DOCUMENTATION

Veille technologique sur la VoIP avantages, solutions et sécurité

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

Les coûts opérationnels et de maintenance d'un système VoIP peuvent être inférieurs à ceux d'un système téléphonique traditionnel pour plusieurs raisons. Tout d'abord, la VoIP utilise le réseau internet existant de l'entreprise, ce qui réduit les coûts d'infrastructure tels que les lignes téléphoniques dédiées et les frais d'interurbain. En outre, les systèmes VoIP sont souvent plus faciles à gérer et à entretenir que les systèmes téléphoniques traditionnels. Les mises à jour logicielles et les modifications de configuration peuvent être effectuées à distance, ce qui réduit les coûts de déplacement et de maintenance sur site.

De plus, les coûts de matériel peuvent également être réduits avec la VoIP, car les téléphones IP peuvent être connectés directement au réseau de l'entreprise, éliminant ainsi le besoin de cartes d'interface téléphonique coûteuses. Les entreprises peuvent également économiser de l'argent en utilisant des logiciels de communication unifiée qui intègrent la voix, la vidéo et la messagerie instantanée en une seule plateforme.

Cependant, il est important de noter que les coûts opérationnels et de maintenance d'un système VoIP peuvent varier en fonction de la taille et de la complexité de l'architecture de l'entreprise. Les entreprises doivent donc évaluer leurs besoins spécifiques en matière de communication et choisir une solution VoIP qui répond à ces besoins tout en étant rentable à long terme.

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 ?

La configuration VoIP d'un call center diffère de celle d'un standard téléphonique d'entreprise en raison du volume d'appels plus élevé et des fonctionnalités avancées requises, telles que la distribution automatique des appels et l'intégration CRM. Les standards téléphoniques d'entreprise ont des besoins plus simples, tels que la messagerie vocale et le transfert d'appels. La configuration dépendra des besoins spécifiques de chaque entreprise et de leur volume d'appels. Il est important de travailler avec un fournisseur de services VoIP expérimenté pour concevoir une solution personnalisée.

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.

Uber et Amazon sont des exemples de sites marchands et de services utilisant des services VoIP pour leur customer service. Les clients peuvent contacter Uber pour des problèmes liés aux courses, tandis qu'Amazon traite les questions concernant les commandes, les retours et les remboursements.

Architecture possible d'un système VoIP pour le customer service :

1. Interface utilisateur : Accessible via le site web, une application mobile ou un logiciel dédié, l'interface utilisateur doit être conviviale.
2. Passerelle VoIP : Responsable de la conversion des signaux vocaux analogiques en signaux numériques et de la connexion entre le réseau téléphonique traditionnel (PSTN) et le réseau VoIP.
3. Serveur VoIP : Gère la communication entre les clients et les agents du service client, prenant en charge plusieurs appels simultanés et offrant des fonctionnalités telles que la mise en attente et le transfert d'appel.
4. Logiciel de centre d'appels : Permet aux agents de gérer les appels entrants et sortants, de consulter les informations sur les clients et de suivre les interactions.
5. Base de données client : Stocke les informations sur les clients pour un accès rapide par les agents du service client.
6. Intégration CRM : Le système VoIP peut être intégré à un logiciel de gestion de la relation client (CRM) pour améliorer la productivité des agents.
7. Sécurité et qualité de service : Des mesures de sécurité et de qualité de service doivent être mises en place pour garantir la confidentialité des communications et la fiabilité du système VoIP.

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

SRTP

SRTP (Secure Real-time Transport Protocol) est une extension d'un profil RTP (Real-time Transport Protocol) qui ajoute des fonctionnalités de sécurité supplémentaires pour les communications en temps réel, telles que la VoIP. Les deux principales fonctionnalités de sécurité ajoutées par SRTP sont l'authentification des messages et la protection anti-replay.

L'authentification des messages permet de vérifier que les données transmises proviennent bien de l'expéditeur prétendu et qu'elles n'ont pas été modifiées pendant la transmission. La protection anti-replay, quant à elle, empêche les attaquants de capturer et de rejouer des messages précédemment transmis dans le but de tromper le destinataire.

