République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Bejaïa

Faculté des Sciences Exactes

Département Informatique

Mémoire de Fin de Cycle

En vue de l'obtention du diplôme de Master Professionnel en Administration et Sécurité des Réseaux

Thème

Etude et Mise en Place D'une Solution Voix sur IP Sécurisée

Réalisé par :

M^lle^ Tigrine Ikram

M^lle^ Boughebri Thanina

Evalué par le jury :

M^me^ L. Bachiri	**MCA**	**U.A/Mira Bejaia**	**Présidente**
M^me^ S. Lahlah	**MCB**	**U.A/Mira Bejaia**	**Examinatrice**
M^me^ L. Chelouah	**MCB**	**U.A/Mira Bejaia**	**Encadrante**

Promotion 2020-2021

Remerciement

Nous remercions tout d'abords, dieu le tout puissant de nous avoir accordé la force, la volonté et la connaissance pour accomplir ce travail ;

Nous tenons à remercier notre encadrante M^me^ Chelouah Leila, pour ces précieux conseils et orientations ;

On tient aussi à remercier vivement les membres du jury qui ont accepté d'évaluer notre projet. Nous leurs présentons toute nos gratitudes et nos profonds respects.

On souhaite exprimer enfin notre gratitude et nos vifs remerciements à nos familles et nos amis pour leurs soutiens. Et à M^r^ Ali Larbi qui a participé pour la réalisation de ce travail avec ces conseils.

Dédicace

Du profond de mon cœur, je dédie ce travail à tous ceux qui me sont chers,

A MA CHERE MERE

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.

Je vous remercie pour tout le soutien et l'amour que vous me portez depuis mon enfance et j'éspère que votre bénédiction m'accompagne toujours.

A LA MEMOIRE DE MON PERE

Ce travail est dédié à mon père, décédé trop tôt, qui m'a toujours poussé et motivé dans mes études.

J'espère que, le monde qui est sien maintenant, il apprécie cet humble geste comme preuve de reconnaissance de la part d'une fille qui a toujours prié pour le salut de son âme. Puisse Dieu, le tout puissant, l'avoir en sa sainte miséricorde !

Je dédie également ce projet à ma chère sœur TIGRINE INES et mon chèr frère TIGRINE LYES que DIEU vous procure santé et joie pour le restant de la vie……Je vous aime !

Et à mr Hadji qui a participé pour la réalisation de ce travail.

A tous ceux qui m'ont dispensé le savoir.

A tous ceux qui ont participés de près ou de loin à l'élaboration de ce travail.

Merci beaucoup.

Mlle Tigrine Ikram

Dédicace

A mes chèr parents, ma sœur ''Rosa''et mes frères '' Rayan'' et ''Ghilas'' ; je suis fière de vous avoir à mes cotés. Vous etes ma raison de vivre, merci de m'avoir accompagné et encouragé tout au long de cette épreuve, j'espère pouvoir un jour, vous rendre ne serait-ce qu'une petite partie de ce que vous m'avez donné.

Je dédie ce travail à mon ami''VIP'' qui a toujours été disponible à n'importe quel moment et qui a vécu cette période avec moi avec tous les hauts et les bas.

Je remercie particulièrement mon encadrante Mme Chelouah Leila pour son soutien, ses précieux conseils et surtout sa disponibilité et ses encouragements.

Je dédie également à tous mes amis proches (Bahia, Souad et Zahia à mes chères partenaires merci encore de m'avoir soutenu tout au long de la réalisation de ce mémoire.

Je remercie également ma binome avec qui on a pu réussir à élabiorer ce travail.

Mlle Boughebri Thanina

Table des matières

Table des matières

Table des matières

Table des matières

Table des matières

Table des figures

Chapitre1

Chapitre2

Chapitre3

Table des figures

Chapitre4

Liste des tableaux

Chapitre2

Chapitre3

Acronyme

ACE : Access Control Entry

ACL : Access Control List

AH : Authentification Header

ARP : Address Resolution Protocol

CAN : Convertisseur analogique numérique

CLI : Command Line Interface

CSR : Corporate Social Responsibility

DDoS : Distributed Denial of Service

DHCP : Dynamic Host Configuration Protocol

DMZ : Démilitarized Zone

DNS : Domain Name System

DoS : Deny of Service

DTMF : Dual-Tone Multi-Frequency

DES : Data Encryption Standard

ESP : Encapsulated Security Payload

FTP : File Transfer Protocol

FXO : Foreign eXchange Office

FXS : Foreign eXchange Subscriber

GSM : Global System for Mobile Communications

HTTP : HyperText Transfer Protocol

HTML : HyperText Markup Language

HMAC : keyed-hash message authentication code

IAX : Inter-Asterisk eXchange

ICMP : Internet Control Message Protocol

IETF : Internet Engineering Task Force

IGMP : Internet Group Management Protocol

IGRP : Interior Gateway Routing Protocol

IM : Instant Message

IP : Internet Protocol

IPX : Internetwork Packet Exchange

ISDN : Integrated Service Data Network

ITU : International Telecommunications Union

Acronyme

LAN : Local Area Network

MD5 : Message Digest 5

MIKEY : Multimedia Internet KEYing

MKI : Master Key Identifier

MGCP : Media Gateway Control Protocol

MEGACO : Media Getway Control Protocols and Megaco

MC : Microphone Controller

NAT : Network Address Translation

OS : Operating System

PABX : Private Automatic Branch eXchange

PBX : Private Branch eXchange

PSTN : Public Switched Telephone Network

PEM : Pain Explosif Malléable

QoS : Quality of Service

RFC : Requests For Comment

RNIS : Réseau Numérique à Intégration de Service

RTC : Réseau Téléphonique Commuté

RTCP : Real-time Transport Control Protocol

RTP : Real-Time Transport Protocol

RTSP : Real Time Streaming Protocol

RSVP : Resource ReSerVation Protocol

SIP : Session Initiation Protocol

SNMP : Simple Network Management Protocol

SRTP : Secure Real-time Transport Protocol

SHA : pour Secure Hash Algorithm

SSL : Secure Socket Layer

TCP : Transport Control Protocol

TDM : Time Division Multiplexing

TFTP : Trivial File Transfert Protocol

TLS : Transport Layer Security

ToIP : Telephony over Internet Protocol

UAC : User Agent Client

UAS : User Agent Server

Acronyme

UDP : User Datagram Protocol

URL : Uniform Resource Locator

USB : Universal Serial Bus

VoIP : Voice over Internet Protocol

VPN : Virtual Private Network

WAN : World Area Network

Introduction Générale

La téléphonie traditionnelle est en voie d'extinction, et la VoIP (acronyme de Voice over Internet Protocol), devient progressivement populaire. Les communications IP (pour Internet Protocol) sont continuellement déployées au sein de l'industrie téléphonie depuis les deux dernières décennies.

Crée à l'origine pour le trafic des données, le protocole IP s'est adapté au trafic vocal grâce à son succès mondial en tant que réseau de données.

Avec les systèmes de téléphonie VoIP, les utilisateurs ne sont pas limités au réseau IP pour passer et recevoir des appels. Les lignes téléphoniques peuvent aussi être utilisées pour garantir une disponibilité et une qualité d'appel supérieure. Avec l'utilisation d'une passerelle VoIP, les lignes téléphoniques RTC peuvent être converties en VoIP/SIP (pour Session Initialisation Protocol). Les passerelles VoIP permet à l'utilisateur de passer et recevoir des appels via le réseau téléphonique traditionnel.

Plusieurs fournisseurs offrent certaines solutions qui permettent aux entreprises de migrer vers le monde IP. Des constructeurs de PABX (pour Private Automatic Branch eXchange) tels que Nortel, Siemens, et Alcatel préfèrent la solution de l'intégration progressive de la VoIP qui ne permet pas de bénéficier de tous les services et la bonne intégration vers le monde de données. Le développement des PABXs software, est la solution proposée par des fournisseurs tels que Cisco et Asterisk. Cette solution, qui est totalement basée sur la technologie IP, est donc affectée par les attaques et les vulnérabilités qui menacent la sécurité de ce protocole et l'infrastructure réseau sur laquelle elle est déployée. Cette dernière est le majeur problème pour les entreprises, et un grand défi pour les développeurs.

Dans ce contexte, nous réalisons une mise en place deux mécanismes de sécurité pour la voix IP à savoir : chiffrement des appels dans lequel nous avons réalisé TLS (Transport Layer Security) à l'aide du protocole SIP (Session Initiation Protocol : l'extension du protocole H.323) et SRTP (Secure Real-time Transport Protocol) à l'aide du protocole RTP (Real-time Transport Protocol) ; et implémentation d'un Pare-Feu simplifié (UFW: Uncomplicates

FireWall) sécurisés contre ses attaques afin de réduire le taux de risque d'attaque sur les réseaux VoIP.

Ce mémoire est organisé en quatre chapitres :

Dans le ***premier chapitre***, nous présentons la voix sur IP, leurs architectures et protocoles, leurs différents composants, leurs avantages et inconvénients et leurs modes de communication.

Le ***second chapitre*** est une étude sur les différentes attaques et vulnérabilités sur les divers composants d'une infrastructure de la VoIP dans les réseaux LAN (pour Local Area Network).

Dans le ***troisième chapitre***, nous réalisons l'installation et la configuration d'Asterisk sous le système d'exploitation Linux, ainsi que l'installation et la configuration de Twinkle et MizuDroid.

Le ***quatrième chapitre***, quant à lui, sera consacré à la mise en place des mécanismes de sécurité pour la VoIP. Des tests ont été réalisés où des résultats ont été présentés.

Enfin, notre travail s'achève par une conclusion générale résumant les grands points qui ont été abordés ainsi que les perspectives que nous souhaitons accomplir prochainement.

CHAPITRE 1
Concepts généraux de la Voix sur IP

Introduction

La voix sur IP constitue actuellement l'évolution la plus importante du domaine de la Télécommunication. Avant 1970, la transmission de la voix s'effectuait de façon analogique sur des réseaux dédiés à la téléphonie. La technologie utilisée, était la technologie électromécanique (Crossbar). Dans les années 80, une première évolution majeure a été le passage a la transmission numérique (TDM). La transmission de la voix sur les réseaux informatiques à commutation des paquets IP constitue aujourd'hui une nouvelle évolution majeure comparable aux précédentes. D'où la voix sur IP, qui est un terme qui désignant les protocoles, les logiciels et le matériel qui permettent la transmission de medias à temps réel, sous la forme de paquets. La voix sur IP est devenue importante pour les entreprises. L'enjeu est de réussir à faire converger le réseau de données IP et le réseau téléphonique actuel.

L'objectif de ce chapitre est l'étude de cette technologie et de ses différents aspects. On parlera en détail de l'architecture de la VoIP, ses éléments et son principe de fonctionnement. On détaillera aussi des protocoles VoIP de signalisation et de transport ainsi que leurs principes de fonctionnement et de leurs principaux avantages et inconvénients.

1. Présentation sur la voix sur IP

1.1. Définition

VoIP signifie Voice Over Internet Protocol ou Voix sur IP. Comme son nom l'indique, la VoIP permet de transmettre des sons dans des paquets IP circulant sur Internet. La VoIP peut utiliser du matériel d'accélération pour réaliser ce but et peut aussi être utilisée en environnement de PC [1].

1.2. Architecture

La VoIP étant une nouvelle technologie de communication, elle n'a pas encore de standard unique. En effet, chaque constructeur apporte ses normes et ses fonctionnalités à ses solutions. Les trois principaux protocoles sont H.323, SIP et MGCP/MEGACO (pour Media Gateway Control Protocol / Media Gateway Control Protocols and Megaco. Il existe donc plusieurs approches pour offrir des services de téléphonie et de visiophonie sur des réseaux IP. Certaines placent l'intelligence dans le réseau alors que d'autres préfèrent une approche peer

to peer avec l'intelligence répartie à la périphérie (terminal de téléphonie IP, passerelle avec le réseau téléphonique commuté, etc). Chacune a ses avantages et ses inconvénients [1].

La figure 1 décrit de façon générale, la topologie d'un réseau de téléphonie IP. Elle comprend toujours des terminaux, un serveur de communication et une passerelle vers les autres réseaux. Chaque norme a ensuite ses propres caractéristiques pour garantir une plus ou moins grande qualité de service. L'intelligence du réseau est aussi déportée soit sur les terminaux, soit sur les passerelles/Gatekeeper (contrôleur de commutation), appelées Gatekeeper. On retrouve les éléments communs suivants [2] :

a. Le routeur : permet d'aiguiller les données et le routage des paquets entre deux réseaux. Certains routeurs, comme les Cisco 2600, permettent de simuler un Gatekeeper grâce à l'ajout de cartes spécialisées supportant les protocoles VoIP.

b. La passerelle : permet d'interfacer le réseau commuté et le réseau IP.

c. Le PABX : est le commutateur du réseau téléphonique classique. Il permet de faire le lien entre la passerelle ou le routeur, et RTC (pour Réseau Téléphonique Commuté). Toutefois, si tout le réseau devient IP, ce matériel devient obslète.

d. Les terminaux : sont généralement de type logiciel (software phone) ou matériel (hardphone). Le SoftPhone est installé dans le PC de l'utilisateur. L'interface audio peut être un microphone et des haut-parleurs branchés sur la carte son, même si un casque est recommandé. Pour une meilleure clarté, un téléphone USB (pour Universal Serial Bus) ou Bluetooth peut être utilisé.

Le hardphone est un téléphone IP qui utilise la technologie de la voix sur IP pour permettre des appels téléphoniques sur un réseau IP tel que l'Internet au lieu de l'ordinaire système PSTN (pour Public Switched Telephone Network). Les appels peuvent parcourir par le réseau internet comme par un réseau privé. Un terminal utilise des protocoles comme le SIP (pour Session Initiation Protocol) ou l'un des protocoles propriétaires tel que celui utilisé par Skype.

e. X-lite : X-Lite est un logiciel très utilisé par les utilisateurs de voix sur IP car il utilise le protocole SIP. Les fournisseurs de voix sur IP proposent parfois leur service de téléphonie sur IP via l'utilisation de X-Lite qui est gratuit. Le logiciel est très design et instinctif. Des sociétés telles que Free IP Call ou Vonage proposent à leurs clients d'effectuer des appels via ce logiciel. Pour l'utiliser, il faut ouvrir un compte chez un fournisseur de Voix sur IP et

configurer le X-Lite avec les informations fournies par votre fournisseur. Il supporte plusieurs formats de codecs audios tes que GSM, G.771a, G771u. Il permet le transfert d'appels et beaucoup des fonctions. Il est libre et gratuit [4].

La VoIP fonctionne par numérisation de la VoIP, puis par reconversion des paquets numériques en voix à l'arrivée. Le format numérique est plus facile à contrôler, il peut être compressé, routé et converti à un nouveau meilleur format. Le signal numérique est plus tolérant au bruit que l'analogique [1].

Il existe plusieurs protocoles de sécurité qui peuvent supporter la VoIP tel que le H.323, SIP et MGCP.

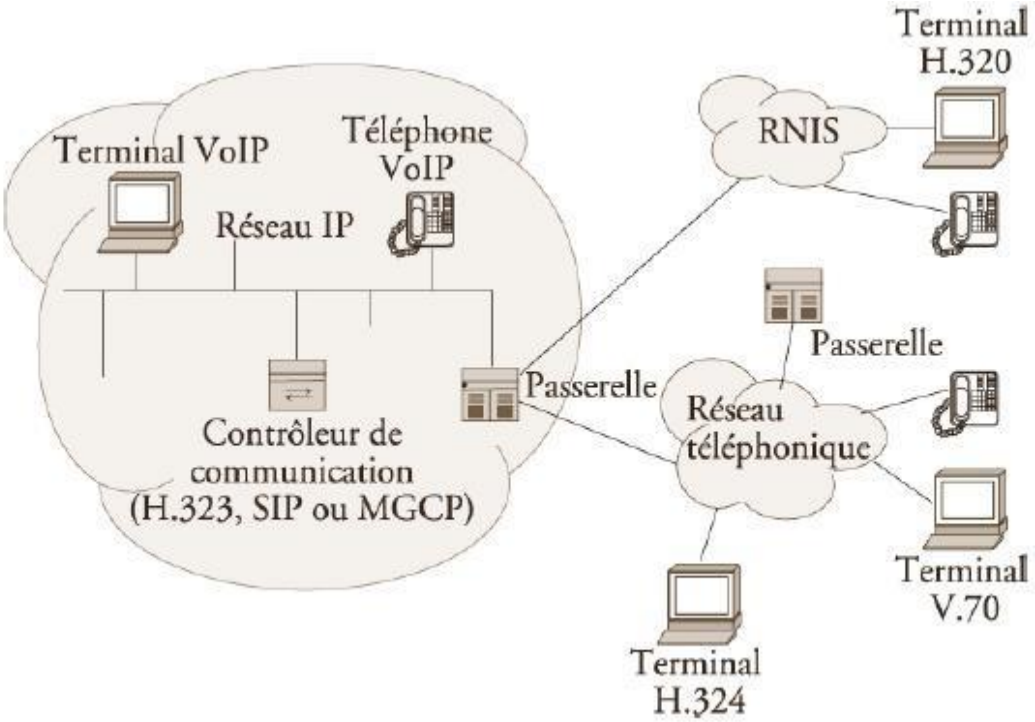

Figure 1 : Architecture générale de la voix sur IP.

1.3. Principe de fonctionnement

Depuis nombreuses années, il est possible de transmettre un signal à une destination éloignée sous forme de données numériques. Avant la transmission, il faut numériser le signal à l'aide d'un CAN (Convertisseur Analogique-Numérique). Le signal est ensuite transmis,

pour être utilisable, il doit être transformé de nouveau en un signal analogique, à l'aide d'un CNA (pour Convertisseur Numériques-Analogique) [1].

L'analogique et le numérique sont deux procédés pour transporter et stocker des données. (de type audio, photo, vidéo, etc). L'analogique est né avec le début de l'électricité tandis que le numérique est apparu plus récemment avec première de l'informatique.
Le principe de l'analogique est de reproduire le signal à enregistrer (audio, vidéo, etc) sous forme similaire sur un support (magnétique en général). Par exemple lorsque l'on enregistre un signal audio sur un système analogique le signal présent sur la bande suivra les mêmes amplitudes (la même courbe) que l'onde sonore (avec plus ou moins de fidélité) : les variations de pressions caractéristiques d'une onde sonore seront traduites en variations d'un signal électrique. Ainsi l'amplitude électrique du signal analogique sera l'image plus ou moins fidèle du signal à enregistrer (audio, vidéo, etc).
Il faut bien garder à l'esprit que le numérique ne sert (dans le cas d'un signal audio ou vidéo) qu'au transport et au stockage des données.

Les réseaux TCP/IP sont des supports de circulation de paquets IP contenant un en-tête (pour controler la communication) et une charge utile pour transporter les données [1].

Il existe plusieurs protocoles qui peuvent supporter la VoIP tels que le H.323, SIP et MGCP. Les deux protocoles les plus utilisés actuellement dans les solutions VoIP présentes sur le marché sont le H.323 et le SIP [1].

2. Les mécanismes de sécurité de la VoIP

2.1. Le protocole H.323

2.1.1. Description générale du protocole H.323

Le standard H.323 fournit, depuis son approbation en 1996, un cadre pour les communications audio, vidéo et de données sur les réseaux IP. Il a été développé par l'ITU (pour International Télécommunication Union) pour les réseaux qui ne garantissent pas une qualité de service (QoS : Quality of Service), tel que IP et IPX (pour Internetwork Packet Exchange) sur Ethernet, Fast Ethernet et Token Ring. Il est présent dans plus de 30 produits et il concerne le contrôle des appels, la gestion multimédia, la gestion de la bande passante pour

les conférences point-à-point et multipoints. H.323 traite également de l'interfaçage entre la LAN et les autres réseaux [2].

Le protocole H.323 fait partie de la série H.23x qui traite de la vidéoconférence au travers différents réseaux. Il inclut H.320 et H.324 liés aux réseaux ISDN (pour Intergrated Service Data Network) et PSTN (pour Public Switched Téléphone Network) [2].

Plus qu'un protocole, H.323 crée une association de plusieurs protocoles différents et qui peuvent être regroupés en trois catégories : La signalisation, La négociation de codec et Le transport de l'information [2].

- Les messages de signalisation sont ceux envoyés pour demander la mise en relation de deux clients, qui indiquent que la ligne est occupée ou que le téléphone sonne. En H.323, la signalisation s'appuie sur le protocole RAS pour l'enregistrement et l'authentification, et le protocole Q.931 pour l'initialisation et le contrôle d'appel [2].

- La négociation est utilisée pour se mettre d'accord sur la façon de coder les informations à échanger. Il est important que les téléphones (ou systèmes) utilisent un langage commun s'ils veulent se comprendre. Il s'agit de codec le moins gourmand en bande passante ou de celui qui offre la meilleure qualité. Il serait aussi préférable d'avoir plusieurs altératives de langages. Le protocole utilisé pour la négociation de codec est le H.245 [2].

- Le transport de l'information s'appuie sur le protocole RTP qui transporte la voix, la vidéo ou les données numérisées par les codecs. Les messages RTCP peuvent être utilisés pour le contrôle de la qualité, ou la négociation des codecs si par exemple la bande passante diminue [2].

Une communication H.323 se déroule en cinq phases :

- L'établissement d'appel ;
- L'échange de capacité et réservation éventuelle de la bande passante à travers le protocole RSVP (pour Resource ReSerVation Protocol) ;
- L'établissement de la communucation audio-visuelle ;
- L'invocation éventuelle de services en phase d'appel (par exemple : transfert d'appel, changement de bande passante, etc) ;
- Et enfin la libération de l'appel.

L'infrastructure H.323 repose sur quatre composants principaux :

- Les terminaux ;
- Les Getways ;
- Les Gatekeepers ;
- Les MCU (Multipoint Control Units).

2.1.2. Rôle des composants

Le support de H.323 est composer de : les terminaux, les Gateways, les Gatekeepers, et les MCU [2].

Comme illustre sur la figure 2.

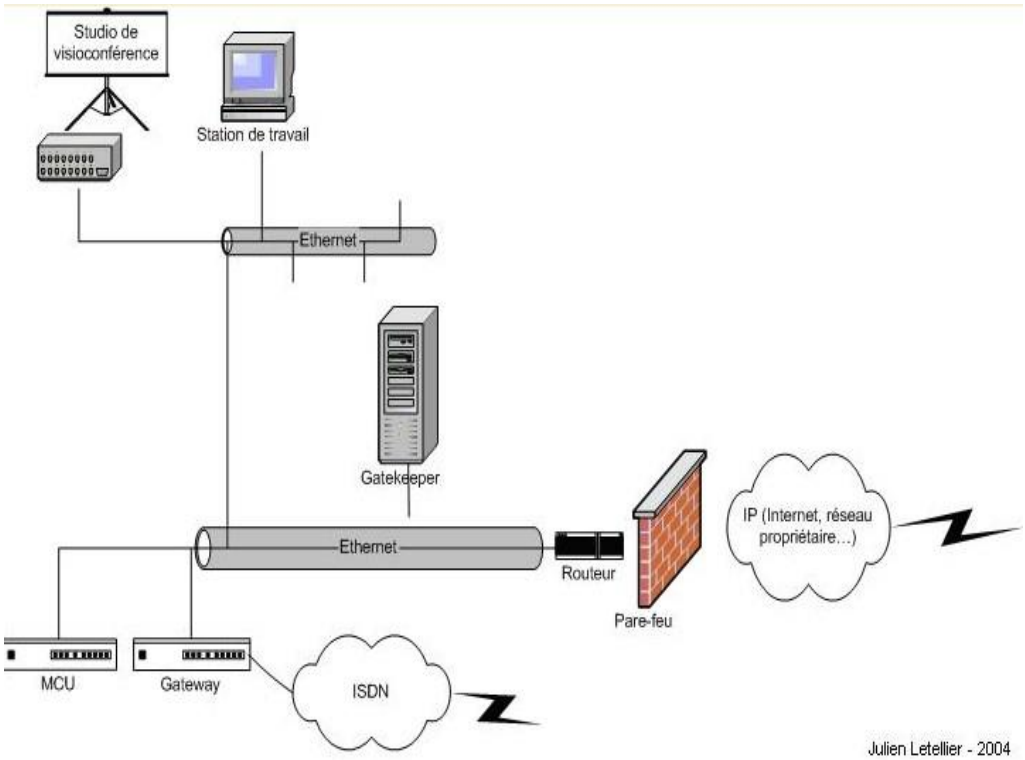

Figure 2 : Les composants de l'architecture H.323.

a. Les terminaux H.323

Le terminal peut être un ordinateur, un combiné téléphonique, un terminal spécialisé pour la vidéoconférence ou encore un télécopieur sur Internet. Le minimum imposé par H.323 est qu'il mette en œuvre la norme de compression de la parole G.711, qu'il utilise le protocole

H.245 pour la négociation de l'ouverture d'un canal et l'établissement des paramètres de la communication, ainsi que le protocole de signalisation Q.931 pour l'établissement et l'arrêt des communications. Le terminal possède également des fonctions optionnelles, notamment, pour le travail en groupe et le partage des documents. Il existe deux types de terminaux H.323, l'un de haute qualité (pour une utilisation sur LAN) ; l'autre optimisé pour des petites largeurs de bandes (28,8/33,6 kbit/s – G.723.1 et H.263) [2].

b. Les passerelles vers des réseaux classiques (RTC, RNIS)

Les passerelles H.323 assurent l'interconnexion avec les autres réseaux, ex :(H.320/RNIS), les modems H.324, téléphones classiques …etc. Elles assurent la correspondance de signalisation de Q.931, la correspondance des signaux de contrôle et la cohésion entre les médias (multiplexage, correspondance de débits, transcodage audio) [2].

c. Gatekeeper ou les portiers

Dans la norme H.323, le Gatekeeper est le point d'entrée au réseau pour un client H.323. Il définit une zone sur le réseau, appelée zone H.323 (voir la figure3 ci-dessous), en regroupant plusieurs terminaux, Gateways et MCU dont il gère le trafic, le routage LAN et l'allocation de la bande passante. Les clients ou les Gateway s'enregistrent auprès de Gatekeeper dès l'activation de celui-ci, ce qui leur permet de retrouver n'importe qu'elle autre utilisateur à travers son identifiant fixe obtenu auprès de son Gatekeeper de rattachement [2].

