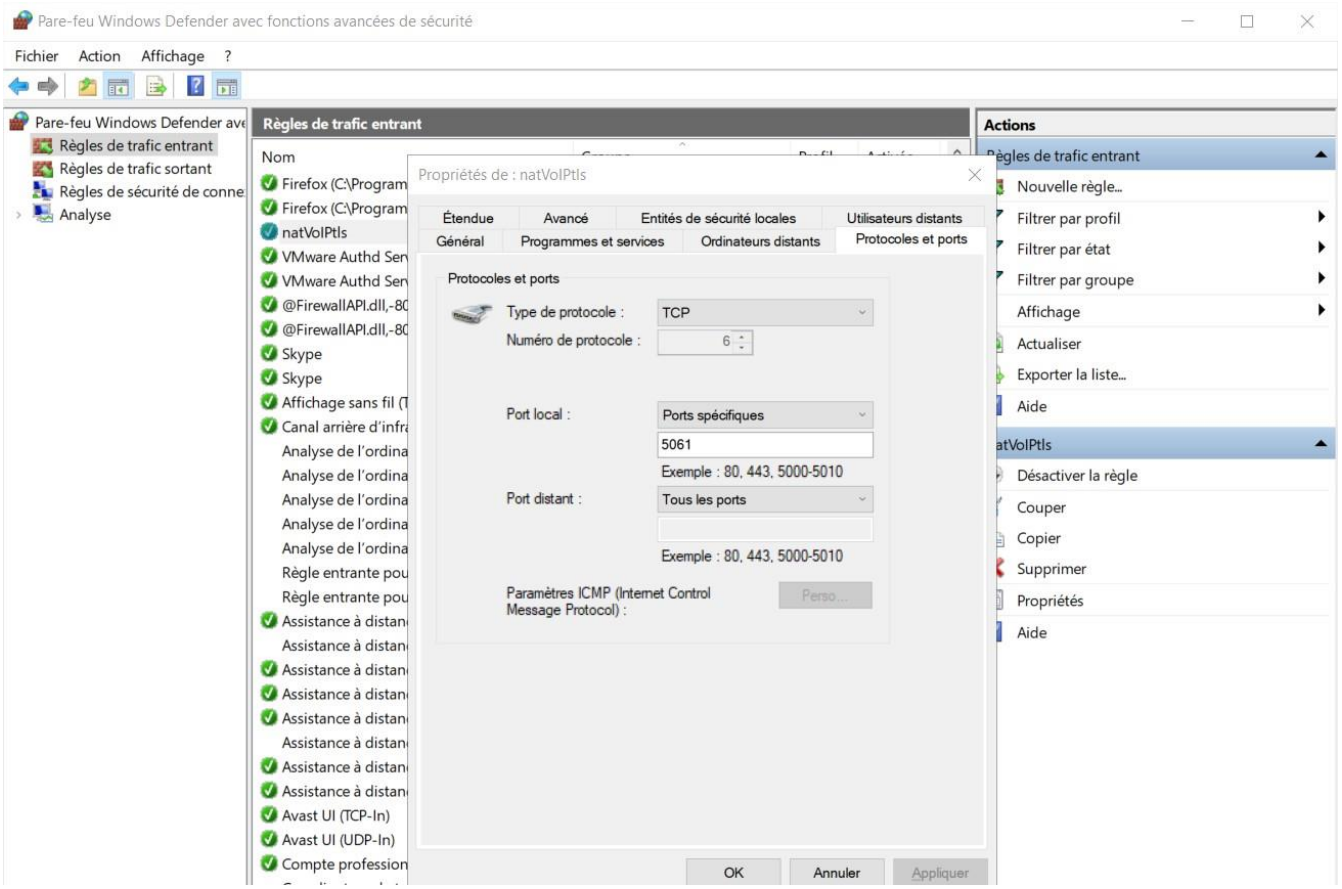
Ou Redirection NAT

Si vous ne voulez pas rajouter une deuxième carte réseau vous pouvez aussi faire une redirection NAT au niveau de VMWare comme ci-dessous.



Ensuite ouvrez le port sips sur votre machine physique.

1



Configuration dns

Permet d'éditer le fichier resolv.conf
adrien@VoIP:~$ sudo nano /etc/resolv.conf

```
nameserver 172.31.254.251 ; dns préférer (active directory).
nameserver 172.31.254.254 ; dns secondaire.
```

Permet de rendre immuable le fichier resolv.conf
adrien@VoIP:~$ sudo chattr +i /etc/resolv.conf
adrien@VoIP:~$ sudo systemctl restart networking

Mise à jour.

Mise a jours des paquets et de la distribution debian.
adrien@VoIP:~$ sudo apt-get update -y && sudo apt-get dist-upgrade -y
L'installation des librairies est nécessaire pour l'installation de tls et srtp.
adrien@VoIP:~$ sudo apt-get install libssl-dev libsrtp2-dev
adrien@VoIP:~$ sudo apt-get install libnon27
adrien@VoIP:~$ sudo updatedb

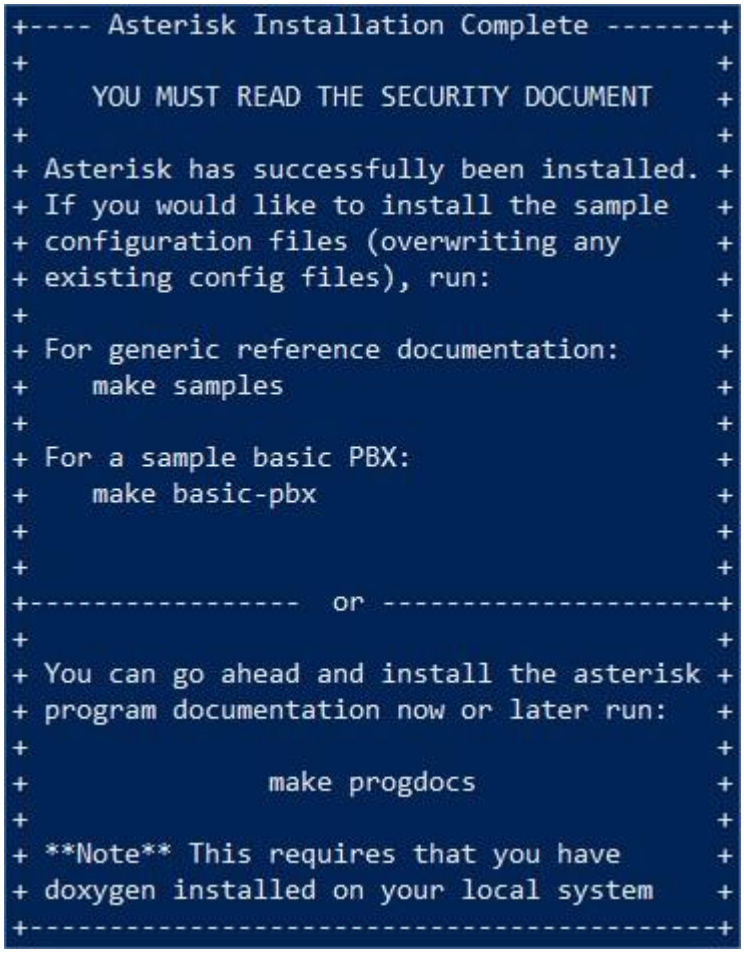
```
adrien@VoIP: $ locate libneon.so.27
/usr/lib/x86_64-linux-gnu/libneon.so.27
/usr/lib/x86_64-linux-gnu/libneon.so.27.5.5
```

adrien@VoIP:~$ sudo apt-get install postgresql

Installation d'asterisk.

adrien@VoIP:~$ sudo wget
https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz
Décompresser le fichier tar d'asterisk.
adrien@VoIP:~$ sudo tar -xvf asterisk-20-current.tar.gz
Ensuite aller dans le répertoire asterisk.
adrien@VoIP:~$ cd asterisk-20.7.0/
Permet d'installer les dépendances nécessaires pour l'installation d'asterisk.
adrien@VoIP:~/asterisk-20.7.0$ sudo ./contrib/scripts/install_prereq install
Permet de configurer asterisk avec ssl et srtp.
adrien@VoIP:~/asterisk-20.7.0$ sudo ./configure --with-crypto --with-ssl --with-srtp

Permet de compiler les sources d'asterisk.
adrien@VoIP:~/asterisk-20.7.0$ sudo make
Permet d'installer asterisk.
adrien@VoIP:~/asterisk-20.7.0$ sudo make install



Compile les échantillons.
adrien@VoIP:~/asterisk-20.7.0$ sudo make samples

Générer le certificat pour le chiffrement tls.