SRTP fonctionne en utilisant l'authentification et le chiffrement pour minimiser les risques d'attaques telles que celles par déni de service (DDoS). Il a été publié en 2004 par l'IETF (Internet Engineering Task Force) en tant que RFC 3711.

SRTP est largement utilisé pour sécuriser les communications VoIP et est pris en charge par de nombreux fournisseurs de services et de matériel VoIP. Il est important de noter que SRTP nécessite l'utilisation de clés cryptographiques pour le chiffrement et l'authentification, il est donc important de mettre en place une gestion de clés sécurisée pour utiliser SRTP de manière efficace.

En résumé, SRTP est une extension de RTP qui ajoute des fonctionnalités de sécurité supplémentaires pour les communications en temps réel, telles que l'authentification des messages et la protection anti-replay. SRTP est largement utilisé pour sécuriser les communications VoIP et est pris en charge par de nombreux fournisseurs de services et de matériel VoIP. Cependant, il est important de mettre en place une gestion de clés sécurisée pour utiliser SRTP de manière efficace.

TLS & SSL

SSL (Secure Sockets Layer) et TLS (Transport Layer Security) sont des protocoles cryptographiques conçus pour fournir une communication sécurisée sur Internet. Ils utilisent une combinaison de chiffrement symétrique et asymétrique, de certificats numériques et d'autres mécanismes de sécurité pour sécuriser les données en transit.

SSL a été initialement développé par Netscape dans les années 1990 et a été largement adopté comme moyen de sécuriser le trafic Web. Cependant, SSL a plusieurs vulnérabilités et faiblesses connues, et il est considéré comme obsolète.

TLS est le successeur de SSL, et il a été normalisé pour la première fois en 1999. TLS est basé sur SSL, mais il comprend plusieurs améliorations et renforcements de sécurité. TLS est maintenant le protocole recommandé pour sécuriser le trafic Web, ainsi que d'autres types de trafic Internet, tels que la VoIP.

Les principales différences entre SSL et TLS sont :

- Sécurité : TLS comprend plusieurs améliorations de sécurité par rapport à SSL, telles qu'un échange de clés et une authentification de message améliorés, et une protection contre certains types d'attaques.
- Interopérabilité : TLS est conçu pour être plus interopérable avec d'autres protocoles de sécurité, tels que IPSec et SSH.
- Performances : TLS est conçu pour être plus efficace que SSL, avec des mécanismes de poignée de main et de reprise de session améliorés.

TLS est maintenant largement pris en charge par les navigateurs Web, les serveurs Web et autres périphériques connectés à Internet, et il est le protocole recommandé pour sécuriser le trafic Web. SSL ne doit pas être utilisé dans les nouveaux systèmes, car il est considéré comme non sécurisé.

En résumé, SSL et TLS sont des protocoles cryptographiques utilisés pour sécuriser la communication sur Internet. TLS est le successeur de SSL, et il comprend plusieurs améliorations de sécurité par rapport à SSL. TLS est maintenant le protocole recommandé pour sécuriser le trafic Web et d'autres types de trafic Internet, tandis que SSL ne doit pas être utilisé dans les nouveaux systèmes.

ZRTP

ZRTP (Zimmermann Real-time Transport Protocol) est un protocole de sécurité pour les communications en temps réel, tel que la VoIP, qui fournit un chiffrement de bout en bout et une authentification des appels. Il a été conçu par Phil Zimmermann, le créateur de PGP (Pretty Good Privacy), et est basé sur le protocole SRTP (Secure Real-time Transport Protocol).

ZRTP utilise une méthode de chiffrement de clé publique pour établir une connexion sécurisée entre les deux parties d'un appel, sans avoir besoin d'une infrastructure de clé publique (PKI) ou d'un serveur de confiance tiers. Au lieu de cela, ZRTP utilise une méthode de "short authentication string" (SAS) pour vérifier l'authenticité de la connexion. Les deux parties de l'appel peuvent comparer les SAS et vérifier qu'elles correspondent pour confirmer que la connexion est sécurisée.