Le Gatekeeper a pour fonction :

- La translation d'alias H.323 vers des adresses IP, selon les spécifications RAS (Registration/Admission/Status) ;
- Le contrôle d'accès, en interdisant les utilisateurs et les sessions non autorisés ;
- Et la gestion de la bande passante, permettant à l'administrateur du réseau de limiter le nombre de visioconférences simultanées. Concrètement, une fraction de la bande passante est allouée à la visioconférence pour ne pas gener les applications critiques sur LAN et le support des conférences multipoint adhoc.

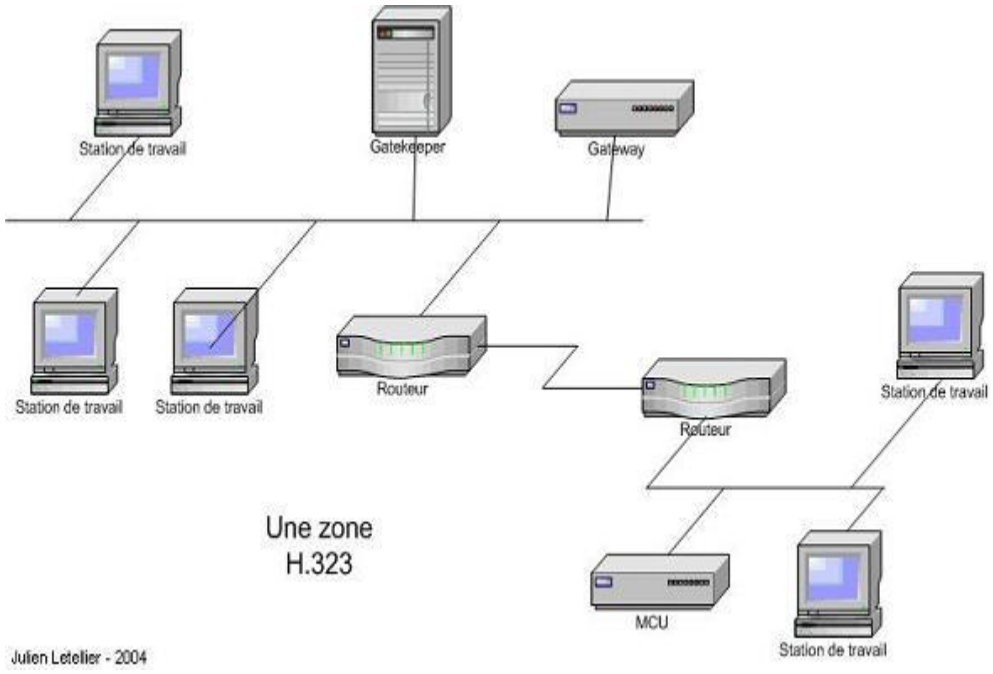

Figure 3 : La zone H.323.

d. Le MCU

Les contrôleurs multipoint ou MCU (pour Multipoint Control Unit) offrent aux utilisateurs la possibilité de faire des visioconférences à trois terminaux et plus en « à la voix ». Une MCU consiste en un Contrôleur Multipoint (MC), auquel est rajouté un ou plusieurs MP (pour Processeurs Multipoints). Le MC prend en charge les négociations H.245 entre tous les terminaux pour harmoniser les paramètres audio et vidéo de chacun. Il contrôle également les ressources utilisées. Mais le MC ne traite pas directement avec les flux audio, vidéo ou données, c'est le MP qui se charge de récupérer les flux et de leurs faire subir les traitements nécessaires. Un MC peut controler plusieurs MP distribués sur le réseau et faisant partie d'autre MCU [2].

2.1.3. Avantages et inconvénients de la technologie H.323

La technologie H.323 possède des avantages et des inconvénients. Parmi les avantages nous citons [5] :

- **Support Multipoint :** H.323 permet de faire des conférences multipoint via une structure centralisée de type MCU ou en mode adhoc.

- **Gestion de la bande passante :** le protocole H.323 permet une bonne gestion de la bande passante en posant des limites au flux audio/vidéo afin d'assurer le bon fonctionnement des applications critiques sur le LAN. Chaque terminal H.323 peut procéder à l'ajustement de la bande passante et la modification du débit en fonction du comportement du réseau en temps réel (latence, perte de paquets et gigue).
- **Support Multicast :** H.323 permet également de faire des transmissions en multicast.
- **Interopérabilité :** H.323 permet aux utilisateurs de ne pas réoccuper de la manière dont se font les communications. Les paramètres (les codecs, le débit…) sont négociés de manières transparentes.
- **Flexibilité :** une conférence H.323 peut inclure des terminaux hétérogènes (studio de visioconférence, PC, téléphone, etc) qui peuvent partager selon le cas de la voix de la vidéo et mêmes des données grâce aux spécifications T.120.

Les inconvénients de la technologie H.323 sont [5] :

- La complexité de mise en œuvre et les problèmes d'architecture en ce qui concerne la convergence des services de téléphonie et d'Internet, ainsi qu'un manque de modularité et de souplesse ;
- Comprend de nombreuses options susceptibles d'être implémentées de façons différentes par les constructeurs et donc de poser des problèmes d'interopérabilité.

2.2. Le protocole SIP

2.2.1. Description générale du protocole SIP

Le protocole SIP est un protocole normalisé et standardisé par l'IETF (décrit par RFC 3261 qui rend obsolète le RFC 2543, le complété par RFC 3265) qui a été conçu pour établir, modifier et terminer des sessions multimédia. Il se charge de l'authentification et de la localisation des multiples participants. Il se charge également de la négociation sur les types des médias utilisables par les différents participants en encapsulant des messages SDP (pour Session Description Protocol). SIP ne transporte pas les données échangées durant la session comme la voix ou la vidéo. SIP étant indépendant de la transmission des données, tout type de données et de protocoles peut être utilisé pour cet échange. Cependant, le protocole RTP

(RTP : Real-time Transport Protocol) assure le plus souvent les sessions audios et vidéo. SIP remplace progressivement H.323 [5].

Le SIP est le standard ouvert de VoIP, interopérable, le plus étendu et vise à devenir le standard des télécommunications multimédia (son, image …). Par exemple, le Skype qui utilise un format propriétaire, ne permet pas l'interopérabilité avec un autre réseau de VoIP et ne fournit que des passerelles payantes vers la téléphonie standard. SIP n'est donc pas seulement destiné à la VoIP, mais pour de nombreuses autres applications, telles que la visiophonie ; la messagerie instantanée ; la réalité virtuelle ou même les jeux vidéo [5].

2.2.2. Principe de fonctionnement

Puisque on choisira le protocole SIP pour effectuer notre travail, on s'approfondira à expliquer les différents aspects, caractéristiques qui font le protocole SIP un bon choix pour l'établissement de la session. Les principales caractéristiques du protocole SIP sont [5] :

a. Fixation d'un compte SIP : il est important de s'assurer que la personne appelée soit toujours joignable. Pour cela, un compte SIP sera associé à un nom unique. Par exemple, si un utilisateur d'un service de VoIP dispose d'un compte SIP et que chaque fois qu'il redémarre son ordinateur, son adresse IP change, il doit cependant toujours être joignable. Son compte SIP doit donc être associé à un serveur SIP (Proxy SIP) dont l'adresse IP est fixe. Ce serveur lui allouera un compte et il permettra d'effectuer ou de recevoir des appels quelques soit son emplacement. Ce compte sera identifiable via son nom (ou pseudo).

b. Changement des caractéristiques durant une session : un utilisateur doit pouvoir modifier les caractéristiques d'un appel en cours. Par exemple, un appel initialement configuré en "voix uniquement " peut être modifié en "voix+vidéo".

c. Différents modes de communication : avec SIP, les utilisateurs qui ouvrent une session peuvent communiquer en mode point à point, en mode diffusif ou dans un mode combinant ceux-ci [5] :

- **Mode point à point :** on parle dans ce cas, de l'unicast qui correspond à la communication entre deux machines ;
- **Mode diffusif :** on parle dans ce cas-là de multicast (plusieurs utilisateurs via une unité de contrôle MCU ;

- **Combinatoire :** combine les deux modes précédents. Plusieurs utilisateurs interconnectés en multicast via un réseau à maillage complet de connexion.

d. Gestion des participants : durant une session d'appel, de nouveaux participants peuvent joindre les participants d'une session déjà ouverte en participant directement, en étant transférés ou en étant mise en attente (cette particularité rejoint les fonctionnalités d'un PABX par exemple, ou l'appelant peut être transféré vers un numéro donné ou être mise en attente) [5].

e. Négociation des médias supportés : il permet à un groupe durant un appel de négocier sur les types de médias supportés. Par exemple, la vidéo peut être ou ne pas être supportée lors d'une session [5].

f. Modèle d'échange : le protocole SIP repose sur un modelé Requête/Réponse. Les échanges entre un terminal appelant et un terminal appelé se font par l'intermédiaire de requête. La liste des requêtes échangées est la suivante [5] :

- **Invite :** cette requête indique que l'application (ou l'utilisateur) correspondante a l'url SIP spécifié est invité à participer à une session. Le corps du message décrit cette session (par exemple, média supporté par l'applant). En cas de réponse favorable, l'invité doit spécifier les médias qu'il supporte ;
- **Ack :** cette requête permet de confirmer que le terminal appelant à bien reçu une réponse définitive à une requête Invite ;
- **Options :** un proxy server en mesure de contacter l'UAS (pour User Agent Server) appelé, doit répondre à une requête Options en précisant ses capacités à contacter le même terminal ;
- **Bye :** cette requête est utilisée par le terminal de l'applé afin de signaler qu'il souhaite mettre un terme à la session ;
- **Cancel :** cette requête est envoyée par un terminal ou par un proxy server afin d'annuler une requête non validée par une réponse finale, comme par exemple, si une machine ayant été invitée à participer à une session, et ayant accepté l'invitation ne reçoit pas de requête Ack, alors elle émet une requête Cancel ;
- **Registre :** cette méthode est utilisée par le client pour enregistrer l'adresse listée dans l'URL (pour Uniforme Resource Locator) par le serveur auquel il est relié.

g. Code d'erreur : une réponse à une requête est caractérisée par un code et un motif, appelés respectivement code d'état et raison phrase. Un code d'état est un entier codé sur 3 digits indiquant un résultat à l'issue de la réception d'une requête. Ce résultat est précisé par une phrase, textbased (UTF-8), expliquant le motif de refus ou de l'acceptation de la requête. Le code d'état est donc destiné à l'automate gérant l'établissement des sessions SIP et les motifs aux programmeurs. Il existe 6 classes de réponses et donc il s'agit de codes d'état, représentées par le premier digit [5] :

- 1xx=Information - La requête a été reçue et continue à être traitée ;
- 2xx=Succès - L'action a été reçue avec succès, comprise et acceptée ;
- 3xx=Redirection - Une autre action doit être menée afin de valider la requête ;
- 4xx=Erreur du client - La requête contient une syntaxe erronée ou ne peut pas être traitée par ce serveur ;
- 5xx=Erreur du serveur - Le serveur n'a pas réussie à traiter une requête apparemment correcte ;
- 6xx=Echec général-La requête ne peut pas être traitée par aucun serveur.

2.2.3. Rôle des composants

Dans un système SIP on trouve deux types de composantes, les agents utilisateurs (UAS, UAC) et un réseau de serveurs (Registrar, Proxy).

L'UAS (pour User Agent Server) : représente l'agent de la partie appelée. C'est une application de type serveur qui contacte l'utilisateur lorsqu'une requête SIP est reçue. Et elle renvoie une réponse au nom de l'utilisateur [5].

L'UAC (pour User Agent Client) : représente l'agent de la partie appelante. C'est une application de type client qui intile les requêtes [5].

Le Registrar : est un serveur qui gère les requêtes REGISTER envoyées par les Users Agents pour signaler leur emplacement courant. Ces requêtes contiennent donc une adresse IP, associée à une URI, qui seront stockées dans une base de données (voir Figure 4) [5].

Les URI SIP : sont très similaires dans leur forme, à des adresses e-mail : sip : utilisateur@domaine.com. Généralement, des mécanismes d'authentification qui permettent d'éviter que quiconque puisse ce sont enregistrer avec n'importe quelle URI [5].

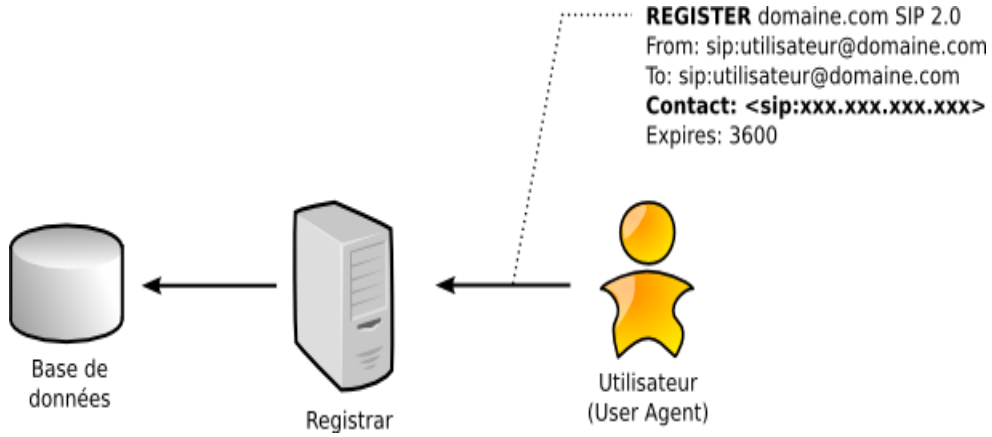

Figure 4 : Enregistrement d'un utilisateur.

Les User Agents désignent les agents que l'on retrouve dans les téléphones SIP, les SoftPhones (logiciels de téléphonie sur IP) des ordinateurs et PDA ou les passerelles SIP. En théorie, on peut établir des sessions directement entre deux User Agents, deux téléphones par exemple. Mais cela nécessite de connaître l'adresse IP du destinataire. Cela n'est pas l'idéal car une adresse IP peut ne pas être publique (derrière un NAT) ou changer et elle est bien plus compliquée à retenir qu'une URI (pour Uniform Resource Identifier). Les User Agents peuvent donc s'enregistrer auprès de Registrars pour signaler leur emplacement courant, c'est-à-dire leur adresse IP.

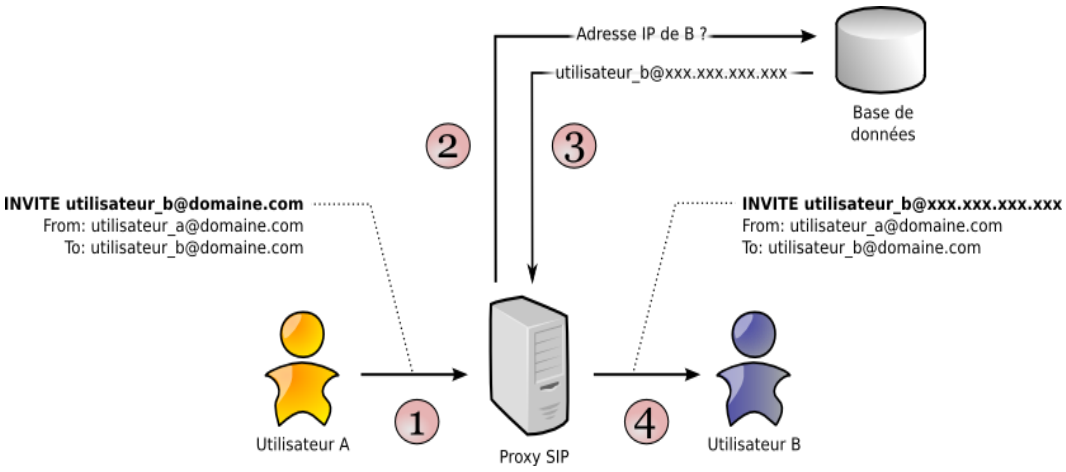

Figure 5 : Principe du protocole SIP.

1. Envoi d'une requête INVITE au proxy ;
2. Le proxy interroge la base de données ;
3. La base de données renvoie l'adresse IP du destinataire ;
4. Le proxy relaie le message au destinataire.

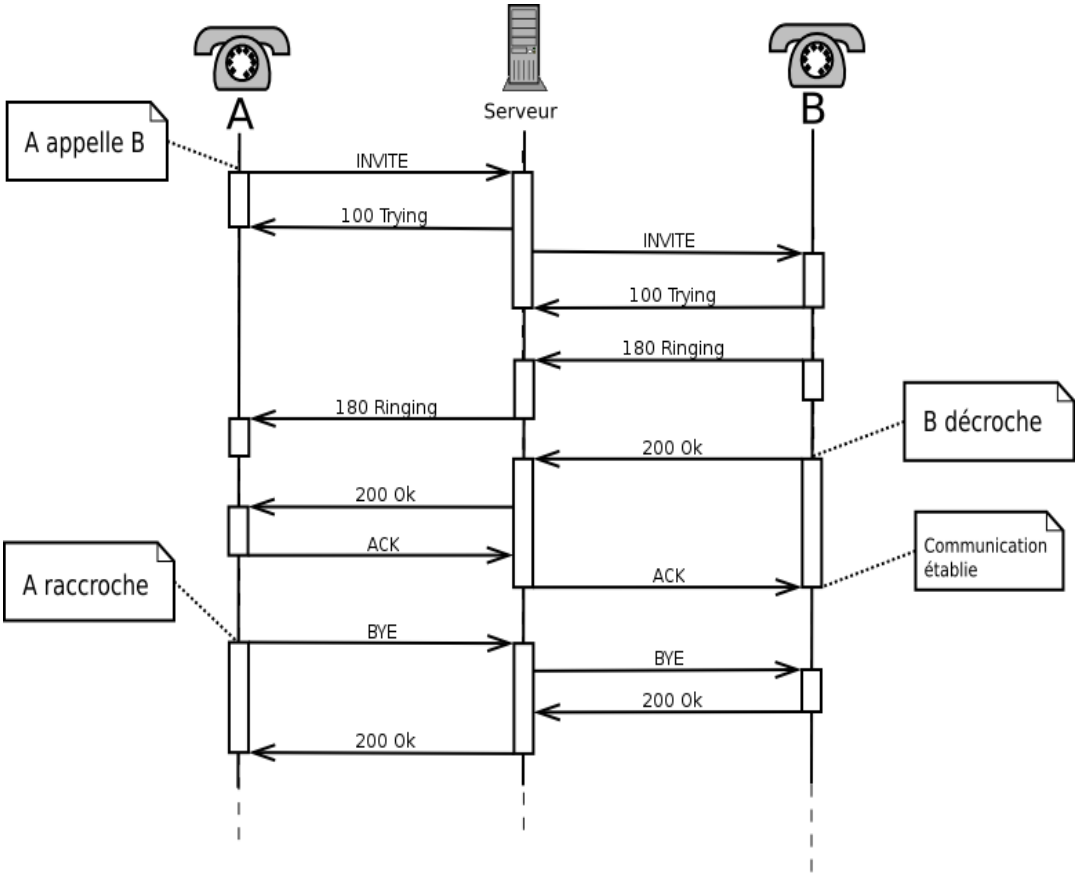

Figure 6 : Session SIP à travers un Proxy.

Le Proxy se contente de relier uniquement des messages SIP pour établir, contrôler et terminer la session (voir Figure6). Une fois la session établie, les données, par exemple, un flux RTP pour la VoIP, ne transitent pas par le serveur Proxy. Elles sont échangées directement entre les User Agents [5].

2.2.4. Avantages et inconvénients du protocole SIP

Ouvert, standard, simple et flexible sont les principaux atouts du protocole SIP. Voilà en détails ces différents avantages [5] :

- **Ouvert** : les protocoles et documents officiels sont détaillés et accessibles à tous en téléchargement ;
- **Standard** : LIETF (pour Internet Engineering Task Force) à normalisé le protocole et son évolution continue par la création ou l'évolution d'autres protocoles qui fonctionnent avec SIP ;
- **Simple** : SIP est simple et très similaire à http ;
- **Flexible** : SIP est également utilisé pour tout type de sessions multimédia (voix, vidéo, mais aussi musique, réalité virtuelle…) ;
- **Téléphonie sur réseaux publics** : il existe de nombreuses passerelles (services payants) vers le réseau public de téléphonie (RTC, GSM…) permettant d'émettre ou de recevoir des appels vocaux ;
- **Points communs avec H.323** : l'utilisateur du protocole RTP et quelques codecs sons et vidéos sont en commun.

Par contre une mauvaise implémentation ou une implémentation incomplète du protocole SIP dans les User Agents peut perturber le fonctionnement ou générer du trafic superflu sur le réseau. Un autre inconvénient est le faible nombre d'utilisateurs : SIP est encore peu connu est utilisé par le grand public, n'ayant pas atteint une masse critique, il ne bénéficie pas de l'effet réseau.

2. 3. La comparaison des deux protocoles H.323 et SIP :

Voici les avantages du protocole H.323 :

- Il existe de nombreux produits (plus de 30) utilisant ce standard adopté par de grandes entreprises telles Cisco, IBM, Intel, Microsoft, Netscape, etc ;
- Les cinq principaux logiciels de visioconférence Picturel 550, Proshare 500, Trinicon 500, Smartstation et Cruiser 150 utilisent sur IP la norme H.323 ;
- Un niveau d'interopérabilité très élevé, ce qui permet à plusieurs utilisateurs d'échanger des données audio et vidéo sans faire attention aux types de média qu'ils utilisent.

Voici les avantages du protocole SIP :

- SIP est un protocole plus rapide. La séparation entre ses champs d'entête et son corps du message facilite le traitement des messages et diminue leur temps de transition dans le réseau ;

- Nombre des entêtes est limité (36 au maximum et en pratique, moins d'une dizaine d'entêtes sont utilisées simultanément), ce qui allège l'écriture et la lecture des requêtes et réponses ;
- Sip est un protocole indépendant de la couche transport. Il peut aussi bien s'utiliser avec TCP que UDP ;
- De plus, il sépare les flux de données de ceux la signalisation, ce qui rend plus souple l'évolution « en direct » d'une communication (arrivée d'un nouveau participant, changement de paramètres…).

Voici la comparaison entre le protocole SIP et H323 :

- Nombre échanges pour établir la connexion

  SIP : 1,5 aller-retour

  H.323 : 6 à 7 aller-retour ;
- Maintenance du code protocolaire

  SIP : Simple par sa nature textuelle à l'exemple de http

  H.323 : Complexe et nécessitant un compilateur ;
- Evolution du protocole

  SIP : Protocole ouvert à de nouvelles fonctions

  H.323 : Ajout d'extensions propriétaires sans concentration entre vendeurs ;
- Fonction de conférence

  SIP : Distribuée

  H.323 : Centralisée par l'unité MC (pour Microphone Controller) ;
- Fonction de téléservices

  SIP : Oui, par défault

  H.323 : H.323 v2 + H.450 ;
- Détection d'un appel en boucle

  SIP : Oui

  H.323 : Inexistante sur la version 1

  Un appel routé sur l'appelant provoque une infinité de requêtes ;
- Signalisation multicast

  SIP : Oui, par défault

  H.323 : Non.

La simplicité, la rapidité et la légèreté d'utilisation, tout en étant très complet, du protocole Sip sont autant d'arguments qui pourraient permettre à Sip de convaincre les investisseurs. De

plus, ses avancées en matière de sécurité des messages sont un atout important par rapport à ses concurrents [7].

3. Protocoles de transport

Nous décrivons deux autres protocoles de transport utilisés dans la voix sur IP à savoir RTP et RTCP (pour Real-time Transport Control Protocol).

3.1. Le protocole RTP

3.1.1. Description générale de RTP

RTP (pour Real time Transport Protocole), standardisé en 1996, est un protocole qui a été développée par l'IETF afin de faciliter le transport temps réel de bout en bout des flots de données audio et vidéo sur les réseaux IP. C'est-à-dire sur les réseaux des paquets. RTP est un protocole qui se situe au niveau de l'application et qui utilise les protocoles sous-jacents de transport TCP (pour Transmission Control Protocol) ou UDP (pour User Datagramme Protocol). Mais l'utilisateur de RTP se fait généralement au-dessus d'UDP, ce qui permet d'atteindre plus facilement le temps réel. Les applications temps réels comme la parole numérique où la visioconférence constitue un véritable problème pour internet. Qui dit application temps réel, dit présence d'une certaine QoS que RTP ne garantie pas du fait qu'il fonctionne au niveau applicatif. De plus RTP est un protocole qui se trouve dans un environnement multipoint, donc on peut dire que RTP possède à sa charge, la gestion du temps réel, mais aussi l'administration de la session multipoint [5].

3.1.2. Les fonctions de RTP

Le protocole RTP a pour but d'organiser les paquets à l'entrée du réseau et de les contrôler à la sortie. Ceci de façon à réformer les flux avec ses caractéristiques de départ. RTP est un protocole de bout en bout, volontairement incomplet et malléable pour s'adapter aux besoins des applications. Il sera intégré dans le noyau de l'application. Il laisse la responsabilité du contrôle aux équipements d'extrémité. Il est aussi un protocole adapté aux applications présentant des propriétés temps réel. Il permet ainsi de [5] :

- Mettre en place un séquencement des paquets par une numérotation et ce afin de permettre ainsi la détection des paquets perdus. Ceci est un point primordial dans la reconstitution des données. Mais il faut savoir quand même que la perte d'un paquet n'est pas

un gros problème si les paquets ne sont pas perdus en trop grands nombres. Cependant, il est très important de savoir quel est le paquet qui a été perdu afin de pouvoir pallier à cette perte ;

- Identifier le contenu des données pour leur associer un transport sécurisé et reconstituer la base de temps des flux (horodatage des paquets : Possibilité de resynchronisation des flux par le récepteur) ;
- L'identification de la source c'est-à-dire l'identification de l'expéditeur du paquet. Dans un multicast l'identité de la source doit être connue et déterminée ;
- Transporter les applications audio et vidéo dans des trames (avec des dimensions qui sont dépendantes des codecs qui effectuent la numérisation). Ces trames sont incluses dans des paquets afin d'être transportées et doivent, de ce fait, être récupérées facilement au moment de la phase de segmentation des paquets afin que l'application soit décodée correctement.

3.1.3. Avantages et inconvénients du protocole RTP

Le protocole RTP permet de reconstituer la base de temps des différents flux multimédia (audio, vidéo, …), de détecter les pertes de paquets, et d'identifier le contenu des paquets pour leur transmission sécurisée [5].