Créer un dossier keys dans /etc/asterisk/
adrien@VoIP:~$ sudo mkdir /etc/asterisk/keys/
Allez dans contrib/script.
adrien@VoIP:~$ cd /asterisk-20.7.0/contrib/scripts/
Génère les certificats.
adrien@VoIP:~/asterisk-20.7.0/contrib/scripts$ sudo ./ast_tls_cert -C 172.31.254.250 -O "My Super Company" -d /etc/asterisk/keys -b 2048.

Configurer PJSIP

Backup du fichier pjsip.conf

Permet de faire un backup du fichier de configuration original.
adrien@VoIP:~$ sudo mv /etc/asterisk/pjsip.conf /etc/asterisk/pjsip.conf.backup
Permet d'éditer le fichier de configuration pjsip.
adrien@VoIP:~$ sudo nano /etc/asterisk/pjsip.conf

Configuration du mode de transport

```
[transport-tls] ; Cette section configure les paramètres du transport TLS.
type=transport ; Indique qu'il s'agit d'un type de transport.
protocol=tls ; Spécifie que le protocole utilisé est TLS.
bind=0.0.0.0:5061 ; Indique que le serveur écoutera sur toutes les interfaces (0.0.0.0) sur le port 5061.
cert_file=/etc/asterisk/keys/asterisk.crt ; Chemin vers le fichier de certificat TLS.
priv_key_file=/etc/asterisk/keys/asterisk.key ; Chemin vers le fichier de clé privée TLS.
allow_reload=yes ; Permet le rechargement des paramètres.
tos=cs3 ; Type de service, qui indique une priorité de service.
cos=3 ; Classe de service, utilisée pour la qualité de service.
local_net=172.31.0.0/255.255.0.0 ; Réseau local autorisé à se connecter.
method=tlsv1_2 ; Méthode TLS utilisée, ici TLS version 1.2.
```

Configuration des endpoints

```
[7001] ; Ces sections configurent les endpoints (extrémités) pour les utilisateurs 7001 respectivement.
context=default ; Redirige vers le label défaut se trouvant dans extensions.conf
type=endpoint ; Spécifie qu'il s'agit d'une extrémité (endpoint)
transport=transport-tls ; Utilise le transport TLS défini précédemment.
disallow=all ; Désactive tous les codecs par défaut.
allow=ulaw ; Autorise le codec ulaw pour la communication audio.
auth=7001-auth ; Utilise l'authentification définie pour cet utilisateur.
aors=7001 ; Associe cet endpoint à l'AOR correspondant.
media_encryption=sdes ; Active le chiffrement du média en utilisant Secure RTP avec SDES.

[7001-auth] ; Ces sections configurent l'authentification pour les utilisateurs 7001.
```

```
type=auth ; Spécifie qu'il s'agit d'une configuration
d'authentification.
auth_type=userpass ; Utilise l'authentification par nom
d'utilisateur et mot de passe.
username=7001 ; Nom d'utilisateur pour l'authentification.
password=7001 ; Mot de passe correspondant pour l'authentification.


[7001] ; Ces sections configurent l'AOR (Address of Record) pour les
utilisateurs 7001 et 7002 respectivement.
type=aor ; Spécifie qu'il s'agit d'un enregistrement d'adresse.
max_contacts=1 ; Limite le nombre de contacts autorisés à un seul.
remove_existing=yes ; Supprime les contacts existants lorsqu'un
nouvel enregistrement est effectué.

[7002] ; Ces sections configurent les endpoints (extrémités) pour
les utilisateurs 7002 respectivement.
context=default ; redirige vers le label défaut se trouvant dans
extensions.conf
type=endpoint ; Spécifie qu'il s'agit d'une extrémité (endpoint)
transport=transport-tls ; Utilise le transport TLS défini
précédemment.
disallow=all ; Désactive tous les codecs par défaut.
allow=ulaw ; Autorise le codec ulaw pour la communication audio.
auth=7002-auth ; Utilise l'authentification définie pour cet
utilisateur.
aors=7002 ; Associe cet endpoint à l'AOR correspondant.
media_encryption=sdes ; Active le chiffrement du média en utilisant
Secure RTP avec SDES.

[7002-auth] ; Ces sections configurent l'authentification pour les
utilisateurs 7002.
type=auth ; Spécifie qu'il s'agit d'une configuration
d'authentification.
auth_type=userpass ; Utilise l'authentification par nom
d'utilisateur et mot de passe.
username=7002 ; Nom d'utilisateur pour l'authentification.
password=7002 ; Mot de passe correspondant pour l'authentification.

[7002] ; Ces sections configurent l'AOR (Address of Record) pour les
utilisateurs 7001 et 7002 respectivement.
type=aor ; Spécifie qu'il s'agit d'un enregistrement d'adresse.
```

`max_contacts=2` ; **Limite le nombre de contacts autorisés à un seul.**
`remove_existing=yes` ; **Supprime les contacts existants lorsqu'un nouvel enregistrement est effectué.**

Configurer le fichier extension.conf

Backup extension.conf

Permet de faire un backup du fichier extensions.conf
adrien@VoIP:~$ sudo mv /etc/asterisk/extensions.conf /etc/asterisk/extensions.conf.backup
Permet d'éditer le fichier extensions.conf
adrien@VoIP:~$ sudo nano /etc/asterisk/extensions.conf

Configuration DND

[default]
`exten => _.!,1`**; Cette ligne définit une extension générique. _.! signifie qu'elle correspond à n'importe quel numéro composé (_ signifie "match anything") suivi d'un ou plusieurs chiffres (!).**
`GotoIfTime(9:00-18:00,mon-fri,,?local-context,${EXTEN},1)`**; C'est une application qui vérifie si l'heure actuelle est comprise entre 9h et 18h du lundi au vendredi. Si c'est le cas, l'appel est redirigé (Goto) vers le contexte `local-context` avec le même numéro composé que l'appelant (${EXTEN}) et la priorité 1. Le ? signifie "si vrai", donc si la condition est vraie, l'appel est redirigé.**
same => n,2,Hungup() **; Cette commande demande à Asterisk de raccrocher l'appel immédiatement. Cela signifie que si l'appel n'a pas été redirigé vers "local-context" pendant les heures de bureau, alors il sera automatiquement raccroché, sans autre traitement.**