ZRTP est conçu pour être facile à utiliser et ne nécessite pas de configuration ou de connaissances techniques spécialisées. Il est également conçu pour être résistant aux attaques de type "man-in-the-middle" (MITM) et pour fournir une confidentialité persistante, ce qui signifie que les clés de chiffrement utilisées pour un appel ne peuvent pas être utilisées pour déchiffrer les appels précédents ou suivants.

ZRTP est un protocole ouvert et a été publié en tant que RFC 6189 par l'IETF (Internet Engineering Task Force) en 2011. Il est pris en charge par plusieurs fournisseurs de services et de matériel VoIP, ainsi que par des logiciels open source tels que Jitsi et Linphone.

En résumé, ZRTP est un protocole de sécurité pour les communications en temps réel, tel que la VoIP, qui fournit un chiffrement de bout en bout et une authentification des appels. Il utilise une méthode de chiffrement de clé publique pour établir une connexion sécurisée entre les deux parties d'un appel, sans avoir besoin d'une infrastructure de clé publique (PKI) ou d'un serveur de confiance tiers. ZRTP est conçu pour être facile à utiliser et est pris en charge par plusieurs fournisseurs de services et de matériel VoIP, ainsi que par des logiciels open source.

IPSec

IPsec (Internet Protocol Security) est un protocole de sécurité réseau qui permet de sécuriser les communications IP en fournissant des services de confidentialité, d'intégrité et d'authentification des données. Il peut être utilisé pour sécuriser les communications entre deux hôtes, entre deux réseaux ou entre un hôte et un réseau.

IPsec fonctionne en utilisant deux modes : le mode transport et le mode tunnel. Le mode transport chiffre les données de la charge utile IP et ajoute une en-tête d'authentification, tandis que le mode tunnel chiffré l'ensemble du paquet IP et ajoute une nouvelle entête IP. Le mode tunnel est souvent utilisé pour sécuriser les communications VPN (Virtual Private Network).

IPsec utilise des algorithmes de chiffrement tels que AES (Advanced Encryption Standard) et DES (Data Encryption Standard) pour chiffrer les données, et des algorithmes d'authentification tels que HMAC (Hash-based Message Authentication Code) pour vérifier l'intégrité des données. Il utilise également des certificats numériques pour authentifier les hôtes et les réseaux.

IPsec est largement pris en charge par les routeurs, les pare-feu et les systèmes d'exploitation modernes, et il est souvent utilisé pour sécuriser les communications entre les réseaux d'entreprise et les réseaux distants, ainsi que pour sécuriser les communications VPN.

En résumé, IPsec est un protocole de sécurité réseau qui permet de sécuriser les communications IP en fournissant des services de confidentialité, d'intégrité et d'authentification des données. Il fonctionne en utilisant deux modes : le mode transport et le mode tunnel, et utilise des algorithmes de chiffrement et d'authentification pour sécuriser les données. IPsec est largement pris en charge et est souvent utilisé pour sécuriser les communications entre les réseaux d'entreprise et les réseaux distants, ainsi que pour sécuriser les communications VPN.

POUR RÉSUMER

En résumé, SRTP et ZRTP sont des protocoles spécifiquement conçus pour sécuriser les communications VoIP en fournissant un chiffrement de bout en bout et une authentification des appels. SRTP utilise une infrastructure de clé publique (PKI) ou un serveur de confiance tiers pour établir une connexion sécurisée, tandis que ZRTP utilise une méthode de "short authentication string" (SAS) pour vérifier l'authenticité de la connexion, ce qui le rend plus facile à utiliser et ne nécessite pas de configuration ou de connaissances techniques spécialisées.

IPsec, d'autre part, est un protocole de sécurité réseau plus général qui peut être utilisé pour sécuriser les communications entre deux hôtes, entre deux réseaux ou entre un hôte et un réseau. Il peut être utilisé pour sécuriser les communications VoIP en utilisant le mode tunnel pour chiffrer l'ensemble du paquet IP, mais il n'est pas spécifiquement conçu pour cela et peut nécessiter une configuration supplémentaire pour fonctionner correctement avec les applications VoIP.