Par contre, il ne permet pas de réserver des ressources dans le réseau ou d'apporter une fiabilité dans le réseau. Ainsi, il ne garantit pas le délai de livraison [5].

3.2. Le protocole RTCP

3.2.1. Description générale de RTCP

Le protocole RTCP est fondé sur la transmission périodique de paquêts de contrôle à tous les participants d'une session. C'est le protocole UDP (par exemple) qui permet le multiplexage des paquets de données RTP et des paquets de contrôle RTCP. Le protocole RTP utilise le protocole RTCP, qui transporte les informations supplémentaires suivantes pour la gestion de la session :

- Les récepteurs utilisent RTCP pour envoyer vers les émetteurs un rapport sur la QoS. Ces rapports comprennent le nombre de paquets perdus, le paramètre indiquant la variance d'une distribution (plus communément appelée " la gigue" : c'est-à-dire les paquets qui arrivent régulièrement ou irrégulièrement) et le délai aller-retour. Ces informations permettent

à la source de s'adapter, par exemple, de modifier le niveau de compression pour maintenir une QoS ;

- **Une synchronisation supplémentaire entre les médias :** Les applications multimédias sont souvent transportées par des flots distincts. Par exemple, la voix, l'image ou même des applications numérisées sur plusieurs niveaux hiérarchiques peuvent voir les flots gérés et suivre des chemins différents ;[5]
- **L'identification des participants à une session :** En effet le protocole RTCP permet aux participants d'indiquer leus départ d'une conférence téléphonique (paquet Bye de RTCP) ou simplement de fournir une indication sur leur comportement.

Le protocole RTCP demande aux participants de la session d'envoyer périodiquement les informations citées ci-dessus. La périodicité est calculée en fonction du nombre de participants de l'application. On peut dire que les paquets RTCP ne transportent que les données des utilisateurs. Tandis que les paquets RTCP ne transportent en temps réel, que de la supervision.

On peut détailler les paquets de supervision en 5 types [5] :

- **200-SR (Sender Report)** : ce rapport regroupe des statistiques concernant la transmission (pourcentage de perte, nombre cumulé de paquets perdus, variation de délai (gigue, etc). Ces rapports sont issus d'émetteurs actifs d'une session ;
- **201-RR (Receiver Report)** : ensemble de statistiques portant sur la communication entre les participants. Ces rapports sont issus de récepteurs d'une session ;
- **202-SDES (Source Description)** : carte de visite de la source (nom, e-mail, localisation) ;
- **203-BYE** : message de fin de participation à une session ;
- **204-APP** : fonctions spécifiques à une application.

Le protocole RTCP est adapté pour la transmission de données temps réel. Il permet d'effectuer un contrôle permanant sur une session et ces participants. Par contre il fonctionne en stratégie bout à bout. Et il ne peut pas contrôler l'élément principal de la communication " le réseau ".

3.2.2. Points forts et limites de la voix sur IP

Différentes sont les raisons qui peuvent pousser les entreprises à s'orienter vers la VoIP comme solution pour la téléphonie. Les avantages les plus marqués sont [5] :

- **Réduction des couts :** en effet le trafic véhiculé à travers le réseau RTP est plus couteux que sur un réseau IP. Réductions importantes pour des communications internationales en utilisant le VoIP ; et ces réductions deviennent encore plus intéressantes dans la mutualisation voix/données du réseau IP intersites. Dans ce dernier cas, le gain est directement proportionnel au nombre de sites distants ;
- **Standars ouvets :** la VoIP n'est plus uniquement H.323, mais un usage multi-protocole selon les besoins de services nécessaires. Par exemple, le protocole H.323 fonctionne en mode égale à égale alors que MGCP fonctionne en mode centralisé. Ces différences de conception offrent immédiatement une différence dans l'exploitation des terminaisons considérées ;
- **Un réseau voix, vidéo et données (à la fois) :** grâce à l'intégration de la voix comme une application supplémentaire dans un réseau IP, ce dernier va simplifier la gestion des trois applications (voix, réseau et vidéo) par un seul transport IP. Une simplification de gestion, mais également une mutualisation des efforts financiers vers un seul outil ;
- **Un service PABX distribué ou centralisé :** le PABX en réseau bénéficient de services centralisés tel que la messagerie vocale et la taxation. Cette même centralisation continue à être assurée sur un réseau VoIP sans limitation du nombre de canaux. Il convient pour en assurer une bonne utilisation de dimensionner convenablement le lien réseau. L'utilisation de la VoIP met en commun un média qui peut à la fois offrir à un moment précis une bande passante maximum à la donnée, et dans une autre période une bande passante maximum à la voix, garantissant toujours la priorité à celle-ci ;

Les points faibles de la voix sur IP sont [5] :

- **Fiabilité et qualité sonore :** un des problèmes les plus importants de la téléphonie sur IP est la qualité de la retransmission qui n'est pas encore optimale. En effet, des désagréments telle la qualité de la reproduction de la voix du correspondant ainsi que le délai entre le moment ou l'un des interlocuteurs parle et le moment ou l'autre entend peuvent être extrêmement problématiques. De plus, il se peut que des morceaux de la conversation manquent (des paquets perdus pendant le transfert) sans être en mesure de savoir si des paquets ont été perdus et à quel moment ;
- **Dépendance de l'infrastructure technologique et support administratif exigeant :** les centres de relations IP peuvent être particulièrement vulnérables en cas d'improductivité de l'infrastructure. Par exemple, si la base de données n'est pas disponible, les centres ne peuvent tout simplement pas recevoir d'appels. La convergence de la voix et des données

dans un seul système signifie que la stabilité du système devient plus importante que jamais et l'organisation doit être préparée à travailler avec efficience ou à encourir les conséquences ;

- **Vol :** les attaquants qui parviennent à accéder à un serveur VoIP peuvent également accéder aux messages vocaux stockés et au même temps au service téléphonique pour écouter des conversations ou effectuer des appels gratuits aux noms d'autres comptes ;
- **Attaque de virus :** si un serveur VoIP est infecté par un virus, les utilisateurs risquent de ne plus pouvoir accéder au réseau téléphonique. Le virus peut également infecter d'autres ordinateurs connectés au système.

Conclusion

Comme on a pu le voir tout au long de ce chapitre, la VoIP est la solution la plus rentable pour effectuer des conversations et des communications téléphoniques en entreprise. Actuellement, il est évident que la VoIP va continuer à évoluer.

La téléphonie IP est une bonne solution en matière d'intégration, fiabilité et de coût. On a vu que la voix sur IP étant une nouvelle technologie de communucation, elle n'a pas encore de standard unique. Chaque standard possède ses propres caractéristiques pour garantir une bonne qualité de service. En effet, le respect des contraintes temporelles est le facteur le plus important lors de transport de la voix.

Malgré que la normalisation n'ait pas atteint la maturité suffisante pour sa généralisation au niveau des réseaux IP, il n'est pas dangereux de miser sur ces standards vu qu'ils ont été acceptés par l'ensemble de la communauté de la téléphonie.

Pour finir, lors de la mise en œuvre de cette technologie, il faut poser la question suivante : le développement de cette technologie représentent'il un risque ou une opportunité pour les utilisateurs et les opérateurs téléphoniques ?

CHAPITRE 2
Etude des differentes attaques et des vulnérabilités contre la VoIP, et mécanismes de sécurité

Chapitre 2 : Etude des différentes attaques et des Vulnérabilité contre VoIP, et mécanismes de sécurité

Introduction

L'opportunité de migrer de la téléphonie classique vers la téléphonie IP, a offert plusieurs avantages pour les entreprises, et les a permis de bénéficier de nouveaux services tel que la vidéoconférence et la transmission des données. L'intégration de ces services dans une seule plateforme nécessite plus de sécurité.

Dans ce chapitre, nous dériverons des attaques qui menacent la VoIP, et nous détaillerons quelques-uns. Nous finirons par une description des bonnes pratiques pour sécuriser les communications de type voix sur IP.

Le système VoIP utilise l'Internet, et particulièrement le protocole IP. De ce fait, les vulnérabilités de celui-ci.

Les attaques sur les réseaux VoIP peuvent être classées en deux types : les attaques internes et les attaques externes. Les attaques externes sont lancées par des personnes autres que celles qui participent à l'appel, et ils se produisent généralement quand les paquets VoIP traversent un réseau peu fiable et/ou l'appel passe par un réseau tiers durant le transfert des paquets. Les attaques internes s'effectuent directement du réseau local dans lequel se trouve l'attaquant.

Il existe deux principales classes de vulnérabilités sur un environnement VoIP. La première dépend des protocoles utilisés (SIP, H.323…) ; et la deuxième est reliée aux systèmes sur lesquels les éléments VoIP sont implémentés. Chaque protocole ou service a ses propres vulnérabilités.

1. Attaques sur les protocoles

Un appel téléphonique VoIP est constitué de deux parties : la signalisation, qui instaure l'appel ; et les flux de media, qui transportent la voix.

La signalisation, en particulier SIP, transmet les entêtes et la charge utile (Payload) du paquet en texte clair, ce qui permet à un attaquant de lire et falsifier facilement les paquets. Elle est donc vulnérable aux attaques qui essaient de voler ou perturber le service téléphonique, et à l'écoute clandestine qui recherche des informations sur un compte utilisateur valide, pour passer des appels gratuits par exemple. La signalisation utilise, en général, le port par défaut UDP/TCP 5060. Le firewall doit être capable d'inspecter les paquets de signalisation et ouvre ce port afin de leurs autoriser l'accès au réseau. Un firewall,

qui n'est pas compatible aux protocoles de la VoIP, doit être configuré manuellement pour laisser le port 5060 ouvert, créant un trou pour des attaques contre les éléments qui écoutent l'activité sur ce port [5].

Le protocole RTP, utilisé pour le transport des flux multimédia, présente également plusieurs vulnérabilités dues à l'absence d'authentification et de chiffrage. Chaque entête d'un paquet RTP contient un numéro de séquence qui permet au destinataire de reconstituer les paquets de la voix dans l'ordre approprié.

Cependant, un attaquant peut facilement injecter des paquets artificiels avec un numéro de séquence plus élevé. En conséquence, ces paquets seront diffusés à la place des vrais paquets.

Généralement, les flux multimédias contournent les serveurs proxy et circulent directement entre les points finaux. Les menaces habituelles conte le flux de la voix sont l'interruption de transport et l'écoute clandestine.

Les protocoles de la VoIP utilisent TCP et UDP comme moyens de transport, et par conséquent sont aussi vulnérables à toutes les attaques contre ces protocoles, tel que le détournement de session Hijacking et la mystification Spoofing, etc. Les types d'attaques les plus fréquentes contre un system VoIP sont :

1.1. Sniffing

Le sniffing (renfilage) peut avoir comme conséquence un vol d'identité et la révélation d'informations confidentielles. Il permet également aux utilisateurs malveillants perfectionnés de rassembler des informations sur les systèmes VoIP. Ces informations peuvent par exemple être employées pour mettre en place une attaque contre d'autres systèmes ou données.

Plusieurs outils requis pour le sniffing, y compris pour le protocole H.323 et des plugins SIP, sont disponibles en open source [3].

1.2. Suivie des appels

Appelé aussi Call tracking ; cette attaque se fait au niveau du réseau LAN/VPN et cible les terminaux (soft/hard phone). Elle a pour but de connaître qui est en train de communiquer et quelle est la période de la communication. L'attaquant doit récupérer les messages INVITE et BYE en écoutant le réseau et peut ainsi savoir qui communique, à quelle heure, et pendant combien de temps.

Pour réaliser cette attaque, L'attaquant doit être capable d'écouter le réseau et récupérer les messages INVITE et BYE [3].

1.3. Injection de paquet RTP

Cette attaque se fait au niveau du réseau LAN ou VPN (pour Virtual Private Network). Elle cible le serveur registrar, et a pour but de perturber une communication en cours [3].

L'attaquant devra tout d'abord écouter un flux RTP de l'appelant vers l'appelé, analyser son contenu et générer un paquet RTP contenant un en-tête similaire mais avec un plus grand numéro de séquence et timestamp afin que ce paquet soit reproduit avant les autres paquets (s'ils sont vraiment reproduits). Ainsi, la communication sera perturbée et l'appel ne pourra pas se dérouler correctement [5].

Pour réaliser cette attaque, l'attaquant doit être capable d'écouter le réseau afin de repérer une communication et ainsi repérer les timestamps des paquets RTP.

Il doit aussi être capable d'insérer des messages RTP qu'il a généré ayant un timestamp modifié.

1.4. Les Spam

Trois formes principales de spams sont jusqu'à maintenant identifiés dans SIP [3] :

- **Call** : ce type de spam est défini comme une masse de tentatives d'initiation de session (des requêtes INVITE) non sollicitées. Généralement c'est un UAC qui lance, en parallèle, un grand nombre d'appels. Si l'appel est établi, l'application génère un ACK, rejoue une annonce préenregistrée, et ensuite termine l'appel ;
- **IM** (Instant Message) **:** ce type de spam est semblable à celui de l'e-mail. Il est défini comme une masse de messages instantanés non sollicitées. Les IM spams sont, pour la plupart, envoyés sous forme de requête SIP. Ça pourrait être des requêtes INVITE avec un entête « Subject » très grand, ou des requêtes INVITE avec un corps en format texte ou HTML (pour HyperText Markup Language).

Bien-sûr, l'IM spam est beaucoup plus intrusif que le spam email. Car dans les systèmes actuels, les IMs apparaissent automatiquement sous forme de pop-up à l'utilisateur ;

- **Presence :** ce type de spam est semblable à l'IM spam. Il est défini comme une masse de requêtes de présence (des requêtes subscribe) non sollicitées. L'attaquant fait ceci dans le but d'appartenir à la " white list " d'un utilisateur afin de lui envoyer des messages instantanés ou d'initier avec lui d'autres formes de communication. L'IM Spam est différent du Presence Spam dans le fait que ce dernier ne transmet pas réellement de contenu dans les messages.

1.5. Le déni de service (DoS : Denial of service)

C'est d'une manière générale, l'attaque qui vise à rendre une application informatique ou un équipement informatique incapable de répondre aux requêtes de ses utilisateurs et donc hors d'usage [5].

Une machine serveur offrant des services à ses clients (par exemple un serveur web) doit traiter des requêtes provenant de plusieurs clients. Lorsque ces derniers ne peuvent en bénéficier, pour des raisons délibérément provoquées par un tiers, il y a déni de service.

Dans une attaque de type DoS flood attack, les ressources d'un serveur ou d'un réseau sont épuisées par un flot de paquets. Un seul attaquant visant à envoyer un flot de paquets peut être identifié et isolé assez facilement. Cependant l'approche de choix pour les attaquants a évolué vers un déni de service distribué (DDoS). Une attaque DDoS repose sur une distribution d'attaques DoS, simultanément menées par plusieurs systèmes contre un seul. Cela réduit le temps nécessaire à l'attaque et amplifie ses effets. Dans ce type d'attaque, les pirates se dissimulent parfois grâce à des machines-rebonds (ou machines zombies), utilisées à l'insu de leurs propriétaires. Un ensemble de machines-rebonds, est contrôlable par un pirate après infection de chacune d'elles par un programme de type porte dérobée (backdoor).

Une attaque de type DoS peut s'effectuer à plusieurs niveaux soit :

Couche réseau :

- **IP Flooding** : le but de l'IP Flooding est d'envoyer une multitude de paquets IP vers une même destination de telle sorte que le traitement de ces paquets empêche une entité du réseau (un routeur ou la station destinatrice) de traiter les paquets IP légitimes. Si l'IP Flooding est combiné avec l'IP Spoofing, il est impossible, pour le destinataire, de connaître l'adresse source exacte des paquets IP. De ce fait, sauf que le destinataire ne limite ses échanges avec certaines stations, il lui est impossible de contrer ce type d'attaques [3] ;

- **Fragmentation des paquets IP** : par la fragmentation des paquets, il est possible de rendre hors service de nombreux systèmes d'exploitation et dispositif VoIP par le biais de la consommation des ressources. Il existe de nombreuses variantes d'attaques par fragmentation, parmi les plus populaires : Teardrop, Opentear, Nestea, Jolt, Boink, et Ping of death [3].

Couche transport

L'UDP Flooding Attacks : le principe de cette attaque est qu'un attaquant envoie un grand nombre de requêtes UDP vers une machine. Le trafic UDP étant prioritaire sur le trafic TCP, ce type d'attaque qui peut vite troubler et saturer le trafic transitant sur le réseau et donc de perturbe davantage la bande passante. Presque tous les dispositifs utilisant le protocole SIP fonctionnent au-dessus du protocole UDP, ce qui en fait d'elles des cibles. De nombreux dispositifs de VoIP et de systèmes d'exploitation peuvent être paralysés grâce à des paquets UDP Flooding visant l'écoute du port SIP (5060) ou d'autres ports [3].

TCP SYN floods est une attaque visant le protocole TCP, et plus exactement la phase d'établissement de connexion. Celle-ci consiste en trois sous-étapes :

- Le client envoie un paquet SYN au serveur ;
- Le serveur répond avec un paquet SYN-ACK ;
- Le client envoie un paquet ACK au serveur.

L'attaque consiste à l'envoie d'un grand nombre de paquets SYN. La victime va alors répondre par un message SYN-ACK d'acquittement. Pour terminer la connexion TCP, la victime ensuite va attendre pendant une période de temps la réponse par le biais d'un paquet ACK. C'est en cela que réside la source de l'attaque, parce que les ACK finaux ne sont jamais envoyés, et par la suite, la mémoire système se remplit rapidement et consomme toutes les ressources disponibles à ces demandes non valides. Le résultat final est que le serveur, le téléphone, ou le routeur ne sera pas en mesure de faire la distinction entre les faux SYN et les SYN légitimes d'une réelle connexion VoIP [3].

Couche application

SIP Flooding : dans le cas de SIP, une attaque DoS peut être directement dirigée contre les utilisateurs finaux ou les dispositifs tels que téléphones IP, routeurs et proxy SIP, ou contre les serveurs concernés par le processus, en utilisant le mécanisme du protocole SIP ou d'autres techniques traditionnelles de DoS.

Voyons maintenant en détail les différentes formes d'attaque DoS :

- **Attaque par la méthode du CANCEL**

C'est un type de déni de service, lancé contre l'utilisateur. L'attaquant surveille l'activité du proxy SIP et attend qu'un appel arrive pour un utilisateur spécifique. Une fois que le dispositif de l'utilisateur reçoit la requête INVITE, l'attaquant envoie immédiatement une requête CANCEL. Cette requête produit une erreur sur le dispositif de l'appelé et termine l'appel. Ce type d'attaque est employé pour interrompre la communication [5] ;

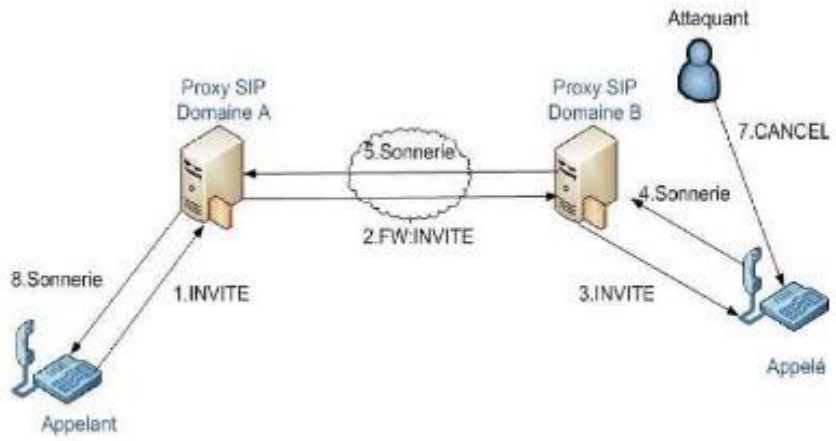

Figure 7 : Attaque DoS via une requête CANCEL.

La figure suivante montre un scénario d'attaque DoS CANCEL, où l'utilisateur toto initie l'appel, envoie une invitation (1) au proxy auquel il est rattaché. Le proxy du domaine A achemine la requête (2) au proxy qui est responsable de l'utilisateur titi. Ensuite, c'est le proxy du domaine B qui prend le relais et achemine la requête INVITE (3), qui arrive enfin à destination. Le dispositif de titi, quand il reçoit l'invitation, sonne (4). Cette information est réacheminée jusqu'au dispositif de toto. L'attaquant qui surveille l'activité du proxy SIP du domaine B envoie une requête CANCEL (7) avant que titi n'ait pu envoyer la réponse OK, qui accepte l'appel. Cette requête annulera la requête en attente (l'INVITE), l'appel n'a pas eu lieu [5] ;

- **Attaque par la méthode du BYE**

L'attaque par la méthode du BYE est dirigée contre les usagers. L'attaquant génère un BYE et interrompt une conversation. Pour réaliser cette attaque, le pirate écoute le trafic et prend les informations nécessaires (comme par exemple le Call-Id, le From ou encore le To) pour

générer un BYE frauduleux correspondant à la session qui est injectée sur le réseau. Le BYE n'étant pas authentifié, celui qui reçoit l'information l'exécute. La figure III.2 représente une attaque DoS via une requête BYE [3] ;

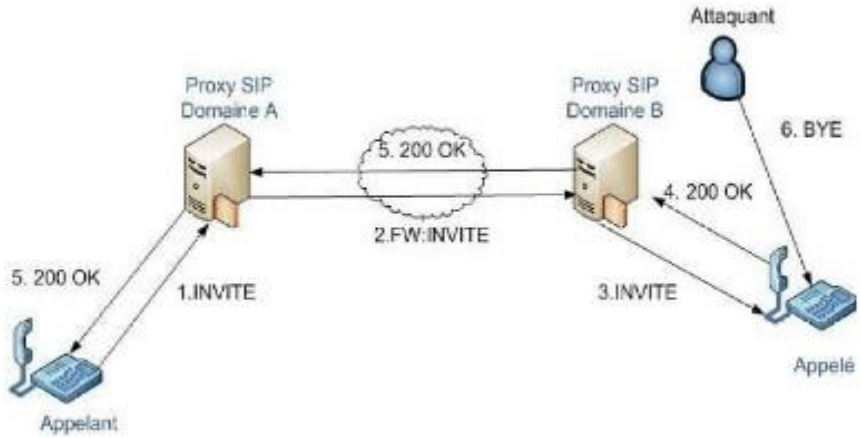

Figure 8 : Attaque DoS via une requête BYE.

- **Attaque par la méthode du REGISTRE**

Le serveur d'enregistrement lui-même est une source potentielle de déni de service pour les utilisateurs. En effet, ce serveur peut accepter des enregistrements de tous les dispositifs. Un nouvel enregistrement avec une «*» dans l'entête remplacera tous les précédents enregistrements pour ce dispositif. Les attaquants, de cette façon, peuvent supprimer l'enregistrement de quelques-uns des utilisateurs, ou tous, dans un domaine, empêchant ainsi ces utilisateurs d'être invités à de nouvelles sessions.

Notez que cette fonction de suppression d'enregistrement d'un dispositif au profit d'un autre est un comportement voulu en SIP, afin de permettre le transfert d'appel. Le dispositif de l'utilisateur doit pouvoir devenir le dispositif principal quand il vient en ligne. C'est un mécanisme très pratique pour les utilisateurs, mais également pour les pirates [3].

1.6. Détournement d'appel (Call Hijacking)

Le Call Hijacking consiste à détourner un appel. Plusieurs fournisseurs de service VoIP utilisent le web comme interface permettant à l'utilisateur d'accéder à leur système téléphonique. Un utilisateur authentifié peut changer les paramètres de ses transferts d'appel à travers cette interface web. C'est peut-être pratique, mais un utilisateur malveillant peut utiliser le même moyen pour mener une attaque.

Exemple : quand un agent SIP envoie un message INVITE pour initier un appel, L'attaquant envoie un message de redirection 3xx indiquant que l'appelé s'est déplacé et par la même occasion donne sa propre adresse comme adresse de renvoi. À partir de ce moment, tous les appels destinés à l'utilisateur sont transférés et c'est l'attaquant qui les reçoit.

Un appel détourné en lui-même est un problème, mais c'est encore plus grave quand il est porteur d'informations sensibles et confidentielles [3].

1.7. Attaque par écoute Clandistine

L'eavesdropping est l'écoute clandestine d'une conversation téléphonique. Un attaquant avec un accès au réseau VoIP peut sniffer le trafic et décoder la conversation vocale. Des outils tels que VOMIT (pour Voice Over Misconfigured Internet Telephones) permettent de réaliser cette attaque. VOMIT convertit les paquets sniffés en fichier .wav qui peut être réécouté avec n'importe quel lecteur de fichiers son [3].

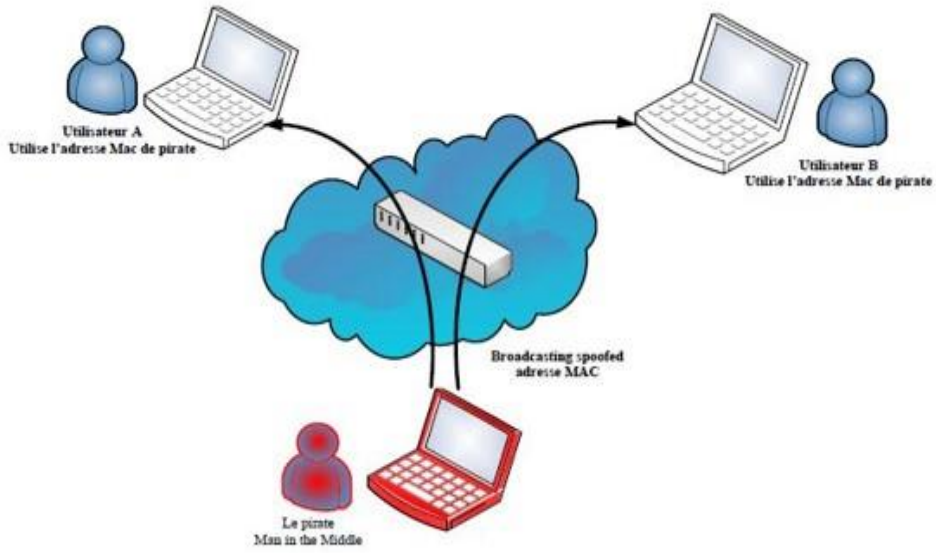

Figure 9 : Exemple de détournement d'appel « Man in the middle ».