Configuration des appelles

[local-context]
`exten => 7001,1,Answer()` ; **Cette ligne indique que lorsque l'extension 7001 est composée, la première étape est de répondre à l'appel.**
`exten => 7001,2,Dial(PJSIP/7001,60)` ; **Ensuite, l'appel est composé vers l'utilisateur ou le périphérique SIP avec l'identifiant 7001. La durée maximale du temps d'attente est de 60 secondes.**

```
exten => 7001,3,Playback(vm-nobodyavail) ; Si l'appel vers 7001 ne peut pas être établi (par exemple, si l'utilisateur 7001 ne répond pas), le système lit un message préenregistré (dans ce cas, "vm-nobodyavail").
exten => 7001,4,VoiceMail(7001@main) ; Ensuite, l'appel est envoyé à la boîte vocale de l'utilisateur 7001 (par exemple, s'il ne répond pas ou si la ligne est occupée).
exten => 7001,5,Hangup() ; Enfin, l'appel est terminé (raccroché).

exten => 8001,1,VoicemailMain(7002@main) ; Lorsque l'extension 8001 est composée, l'appel est envoyé au menu principal de la boîte vocale de l'utilisateur 7002.
exten => 8002,2,Hangup() ; Lorsque l'extension 8002 est composée, l'appel est simplement terminé (raccroché) sans autre action.

exten => 7002,1,Answer() ; Cette ligne indique que lorsque l'extension 7001 est composée, la première étape est de répondre à l'appel.
exten => 7002,2,Dial(PJSIP/7002,60) ; Ensuite, l'appel est composé vers l'utilisateur ou le périphérique SIP avec l'identifiant 7001. La durée maximale du temps d'attente est de 60 secondes.
exten => 7002,3,Playback(vm-nobodyavail) ; Si l'appel vers 7001 ne peut pas être établi (par exemple, si l'utilisateur 7001 ne répond pas), le système lit un message préenregistré (dans ce cas, "vm-nobodyavail").
exten => 7002,4,VoiceMail(7002@main) ; Ensuite, l'appel est envoyé à la boîte vocale de l'utilisateur 7001 (par exemple, s'il ne répond pas ou si la ligne est occupée).
exten => 7002,5,Hangup() ; Enfin, l'appel est terminé (raccroché).

exten => 8001,1,VoicemailMain(7001@main) ; Lorsque l'extension 8001 est composée, l'appel est envoyé au menu principal de la boîte vocale de l'utilisateur 7001.
exten => 8001,2,Hangup() ; Lorsque l'extension 8002 est composée, l'appel est simplement terminé (raccroché) sans autre action.


exten => 8001,1,VoicemailMain(7002@main) ; Lorsque l'extension 8001 est composée, l'appel est envoyé au menu principal de la boîte vocale de l'utilisateur 7002.
exten => 8001,2,Hangup() ; Lorsque l'extension 8002 est composée, l'appel est simplement terminé (raccroché) sans autre action.
```

Configuration du menu

```
Menu [comptabilite-menu]  ;
exten => 1,1,Background(service-comptabilite) ; Si l'appelant choisit l'option 1, Joue l'annonce audio qui décrit les services comptables.
same => n,Goto(service-comptabilite,s,1) ; Ensuite, continué à l'étape suivante (n) et va au sous-menu "service-comptabilite" avec la priorité 1.
exten => 2,1,Background(service-ventes) ; Si l'appelant choisit l'option 2, Joue l'annonce audio qui décrit les services de vente.
same => n,Goto(service-ventes,s,1) ; Ensuite, continué à l'étape suivante (n) et va au sous-menu "service-ventes" avec la priorité 1.

Menu [rh-menu] ;
exten => 1,1,Background(service-rh) ; Si l'appelant choisit l'option 1, Joue l'annonce audio qui décrit les services des ressources humaines.
same => n,Goto(service-rh,s,1) ; Ensuite, continué à l'étape suivante (n) et va au sous-menu "service-rh" avec la priorité 1.
exten => 2,1,Background(service-formation) ; Si l'appelant choisit l'option 2,
Joue l'annonce audio qui décrit les services de formation.
same => n,Goto(service-formation,s,1) ; Ensuite, continué à l'étape suivante (n) et va au sous-menu "service-formation" avec la priorité 1.1
```

Configuration automatique de prospection.

```
NoOp(Démarrage de la prospection automatique) ; C'est une instruction qui n'effectue aucune opération (No Operation). Elle sert souvent de marqueur ou de point de référence dans un script, dans ce cas-ci, indiquant le début de la prospection automatique.
Set(NUMERO=${RAND(1, NOMBRE_DE_CONTACTS)}) ; Cette ligne attribue une valeur aléatoire entre 1 et le nombre total de contacts dans votre liste à la variable NUMÉRO. La fonction RAND() génère un nombre aléatoire.
Set(TARGET=${FIELDNUM(mon.csv,2)}) ; Cette ligne extrait le numéro
```

de téléphone correspondant à l'entrée sélectionnée aléatoirement dans votre liste de contacts. FIELDNUM() est utilisée pour extraire une valeur spécifique d'un fichier CSV (dans ce cas, le numéro de téléphone), en utilisant le numéro de ligne (NUMÉRO) et le numéro de colonne (2) comme références.
`Dial(PJSIP/${TARGET}@172.31.254.250)` ; Cette ligne compose le numéro de téléphone extrait dans l'étape précédente via votre tronc SIP PJSIP (un protocole de voix sur IP). Elle tente de connecter l'appel au numéro de téléphone.
`Hangup()` ; Cette ligne termine l'appel après qu'il a été composé, qu'il soit connecté ou non.

Configuration Voicemail

Backup de voicemail.conf

adrien@VoIP:~$ sudo mv /etc/asterisk/voicemail.conf /etc/asterisk/voicemail.conf.backup
adrien@VoIP:~$ sudo nano /etc/asterisk/voicemail.conf

Configuration de mot de passe de la boîte vocale

`7001 => 7001` ; Cette ligne indique que l'utilisateur 7001 dispose d'une boîte vocale avec le même numéro, c'est-à-dire que son numéro d'extension est également son numéro de boîte vocale. Cela signifie que lorsque vous appelez l'extension 7001 et que l'utilisateur ne répond pas ou est occupé, vous pouvez laisser un message dans sa boîte vocale en utilisant le même numéro d'extension.
`7002 => 7002` ; De même, cette ligne indique que l'utilisateur 7002 dispose également d'une boîte vocale avec le même numéro d'extension.

Configurer modules.conf

Backup modules.conf

Copi module.conf dans /etc/asterisk/ et le renomme en modules.conf.backup.

adrien@VoIP:~$ sudo cp /etc/asterisk/modules.conf /etc/asterisk/modules.conf.backup

Configuration module.conf

adrien@VoIP:~$ sudo nano /etc/asterisk/module.conf

```
load = res_pjsip.so ; charge le module pjsip qui permet d'activer le canal pjsip.conf.
load = res_srtp.so ; charge le module pour activer le srtp.
noload = cel_tds.so ; Empêche le chargement du module cel_tds.so
noload = cel_radius.so ; Empêche le chargement du module cel_radius.so
noload = cdr_pgsql.so ;  Empêche le chargement du module cdr_pgsql.so
noload = cel_sqlite3_custom.so ; Empêche le chargement du module cel_sqlite3_custom.so
noload = cdr_sqlite3_custom.so ; Empêche le chargement du module cdr_sqlite3_custom.so
noload = cdr_tds.so ; Empêche le chargement du module cdr_tds.so
noload = cdr_radius.so ; Empêche le chargement du module cdr_radius.so
noload = pbx_dundi ; Empêche le chargement du module pbx_dundi
```

Backup ari.conf

Copi ari.conf et le renomme en ari.conf.backup dans /etc/asterisk.
adrien@VoIP:~$ sudo cp /etc/asterisk/ari.conf /etc/asterisk/ari.conf.backup
Décommenter c'est ligne pour enlever l'erreur liée à ari.conf

```
[username]
type = user ; Spécifie la configuration de l'utilisateur.
read_only = no ;  Lorsque la valeur est fixée à oui, l'utilisateur n'est autorisé qu'à
```