En fin de compte, le choix entre SRTP, ZRTP et IPsec dépendra des besoins spécifiques de votre entreprise et de votre réseau. Si vous avez besoin d'une solution facile à utiliser et spécifiquement conçue pour la VoIP, SRTP ou ZRTP seraient probablement les meilleurs choix. Si vous avez besoin d'une solution plus générale pour sécuriser toutes les communications réseau, IPsec pourrait être un meilleur choix. Il est important de noter que les trois protocoles peuvent être utilisés ensemble pour fournir une sécurité supplémentaire. Par exemple, IPsec peut être utilisé pour sécuriser la connexion entre les réseaux, tandis que SRTP ou ZRTP peut être utilisé pour sécuriser les communications VoIP entre les hôtes.

Installation d'Asterisk

Téléchargement et installation des mises à jour :

```
sudo apt upgrade && sudo apt update
```

Asterisk sous Debian 12

Installation des fichiers :

```
wget https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-21.2.0.tar.gz
```

Décompression de l'archive :

```
tar -zxvf asterisk-21.2.0.tar.gz
```

Installation des dépendances nécessaires pour VOIP :

```
apt install build-essential
apt install libncurses5-dev libssl-dev libxml2-dev libsqlite3-dev uuid-dev
```

Pour la suite nous allons dans le répertoire du paquet asterisk :

```
cd asterisk-21.2.0
```

Installation d'autres librairies nécessaires au bon fonctionnement d'asterisk :

```
sudo apt install libedit-dev
sudo apt install libjansson-dev
```

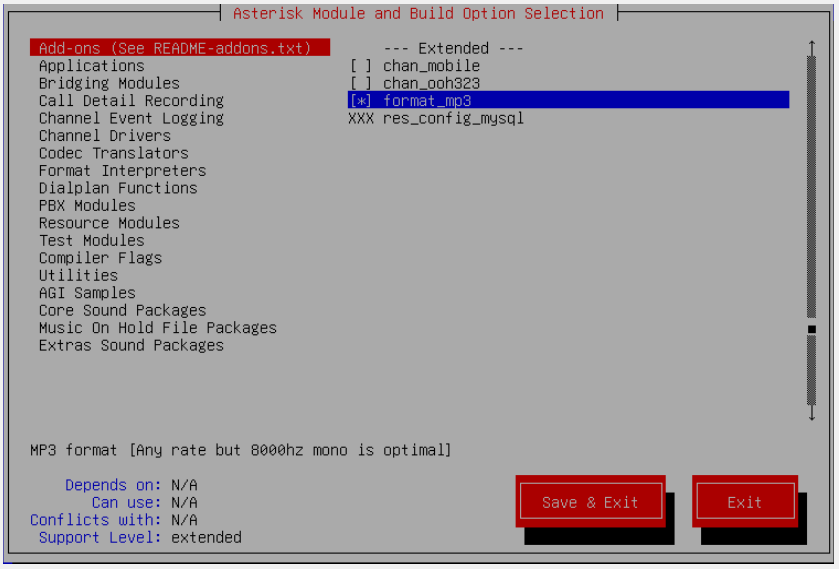
Exécution du script de configuration de la compilation des paquets :

```
./configure --with-jansson-bundled
```

Installation des modules souhaité :

```
sudo make menuselect
```

Le terminal affiche ensuite ce menu :



Sélectionner Add-ons puis le module mp3

Compilation du paquet asterisk :

```
sudo make
```

Inclure les paquets à l'installation d'asterisk & Installation des fichiers de configuration génériques

```
sudo make samples
```

Installation du script d'initialisation Asterisk

```
sudo make config
```

Démarrage d'Asterisk

```
sudo systemctl start asterisk
```

Installation terminée !