Le principe de l'écoute clandestine est montré dans la figure 8 comme suit :

- ✓ Déterminer les adresses MAC des victimes (client-serveur) par l'attaquant ;
- ✓ Envoi d'une requête ARP non sollicitée au client, pour l'informer du changement de l'adresse MAC du serveur VoIP à l'adresse MAC ;

- ✓ Envoi d'une requête ARP non sollicitée au serveur, pour l'informer du changement de l'adresse MAC du client à l'adresse MAC ;
- ✓ Désactiver la vérification des adresses MAC sur la machine d'attaque afin que le trafic puisse circuler entre les 2 victimes.

2. Les vulnérabilités de l'infrastructure (Hardware et Software)

Une infrastructure VoIP est composée de téléphones IP, Gateway, serveurs (proxy, registre, etc.). Chaque élément, que ce soit un système embarqué ou un serveur standard tournant sur un système d'exploitation, est accessible via le réseau comme n'importe quel ordinateur. Chacun comporte un processeur qui exécute des logiciels qui peuvent être attaqués ou employés en tant que points de lancement d'une attaque plus profonde [3].

2.1. Faiblesses dans la configuration des dispositifs de la VoIP

Plusieurs dispositifs de la VoIP, dans leur configuration par défaut, peuvent avoir une variété de ports TCP et UDP ouverts. Les services fonctionnant sur ces ports peuvent être vulnérables aux attaques DoS ou buffer overflow.

Plusieurs dispositifs de la VoIP exécutent également un serveur web pour la gestion à distance, qui peut être vulnérable aux attaques buffer overflow et à la divulgation d'informations.

Si les services accessibles ne sont pas configurés avec un mot de passe, un attaquant peut acquérir un accès non autorisé à ce dispositif.

Les services SNMP (pour Simple Network Management Protocol) offerts par ces dispositifs peuvent être vulnérables aux attaques de reconnaissance ou attaques d'overflow.

Plusieurs dispositifs de la VoIP sont configurés pour télécharger périodiquement un fichier de configuration depuis un serveur par TFTP (pour Trivial File Transfer Protocol) ou d'autres mécanismes. Un attaquant peut potentiellement détourner ou mystifier cette connexion et tromper le dispositif qui va télécharger un fichier de configuration malveillant à la place du véritable fichier.

2.2. Infrastructure Hardware

2.2.1. Les téléphones IP

Un pirate peut compromettre un dispositif de téléphonie sur IP, par exemple un téléphone IP, un SoftPhone et autres programmes ou matériels clients. Généralement, il obtient les privilèges qui lui permettent de commander complètement la fonctionnalité du dispositif [3].

Compromettre un point final (téléphone IP) peut être fait à distance ou par un accès physique au dispositif. Le pirate pourrait modifier les aspects opérationnels d'un tel dispositif :

La pile du système d'exploitation peut être changée. Ainsi, la présence de l'attaquant ne sera pas remarquée.

Aussi, un firmware modifié de manière malveillante peut être téléchargé et installé. Les modifications faites à la configuration des logiciels de téléphonie IP peuvent permettre :

- Aux appels entrants d'être réorientés vers un autre point final sans que l'utilisateur soit au courant ;
- Aux appels d'être surveillés ;
- À l'information de la signalisation et/ou les paquets contenant de la voix d'être routés vers un autre dispositif et également d'être enregistrés et/ou modifiés.

De compromettre la disponibilité du point final. Par exemple, ce dernier peut rejeter automatiquement toutes les requêtes d'appel, ou encore, éliminer tout déclenchement de notification tel qu'un son, une notification visuelle à l'arrivée d'un appel. Les appels peuvent également être interrompus à l'improviste (quelques téléphones IP permettent ceci via une interface web).

D'autres conséquences possibles sont :

- Des backdoors pourraient été installés ;
- Toutes les informations concernant l'utilisateur qui sont stockées sur le dispositif pourraient été extraites.

L'acquisition d'un accès non autorisé sur un dispositif de téléphonie IP peut être le résultat d'un autre élément compromis sur le réseau IP, ou de l'information récoltée sur le réseau.

Les SoftPhones ne réagissent pas de la même façon aux attaques comparés à leur homologues téléphones IP. Ils sont plus susceptibles aux attaques dues au nombre de vecteurs inclus dans le système, à savoir les vulnérabilités du système d'exploitation, les vulnérabilités de l'application, les vulnérabilités du service, des vers, des virus, etc. En plus, le softphone

demeure sur le segment de données, est ainsi sensible aux attaques lancées contre ce segment et pas simplement contre l'hôte qui héberge l'application softphone [3].

Les téléphones IP exécutent, quant à eux, leurs propres systèmes d'exploitation avec un nombre limité de services supportés, et possèdent donc moins de vulnérabilités [3].

2.2.2. Les serveurs VoIP

Un pirate peut viser les serveurs qui fournissent le réseau de téléphonie sur IP. Compromettre une telle entité mettra généralement en péril tout le réseau de téléphonie dont le serveur fait partie.

Par exemple, si un serveur de signalisation est compromis, un attaquant peut contrôler totalement l'information de signalisation pour différents appels. Ces informations sont routées à travers le serveur compromis. Avoir le contrôle de l'information de signalisation permet à un attaquant de changer n'importe quel paramètre relatif à l'appel.

Si un serveur de téléphonie IP est installé sur un système d'exploitation, il peut être une cible pour les virus, les vers, ou n'importe quel code malveillant.

2.3. Les vulnérabilités de système d'exploitation (Infrastructure Software)

Ces vulnérabilités sont pour la plupart relatives au manque de sécurité lors de la phase initiale de développement du système d'exploitation et ne sont découvertes qu'après le lancement du produit.

Une des principales vulnérabilités des systèmes d'exploitation est le buffer overflow. Il permet à un attaquant de prendre le contrôle partiel ou complet de la machine [3].

Les dispositifs de la VoIP tels que les téléphones IP, Call Managers, Gateway et les serveurs proxy, héritent les mêmes vulnérabilités du système d'exploitation ou du firmware sur lequel ils tournent.

Il existe une centaine de vulnérabilités exploitables à distance sur Windows et même sur Linux. Un grand nombre de ces exploits sont disponibles librement et prêts à être téléchargés sur l'Internet.

Une application de la VoIP s'avère être sûre, celle-ci devient menacé si le système d'exploitation sur lequel elle tourne est compromis.

3. Sécurisation et bonnes pratiques

On a déjà vu que les vulnérabilités existent au niveau protocolaire, application et systèmes d'exploitation. Pour cela, on a découpé la sécurisation aussi en trois niveaux : Sécurisation protocolaire, sécurisation de l'application et sécurisation du système de l'exploitation.

3.1. Sécurisation protocolaire

La prévalence et la facilité de sniffer des paquets et d'autres techniques pour la capture des paquets IP sur un réseau pour la voix sur IP fait que le cryptage soit une nécessité. La sécurisation de la VoIP est à la protection des personnes qui sont interconnectées [3].

IPsec peut être utilisé pour réaliser deux objectifs. Au premier lieu, pour garantir l'identité des deux points terminaux et protéger la voix une fois que les paquets quittent l'Intranet de l'entreprise. VoIPsec (VoIP utilisant IPsec) contribue à réduire les menaces, les sniffeurs de paquets, et de nombreux types de trafic « vocal analyze ». Une fois combiné avec un pare-feu, IPsec fait en sorte que la VoIP soit plus sûr qu'une ligne téléphonique classique. Il est important de noter, toutefois, que IPsec n'est pas toujours un bon moyen pour certaines applications, et que certains protocoles doivent continuer à compter sur leurs propres dispositifs de sécurité.

3.1.1. VoIP VPN

Un VPN VoIP combine la voix sur IP et la technologie des réseaux virtuels privés pour offrir une méthode assurant la préservation de la prestation vocale. Puisque la VoIP transmet la voix numérisée en un flux de données, la solution VPN VoIP semble être la plus appropriée vue qu'elle offre le cryptage des données grâce à des mécanismes de cryptages, puisqu'elle permet d'offrir l'intégrité des paquets VoIP [3]

.

Cryptage aux points terminaux

Vu que notre objectif est d'assurer la confidentialité et l'intégrité des clients, le mode choisi est donc le mode tunnel. Ce choix est dû au fait qu'il sécurise le paquet comme un tout

(contrairement au mode transport, qui ne sécurise que le payload IP). Le mode tunnel se base sur l'encapsulation de tout le paquet IP et ajoute un nouvel entête pour l'acheminement de ce dernier. Ce mode est généralement utilisé pour les routeur-to-routeur. En plus du mode tunnel, on a choisi le protocole ESP (pour Encapsulating Security Payload), qui va à son tour assurer le cryptage des données et donc la confidentialité ; contrairement au protocole AH (pour Authentification Header), qui ne permet que l'authentification des paquets et non le cryptage.

Dans ce cas, la solution qu'on propose est ESP mode tunnel, qui sera appliquée uniquement sur les points de terminaison à la voix IP, c'est-à-dire le routeur. Ceci nous permettra donc de minimiser le nombre de machines qui seront impliquées dans le traitement engendré par la sécurité. De plus, le nombre des clés nécessaires sera réduit.

3.1.2. Secure RTP ou SRTP

SRTP est conçu pour sécuriser la multiplication à venir des échanges multimédias sur les réseaux. Il couvre les lacunes de protocoles de sécurité existants comme IPsec (IP Security), dont le mécanisme d'échanges de clés est trop lourd. Il est aussi bâti sur le protocole temps réel RTP. Il associe aussi une demi-douzaine de protocoles complémentaires. Il est donc compatible à la fois avec des protocoles d'initiation de session de voix sur IP tel que SIP, ainsi que le protocole de diffusion de contenu multimédia en temps réel RTSP (Real Time Streaming Protocol). Mais, il s'adjoint surtout les services du protocole de gestion de clé MIKEY (Multimedia Internet KEYing).

Service de sécurités offertes par SRTP

Les principaux services offerts par SRTP sont :

- Rendre confidentielles les données RTP, que ce soit l'en-tête et la charge utile ou seulement la charge utile ;
- Authentifier et vérifier l'intégrité des paquets RTP. L'émetteur calcule une empreinte du message à envoyer, puis il l'envoie avec le message même ;
- La protection contre le rejeu des paquets. Chaque récepteur tient à jour une liste de tous les indices des paquets reçus et bien authentifiés.

Principe de fonctionnement de SRTP

Avec une gestion de clé appropriée, SRTP est sécurisé pour les applications unicast et multicast de RTP. En théorie, SRTP est une extension du protocole RTP dans lequel a été rajoutée des options de sécurité. En effet, il a pour but d'offrir plusieurs implémentations de cryptographie, tout en limitant l'overhead lié à l'utilisation des chiffrements. Il propose des algorithmes, qui monopoliseront au minimum les ressources et l'utilisation de la mémoire. Surtout, il permet de rendre RTP indépendant des autres couches, en ce qui concerne l'application de mécanismes de sécurité.

Pour implémenter les différents services de sécurité précités, SRTP utilise les composants principaux suivants :

- **Une clé maîtresse** utilisée pour générer des clés de session ; Ces dernières seront utilisées pour chiffrer ou pour authentifier les paquets ;
- **Une fonction** utilisée pour calculer les clés de session à partir de la clé maîtresse ;
- **Des clés aléatoires** utilisées pour introduire une composante aléatoire afin de contrer les éventuels rejeu ou effets de mémoire.

SRTP utilise deux types de clés : clef de session et clef maîtresse. Par "clef de session", nous entendons une clef utilisée directement dans les transformations cryptographiques ; et par « clef maîtresse », nous entendons une chaîne de bit aléatoire à partir desquelles les clefs de sessions sont dérivées par une voie sécurisée avec des mécanismes cryptographiques [3].

Format du paquet SRTP

Un paquet SRTP est généré par transformation d'un paquet RTP, grâce à des mécanismes de sécurité. Donc, le protocole SRTP effectue une certaine mise en forme des paquets RTP avant qu'ils ne soient sur le réseau. La figure suivante présente le format d'un paquet SRTP [3].

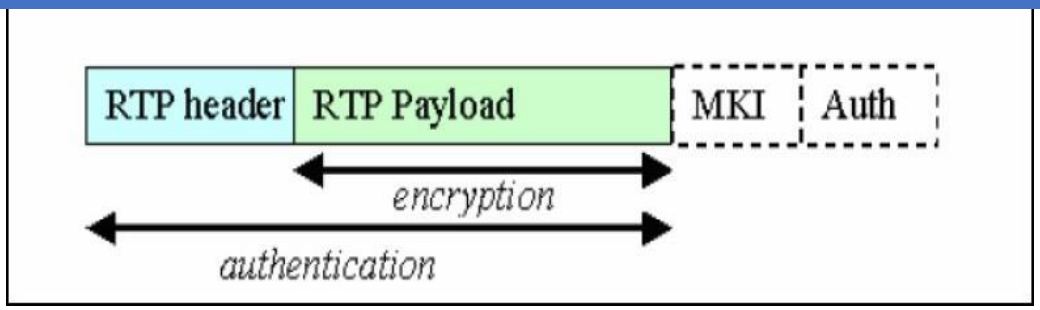

Figure 10 : Format d'un paquet SRTP.

On remarque que le paquet SRTP est réalisé en rajoutant deux champs au paquet RTP :

- **SRTP MKI (SRTP Master Key Identifier)** : sert à ré-identifier une clef maîtresse particulière dans le contexte cryptographique. Le MKI peut être utilisé par le récepteur pour retrouver la clef primaire correcte quand le besoin d'un renouvellement de clefs survient ;
- **Authentification tag** : est un champ inséré lorsque le message a été authentifié. Il est recommandé d'en faire usage. Il fournit l'authentification des en-têtes et données RTP et indirectement fournit une protection contre le rejeu de paquets en authentifiant le numéro de séquence.

3.1.3. Protocole TLS

C'est un protocole de sécurisation des échanges au niveau de la couche transport (**TLS : Transport Layer Security**). TLS, anciennement appelé **Secure Sockets Layer** (**SSL**), est un protocole de sécurisation des échanges sur Internet. C'est un protocole modulaire, dont le but est de sécuriser les échanges des données entre le client et le serveur indépendamment de tout type d'application. TLS agit comme une couche supplémentaire au-dessus de TCP [3].
Le protocole TLS est subdivisé en quatre sous protocoles

a). Le protocole Hanshake

C'est un protocole qui permet au client et au serveur de s'authentifier mutuellement, de négocier les algorithmes de chiffrement, de négocier les algorithmes de MAC (pour Message Authentification Code) et enfin de négocier les clés symétriques qui vont servir au chiffrement [6].

b). Le protocole Change Cipher Spec

Ce protocole contient un seul message :

Change_cipher_spec. Il est envoyé par les deux parties au protocole de négociation. Ce message transite chiffré par l'algorithme symétrique précédemment négocié [6].

c). Le protocole Alert

Ce protocole spécifie les messages d'erreur que peuvent s'envoyer les clients et serveurs. Les messages sont composés de deux octets. Le premier est soit warning soit fatal. Si le niveau est fatal, la connexion est abandonnée. Les autres connexions sur la même session ne sont pas coupées, mais on ne peut pas en établir de nouvelles. Le deuxième octet donne le code d'erreur [6].

d). Le protocole Record

Ce protocole chapeaute les autres protocoles de TLS, en fournissant une interface unifiée pour la transmission des données.

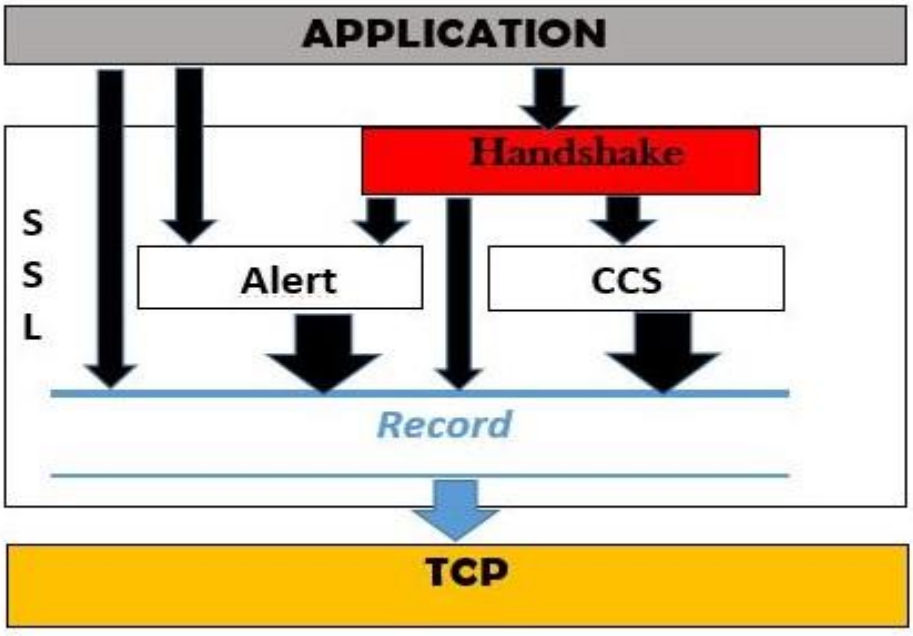

Figure 11 : Empilement des sous-couches protocolaires de SSL.

d.1). Rôles du protocole Record

- **Encapsulation** : permet aux données SSL (pour Secure Socket Layer), et TLS d'être transmises et reconnues sous une forme homogène [6] ;

- **Confidentialité** : Assure que le contenu du message ne peut pas être lu par un tiers : les données sont chiffrées en utilisant les clés produites lors de la négociation [6] ;
- **Intégrité et Identité** : Permet de vérifier la validité des données transmises, grâce aux signatures [6] ;
- **MAC** : cette signature est elle aussi générée à l'aide des clés produites lors de la négociation [6].

d.2). Processus d'encapsulation

- **Segmentation** : les données sont découpées en blocs de taille inférieure à 16 384 octets [6] ;
- **Compression** : les données sont compressées en utilisant l'algorithme choisi lors de la négociation. À partir de SSL 3.0, il n'y a plus de compression [6] :
- **Chiffrement** : le paquet obtenu est chiffré à l'aide de l'algorithme de chiffrement. Le choix peut se faire entre RC2, RC4, DES (pour Data Encryption Standard) avec une clef de taille 40 bits ou de 64 bits, ou l'algorithme de **Fortezza**. Ce dernier algorithme est un algorithme secret défense aux États Unis. Notons qu'il est possible de choisir des échanges en clair.

Algorithme de hachage utilisé, peut être soit le MD5 (pour Message Digest 5), soit le SHA (pour Secure Hash Algorithm). Il est possible de ne choisir aucun algorithme de hachage.

La négociation de cette suite de chiffrement se fait en clair pendant l'établissement de la session.

Le **tableau 1** donne l'ensemble des algorithmes supportés par SSL tandis que le **tableau 2** contient les suites de chiffrement reconnues

Fonction	**Algorithme**
Échanges de clefs	RSA, Fortezza, Diffie-Hellman
Chiffrement symétrique à la volée	RC4 avec clefs de 128 bits ou de 40 bits
Chiffrement symétrique en blocs	DES, DES40, 3DES RC2, IDEA, Fortezza
Hachage	MD5, SHA

Tableau 1 : Algorithmes négociés par le protocole Handshake.

Echange de Clefs	***Chiffrement Symétrique***	***Hachage***	***Signature***
	Sans Chiffrement RC4-40	MD5 ou SHA MD5	
RSA	RC4-128 RC2 CBC 40 IDEA CBC DES40 CBC DES CBC 3DES EDE CBC	MD5 ou SHA MD5 SHA SHA SHA SHA	
Diffie-Hellman	DES40 CBC	SHA	DSS ou RSA
	DES CBC 3DES, EDE CBC	SHA SHA	DSS ou RSA DSS ou RSA
Diffie-hellman Ephémère	DES40 CBC DES CBC 3DES EDE CBC	SHA SHA SHA	DSS ou RSA DSS ou RSA DSS ou RSA

Tableau 2 : Suites de chiffrement reconnues par SSL.

Réception des paquets

À la réception des paquets, le destinataire effectue les opérations suivantes [6] :

1. Vérification de l'entête SSL ;
2. Déchirage du paquet ;
3. Vérification du champ HMAC (pour keyed-hash message authentication code) (en appliquant la même fonction que ci-dessus aux données déchiffrées, puis en comparant le résultat au HMAC reçu) ;
4. Décompression des données ;
5. Réassemblage des parties.

Si au cours de ces vérifications se passe mal, une alarme est générée.

3.2. Sécurisation de l'application

Plusieurs méthodes peuvent être appliquées pour sécuriser l'application, ces méthodes varient selon le type d'application (serveur ou client). Pour sécuriser le serveur il faut [3] :

- L'utilisation d'une version stable, Il est bien connu que toute application non stable contient surement des erreurs et des vulnérabilités. Pour minimiser les risques, il est impératif d'utiliser une version stable ;

- Tester les mises à jour des softwares dans un laboratoire de test. Il est très important de tester toute mise à jour de l'application dans un laboratoire de test avant de les appliquer sur le système en production ;

- Ne pas tester les correctifs sur le serveur lui-même ;

- Ne pas utiliser la configuration par défaut qui sert juste à établir des appels. Elle ne contient aucune protection contre les attaques ;

- Ne pas installer une application client dans le serveur.

Certains paramètres doivent être appliqués de manière sélective. Ces paramètres renforcent la sécurité de l'application. On peut les activer ou les interdire sur la configuration générale de l'application, comme on peut juste utiliser les paramètres nécessaires pour des clients bien déterminés et selon le besoin bien sûr. Ces paramètres protègent généralement contre le déni de service et ces différentes variantes. Il est conseillé d'utiliser les paramètres qu'utilise le hachage des mots de passe, et cela assure la confidentialité.

3.3. Sécurisation du système d'exploitation

Il est très important de sécuriser le système sur lequel est implémenté le serveur de VoIP. En effet, si le système est compromis, l'attaque peut se propager sur l'application serveur. Celle-ci risque d'affecter les fichiers de configuration contenant des informations sur les clients enregistrés [3].

Il y a plusieurs mesures de sécurités à prendre pour protéger le système d'exploitation [3] :

- Utiliser un système d'exploitation stable. Les nouvelles versions contiennent toujours des bugs et des failles qui doivent être corrigés et maîtrisés avant ;

- Mettre à jour le système d'exploitation en installant les correctifs de sécurité recommandé pour la sécurité ;

- Ne pas mettre des mots de passe simples et robustes. Ils sont fondamentaux contre les intrusions. Et ils ne doivent pas être des dates de naissances, des noms, ou des numéros de téléphones. Un mot de passe doit être assez long et formé d'une combinaison de lettre, de chiffres et ponctuations ;

- Ne pas exécuter le serveur VoIP avec un utilisateur privilège. Si un utilisateur malveillant arrive à accéder au système via une exploitation de vulnérabilité sur le serveur VoIP, il héritera tous les privilèges de cet utilisateur ;

- **Asterisk in CHROOT** : empêcher le serveur VoIP d'avoir une visibilité complète de l'arborescence du disque, en l'exécutant dans un environnement sécurisé qui l'empêche d'interagir librement avec le système ;

- **Sauvegarde des fichiers log à distance** : les fichiers log sont très importants. Il est conseillé de les enregistrer sur un serveur distant ;

- **Installer seulement les composants nécessaires** : ainsi pour limiter les menaces sur le système d'exploitation. Il vaut mieux installer sur la machine le système d'exploitation et le serveur ;

- Supprimer tous les programmes logiciels ou des choses qui n'ont pas d'importance et qui peuvent être une cible d'attaque pour accéder au système ;

- Renforcer la sécurité du système d'exploitation en installant des patches qui permettent de renforcer la sécurité générale du noyau.

On peut aussi utiliser les pare-feu ou/et les ACL (pour Access Control List) pour limiter l'accès à des personnes bien déterminées et fermer les ports inutiles et ne laisser que les ports utilisés (5060, 5061, 4569, …). Le pare-feu (firewall) est un software ou hardware, qui a pour fonction de sécuriser un réseau ou un ordinateur contre les intrusions venant d'autres machines. Le pare-feu utilise le système de filtrage de paquet après l'analyse de l'entête des paquets IP, qui s'échange entre les machines [3].

Le firewall peut être implémenté avec une ACL, qui est une liste d'**Access Control Entry** (**ACE**) ou entrée de contrôle d'accès donnant ou supprimant des droits d'accès à une personne ou un groupe. On aura besoin d'ACL pour donner des droits à des personnes bien déterminés selon leurs besoins et leurs autorités [3].

Pour un serveur VoIP, il est important d'implémenter les ACL pour sécuriser le serveur en limitant l'accès à des personnes indésirables. Par exemple, seuls les agents enregistrés peuvent envoyer des requêtes au serveur. Il existe trois catégories d'ACL :

La liste de contrôle d'accès peut être installée en réseau sur les pares-feux ou les routeurs, mais aussi ils existent dans les systèmes d'exploitation.

Conclusion

Dans ce chapitre, il a été question de présenter les différentes vulnérabilités devant lesquelles le déploiement de la voix sur IP fait face. De ce fait, certaines mesures de sécurisation de l'environnement IP doivent être prises en compte afin de garantir la qualité des services.

La voix sur IP devient perpétuellement plus ciblée. Il existe plusieurs autres attaques qui menacent la sécurité du VoIP. Les attaques citées dans ce chapitre sont les plus fameuses et courantes dans les réseaux VoIP. Mais en suivant certaines pratiques, parmi dont on a cité certaines, nous serons dans la mesure de créer un réseau plus sécurisé.

CHAPITRE 3
Installation et configuration d'une solution VoIP basée sur l'outil Asterisk

Chapitre 3 : Installation et configuration d'une solution VoIP basée sur l'outil Asterisk

Introduction

Notre travail de recherche consiste à faire l'étude du maximum de fonctionnalités offertes par un **PBXAsterisk**. Il s'agit de traiter uniquement les possibilités d'Asterisk dans le cadre d'une solution VoIP, dans un réseau local. Donc, toute la partie concernant la connexion de le **PBX** avec le réseau téléphonique classique n'est pas abordé.

1. Présentation d'Asterisk

1.1. Définition

Asterisk est un autocommutateur téléphonique privée (**PABX**) open source pour systémes UNIX, originellement crée en 1999 par Mark Spencer fondateur de la société Digium. Asterisk est publié sous licence GPL. Asterisk permet, entre autres, la messagerie vocale, les conférences, les files d'attente, les agents d'appels, les musiques d'attente et les mises en garde d'appels, ainsi que la distribution des appels. Toutes ces fonctionnalités standards sont intégrées directement au logiciel. Asterisk implémente les protocoles H.320, H.323 et SIP, ainsi qu'un protocole spécifique nommé IAX (IAX : InterAsterisk eXchange).