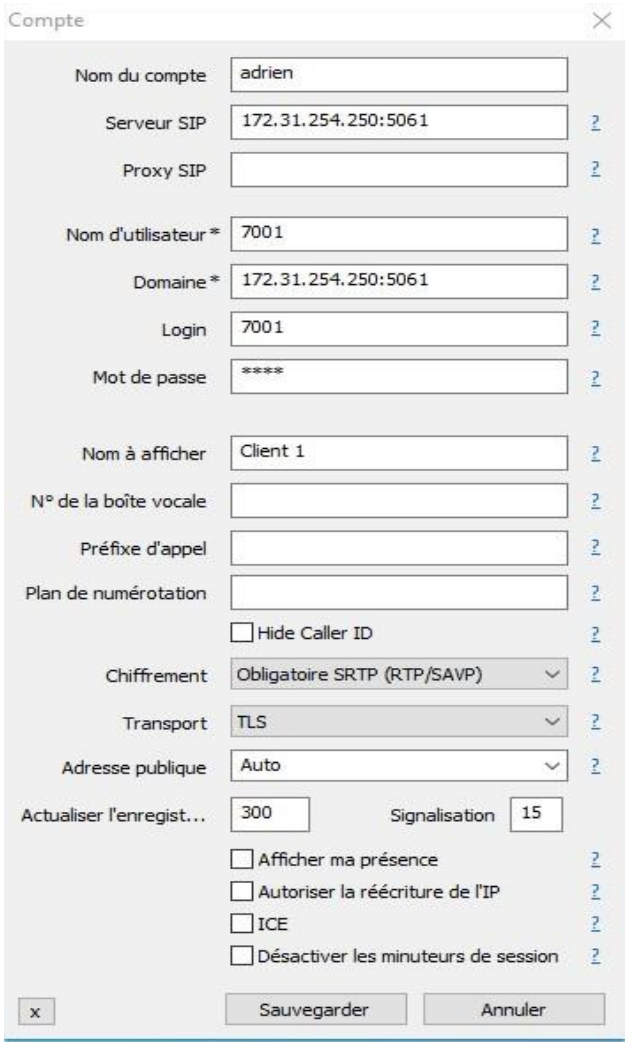
Vérifier la configuration des endpoints

Initialise le service asterisk
adrien@VoIP:~$ sudo asterisk -v
Démarre asterisk
adrien@VoIP:~$ sudo asterisk -r

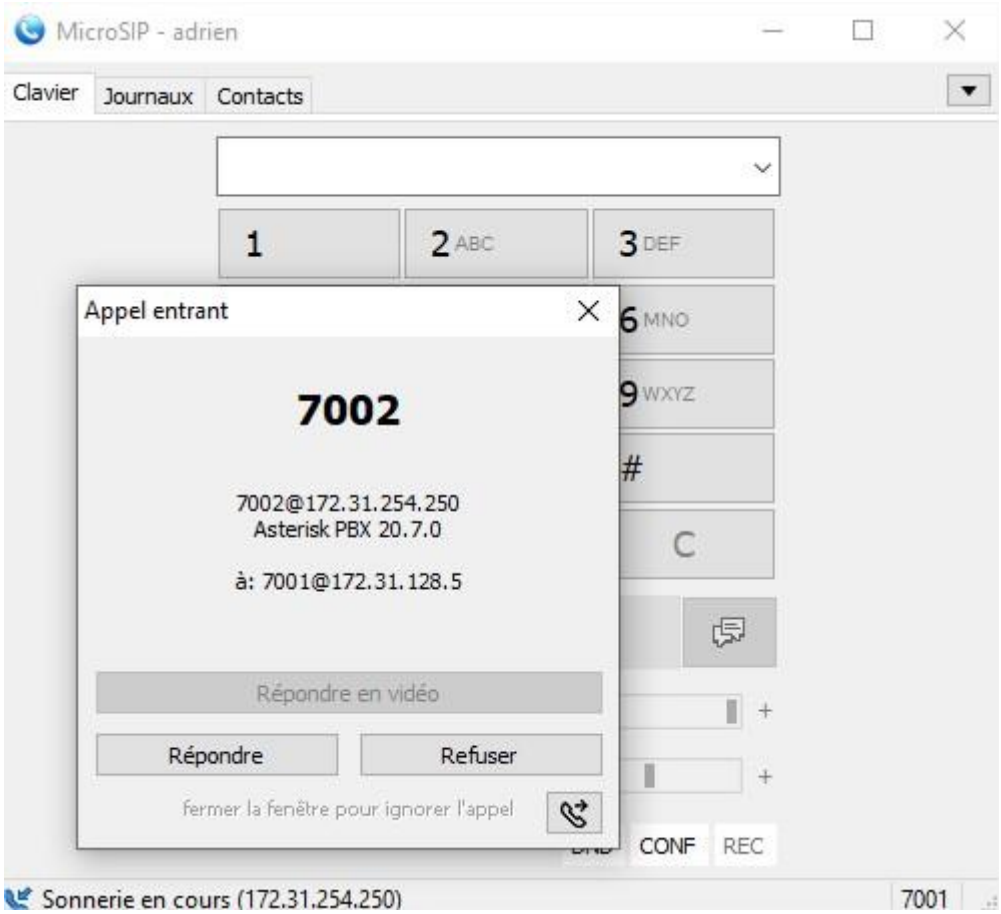
recharge la configuration d’asterisk
VoIP*CLI> reload
recharge la configuration
VoIP*CLI> pjsip show endpoints

```
VoIP*CLI> pjsip show endpoints

 Endpoint:  <Endpoint/CID.....................................>  <State.....>  <Channels.>
    I/OAuth:  <AuthId/UserName.....................................................>
        Aor:  <Aor.............................................>  <MaxContact>
     Contact:  <Aor/ContactUri..........................> <Hash....> <Status> <RTT(ms)..>
  Transport:  <TransportId........>  <Type>  <cos>  <tos>  <BindAddress..................>
   Identify:  <Identify/Endpoint.....................................................>
        Match:  <criteria.........................>
    Channel:  <ChannelId.....................................>  <State.....>  <Time.....>
        Exten: <DialedExten...........>  CLCID: <ConnectedLineCID.......>
==========================================================================================

 Endpoint:  7001                                                 Not in use    0 of inf
    InAuth:  7001-auth/7001
        Aor:  7001                                               1
     Contact:  7001/sip:7001@172.31.128.5:49779;transport e087b34a32 NonQual         nan
  Transport:  transport-tls             tls      3     96  0.0.0.0:5061

 Endpoint:  7002                                                 Not in use    0 of inf
    InAuth:  7002-auth/7002
        Aor:  7002                                               2
     Contact:  7002/sip:7002@192.168.8.112:57386;transpor de81fa82c7 NonQual         nan
  Transport:  transport-tls             tls      3     96  0.0.0.0:5061
```

Configuration Micro SIP.