Configuration d'Asterisk

Vérification de l'activité du service

```
sudo systemctl status asterisk
```

Création des sauvegardes des fichiers de configuration

```
sudo cp /etc/asterisk/extensions.conf /etc/asterisk/extensions.conf.sample
sudo cp /etc/asterisk/pjsip.conf /etc/asterisk/pjsip.conf.sample
```

/etc/asterisk/pjsip.conf

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

[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

[alice](endpoint_internal)
auth=alice
aors=alice
[alice](auth_userpass)
password=alice101
username=alice
[alice](aor_dynamic)

[bob](endpoint_internal)
auth=bob
aors=bob
[bob](auth_userpass)
password=bob102
username=bob
[bob](aor_dynamic)
```

/etc/asterisk/extensions.conf

```
[from-internal]
exten=>101,1,Dial(PJSIP/alice,10)
exten=>102,1,Dial(PJSIP/bob,10)
exten=>199,1,VoiceMailMain()      ; Numéro de téléphone du répondeur

; Règles 2
exten=>101,2,VoiceMail(101)      ; Appel répondeur compte 101
exten=>102,2,VoiceMail(102)      ; Appel répondeur compte 102
```

Ce code définit les règles de gestion des appels pour trois numéros de téléphone internes : 101, 102 et 199. Les deux premiers numéros (101 et 102) correspondent à des extensions internes, tandis que le troisième (199) est le numéro du répondeur.

Pour l'extension 101, la règle 1 indique que le serveur doit composer le numéro PJSIP/alice et attendre pendant 10 secondes que quelqu'un décroche. Si personne ne répond, la règle 2 est appliquée et l'appel est redirigé vers le répondeur du compte 101.

Pour l'extension 102, la règle 1 indique que le serveur doit composer le numéro PJSIP/bob et attendre pendant 10 secondes que quelqu'un décroche. Si personne ne répond, la règle 2 est appliquée et l'appel est redirigé vers le répondeur du compte 102.

Enfin, le numéro 199 est configuré pour rediriger directement l'appelant vers le répondeur principal du serveur, grâce à la commande VoiceMailMain().

En résumé, ce code permet de gérer les appels entrants pour les extensions 101 et 102, en redirigeant les appels non répondus vers les répondeurs respectifs de ces extensions, et en fournissant un numéro de téléphone dédié pour accéder au répondeur principal du serveur.

/etc/asterisk/voicemail.conf

```
[general]
format=wav49|gsm|wav|ulaw
maxmsg=30               ; Max messages
maxsecs=0               ; Durée max message
minsecs=1               ; Durée min message
maxlogins=3             ; Nombre tentative login
review=yes              ; Permet à l'appelant de réécouter son message