Ce protocole IAX permet la communication entre deux serveurs Asterisk. Asterisk peut également jouer le rôle de « registrar » et passerelle avec les réseaux publics (RTC, GSM, etc.) [10].

1.2. Historique

Asterisk est né en 1999, créé par un étudiant de l'université d'Auburn (Etats-Unis - Alabama). À la recherche d'un commutateur téléphonique privé pour créer un centre de support technique sur Linux, il est dissuadé par les tarifs trop élevés des solutions existantes, et décide de se créer son propre routeur d'appels sous Linux : le PBX Asterisk. Quelques temps après, il a créé la société Digium, fournisseur de cartes FXO et FXS compatibles avec Asterisk [10].

1.3. Intérêt du choix d'Asterisk

Asterisk joue un rôle important dans le monde de la téléphonie, et c'est un commutateur privé. Il a été conçu principalement pour sa compatibilité avec les équipements numériques et

analogiques de la VoIP de base standard, ainsi que pour son moindre coût. Mais aussi, pour sa flexibilité. C'est un serveur qui évolue régulièrement en fournissant davantage de nouvelles fonctionnalités, il supporte tous les protocoles de la téléphonie et fonctionne sur plusieurs plateformes (Linux, Windows, Mac) [11].

1.4. Fonctionnalités

Asterisk compte un nombre très élevé de fonctions permettant de répondre à la majorité des besoins en téléphonie, nous citons :

- Messagerie vocale ;
- Conférence téléphonique ;
- Répondeur vocal interactif ;
- Mise en attente d'appels ;
- VoIP. . . etc.

1.5. Architecture

Asterisk est soigneusement conçu pour une flexibilité maximale. Les APIs spécifiques sont définies autour d'un système PBX central. Ce noyau avancé manipule l'interconnexion interne du PBX, proprement soustrait des protocoles spécifiques des codecs et des interfaces matérielles des applications de téléphonie. Cela permet à Asterisk d'utiliser n'importe quel matériel approprié et technologie disponible (maintenant ou à l'avenir) pour exécuter ces fonctions essentielles, en connectant le matériel et les applications [12].

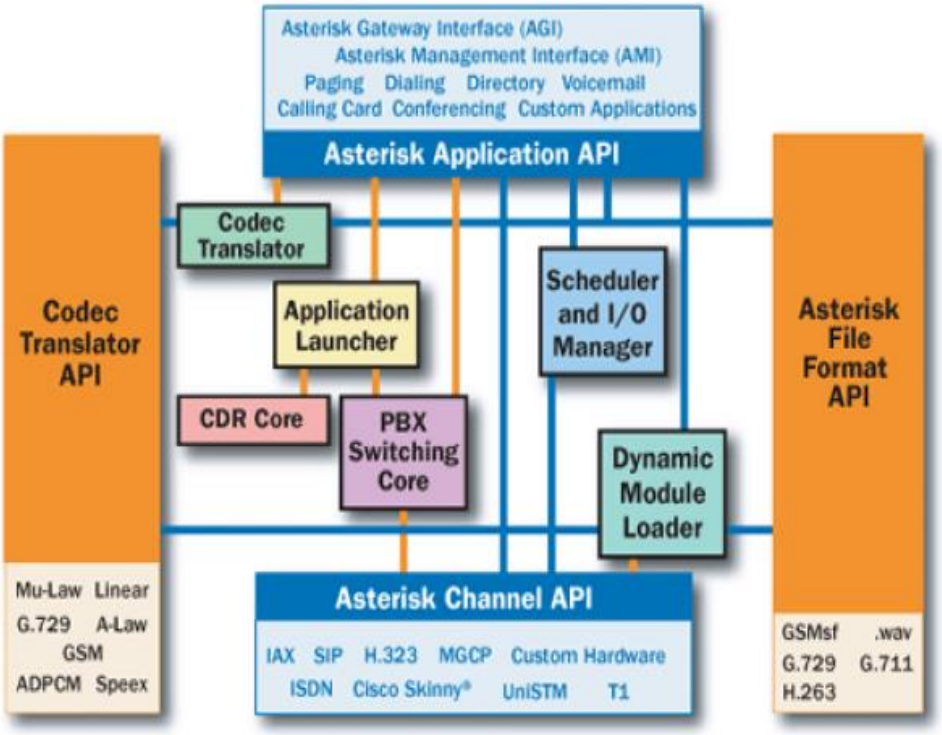

Figure 12: Architecture d'asterisk.

2. Installation d'Asterisk 18

Avant d'installer Asterisk, il faut préparer le système sous lequel on installera notre serveur. Pour cela, il faut installer tout d'abord les pré-requis nécessaires.

2.1. Détermination des pré-requis

Les pré-requis nécessaires pour que l'installation du serveur Asterisk s'accomplit avec succès, sont classés dans le tableau ci-dessous :

Nom du paquetage	Commande d'installation	Note
GCC 3.x	yum install –y gcc	Nécessaire pour compiler zaptel, libpri, et asterisk
Ncurses-devel	yum install –y ncurses-devel	Nécessaire pour menuselect
libtermcap-devel	yum install –y libtermcap-devel	Nécessaire pour asterisk
Kernel Development Headers	yum install -y kernel-devel	Nécessaire pour compiler zaptel
Kernel Development Headers (SMP)	yum install -y kernel-smp-devel	Nécessaire pour compiler zaptel
GCC C++ 3.x	yum install –y gcc-c++	Nécessaire pour asterisk
OpenSSL (optionnel)	yum install –y openssl-devel	Dépendance de OSP, IAX2 encryption, res_crypto (RSA key support) Nécessaire pour asterisk
zlib-devel (optionnel)	yum install –y zlib-devel	Dépendance de DUNDi Nécessaire pour asterisk
unixODBC; unixODBC-devel (optionnel)	yum install –y unixODBC-devel	Dépendance de func_odbc, cdr_odbc, res_config_odbc, res_odbc, ODBC_STORAGE
Libtool (optionnel; recommandé)	yum install –y libtool	Dependence de ODBC-related modules
GNU make (version 3.80 ou plus)	yum install –y make	Nécessaire pour compiler zaptel et asterisk

Tableau 3 : Liste de paquetages nécessaires pour compiler Asterisk et Libpri [13].

2.2. Téléchargement des codes sources

Les lignes de commandes nécessaires pour le téléchargement d'Asterisk et On identifie l'url. Après on télécharge via la commande wget

- Les mises à jour en ligne de commande :

APT est une interface de gestion des paquets des systèmes Linux Debian. **apt-get** est le programme de base permettant d'installer, de mettre à jour ou de supprimer des paquets. Depuis Debian 8 Jessie, un équivalent apt existe avec une interface colorée, plus agréable pour l'utilisateur.

L'utilisation d'apt-get reste toutefois recommandée dans les scripts Shell, sinon c'est apt qu'il convient maintenant d'utiliser. D'autres gestionnaires existent, notamment aptitude proposant une interface pour la gestion.

sudo apt-get update = Télécharge les informations des paquets à partir des sources configurées.

sudo apt-get upgrade = Mets à jour les paquets installés sans en supprimer.

sudo apt-get dist-upgrade = Installe les versions candidates des paquets installés en installant ou en supprimant d'autres paquets si nécessaire.

- Nous procédons ensuite à l'installation des dépendances :

```
sudo apt-get install Linux-headers$(uname -r)
sudo apt-get install build-essential libtonezone-dev autoconf pkg-config libtool
```

sudo : exécute l'instruction avec des privilèges élevés.

apt-get : est une commande permettant de récupérer un package / programme spécifique

linux-headers : est le début d'un nom de package.

- Après on va créer le répertoire Asterisk par cette commande :

```
mkdir /downloads/asterisk -r
```

- Et après, on va se déplacer sur ce répertoire avec cette commande :

```
cd /downloads/asterisk
```

- Et on télécharge le logiciel avec cette commande :

```
wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-18-current.tar.gz
```

2.3. Extraction des paquettages

- Les paquetages téléchargés sont des archives compressées, qui contiennent le code source, on aura besoin de les extraire, en utilisant la commande 'tar', avant de les compiler :

```
tar -zxvf asterisk-18-current.tar.gz
```

- Après, on va se positionner sur asterisk-18.3.0 :

```
cd /downloads/asterisk/asterisk-18.3.0/contrib/scripts/
```

- Et enfin, on va installer les pré-requis avec la commande ci-dessous :

```
./install_pre_req install
```

2.4. Installation des paquets

```
sudo apt install libedit-dev
sudo apt install uuid-dev
sudo apt install -y libjansson-dev
sudo apt install -y libxml2-dev
sudo apt install libsqlite3-dev
```

2.5. La configuration

```
./configure
```

Après avoir tapé la commande de configuration sur la terminale, il va nous afficher le symbole du logiciel Asterisk, ce qui signifie que le logiciel est bien installé.

2.6. Compilation et installation

Pour les distributions Linux on utilisent les commandes suivantes :

make	= compilation du code source
make install	= éxecution de la partie install dans makefile
make samples	= installer les exemples de fichiers de configuration
make progdocs	=
make config	= cette commande charge le serveur Asterisk au démarrage du système et installe les fichiers de configuration
make install-logrotate	= installe les fichiers de configuration

2.7. Finalisation de l'installation

Ainsi, Asterisk est installé. Il suffit maintenant de lancer le serveur et de se connecter à la console CLI (Commande Line Interface) via la commande :

```
sudo asterisk -rvv
```

Le serveur Asterisk permet d'interagir directement avec le système sans avoir à modifier les fichiers de configuration avec la CLI.

2.8. Commandes utiles

Une fois connecté à Asterisk via la console, plusieurs commandes utiles, internes à la console sont disponibles :

help : affiche la liste des commandes et aide associée ;

reload : recharge tous les fichiers de configurations ;

restart now : relance complètement et immédiatement Asterisk ;

sip reload : recharge le fichier sip.conf ;

sip show peers : voir le status des peers SIP ;

sip show channels : permet de voir les canaux actifs ;

sip set debug : permet de voir les messages SIP qui passent par le serveur ;

dialplan reload : recharge le fichier extensions.conf ;

agent show : voir le status des agents ;

agent logoff name : déconnecter l'agent name ;

sip show users : voir le status des utilisateurs SIP.

- Avec la commande **CLI>help**, on peut alors voire rapidement l'ensemble des commandes disponibles via l'interface CLI (Command Line Interface). D'où il est nécessaire d'avoir les fichiers qui commencent par **sip**, comme illustré sur la figure suivante :

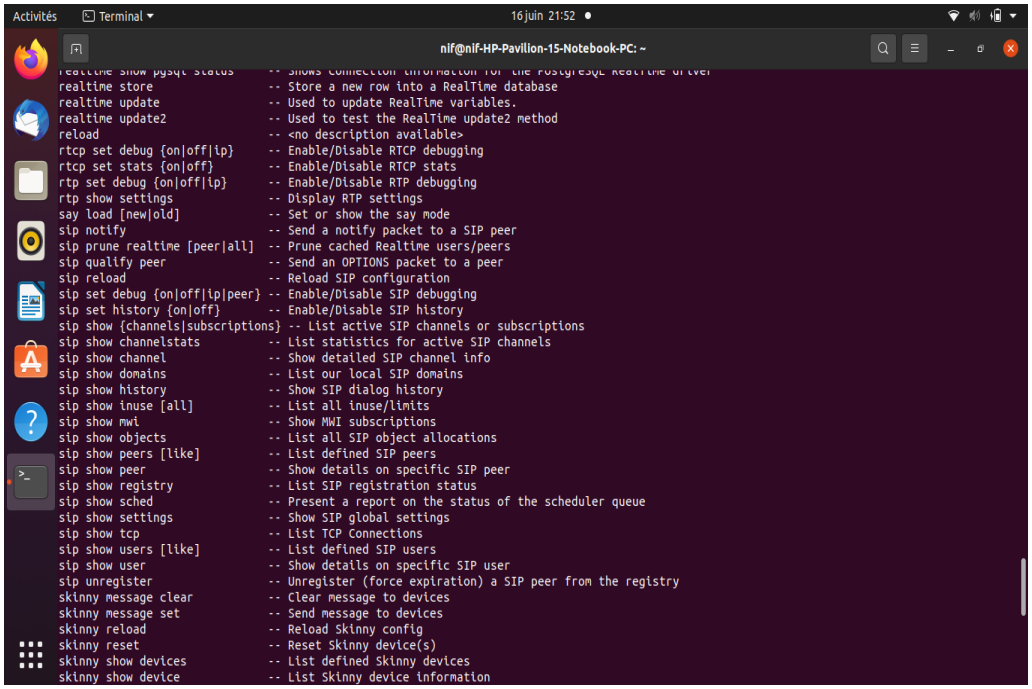

Figure 13 : Capture de la commande help.

3. Configuration d'Asterisk

La configuration du serveur asterisk est réalisée à travers plusieurs fichiers textes, qui se situent dans le répertoire **/etc/asterisk**. Parmi eux, le fichier **sip.conf** qui permet la déclaration des utilisateurs.

- Pour éditer le fichier **sip.conf** :

sudo gedit /etc/asterisk/sip.conf

- Pour ajouter les deux utilisateurs, il faut configurer ce fichier en insérant par exemple à la fin les lignes suivantes (le texte après le point-virgule est un commentaire) :

```
[hakim]                          ; nom du téléphone
type=friend                      ; type de téléphone
host=dynamic                     ; enregistrement dynamique
username=hakim                   ;nom d'utilisateur associé
secret=toto                      ; mot de passe
callerid='hakim' <555>           ; association user et numéro de téléphone
context=defaut

[zak]                            ; nom du téléphone
type=friend                      ; type de téléphone
host=dynamic                     ; enregistrement dynamique
username=zak                     ;nom d'utilisateur associé
secret=toto                      ; mot de passe
callerid='zak ' <556>            ; association user et numéro de téléphone
context=defaut
```

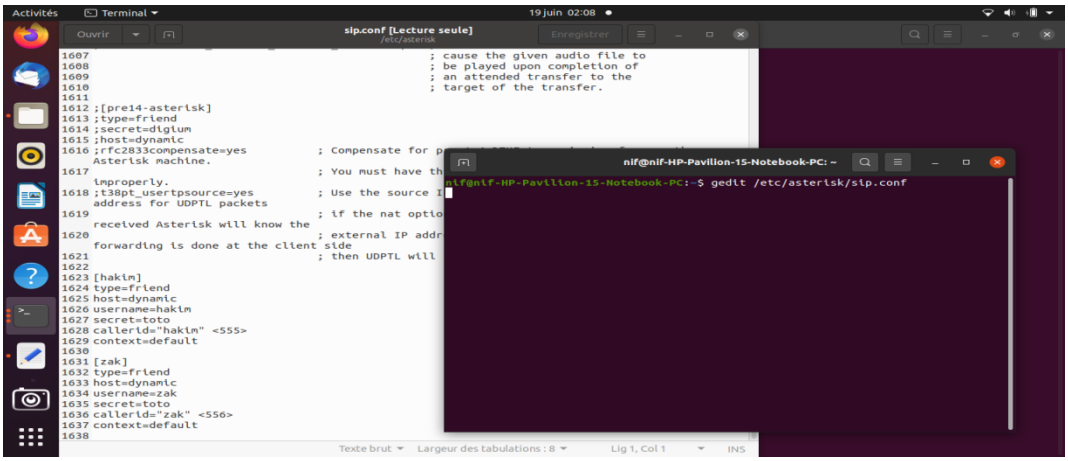

Figure 14 : Capture du fichier sip.conf

A chaque fois qu'on modifie le fichier 'conf ', il faut recharger ce fichier de configuration dans Asterisk.

- Pour recharger le module SIP afin de relier la configuration modifiée, il suffit de taper la commande :

CLI>sip reload

Dans la console d'Asterisk, il nous suffit de taper la commande : **reload** cette commande permet de recharger les fichiers de configurations d'Asterisk sans redémarrer le serveur.

Plusieurs options permettent de définir et de paramétrer un client :

- **type** : type de client (peer, user ou friend) ;
- **username** : identifiant de l'utilisateur ;
- **secret** : mot de passe de l'utilisateur ;
- **host** : méthode pour trouver le client (dynamic, nom d'hôte ou adresse IP) ;
- **callerid** : identité de l'utilisateur ;
- **Language** : langue par défaut pour l'utilisateur.

Description des paramètres : Pour chacun des paramètres précédents, plusieurs valeurs sont disponibles selon la configuration désirée.

Type :

- **peer :** client SIP auquel Asterisk pourra envoyer des appels ;
- **user :** client SIP qui pourra passer des appels via Asterisk ;
- **friend :** client qui sera à la fois en mode 'peer' et 'user'.

Host :

- **dynamic :** le client s'enregistre auprès du serveur ;
- **nom d'hôte :** nom d'hôte du client ;
- **adresse IP :** adresse IP du client.

L'exemple suivant du fichier **sip.conf** avec deux utilisateurs **hakim** et **zak.**

Dans ce fichier de configuration, nous avons créer deux utilisateurs. Une fois le fichier **sip.conf** est enregistré, nous allons dans la console Asterisk, taper **reload** puis on tape la commande :

CLI> sip show users.

Les deux comptes utilisateurs que nous venons de créer devrait y apparaitre.

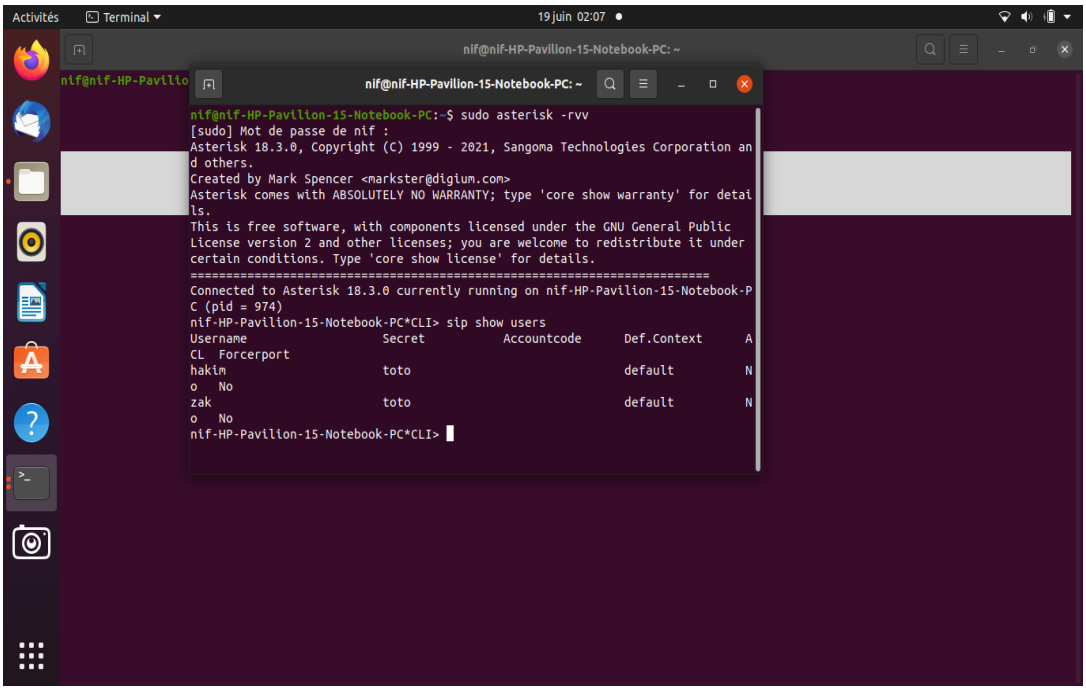

Figure 15 : Capture sur les utilisateurs créer.

Dans un autre terminal, on doit saisir la commande suivante pour avoir l'adresse IP :

ifconfig

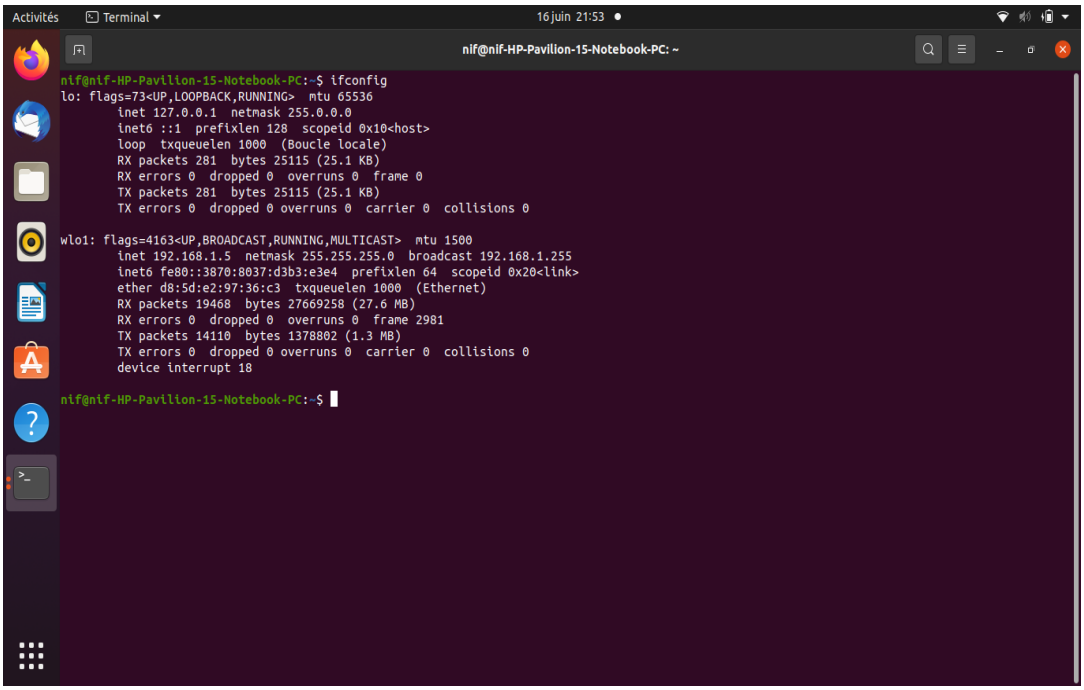

Figure 16 : Capture de la commande ifconfig.

Nos utilisateurs sont créés, mais ils n'ont pas encore la possibilité de s'appeler.

Pour éditer le fichier **extensions.conf** :

sudo gedit/etc/asterisk/extensions.conf

Il suffit maintenant d'attribuer un numéro de téléphone à chacun des deux utilisateurs que nous venons de déclarer. Le fichier **extensions.conf** permet d'associer à chaque numéro de téléphone une suite de commandes.

Pour se faire, il suffit d'ajouter les deux lignes suivantes dans le contexte **[default]** de ce fichier :

exten => 555,1,Dial(SIP/hakim,10,tr)

555 appels hakim en priorité (1) avec un timeout de dix secondes pour raccrocher

exten => 556,1,Dial(SIP/zak,10,tr)

556 appels le téléphone zak.

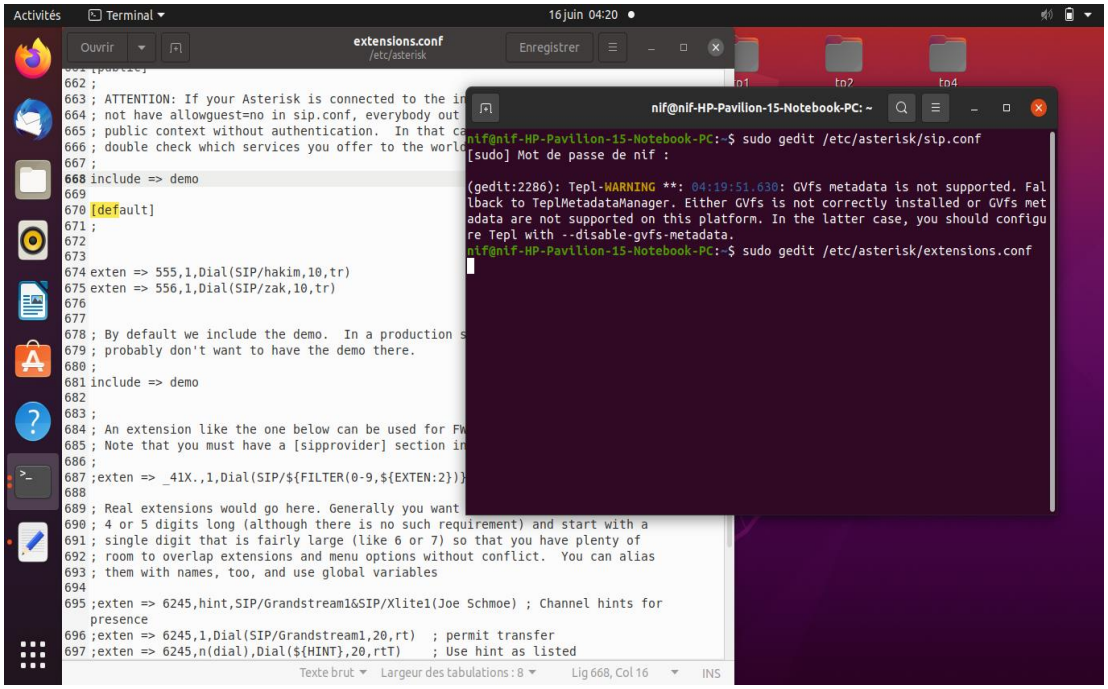

Figure 17 : Capture du fichier extensions.conf.

A la fin, on doit vérifier l'éxistance de ces deux numéros sur le serveur Asterisk avec la commande :

CLI> dialplan show

Voilà, le serveur Asterisk est maintenant prêt à enregistrer les deux utilisateurs hakim et zak qui disposent respectivement des numéros de ligne 555 et 556.