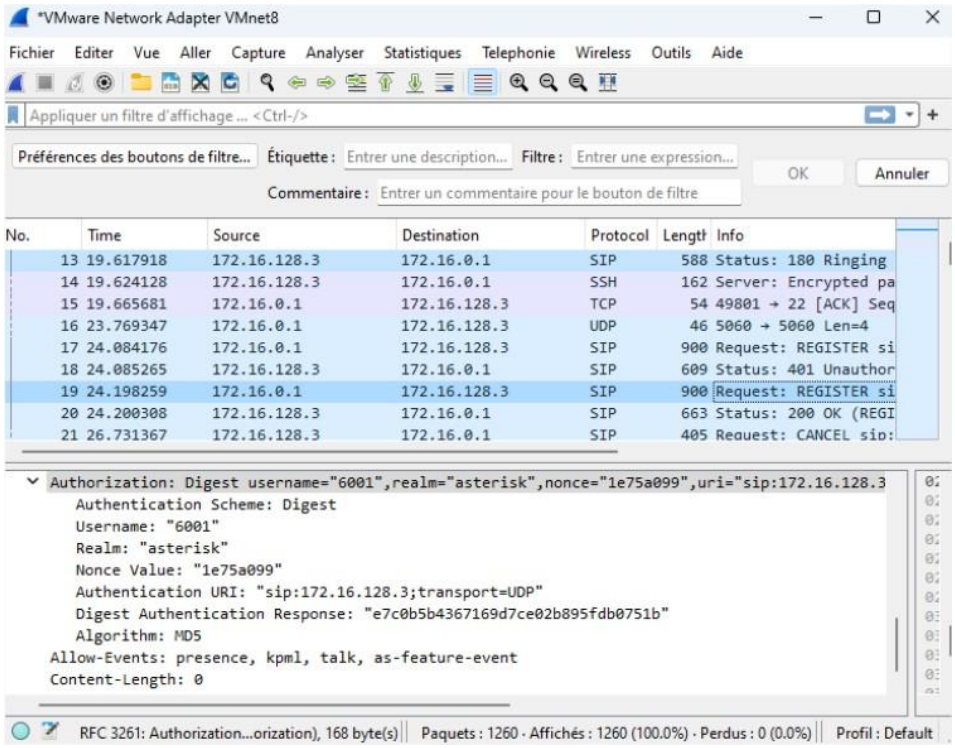


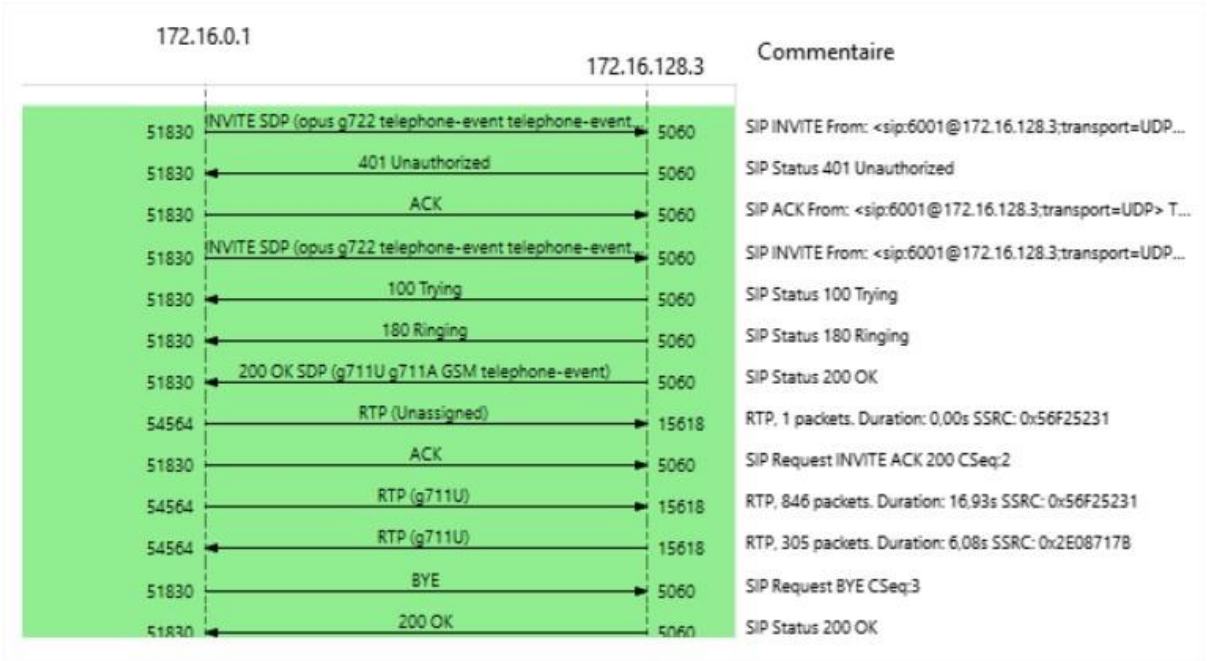
Ci-dessous vous pouvez voir que la fonctionnalité d'appel fonctionne.



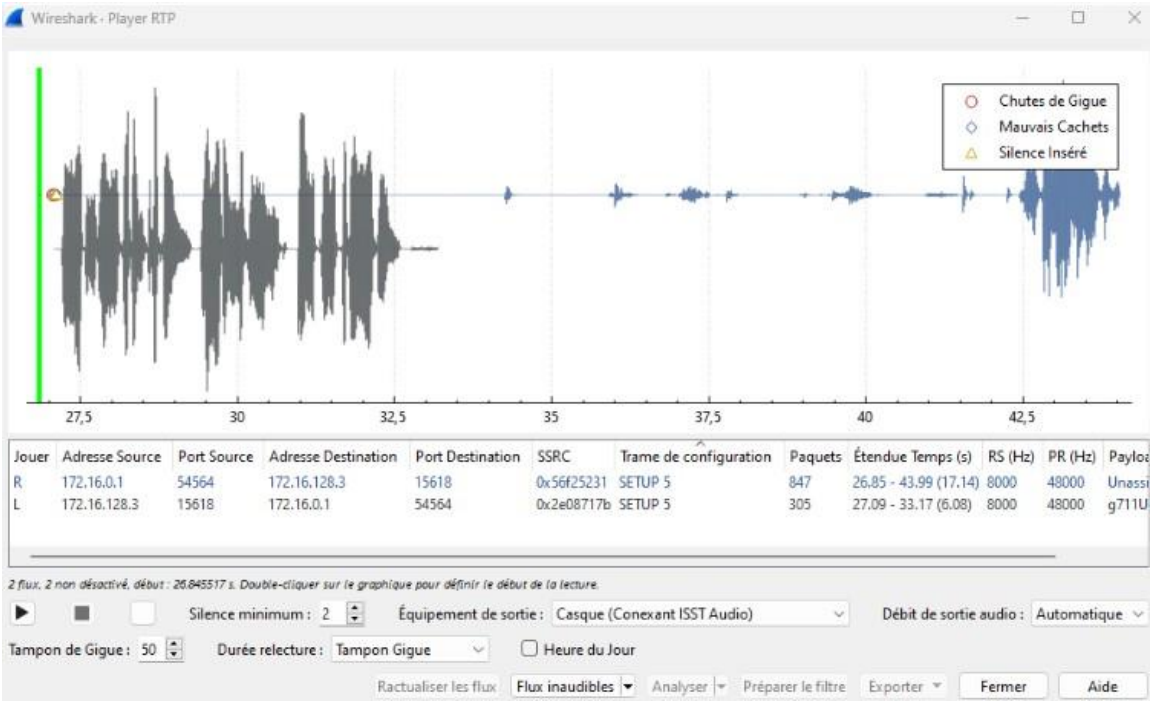
Chiffrement transport layer security

Sans chiffrement tls.

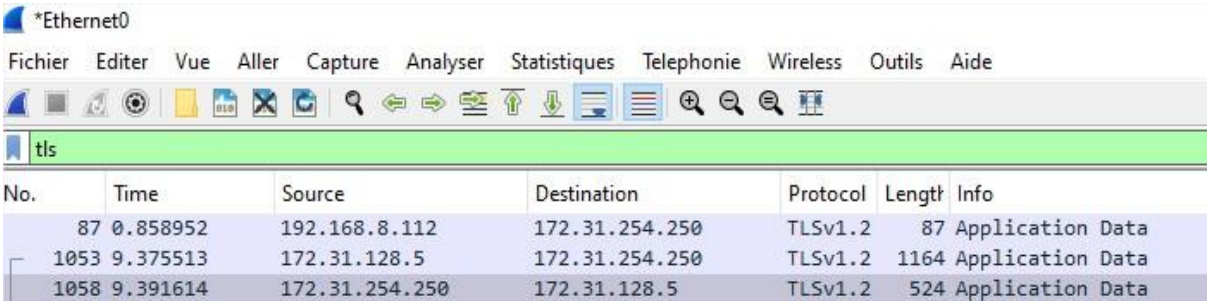




Sans chiffrement srtp.