[default]
; Numéro de messagerie => mot de passe, nom d'utilisateur
101 => 1234, alice
102 => 1234, bob
```

Voici une explication des différentes lignes de ce fichier :

- La section `[general]` contient des paramètres généraux pour le système de messagerie vocale.
- La ligne `format=wav49|gsm|wav|ulaw` spécifie les formats audio pris en charge pour les messages vocaux. Dans ce cas, les formats wav49, gsm, wav et ulaw sont supportés.
- La ligne `maxmsg=30` définit le nombre maximum de messages que chaque boîte vocale peut contenir.
- La ligne `maxsecs=0` définit la durée maximale d'un message vocal en secondes. La valeur 0 signifie qu'il n'y a pas de limite de durée.
- La ligne `minsecs=1` définit la durée minimale d'un message vocal en secondes. Dans ce cas, un message vocal doit durer au moins 1 seconde.
- La ligne `maxlogins=3` définit le nombre maximum de tentatives de connexion autorisées avant que le système ne verrouille la boîte vocale.
- La ligne `review=yes` permet à l'appelant de réécouter son message vocal avant de l'enregistrer définitivement.
- La section `[default]` contient des paramètres pour les boîtes vocales individuelles.
- La ligne `101 => 1234, alice` définit une boîte vocale avec le numéro 101, le mot de passe 1234 et le nom d'utilisateur "alice".
- La ligne `102 => 1234, bob` définit une boîte vocale avec le numéro 102, le mot de passe 1234 et le nom d'utilisateur "bob".

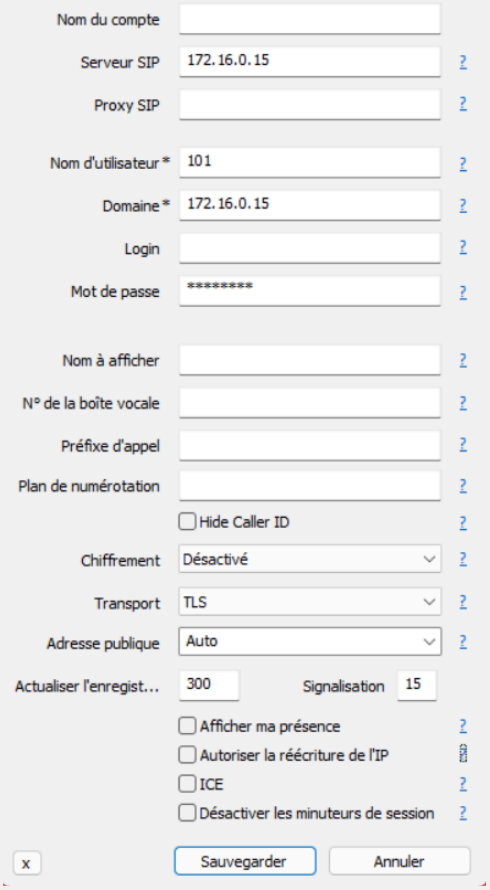
En résumé, ce fichier de configuration définit les paramètres généraux pour un système de messagerie vocale et les paramètres pour deux boîtes vocales individuelles.

Configuration des clients

Ordinateurs (MicroSIP)

Télécharger MicroSIP : https://www.microsip.org/download/MicroSIP-3.21.3.exe

Configuration MicroSIP : Ctrl+M puis



Mobile

Sécurisation de la communication

Transport Layer Security (TLS)

Transport Layer Security (TLS) fournit un cryptage pour la signalisation des appels. Il s'agit d'un moyen pratique d'empêcher les personnes extérieures à Asterisk de savoir qui vous appelez. La configuration de TLS entre Asterisk et un client SIP implique la création de fichiers clés, la modification de la configuration SIP ou PJSIP d'Asterisk pour activer TLS, la création d'un pair SIP capable de TLS et la modification du client SIP pour qu'il se connecte à Asterisk via TLS.

Génération d'un certificat SSL "auto-signé" + Fichiers clés

Création du répertoire qui va contenir tous nos fichiers :

```
mkdir /etc/asterisk/keys
```

Utilisation du script ast_tls_cert pour créer nos fichiers dont notre certificat

```
./usr/src/asterisk/ast_tls_cert -C 172.16.0.15 -O "VoipCo" -d /etc/asterisk/keys -b 2048
```

option -C : définit le nom de domaine ou l'adresse IP du serveur VoIP.
option -O : définit le nom de l'organisation.
option -d : répertoire qui va contenir les fichiers.
option -b : spécifie la taille du fichier de clé privée, par défaut 1024.

Durant le déroulement du script, il est demandé de renseigner une phrase de passe pour le fichier /etc/asterisk/keys/ca.key qu'il faut garder en mémoire le temps de la génération. Il faut renseigner la phrase 4 fois au total.

Les fichiers clés ne doivent être disponibles uniquement en lecture, dans le cas contraire :

```
chmod 600 /etc/asterisk/keys/*.*
```

Configuration du canal PJSIP

```
sudo nano /etc/asterisk/sip.conf
```

Modifier la section [general] de la façon suivante :

```
[general]
context=public
bindaddr=0.0.0.0
bindport=5060
tlsbindaddr=0.0.0.0:5061
transport=udp,tls
deny=all
allow=alaw,g722
dmtf=rfc2833
tlsenable=yes
tlscertfile=/etc/asterisk/keys/asterisk.pem

[transport-tls]
type = transport
protocol = tls
bind = 0.0.0.0:5061
cert_file=/etc/asterisk/keys/asterisk.crt
priv_key_file=/etc/asterisk/keys/asterisk.key
```

Redémarrer ensuite le service :

```
sudo systemctl restart asterisk.service
```