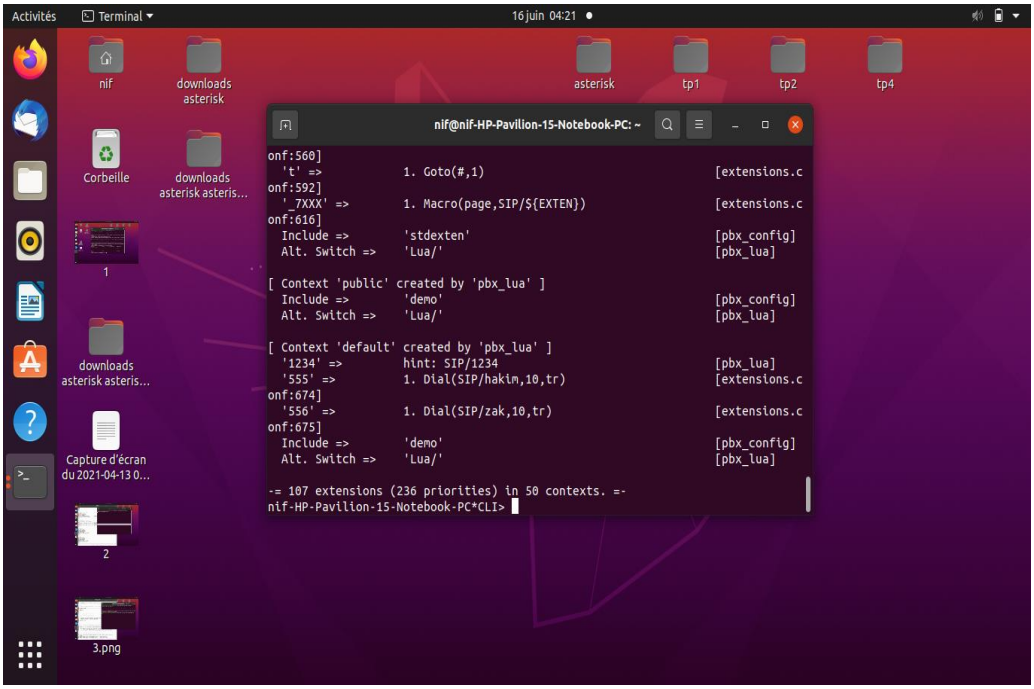

Figure 18 : Capture qui affiche les utilisateurs enregistrés avec leurs numéros de ligne.

On va tester le fonctionnement de la communication téléphonique avec un appel des deux utilisateurs, d'où on doit servir de deux SoftPhone (téléphone logiciels) pour se connecter en tant que hakim et zak. Il faut que les deux SoftPhones et le serveur Asterisk doivent appartenir au même réseau local (adresse IP). Puis il suffit que l'un des utilisateurs tape le numéro de l'autre pour effectuer l'appel.

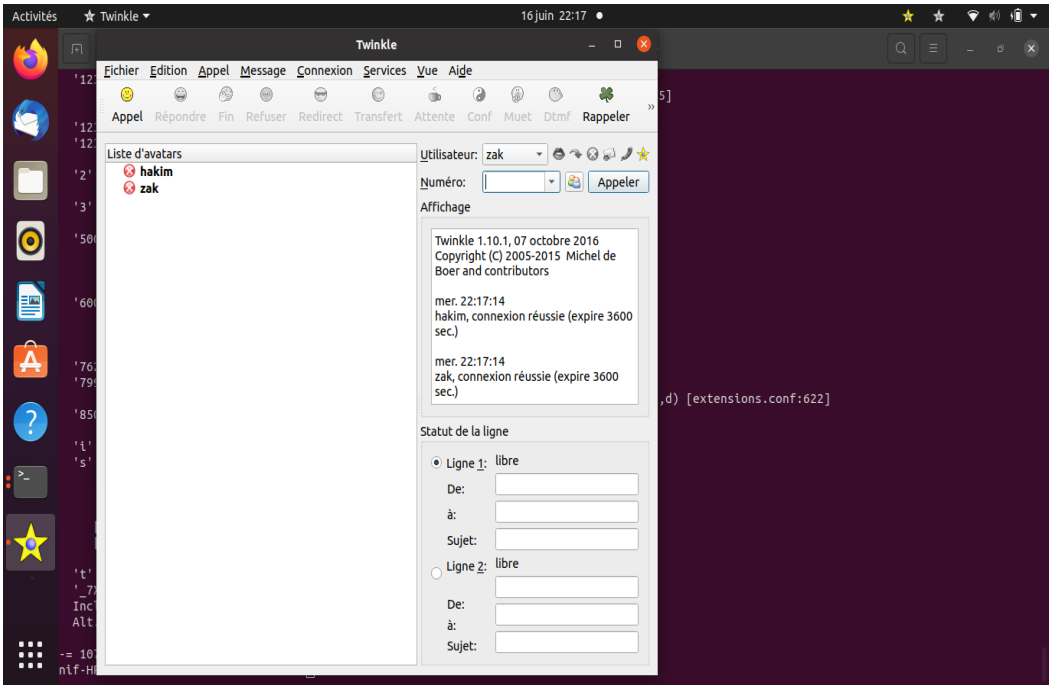

Figure 19 : Capture sur le teste d'appel avec Twinkle.

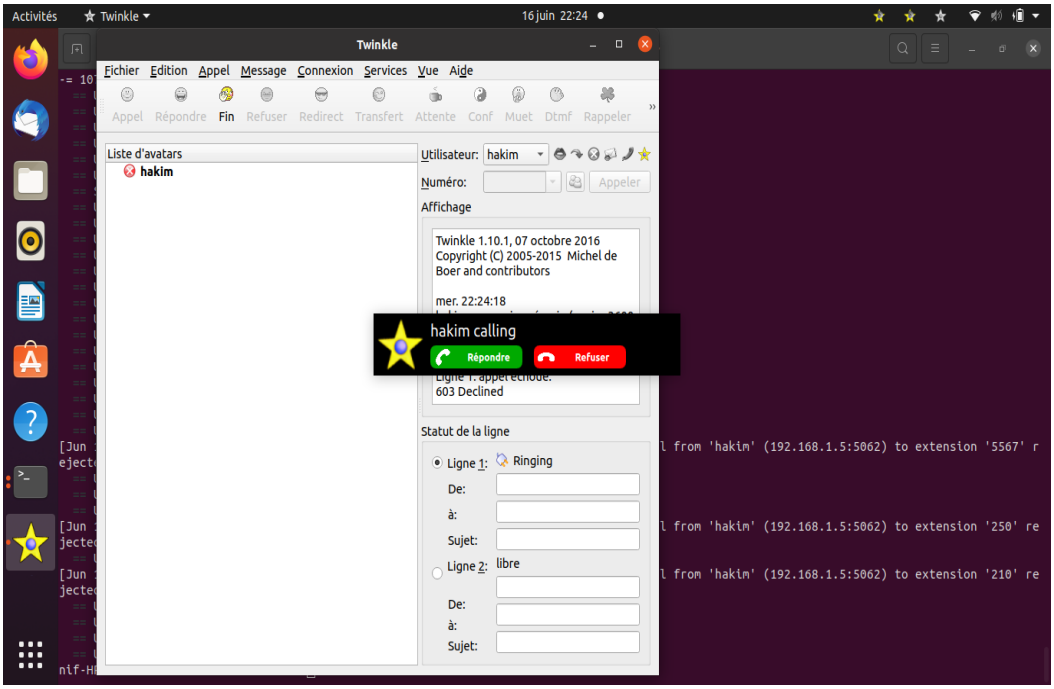

Figure 20 : Capture d'appel hakim vers zak.

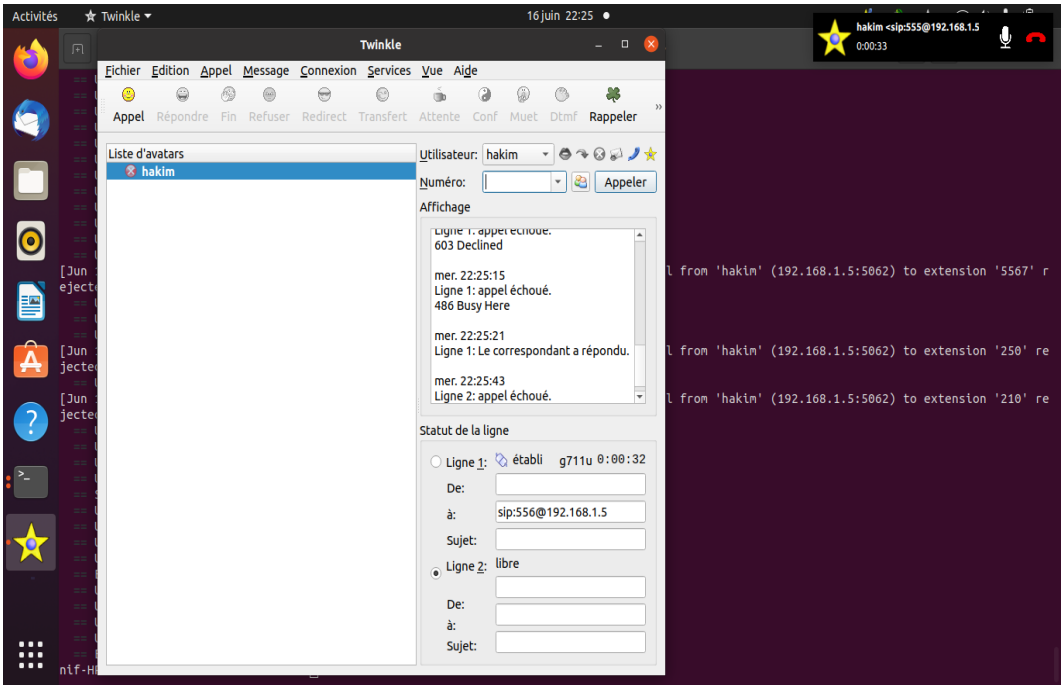

Figure 21 : Capture affiche que zak a répondu.

4. Mise en place des boîtes vocales

Les deux fichiers, que nous allons éditer pour effectuer ce test, sont les fichiers **voicemail** et **extensions.conf** se trouvant dans **/etc/asterisk**.

4.1. Configuration de la messagerie vocale d'un utilisateur

Maintenant nous allons éditer le fichier **extensions.conf** pour configurer deux choses :

- Le fait qu'au bout d'un certain temps Asterisk bascule sur la boite vocale de l'utilisateur dû celui-ci ne répond pas ;
- Création d'une extension pour créer le numéro qui servira à consulter la boite vocale.

Dans l'exemple suivant, les appels arrivant sur le serveur Asterisk à destination du numéro 201 sont envoyés vers le téléphone de Poste1 pendant 10 secondes puis sont envoyés sur la messagerie de Poste1.

- Sur le fichier extensions.conf, il faut mettre la configuration suivante :

```
[default]
exten => 201,1,Dial(SIP/Poste1,10)
exten => 201,2,VoiceMail(222@test_voicemail)
```

Signification :

- **exten => :** on déclare l'extension ;
- **201 :** numéro ;
- **2 :** priorité ;
- **VoiceMail :** on lance l'application Voicemail ;
- **222@test_voicemail :** on récupère le numéro mis en variable (222) et on se connecte à la boite vocale associée du contexte work (test_Voicemail) (comme précisé dans le fichier **voicemail.conf**).

4.1.1. Modification dans le fichier voicemail.conf

- Commencer par indiquer le numéro de la messagerie, associé à la mailbox de la manière suivante :

```
<Numéro de voicemail> => <mot de passe>, <nom d'utilisateur>, <email>, <option(s)>.
```

- Sur le fichier **voicemail.conf**, il faut mettre la configuration suivante :

```
[test_voicemail]
222 => 0000,Poste1,Poste1@asterisktest.com
```

4.1.2. Ecouter sa messagerie

A présent, il n'y a plus qu'à ajouter un plan de numérotation, pour consulter sa messagerie. Notez que le numéro de répondeur sera le même pour tout le monde, et qu'un serveur vocal vous demandera votre numéro de messagerie, ainsi que le mot de passe. Cela peut être ajouté par l'entrée suivante, sur le fichier **extensions.conf,** dont il faut mettre la configuration suivante :

```
exten => 666,1,VoiceMailMain(222@test_voicemail)
```

En appelant le 666 on aura la boite vocale 222 qui se trouve dans le contexte **[test_voicemail],** qui nous demandera de saisir le mot de passe de la boite vocale.

5. Routage d'appels vers un groupe d'utilisateurs

Dans l'exemple suivant, les appels arrivant sur le serveur Asterisk, à destination du numéro 205, sont envoyés vers le téléphone de Poste1, puis vers le téléphone de Poste2.

- Dans le fichier **extensions.conf**, il faut ajouter la configuration suivante :

```
[default]
exten => 205,1,Dial(SIP/Poste1,10)
exten => 205,2,Dial (SIP/Poste2,10)
exten => 205,3,Goto(default,205,1)
```

Remarque :

L'instruction **Goto()** permet de renvoyer l'appel où l'on veut dans le fichier **extensions.conf**. Dans notre cas, l'appel basculera au téléphone de Poste1 au téléphone de Poste2 jusqu'à ce qu'un des deux décroche.

6.Routage vers plusieurs téléphones en même temps

L'exemple suivant montre comment faire sonner deux téléphones en même temps. Quand on compose le 206, les téléphones de Poste1 et de Poste2 sonneront.

- Dans le fichier **extensions.conf,** on doit ajouter cette configuration :

```
[default]
exten => 206,1,Dial(SIP/Poste1&SIP/Poste2,10)
```

7. Standard automatique

Le standard automatique permet à un utilisateur d'écouter un message lui indiquant les choix possibles. Après, il lui suffit de presser une des touches pour effectuer l'action voulue. Il est possible de combiner les menus pour développer une architecture plus complexe. Dans l'exemple suivant, quand l'utilisateur compose le 210, il entend un message vocal qui l'invite à taper 1,2 ou 9 sur son clavier. S'il tape 1, l'appel est envoyé au Poste1. S'il tape 2, l'appel est envoyé au Poste2. S'il tape 9, Asterisk raccroche. Si l'utilisateur ne fait rien, le message est joué en boucle. D'où on doit ajouter ces configurations dans le fichier **extensions.conf** :

```
[default]
exten => 210,1,Goto(Menu,s,1)              ;appel du standard automatique

[Menu]                                     ;standard automatique
exten => s,1,Background(var/msg/Menu)      ;le message audio enregistré dans
                                           ; /var/msg/Menu.gsm et joue
exten => s,2,WaitExten(2)                  ;on attend 2 secondes
exten => s,3,Goto(Menu,s,1)                ;on recommence le tout
exten => 1,1,SayNumber(1)
exten => 1,2,goto(default,200,1)           ;1 appel Poste1
exten => 2,1,SayNumber(2)
exten => 2,2,Goto(default,201,1)           ;2 appel Poste2
exten => 9,1,SayNumber(9)
exten => 9,2,Hangup()                      ;9 on raccroche
```

Dans chaque ajout ou modification d'un client, il faut mettre à jour le serveur Asterisk en utilisant les commandes suivantes :

```
CLI> sip reload
CLI> dialplan reload
CLI> reload
```

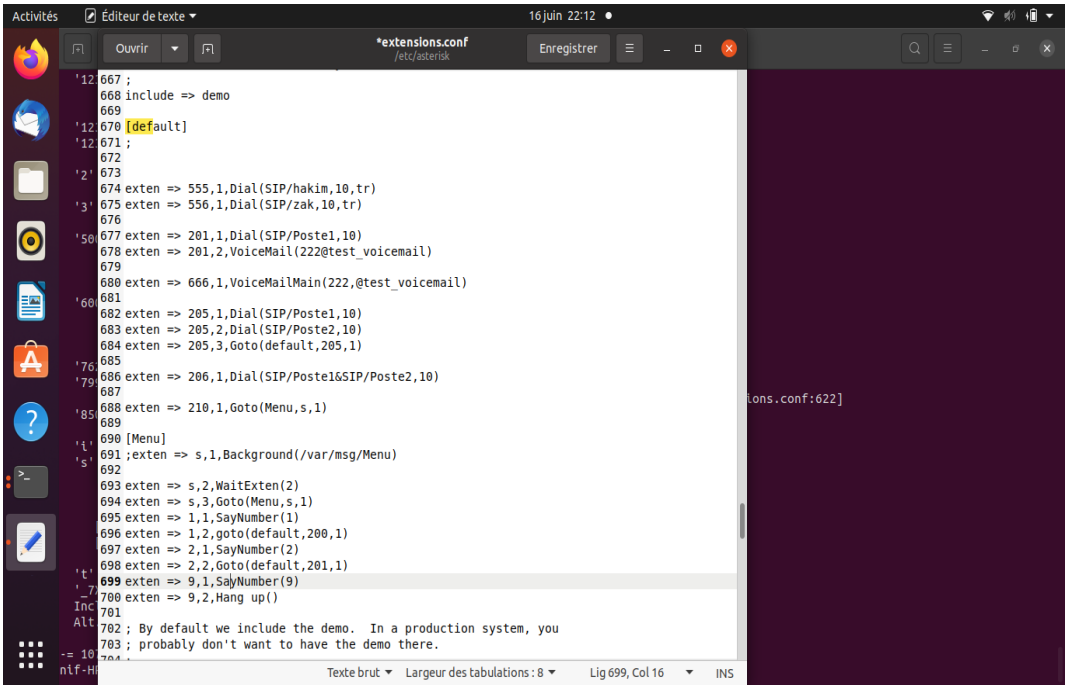

Figure 22 : Capture affiche les configurations ajouter au fichier extensions.conf.

On va tester les appels

8. Installation et configuration de MizuDroid

8.1. Présentation MizuDroid

MizuDroid est un softphone VoIP pour téléphones mobiles Android, basé sur les normes du protocole SIP, permettant aux utilisateurs de se connecter à un serveur VoIP et de passer des appels vers d'autres utilisateurs VoIP ou vers une ligne fixe et réseaux mobiles, généralement à des prix inférieurs à ceux des appels GSM natifs. Il fonctionne avec n'importe quel SIP conforme au Softswitch, ProxyVoIP, Softphone ou téléphone IP.

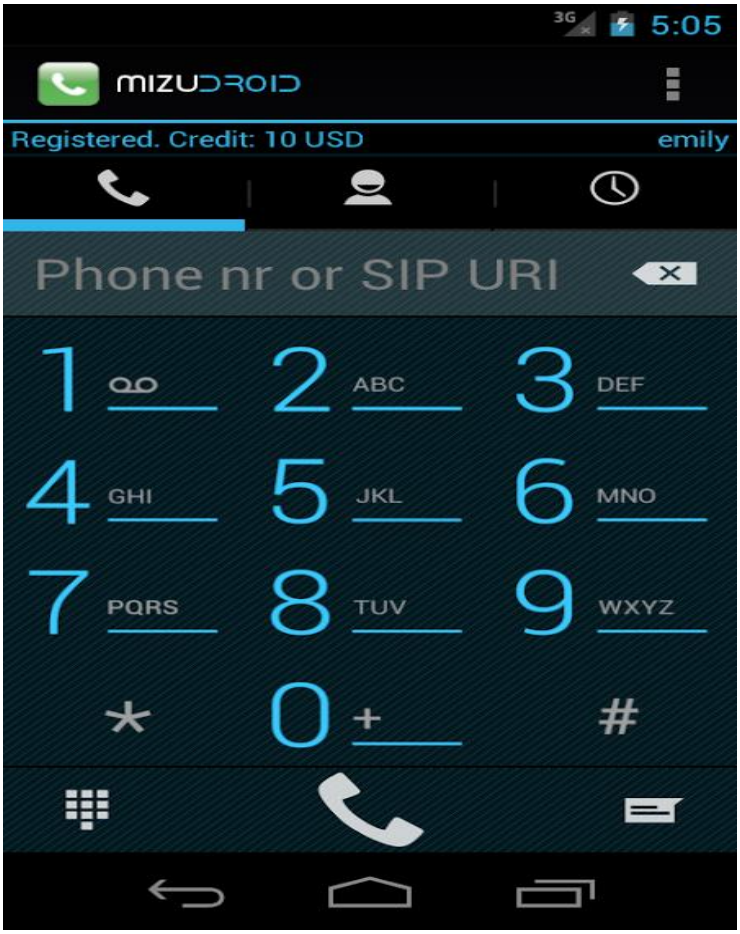

Figure 23 : MizuDroid softphone.

8.2. Configuration de MizuDroid

Contactez vos proches en local où à l'international avec la téléphonie VoIP de votre forfait internet Nautile [14].

Pour configurez sur votre téléphonie android l'application MizuDroid il faut suivre ces étapes :

1. Téléchargez l'application « MizuDroid SIP VoIP » sur Play store ;
2. Lancez l'application « MizuDroid SIP VoIP » ;
3. Entrez 'sip.nautile.nc' dans 'server', votre login VoIP dans 'usename', ainsi que votre mot de passe dans 'password' puis OK. Ces informations sont inscrites sur la fiche récapitulative Nautile ou sur votre espace 'Mon nautile' dans la section 'Téléphonie VoIP'.

4. La pastille en haut à gauche passe au vert, vous pouvez désormais composer le numéro de téléphone correspondant.

Vérifiez si la configuration est correcte en appelant le 68 99 99 [14].

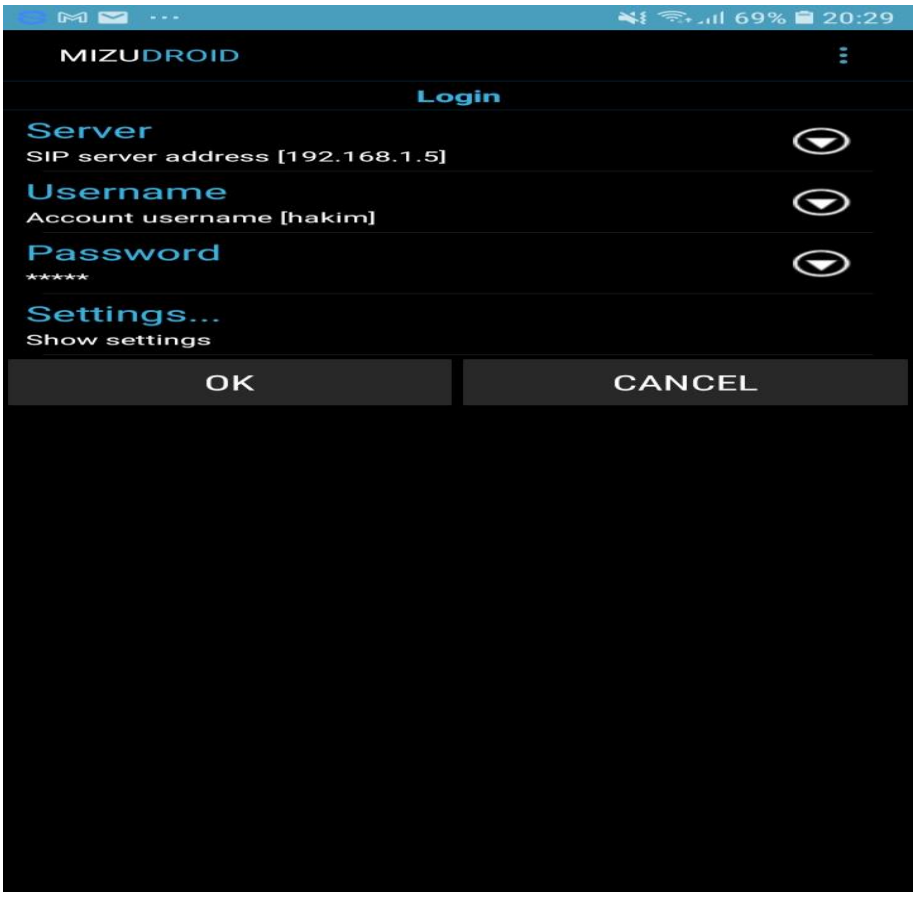

Figure 24 : Configuration du compte de l'appelant « client ».

Faire des appels :

Vous disposez des options suivantes :

- Sur la page du clavier de numérotation, appuyez sur les chiffres pour entrer le numéro de destination ;
- Sur le pavé numérique, entrez n'importe quel numéro, ainsi que le nom d'utilisateur SIP ou URI ;

- Appel de vos contacts ;
- Appel depuis la page d'historique.

Pendant l'appel, vous avez les options suivantes :

- Ajouter d'autres parties (conférence) ;
- Muet ;
- Tenir ;
- Envoyer DTMF ;
- Définir le haut-parleur ;
- Transférer l'appel.

Nous pouvons maintenent faire un appel test de l'utilisateur « 555 » (hakim) vers l'utilisateur « 556 » (zak) comme ceci :

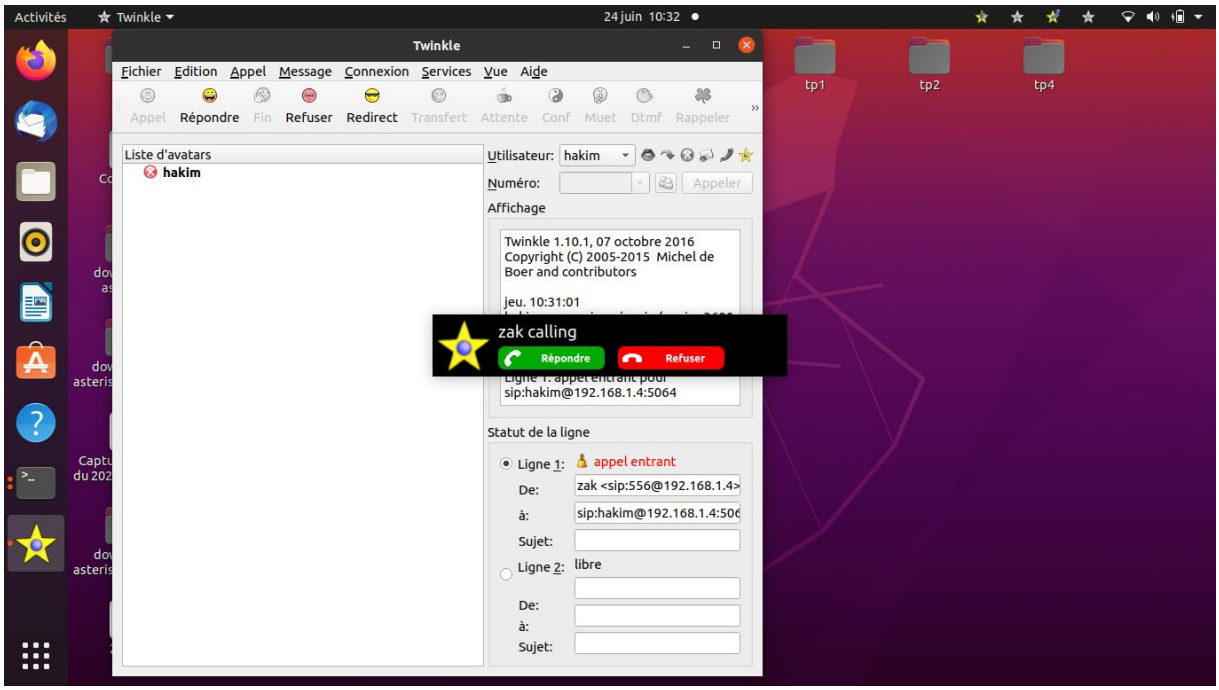

Figure 25 : Appel test entre l'utilisateur « 555 » et « 556 ».