Avec chiffrement tls.



Installation fail2ban

Installation de fail2ban les paquets.
adrien@VoIP:~/asterisk-20.7.0$ sudo apt-get install fail2ban

Backup fail2ban

Permet de faire un backup de fail2ban.conf
adrien@VoIP:~/asterisk-20.7.0$ sudo cp /etc/fail2ban/fail2ban.conf /etc/fail2ban/fail2ban.conf.backup
Permet de faire un backup de jail.conf
adrien@VoIP:~/asterisk-20.7.0$ sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.conf.backup

Configuration fail2ban

Permet d'éditer le fichier jail.local.
adrien@VoIP:~$ sudo nano /etc/fail2ban/jail.local

```
enabled = true ; activé (true) signifie que cette configuration est activée.
port = 5061 ; le port utilisé par le service Asterisk.
action : actions à prendre en cas de détection d'une tentative d'intrusion.
logpath = /var/log/asterisk/full ; chemin du fichier journal où Fail2Ban surveille les activités.
maxretry = 5 ; le nombre maximal de tentatives de connexion autorisées avant que l'adresse IP soit bannie.
bantime = 3m ; durée pendant laquelle une adresse IP est bannie après avoir atteint le nombre maximal de tentatives de connexion (3 minutes dans ce cas).
```

Redémarrer le daemon fail2ban

Initie le service fail2ban.
adrien@VoIP:~$ sudo systemctl enable fail2ban
Démarre le service fail2ban.
adrien@VoIP:~$ sudo systemctl start fail2ban
Créer le fichier de log pour fail2ban
adrien@VoIP:~/asterisk-20.7.0$ sudo touch /var/log/auth.log

```
adrien@VoIP: $ sudo systemctl status fail2ban
[sudo] Mot de passe de adrien :
● fail2ban.service - Fail2Ban Service
     Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
     Active: active (running) since Sun 2024-05-05 18:33:43 CEST; 5h 33min ago
       Docs: man:fail2ban(1)
   Main PID: 658 (fail2ban-server)
      Tasks: 7 (limit: 2265)
     Memory: 27.9M
        CPU: 9.594s
     CGroup: /system.slice/fail2ban.service
             └─658 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

mai 05 18:33:43 VoIP systemd[1]: Started fail2ban.service - Fail2Ban Service.
mai 05 18:33:44 VoIP fail2ban-server[658]: 2024-05-05 18:33:44,382 fail2ban.configreader
inition'. Using default one: 'auto'
mai 05 18:33:44 VoIP fail2ban-server[658]: Server ready
```

AsteriskOnFly

adrien@VoIP:~$ sudo nano AsteriskOnFly.sh

---

apt-get update -y ; Mets a jours le gestionnaire de paquet.
apt-get dist-upgrade -y ; Mets a jours la distribution debian.
sudo apt-get install libssl-dev libsrtp2-dev ; Permet d'installer les librairies utilisées pour le chiffrement du protocol sip et rtp.

wget https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz ; Télécharge asterisk
tar -zxvf asterisk-20-current.tar.gz ; Décompresse le fichier tar.gz

./contrib/scripts/install_prereq install ; Installe les dépendances nécessaire pour l'installation.
make ; Permet de compiler asterisk depuis la source.
make install ; Install asterisk.
make samples ; Compile les échantillons

mkdir /etc/asterisk/keys/ ; Créer le répertoire keys dans /etc/asterisk/.
cd /asterisk-20.7.0/contrib/scripts/ ; Change de repertoire pour aller dans /asterisk-20.7.0/contrib/scripts/
./ast_tls_cert -C 172.31.254.250 -O "My Super Company" -d /etc/asterisk/keys -b 2048. ; Génére le certificat tls.
cd ~ ; Nous ramène dans le répertoire courant

asterisk -v ; initialise asterisk.

```
echo "Asterisk a été installé avec succès."; Affiche le message (Asterisk a été installé avec succès).
```

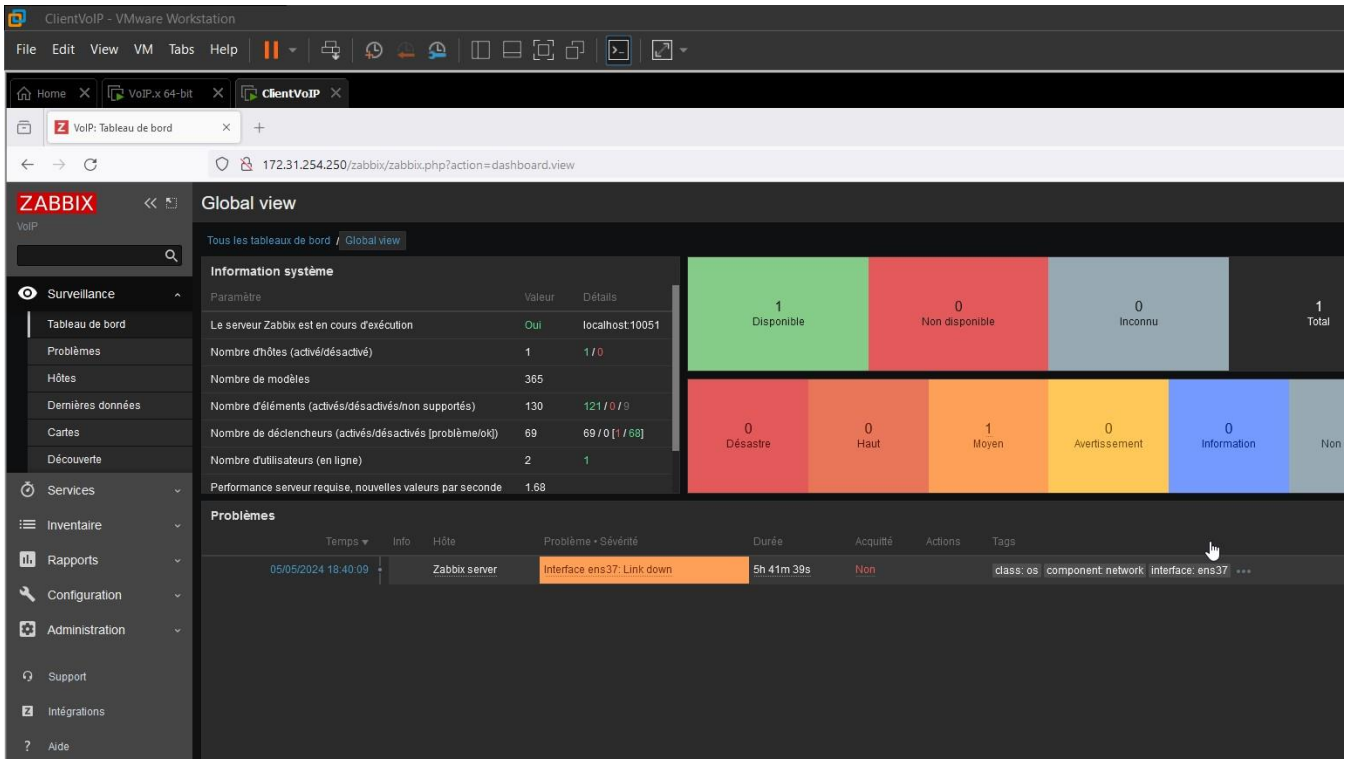
Pour le lancer, tapez.
adrien@VoIP:~$ sudo ./AsteriskOnFly.sh

Configuration Zabbix

Permet de télécharger zabbix.
adrien@VoIP:~$ wget
https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release_6.0-5+debian12_all.deb
Permet d'ajouter les paquet zabbix au gestionnaire de paquet dpkg.
adrien@VoIP:~$ dpkg -i zabbix-release_6.0-5+debian12_all.deb
Permet de mettre à jour les paquets.
adrien@VoIP:~$ apt update
Permet d'installer les paquets qui permettent l'installation de zabbix.
adrien@VoIP:~$ apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent
Permet d'installer le serveur de base de données mariadb.
adrien@VoIP:~$ apt-get install mariadb-server
Permet de se connecter à la cli de mariadb en tant que root.
MariaDB [zabbix]> mysql -uroot -p
Permet de créer la base de données nommée zabbix et de configurer l'encodage utf8mb4.
MariaDB [zabbix]> create database zabbix character set utf8mb4 collate utf8mb4_bin;
Permet de créer un utilisateur nommé zabbix.
MariaDB [zabbix]> create user zabbix@localhost identified by 'password';
Permet d'attribuer les privilèges pour sur la base de données zabbix pour l'utilisateur zabbix.
MariaDB [zabbix]> grant all privileges on zabbix.* to zabbix@localhost;
Permet d'activer le fait que MySQL ne vérifie pas si l'utilisateur qui crée une fonction a les privilèges de réplication appropriés.
MariaDB [zabbix]> set global log_bin_trust_function_creators = 1;
Permet de quitter la cli mysql.
MariaDB [zabbix]> quit;
Cela garantit que les données stockées dans la base de données seront traitées correctement avec des caractères multi bits, utiles pour prendre en charge des langues avec des caractères spéciaux.
sudo zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix
Permet de lancer mariadb en tant que root.
MariaDB [zabbix]> mysql -uroot -p
Permet de déactiver le fait que MySQL ne vérifie pas si l'utilisateur qui crée une fonction a les privilèges de réplication appropriés.
MariaDB [zabbix]> set global log_bin_trust_function_creators = 0;
Permet d'éditer le fichier /etc/zabbix/zabbix_server.conf
adrien@VoIP:~$ sudo nano /etc/zabbix/zabbix_server.conf

DBHost=localhost ; Permet d'indiquer que le serveur de base de données se trouve sur le serveur zabbix.
DBName=zabbix ; Permet d'indiquer le nom de la base de données.
DBUser=zabbix ; Permet d'indiquer un user zabbix pour se connecter à l'interface web.
DBPassword=mdp ; Permet de définir un mot de passe pour se connecter à l'interface web.
DBPort=3306 ; Permet de définir le port utilisé par la base de données.

Permet d'initier le daemon (apache2) le serveur zabbix et l'agent.
systemctl enable zabbix-server zabbix-agent apache2
Permet de redémarrer le serveur web (apache2) le serveur zabbix et l'agent zabbix.
systemctl restart zabbix-server zabbix-agent apache2
Ensuite accéder à l'interface web zabbix avec vos identifiant.



Intégration Server LDAP

On va commencer par installer un serveur local OpenLDAP sur une installation fraîche pour ensuite tenter une intégration avec active directory.
sudo apt install slapd ldap-utils
sudo dpkg-reconfigure slapd
On récupère le schéma asterisk pour l'importer dans LDAP
https://github.com/asterisk/asterisk/blob/master/contrib/scripts/asterisk.ldap-schema
sudo cp asterisk.ldap-schema /etc/ldap/schema/asterisk.schema
On restart le serveur LDAP et on importe le fichier asterisk.ldif.
https://github.com/asterisk/asterisk/blob/master/contrib/scripts/asterisk.ldif

sudo /etc/init.d/slapd restart
sudo ldapadd -Y EXTERNAL -H ldapi:/// -f asterisk.ldif
On utilise le fichier suivant pour ajouter un utilisateur de test, il faudra modifier dc=shifteight,dc=org en fonction du nom de domaine que l'on a utilisé lors de la configuration
astuser.ldif
dn: uid=rbryant,dc=shifteight,dc=org
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: AsteriskSIPUser
uid: rbryant
sn: Bryant
givenName: Russell
cn: RussellBryant
displayName: Russell Bryant
uidNumber: 1001
gidNumber: 10001
userPassword: {md5}a7be810a28ca1fc0668effb4ea982e58 # mdp : my_secure_password
homeDirectory: /home/russell
AstAccountCallerID: 1001
AstAccountContext: from-internal
AstAccountType: friend
AstAccountHost: dynamic
AstAccountRealmedPassword: {md5}a7be810a28ca1fc0668effb4ea982e58
On ajoute notre utilisateur de test avec la commande
sudo ldapadd -x -D cn=admin,dc=shifteight,dc=org -f astuser.ldif -W
Le fichier **/etc/asterisk/res_ldap.conf** est configuré pour faire le mapping entre les attributs asterisk et LDAP, il ne reste plus qu'à lui fournir les informations de connexion au serveur
[_general]
host=127.0.0.1 ; LDAP host
port=389
protocol=3 ; Version of the LDAP protocol to use; default is 3.
basedn=dc=shifteight,dc=org ; Base DN
user=cn=admin,dc=shifteight,dc=org ; Bind DN
pass=*******
Pour finir, on édite le fichier /etc/asterisk/extconfig.conf**
sipusers => ldap,"dc=shifteight,dc=org",sip
sippeers => ldap,"dc=shifteight,dc=org",sip
Pour que notre client s'enregistre auprès d'Asterisk, il nous faut rajouter la ligne suivante à **/etc/asterisk/sip.conf**
rtcachefriends=yes
et la ligne suivante à /etc/asterisk/extensions.conf qui permet de passer l'appel avec le "numéro" Russel Bryant.
.exten => _.!,1,Dial(SIP/${EXTEN},10)
Par défaut, Asterisk utilise le common name comme numéro, on peut changer le mapping dans le fichier **/etc/asterisk/res_ldap.conf** en remplaçant la ligne
name = cn
par
name = AstAccountCallerID
Pour la connexion au serveur et l'extension de l'appel on utilise le Common Name de l'utilisateur : RusselBryant et le mot de passe "my_secure_password".
Intégration Active Directory

Il nous faut rajouter des attributs aux utilisateurs qui seront utilisés par Asterisk pour faire un mapping. Pour ce faire, nous avons besoin de modifier le schéma LDAP utilisé par Active Directory. L'application qui permet de faire ces modifications n'est pas directement accessible, la procédure pour débloquer la modification du schéma est expliqué à l'adresse suivante
https://rdr-it.io/troubleshooting/active-directory-acceder-a-la-console-schema-active-directory/
Les attributs nécessaires à l'intégration à Active Directory sont les suivants:

- **AstAccountCallerID: 1001**
- **AstAccountContext: from-internal**
- **AstAccountType: friend**
- **AstAccountHost: dynamic**
- **AstAccountRealmedPassword: {md5}a7be810a28ca1fc0668effb4ea982e58**

La procédure d'ajout des attributs est disponible ici:
https://www.offsitenoc.com/asterisk-active-directory-schema-extension/
Attention: L'attribut **AstAccountRealmedPassword** est obligatoire. Si cet attribut est absent Asterisk n'effectuera pas de vérification de mot de passe et on pourra s'authentifier sans mot de passe.
Références :
https://www.youtube.com/watch?v=zSKXzL_dafs
https://www.youtube.com/watch?v=QfktNKem1xo -> DnD
https://www.networklab.fr/configuration-basique-dasterisk

Documentation attendu

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

Oui, on peut souvent dire la même chose des coûts opérationnels et de maintenance dans de nombreux contextes. Les deux types de coûts sont généralement liés aux activités quotidiennes nécessaires pour faire fonctionner et entretenir un système, un équipement ou une infrastructure.

Les coûts opérationnels sont généralement associés aux dépenses courantes engagées pour maintenir les opérations en cours, telles que les salaires du personnel, les frais de consommables, les services publics, etc. De même, les coûts de maintenance sont liés aux dépenses engagées pour préserver et réparer les équipements et les infrastructures afin de garantir leur bon fonctionnement et leur longévité.