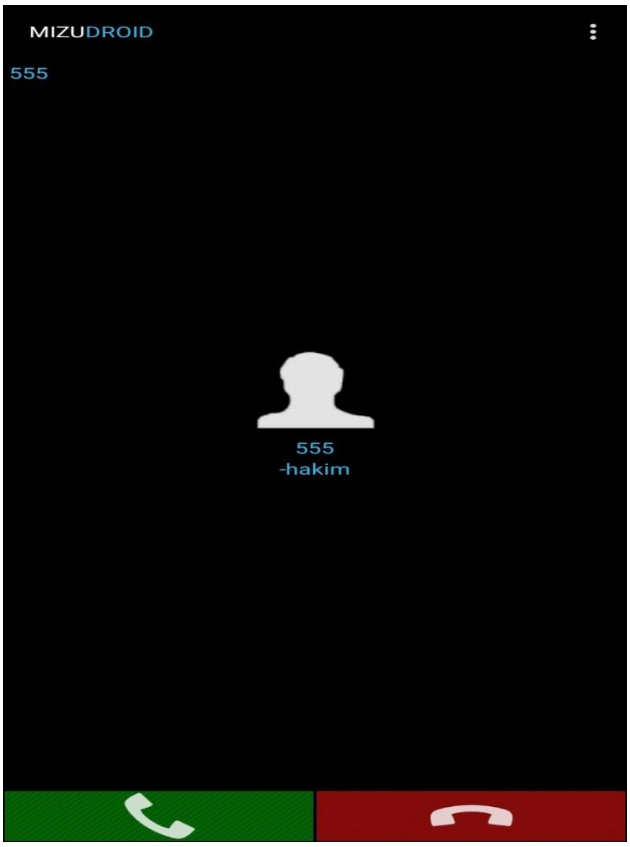

Figure 26 : Appel test entre les utilisateurs hakim et zak.

Conclusion

Nous constatons que la VoIP peut être déployée selon une architecture voulue en fonction du besoin de l'utilisateur. Bien que le choix des serveurs de téléphonie soit multiple, plusieurs choisissent Asterisk pour les nombreux avantages qu'il offre en termes de fiabilité, de coût et surtout de performance. La capacité d'Asterisk à se conformer au besoin des utilisateurs lui confère un avantage unique en son genre. On peut dire aussi qu'Asterisk est un outil ouvert à tous, gratuit et simple d'utilisation.

D'autre part, un serveur sans client est complètement inutile. Ce qui sous-entend qu'Asterisk ne serait rien sans ces précieux clients, tels que MizuDroid. Ce dernier, lorsqu'il est correctement configuré, offre une large gamme de possibilité à l'utilisateur allant du simple appel téléphonique aux messages textes. C'est en travaillant ensemble qu'Asterisk et MizuDroid peuvent donner le meilleur d'eux-mêmes pour les utilisateurs. En effet, si le serveur est défaillant, le softphone ne fonctionnerait pas et vice vers ça, s'il n'y a pas de logiciel client, l'utilisateur ne pourrait rien faire. Cela prouve qu'Asterisk joue un rôle très important au niveau de l'entreprise

Dans le dernier chapitre nous allons intéresser aux techniques, mécanismes et configuration à mettre en place, afin de sécuriser la solution VoIP basée sur le serveur Asterisk.

Chapitre 3 : Installation et configuration d'une solution VoIP basée sur l'outil Asterisk

CHAPITRE 4

Proposition et implémentation des mécanismes de sécurité pour la VoIP

Chapitre 4 : Proposition et implémentation des mécanismes de sécurité VoIP

Introduction

La VoIP peut dans une décennie constituer une des plus puissantes technologies jamais conçues par l'homme. Néomoins, si la qualité des services qu'elle offre ne satisfait pas les clients et les utilisateurs, la VoIP ne pourra jamais émerger et prendre la place qui devrait être la sienne. Pour pallier ce problème, il est nécessaire d'apporter des améliorations aux services offerts. Ceci n'est pas une chose aisée, car la VoIP elle-même ne peut être améliorée que si d'autres ressources viennent s'y ajouter. Le cas du présent travail qui traite la VoIP avec Asterisk, qui offre la possibilité de sécuriser la VoIP en faisant des modifications au cœur même du logiciel.

Ce chapitre a pour objectif d'illustrer les différentes solutions de sécurisation de la VoIP. Pour atteindre ce but, il est nécessaire de faire appel à diverses méthodes spécifiques visant chacune à améliorer la qualité du service afin de satisfaire au mieux l'utilisation, et afin de lui garantir les meilleures protections possibles.

1. Localisation des serveurs VoIP

Toute bonne attaque VoIP commence par une étape, qui établit le profil de la cible connu sous le nom *profiling* ou encore *foot priting*. Une empreinte engloble les informations sur la cible qui déploie le serveur VoIP et ces paramètres de sécurité. Il existe plusieurs méthodes pour la collecte des informations et voici quelques-unes les plus utilisées [12] :

1.1. Utilisations des serveurs Whoi

Les Whois sont des services proposés gratuitement en ligne permettant d'obtenir des informations sur un domaine particulier, sur une adresse de messagerie. Grâce à ses bases de données comme [12] :

- **Whois.ripe.net :** s'occupe d'attribuer des adresses IP pour l'Europe ;
- **Whois.apnic.net :** attribue les adresses IP pour l'Asie ;
- **Whois.nic.mil :** attribue les adresses IP des systèmes militaires américains.

1.2. Utilisations des aspirateurs de sites

Si la cible à un site, le pirate doit le parcourir à la recherche d'adresses emails, de compte et mots de passes ou d'autres informations précises. Parcourir le code source peut aussi

recenser des informations qui pourraient permettre de remonter aux sources. Les aspirateurs de sites permettent d'automatiser ces recherches [12].

1.3. Utilisations des moteurs de recherches et des agents intelligents

Un des grands avantages des moteurs de recherches Internet est leurs énormes potentiels pour découvrir les plus obscurs détails sur l'Internet. L'un des plus grands risques pour la sécurité est aujourd'hui l'énorme potentiel des moteurs de recherche pour découvrir les détails sur l'Internet. Il existe de sorte qu'un hacker peut exploiter, en utilisant simplement les fonctionnalités avancées d'un service tel que Google. Le ciblage des catégories suivantes des résultats de recherche peuvent souvent fournir de riches détails sur la solution VoIP, déployée par un organisme [12] :

- Vendeur de produit VoIP, les communiqués de presse et des études de cas ;
- CV de l'administrateur ou liste de références des vendeurs ;
- Les forums.

1.4. Balayage (Scan) des réseaux VoIP

Pour pouvoir identifier chaque composante du réseau, il faut déchiffrer et comprendre un bon nombre de paquets afin de reconnaitre par exemple leur adresse IP et son ID. D'autant plus qu'un réseau VoIP ne se limite pas à quelques clients et un serveur Asterisk.

Les serveurs TFTP (pour Trivial File Transfert Protocol) par exemple sont d'une nécessité pour un attaquant afin de retrouver les fichiers de configurations des téléphones IP pour leur usurper leurs identités par exemple.

Afin de scanner un réseau, l'outil nécessaire pour cela est un scanner de réseau (sniffer en anglais). C'est un logiciel permettant de découvrir les équipements présents sur un réseau et les services qu'il offre. Le scanner de réseau est souvent utilisé par les administrateurs réseau au cours de test de sécurité. Son principe de fonctionnement est de tester chaque adresse IP et chaque port TCP ou UDP afin de vérifier la présence d'un serveur ou d'un quelconque équipement fonctionnant en TCP/IP [12].

2. Attaques au niveau applicatif

- Les téléphones VoIP disposent d'une interface web de base souvent non protégée, permettant sa programmation à distance. L'assaillant après un scan et identification des
- terminaux, va pouvoir récupérer des informations essentielles (mots de passe, adresses, etc) et détourner à son profit les comptes.
- Les téléphones et IPBX proposent des services qui parfois contiennent des failles de sécurité. Une fois exploitée, l'assaillant pourra prendre le contrôle de tout ou partie du système.
- La configuration ne prenant pas suffisamment en compte la sécurité : mot de passe évident, compte basique, etc [11].

3. Les logiciels d'attaques

3.1. Wireshark

Wireshark est un logiciel d'analyse de protocole, utilisé dans le dépannage et l'analyse de réseaux informatiques, le développement de protocoles, l'éducation et la rétro-ingénierie, ainsi le piratage. C'est l'analyseur réseau le plus populaire du monde. Cet outil extrêmement puissant fournit des informations sur des protocoles réseaux et applicatifs à partir de données capturées sur un réseau.

L'utilisation de Wireshark dans notre projet est pour la détection des vulnérabilités dans le réseau VoIP. Nous essayerons de capturer les paquets qui circulent pour déterminer quelques informations, telles que les adresses IP, les numéros de ports, et d'autres informations qui servent au piratage (vol d'identité, déni de service, etc.). Ainsi que nous pouvons écouter une communication entre deux clients, en décodant les paquets RTP (écoute clandestine).

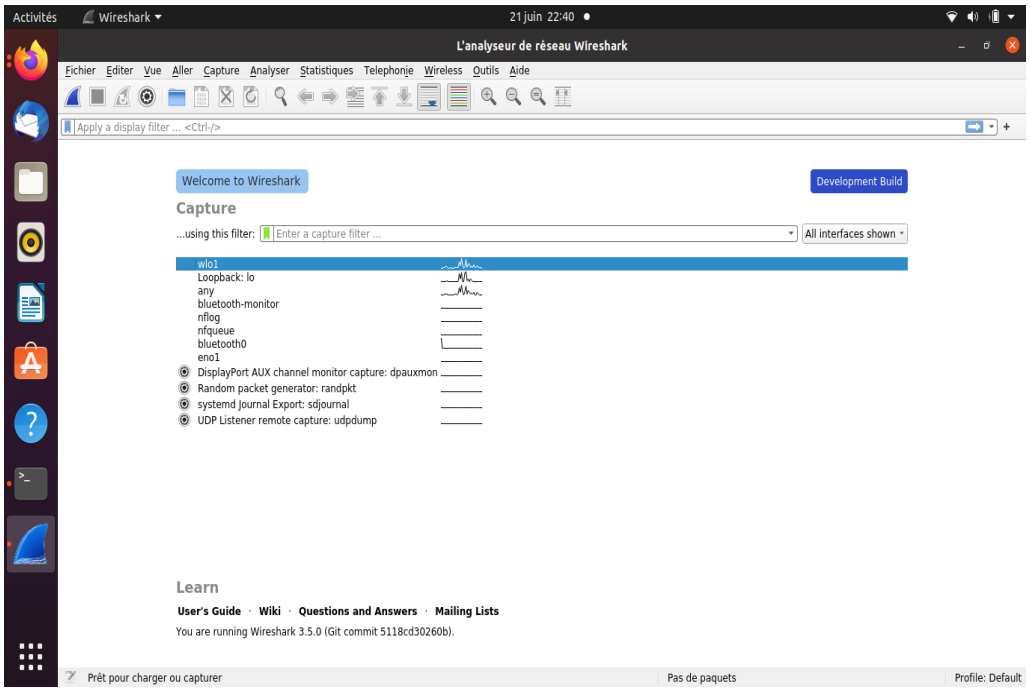

Figure 27 : Ecran d'accueil de Wireshark.

3.2. Captures de trames

Nous avons placé Wireshark dans une 3ème machine, qui va jouer le rôle de l'attaquant. Elle va sniffer tout le trafic circulant dans notre réseau local. Nous avons lancé au début la capture des trames ensuite on a initialisé une connexion entre deux clients, « hakim » et « zak ». On obtient ce résultat :

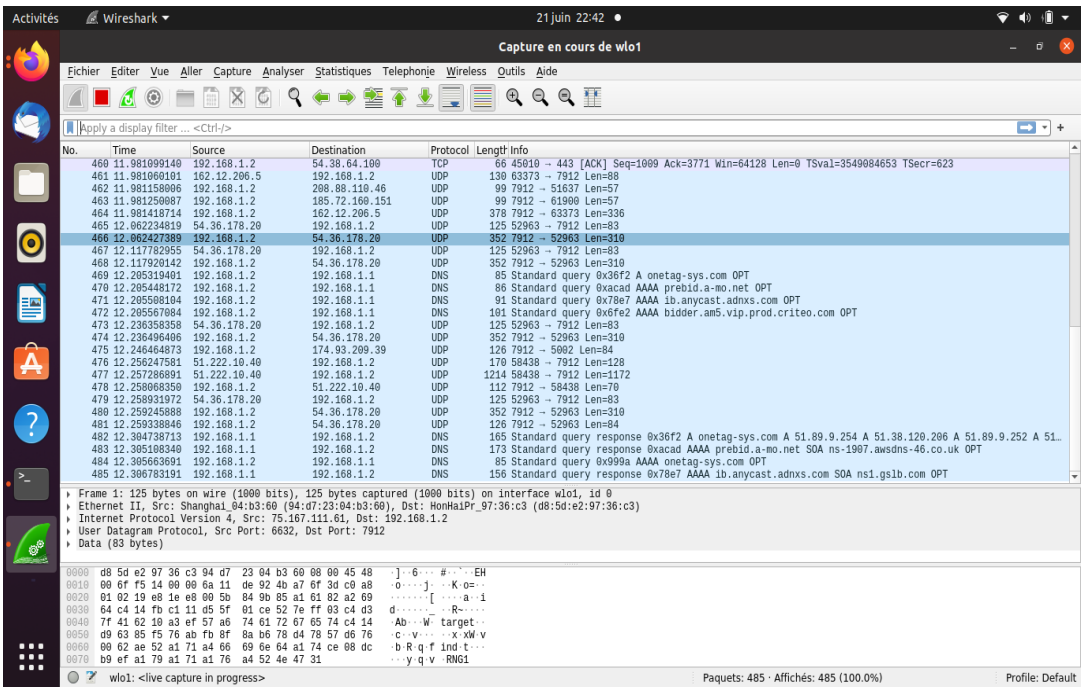

Figure 28 : Ecran de Wireshark.

Comme nous pouvons le voir dans la figure 27, la conversation entre ces deux hôtes à été capturée. La fenêtre principale de Wireshark comprend deux grandes parties. Dans la première partie, nous voyons les différentes étapes de connexion entre les deux clients. Dans la deuxième partie, qui est la plus intéressante, nous pouvons lire le contenu des paquêts et donc collecter des informations très indispensables pour effectuer une bonne attaque.

3.3. Démonstration de l'attaque Clandestine avec Wireshark

Nous utilisons Wireshark dans cette sous-section pour conduire l'attaque d'écoute clandestine. Cette attaque consiste à capturer les trames circulantes entre deux machines effectuant une conversation VoIP, et décoder par la suite les paquêts afin d'écouter la conversation effectuée.

Nous allons maintenant procéder au décodage de l'appel. Dans le menu de Wireshark, nous cliquons sur le bouton « Téléphonie », puis sur le bouton « appels VoIP ».

Une deuxième fenêtre s'ouvre contenant les communications :

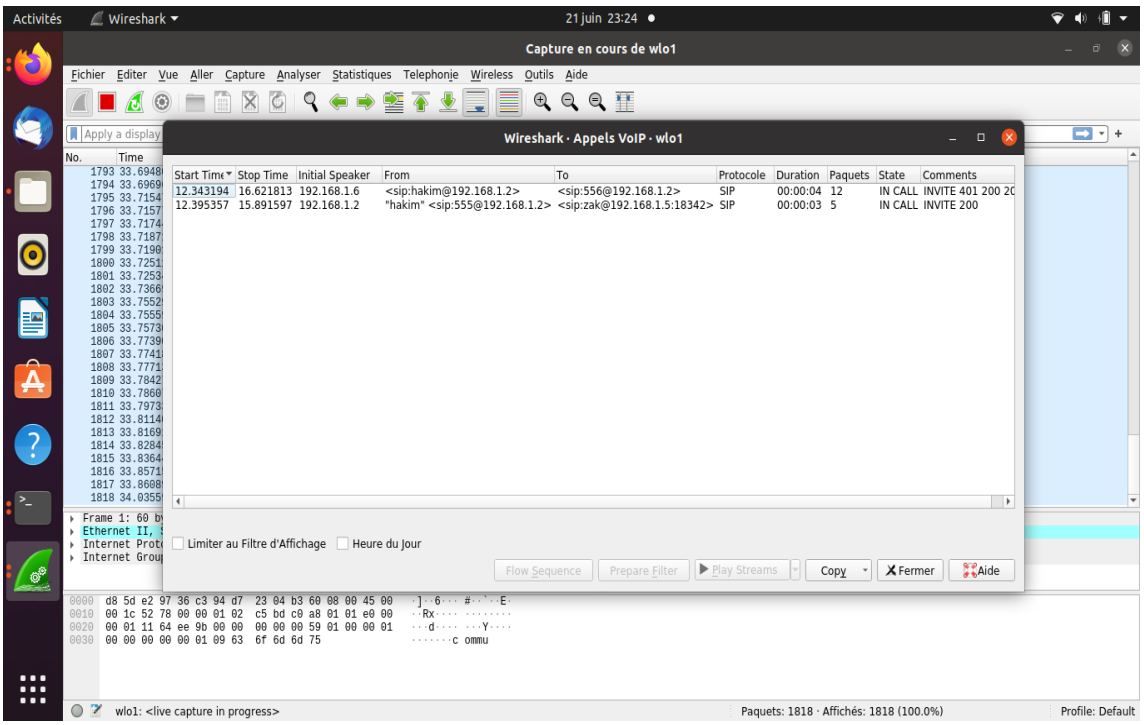

Figure 29 : Communication téléphonique détectée.

Nous cliquons sur le bouton « Player », une fenêtre « RTP Player » s'ouvre pour le décodage nous cliquons sur « Decode » et nous obtenons ceci :

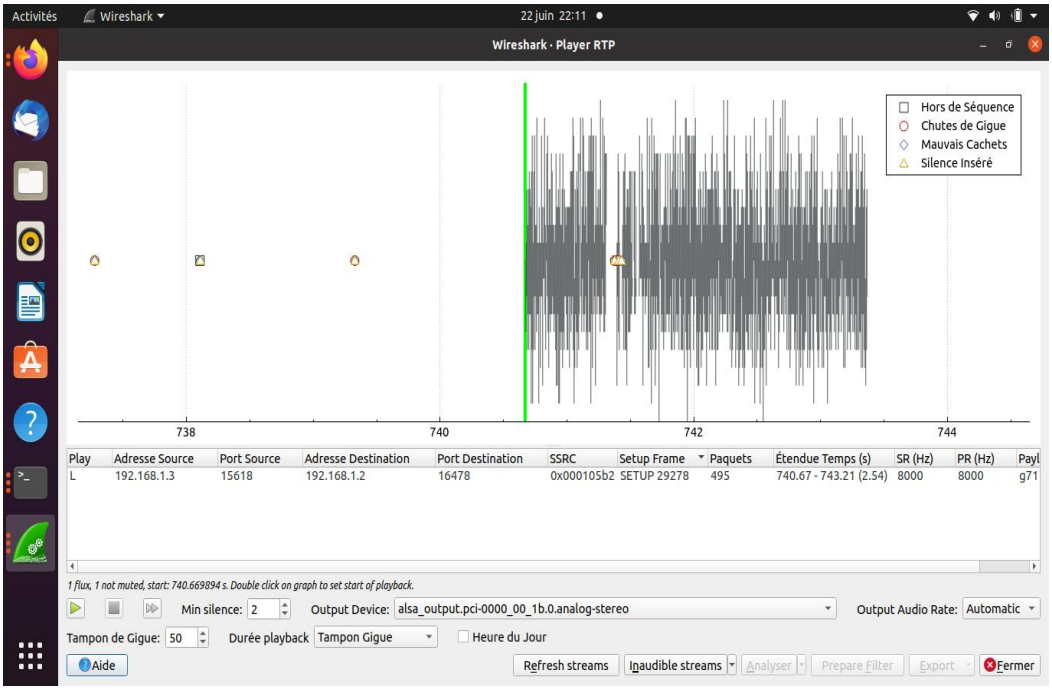

Figure 30 : Communication décodée (RTP Player).

Maintenant que le décodage à abouti nous pouvons aussi voir la figure 31 que le son est décodé et qu'il est prêt à être écouté en appuyant sur le bouton « Play ».

4. Choix et implémentation des bonnes pratiques

Pour se protéger contre des attaques, nous avons choisi un ensemble de solutions qui peuvent aider à minimiser les menaces.

4.1. Chiffrement des appels

La sécurisation des appels peut être intéressante à mettre en place, afin de protéger nos appels téléphoniques. Nous pouvons chiffrer le flux de signalisation ainsi que le flux audio. De cette manière, nous pouvons assurer la confidentialité des appels.

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

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

4.1.1. Chiffrement SIP avec TLS

Commençons donc par le protocole SIP

Pour cela, nous allons faire appel au protocole TLS. Tout d'abord, il va nous falloir créer des clés pour Asterisk et les clients profitant du chiffrement. Ensuite, nous devrons autoriser Asterisk à utiliser TLS pour les échanges SIP. Puis nous devrons choisir les clients à sécuriser. Enfin, il faudra activer le chiffrement sur le poste de client et lui fournir les fichiers de clé.

- Commençons donc par générer les clés. Pour cela, nous créons un dossier qui contiendra les clés :

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

- Le script permettant de créer les clés se trouvant dans le dossier suivant :

```
cd  /downloads/asterisk/asterisk-18.3.0/contrib/scripts/
```

- Le script s'exécute comme ceci :

```
./ast_tls_cert -C asterisk.networklab.com -O « NetworkLab » -d /etc/asterisk/keys
```

Voici le détail des options :

- **C :** permet de spécifier le nom d'hôte du serveur Asterisk. À défaut d'un nom, vous pouvez spécifier un SIP ;
- **O :** permet de définir le nom de l'organisation ;
- **d :** permet de spécifier le dossier de sortie.

À l'exécution du script, il vous sera demandé de spécifier le mot de passe des fichiers.

Retenez bien le mot de passe entré pour chaque fichier, pour utiliser le même à chaque fois.

- Les fichiers suivants devraient être crées après avoir taper cette commande :

```
ls /etc/asterisk/keys/
```

Les fichiers sont **: {asterisk.crt, asterisk.csr, asterisk.key, asterisk.pem ca.cfg, ca.crt, ca.key, tmp.cfg}**

Nous avons donc un certificat auto-signé pour l'autorité de certificat, et un certificat pour le serveur Asterisk. Nous avons aussi une clé privée pour l'autorité de certificat et une pour Asterisk.

Les fichiers PEM (pour Pain Explosif Malléable) regroupent la clé privée et le certificat.

Les fichiers CSR (pour Corporate Social Responsibility) sont des fichiers de requête de certificat (nous n'en avons pas besoin)

- Désormais, nous devons créer les clés de certificats pour les clients :

```
. /ast_tls_cert -m client -c /etc/asterisk/keys/ca.crt -K /etc/asterisk/keys/ca.key -C phone101.networklab.com -O « NetworkLab » -d /etc/astyerisk/keys -o 101
```

Voici le détail des options :

- **m :** indique qu'il faut créer un certificat client ;
- **C :** permet de spécifier le chemin vers le certificat de l'autorité de certificat ;
- **K :** permet de spécifier le chemin vers la clé privée de l'autorité de certificat ;
- **c :** permet de spécifier le nom d'hôte du poste du client. Il est possible de spécifier un IP.
- **O :** permet de définir le nom de l'organisation ;
- **d :** permet de spécifier le dossier de sortie ;
- **o :** permet de choisir le nom de la clé à créer.

- Nous retrouvons 4 nouveaux fichiers après avoir tapé cette commande :

```
ls /etc/asterisk/keys/
```

Les fichiers sont : **{101.crt, 101.csr, 101.key, 101.pem**

asterisk.crt, asterisk.csr, asterisk.key, asterisk.pem

ca.cfg, ca.crt, ca.key, tmp.cfg}

L'opération est à répéter pour tous les clients ayant bénéficier de TLS

À partir de cela, nous devons configurer Asterisk pour autoriser l'utilisation de TLS.

Dans le fichier **sip.conf,** on doit apporter les modifications suivantes :

```
[general]
tlsenable=yes
tlsbindaddr=0.0.0.0
tlscertfile= /etc/asterisk/keys/asterisk.pem
tlscafile=/etc/asterisk/keys/ca.crt
tlscipher=All
tlsclientmethod=tlsv1
```

Ensuite, nous devons autoriser les clients à utiliser TLS.

Dans le fichier **user.conf**, on ajoute la ligne 'transport=tls' pour tous les clients concernés :

```
[hakim]
fullname=hakim
username=hakim
secret=toto
mailbox=555
context=dept_1
transport=tls
```

On n'oublie pas de relancer Asterisk une fois la configuration est terminée :

sudo asterisk -rvvvv

Et enfin, nous pouvons configurer les postes des clients.

Pour cette démonstration, nous avons choisi d'utiliser le logiciel MizuDroid.