Dans de nombreux cas, une gestion efficace des coûts opérationnels peut également entraîner une réduction des coûts de maintenance à long terme, car des opérations bien gérées peuvent contribuer à réduire l'usure et les pannes des équipements. De même, des pratiques de maintenance préventive appropriées peuvent contribuer à réduire les temps d'arrêt et les coûts de réparation imprévus, ce qui peut à son tour avoir un impact positif sur les coûts opérationnels.

Cependant, il convient de noter que dans certains cas, les coûts opérationnels et de

maintenance peuvent différer en fonction de la nature spécifique de l'activité ou de l'industrie concernée. Par exemple, dans certaines industries, les coûts de maintenance peuvent être beaucoup plus élevés que les coûts opérationnels en raison de la nature complexe et coûteuse des équipements nécessitant une maintenance régulière et spécialisée.

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 ?

Nombre de lignes et de postes téléphoniques : Un call center peut nécessiter un plus grand nombre de lignes téléphoniques et de postes agents pour gérer un volume élevé appels entrants et sortants. Par conséquent, la configuration VoIP du call center devrait prendre en compte cette demande accrue en termes de capacité et de gestion des appels simultanés.
Fonctionnalités de routage avancées : Les call centers exigent souvent des fonctionnalités de routage avancées pour distribuer efficacement les appels entrants aux agents disponibles en fonction de critères tels que la compétence, la disponibilité, la priorité de l'appel, etc. La configuration VoIP un call center devrait donc inclure des capacités de routage intelligent et de distribution automatique des appels.
Gestion des files d'attente : Les call centers peuvent avoir besoin de fonctionnalités de gestion des files d'attente pour placer les appelants en attente lors des périodes de forte affluence et pour fournir des annonces d'attente, des estimations de temps d'attente, etc. La configuration VoIP devrait permettre la mise en place et la gestion efficace de ces files d'attente.
Intégration avec les logiciels de centre appels : Les systèmes VoIP des call centers sont souvent intégrés à des logiciels de centre d'appels (CRM, gestion des tickets, etc.) pour offrir une expérience client plus complète et pour permettre aux agents accéder aux informations pertinentes pendant les appels. La configuration VoIP devrait donc prendre en charge cette intégration avec les autres systèmes utilisés dans le centre d'appels.
Surveillance et reporting avancés : Les gestionnaires de call centers ont souvent besoin outils de surveillance et de reporting avancés pour suivre les performances des agents, la durée des appels, les temps d'attente, etc. La configuration VoIP devrait permettre la collecte et analyse de ces données pour améliorer efficacité opérationnelle du centre d'appels.
En résumé, bien que les principes fondamentaux de la technologie VoIP restent les mêmes, la configuration spécifique un système VoIP pour un call center sera adaptée pour répondre aux exigences uniques de gestion des appels dans cet environnement.

Identifiez des sites marchands ou de service dont customer

Le service implique des services VoIP, donnez quelques exemples et
Décrivez une architecture possible de leur système.

Centre d'appels externalisé en ligne : Des entreprises telles que LiveOps ou Arise Virtual Solutions fournit des services de centre d'appels externalisés en ligne. Leur service clientèle implique souvent utilisation de la VoIP pour gérer les appels entrants et sortants des clients. Voici une architecture possible pour un tel système :

Interface utilisateur web : Les agents accèdent à une interface web où ils peuvent voir les informations sur les appels entrants, répondre aux appels, saisir des notes sur les interactions avec les clients, etc.

Plateforme VoIP : Une plateforme VoIP est intégrée à interface web pour permettre aux agents de passé et de recevoir des appels via Internet. Cette plateforme gère également les fonctionnalités de routage des appels, les files d'attente, enregistrement des appels, etc.
Intégration CRM : La plateforme VoIP est souvent intégrée à un système de gestion de la relation client (CRM) pour permettre aux agents accéder aux informations client pertinentes pendant les appels.

Plateforme de réservation en ligne : Des sites comme Booking.com ou Airbnb proposent des services de réservation en ligne pour les voyages et hébergement. Leur service clientèle peut également utiliser la VoIP pour fournir une assistance aux clients. Voici une architecture possible :

Système de réservation en ligne : Les clients effectuent des réservations via le site web ou application mobile de la plateforme.
Centre d'appels virtuel : Un centre d'appels virtuel est mis en place pour gérer les demandes assistance des clients, telles que les questions sur les réservations, les modifications, les annulations, etc.
Plateforme VoIP : Une plateforme VoIP est utilisée pour acheminer les appels des clients vers les agents du centre d'appels. Cette plateforme peut également inclure des fonctionnalités telles que l'enregistrement des appels et les options de rappel.
Intégration système de réservation - CRM : Les informations sur les réservations des clients sont intégrées au système CRM utilisé par les agents pour fournir une assistance personnalisée.
Service de commerce électronique : Des sites de commerce électronique comme Amazon ou eBay peuvent également utiliser la VoIP pour leur service clientèle. Voici une architecture possible :

Plateforme de commerce électronique : Les clients effectuent des achats via le site web ou l'application mobile de la plateforme.

Centre d'appels interne ou externalisé : Un centre d'appels interne ou externalisé est chargé de gérer les questions des clients, telles que les retours de produits, les problèmes de livraison, les demandes de renseignements sur les produits, etc.

Plateforme VoIP : Une plateforme VoIP est utilisée pour gérer les appels des clients et les acheminer vers les agents du centre d'appels. Elle peut également prendre en charge les fonctionnalités de messagerie vocale et de conférence.

Intégration CRM : Les informations sur les clients et leurs achats sont intégrées au système CRM utilisé par les agents pour fournir un service client personnalisé et efficace.

Ces architectures sont des exemples généraux et peuvent varier en fonction des besoins spécifiques de chaque entreprise et de son infrastructure existante.

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

SRTP (Secure Real-time Transport Protocol) : SRTP est une extension du protocole RTP (Real-time Transport Protocol), utilisé pour le transport des flux audio et vidéo en temps réel sur Internet. SRTP fournit le chiffrement des données de média en utilisant des algorithmes tels que AES (Advanced Encryption Standard) pour assurer la confidentialité des appels VoIP.
TLS (Transport Layer Security) : TLS est un protocole de chiffrement utilisé pour sécuriser les communications sur Internet. Dans le contexte de la VoIP, TLS peut être utilisé pour chiffrer les échanges de données de signalisation (comme les messages SIP) entre les clients VoIP et les serveurs, garantissant ainsi la confidentialité et l'intégrité des informations de signalisation.
ZRTP (Zimmermann Real-time Transport Protocol) : ZRTP est un protocole de chiffrement spécifiquement conçu pour sécuriser les communications en temps réel, telles que les appels vocaux sur IP. Il utilise Diffie-Hellman pour établir une clé de chiffrement partagée entre les parties, garantissant ainsi la confidentialité des conversations VoIP sans nécessiter de certificats pré-installés.
DTLS (Datagram Transport Layer Security) : DTLS est une variante de TLS conçue pour sécuriser les communications basées sur UDP (User Datagram Protocol), qui est souvent utilisé dans les applications de VoIP. DTLS peut être utilisé pour chiffrer les flux de médias VoIP en garantissant la confidentialité et d'intégrité des données échangées entre les parties.
SIPS (SIP Secure) : SIPS est une extension du protocole SIP (Session Initiation Protocol) qui utilise TLS pour sécuriser les échanges de données de signalisation SIP entre les clients VoIP et les serveurs. Cela garantit que les informations de signalisation, telles que les invitations d'appels, les réponses et les mises à jour, sont protégées contre les attaques telles que l'interception et modification.
En utilisant ces protocoles de chiffrement, les fournisseurs de services VoIP et les entreprises peuvent garantir la sécurité et la confidentialité des

communications vocales sur IP, réduisant ainsi les risques liés à l'interception des appels, à l'écoute clandestine et à d'autres formes d'attaques.