Le client doit posséder à enlever deux fichiers :

ca.crt **client.pem**
client 555 : ca.crt **555.pem**

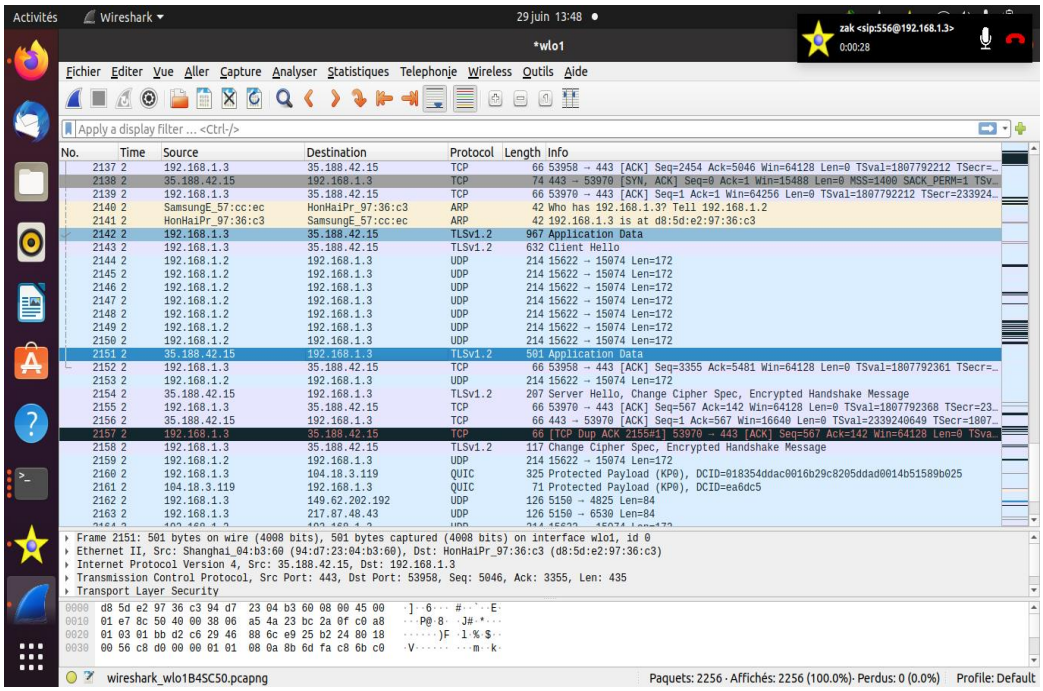

Figure 31 : Présentation du TLS sur Wireshark.

4.1.2. Chiffrement RTP avec SRTP

Une fois que la signalisation est chiffrée, il nous faut encore chiffrer le flux audio. Pour cela, nous allons utiliser le protocole SRTP.

La première étape, consiste à ajouter le support de SRTP à Asterisk. Commencons par télécharger la librairie SRTP avec les commandes suivantes :

```
cd /usr/src
sudo -O srtp-1.4.4.tgz http://sourceforge.net/projects/srtp/files/srtp/1.4.4/rtp-1.4.4.tgz/downloads
sudo tar xvzf srtp-1.4.4.tgz
rm srtp-1.4.4.tgz
```

Procédons à l'instalation:

```
cd /srtp
./configure –prefix=/usr CFLAGS = -fPIC -wall -04 -fexpensive-optimizations -funroll-loops
make
make install
```

Ensuite il faut reconfigurer l'asterisk pour ajouter le protocole SRTP

```
cd  /downloads/asterisk/asterisk-18.4.0
make clean
./configure
make
make install
```

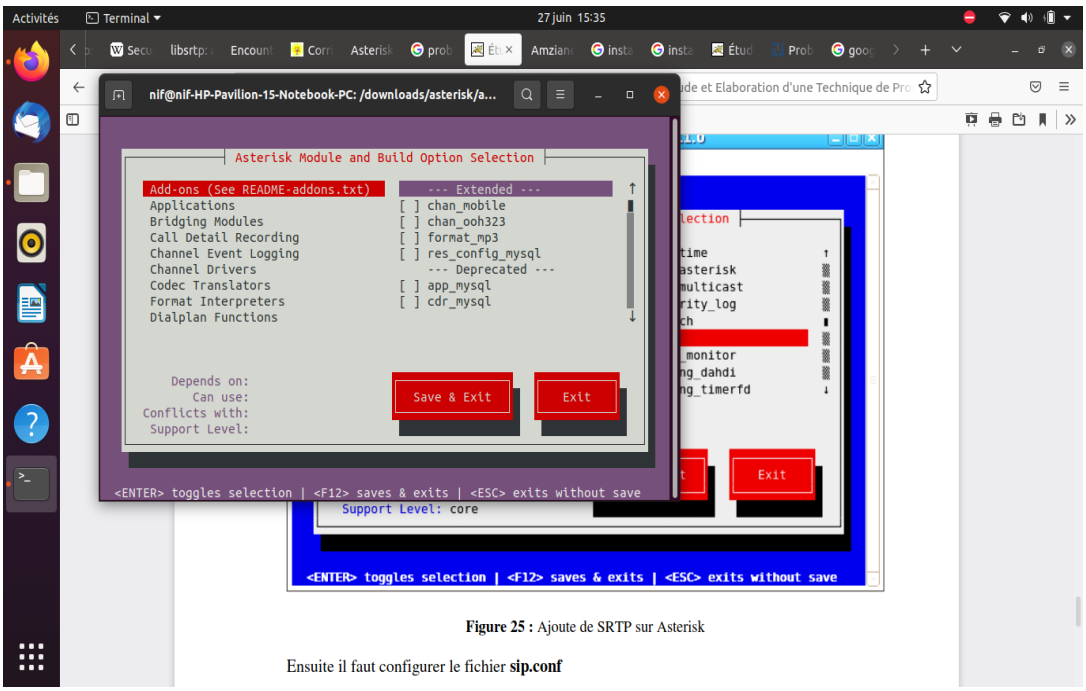

Figure 32 : Ajout de SRTP sur Asterisk.

Pour qu'Asterisk prenne en charge SRTP, il nous faut le redémarrer :

Sudo asterisk -rvvvv **module show like res_srtp.so**

Comme illustré dans la figure suivante :

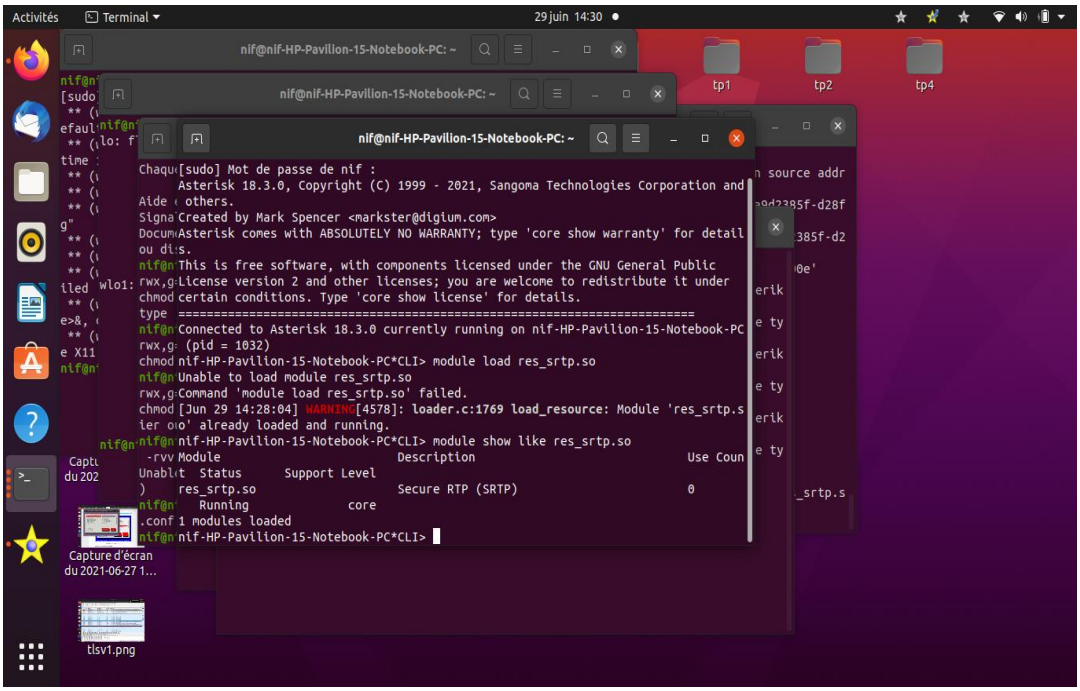

Figure 33 : Le SRTP est ajouté.

À ce niveau, il faut forcer l'utilisation de SRTP sur les clients voulus. Pour cela, on va ajouter la ligne 'encryption=yes' chez les utilisateurs concernés dans **sip.conf.**

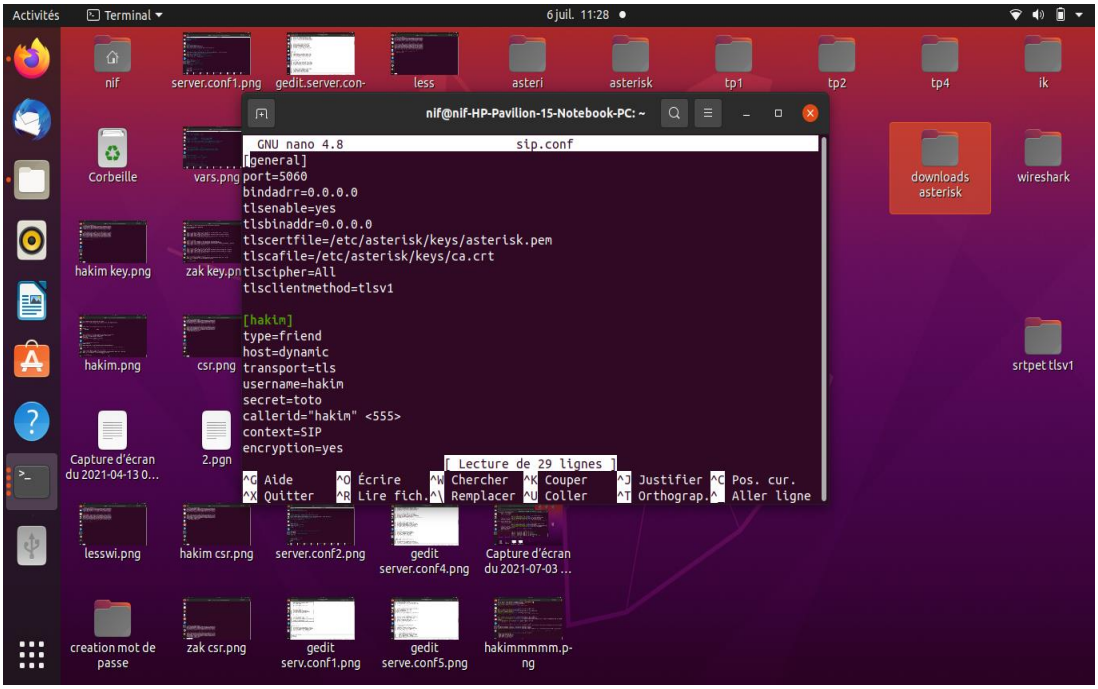

Figure 34 : Ajout de la ligne « encryption=yes ».

À la fin on va relancer Asterisk une fois la configuration est complète.

Nous pouvons à présent configurer le poste du client pour qu'il utilise SRTP.

Pour vérifier notre travail, nous abordons à nouveau en utilisation Wireshark. Lors de l'écoute on n'entend rien du tout.

4.2. Autres solutions de sécurisation

4.2.1. Implémentation d'un Firewall

Dans le cadre de notre projet, le Firewall va nous permettre de minimiser le trafic entrant au serveur Asterisk est cela pour limiter les attaques de types DoS.

En effet, notre projet est de ne laisser passer que le trafic VoIP et plus exactement les paquets basés sur le protocole SIP et le protocole RTP, qui sont utilisés par notre serveur Asterisk pour le trafic VoIP.

- **UFW (Pare-Feu simplifié)**

Le pare-feu C'est un logiciel ou un micrologiciel incorporé dans une grande variété de périphériques en réseau qui filtre le trafic et réduit les risques que des logiciels malveillants puissent avoir une incidence néfaste sur la sécurité d'un réseau privé.

L'outil de configuration de pare-feu par défaut pour Ubuntu est UFW (pour Uncomplicated FirewallWall). Développé pour faciliter la configuration de pare-feu iptables, UFW fournit un moyen convival de créer un pare-feu basé sur l'hôte Ipv4 et Ipv6.

Bien qu'Iptables soit un outil solide et fléxible, UFW peut parfois être difficile pour les débutants d'apprendre à l'utiliser pour configurer correctement un pare-feu. Si un utilisateur souhaite commencer à sécuriser son réseau, UFW peut être la solution appropriée.

Le Iptables est un logiciel libre de l'espace utilisateur Linux grâce auquel l'administrateur système peut configurer les chaîne et règles dans le pare-feu en espace noyau.

Voici les étapes d'installation, configuration et d'utilisation d'UFW :

Étape 1 : Configuration les stratégies par défaut

Pour installer UFW sur Ubuntu, nous pouvons taper la commande suivante :

```
sudo apt-get install ufw
```

UFW est connu pour refuser toutes les connexions entrantes et autorise toutes les connexions sortantes. Cela signifie qu'un client essayant d'atteindre notre serveur ne pourrait pas se connecter. Lorsqu'une application de notre serveur essaie de se connecter à un autre serveur extérieur, cela sera autorisé.

Les commandes suivantes ont pour but :

```
sudo ufw default deny incoming
sudo ufw default allow outgoing
```

Étape 2 : Autorisation les connexions SSH

Par défaut, nous avons restreint toutes les connexions entrantes à notre serveur comme nous pouvons le voir dans notre étape précédante. Pour autoriser les connexions utilisant SSH sécurisé, nous utiliserons la commande suivante :

```
sudo ufw allow ssh
```

La commande ci-dessus créera des règles de pare-feu qui autoriseront toutes les connexions sur le port 2222, qui est le port par defaut sur lequel le demon SSH l'écoute.
Si le démon SSH est configuré sur un autre port que celui par défaut, nous pouvons le spécifier dans notre commande pour écouter ce port.
La commande suivante écoute le port 2222 au cas où SSH serait configuré.

Nous pouvons également spécifier le protocole (TCP ou UDP) dans notre commande ci-dessous.

La commande ci-dessous est utilisée pour les deux protocoles.

```
sudo ufw allow 2222
```

Étape 3 : Autorisation des connexions entrantes spécifiques

Pour autoriser les connexions entrantes sur un port spécifique, nous utiliserons les commandes suivantes pour spécifier la règle pour UFW. Par exemple, si nous voulons que notre serveur écoute http sur le port 80, voici la commande à exécuter :

```
sudo ufw allow http
```

Ce qui signifie :

```
sudo ufw allow 80
```

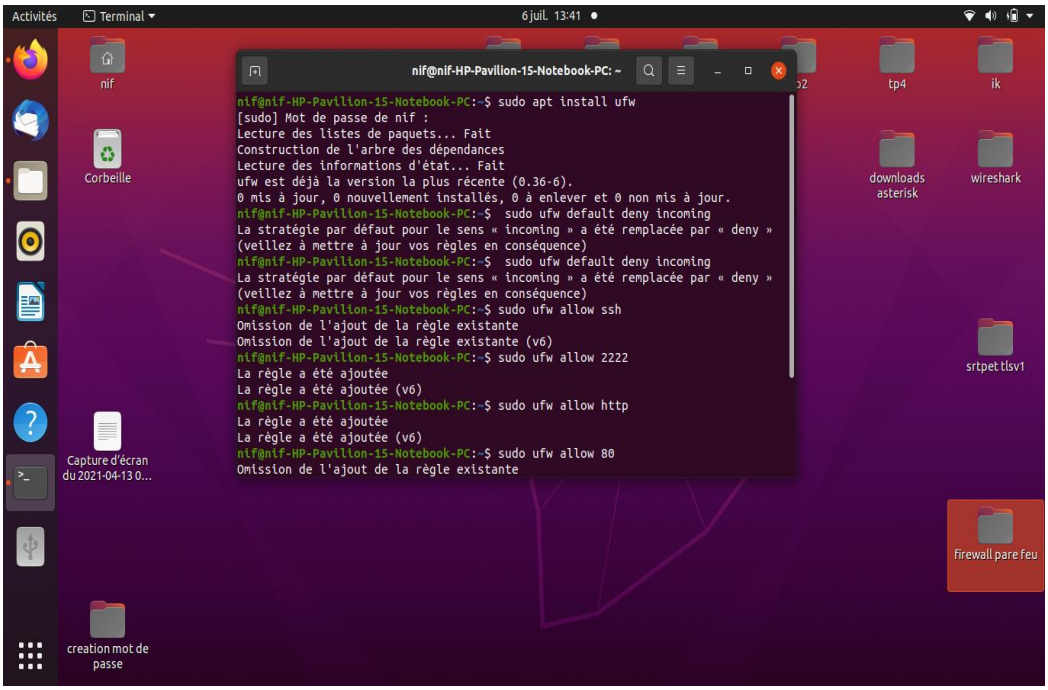

Figure 35 : Installation et configuration d'UFW (1).

Nous pouvons utiliser l'un des éléments ci-dessus pour le port 80. Pour HTTPS, l'une des commandes suivantes servira à autoriser la connexion :

sudo ufw allow https

Ou :

sudo ufw allow 443

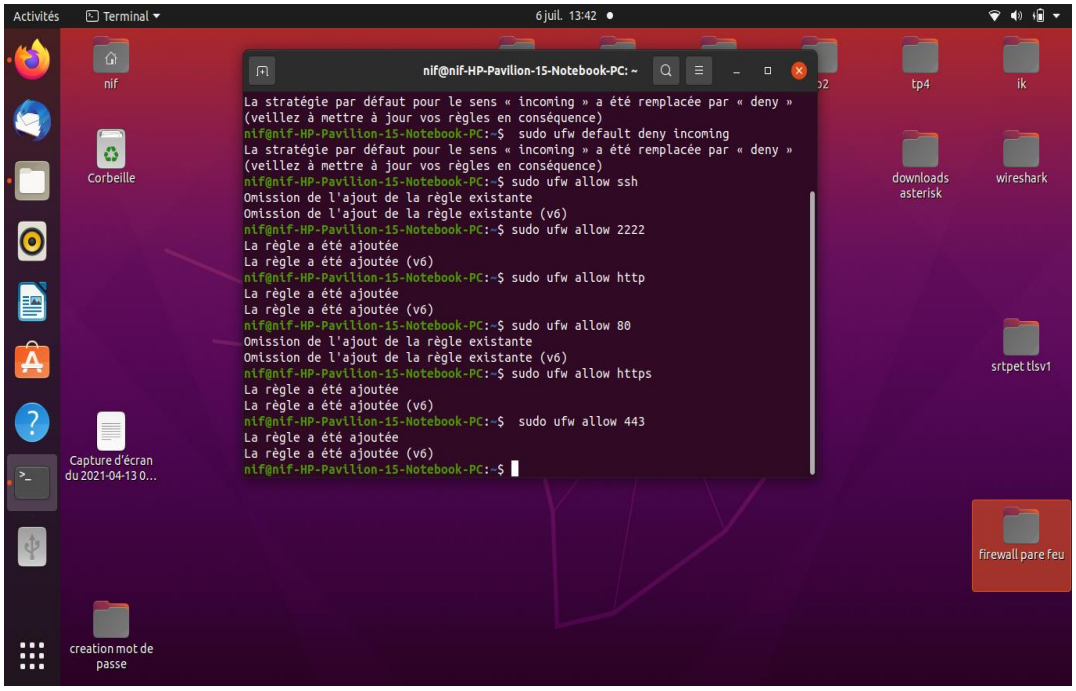

Figure 36 : Installation et configuration d'UFW (2).

Nous pouvons également spécifier une plage de port, ce qui signifie plus d'un port. Une chose à noter est que nous devons spécifier le protocole dans la commande (tcp ou udp).

La commande suivante autorise les connexions des ports 6000 à 6003 pour tcp et udp :

```
sudo ufw allow 6000 :6003/tcp
sudo ufw allow 6000 :6003/udp
```

Et pour ainsi pour les ports 5060 à 6003 :

```
sudo ufw allow 5060 :6003/tcp
sudo ufw allow 5060 :6003/udp
```

Étape 4 : Refus des connexions entrantes

Parfois, nous voulons refuser des connexions spécifiques en fonction de l'adresse IP Source.

C'est parce que nous savons parfois que notre serveur est attaqué à partir de là.

Nous allons donc créer une règle de refuse pour l'adresse IP spécifique.

La commande suivante refuse la connexion à partir d'un adresse IP 203.0.123.5 :

sudo ufw deny from 203.1.123.5

Étape 5: Activation d'UFW

Après toutes les configurations UFW, l'étape suivante consiste à l'activer.

sudo ufw enable

Étape 6 : Vérification l'état de l'UFW

Nous pouvons vérifier l'état avec la commande suivante :

sudo ufw satatus verbose

Ce qui suit est le résultat de la commande précédante lorsqu'elle est inactive :

status : inactive

Dans notre cas, il est actif, ce qui suit sera la sortie et les résultats :

status : active

To	**Action**	**From**
--	----------	----------
22	Allow IN	Anywhere
80	Allow IN	Anywhere
443	Allow IN	Anywhere
2222	Allow IN	Anywhere
6000 :6003/tcp	Allow IN	Anywhere
6000 :6003/udp	Allow IN	Anywhere
Anywhere	Deny	203.0.123.5

Comme illustré dans les figures suivantes :

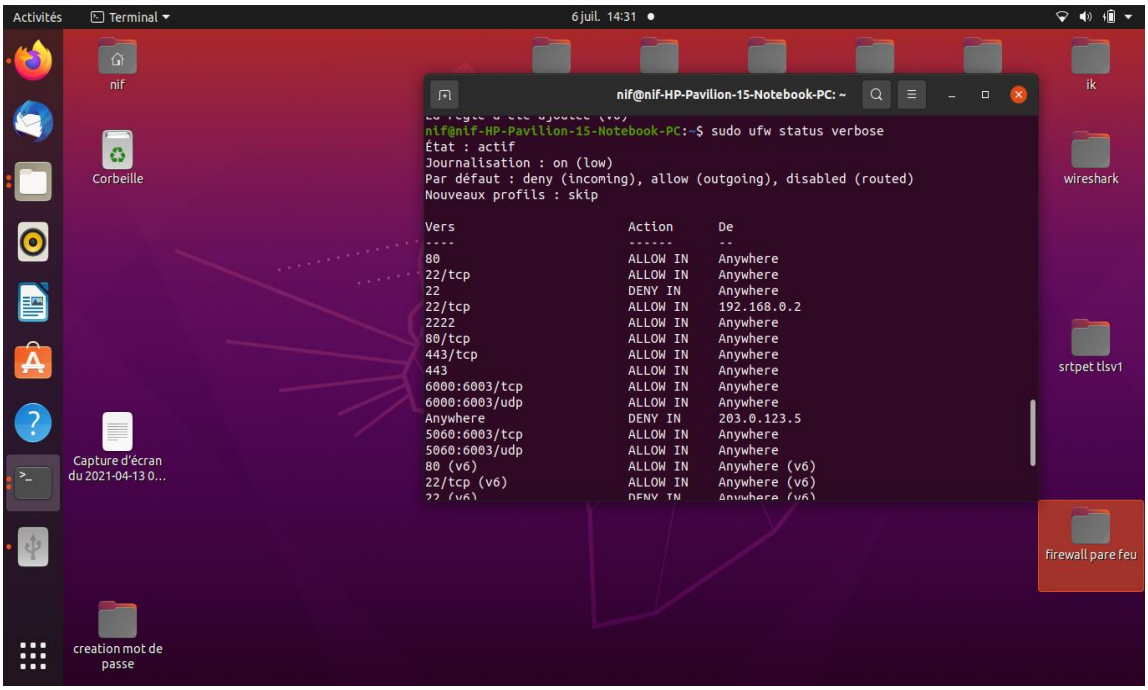

Figure 37 : Vérification de l'état de l'UFW (1).

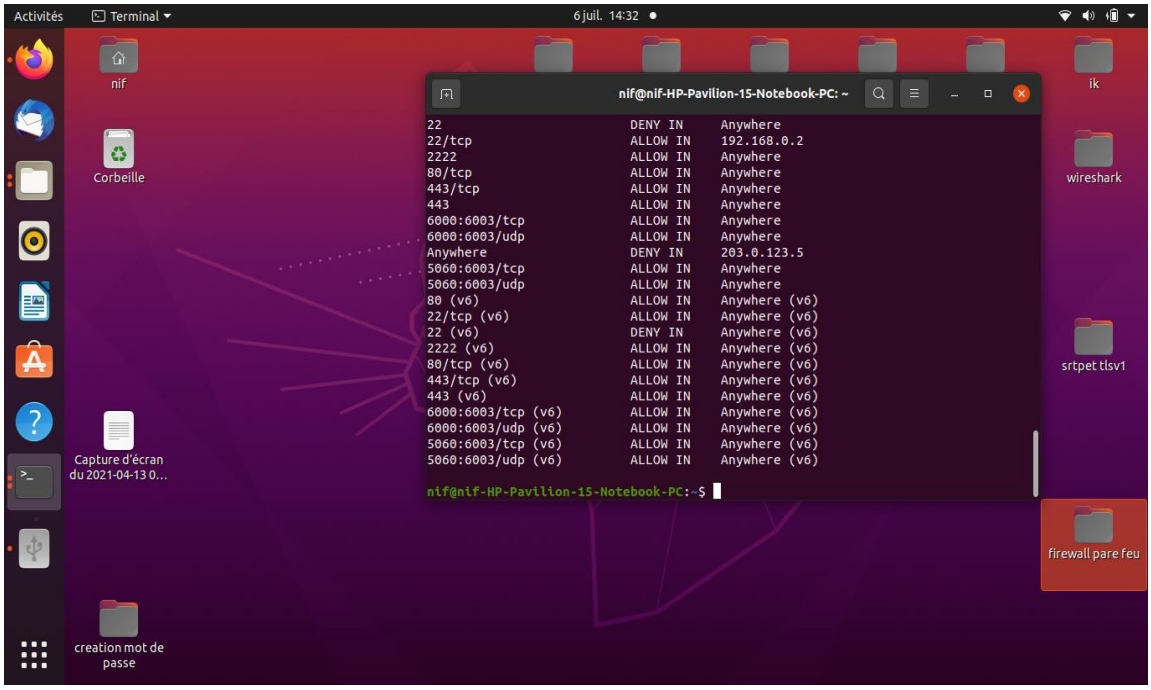

Figure 38 : Vérification de l'état de l'UFW (2).

Conclusion

Dans ce chapitre, nous avons pu exploiter les failles de sécurité existantes dans notre infrastructure, en simulant des attaques connus sur la VoIP. Nous avons ensuite mis en place des politiques de sécurité qui diminueront l'impact des vulnérabilités et offrant un environnement plus protégé pour les clients SIP. Mais il faut savoir qu'il est impossible d'avoir une sécurité parfaite au niveau du réseau VoIP et généralement sur tous les réseaux.

Conclusion générale

Les avantages apportés par les services de téléphonie IP en termes de coûts, de facilité d'utilisation, d'extension et de maintenance, réduisent de plus en plus d'utilisateurs, aussi bien d'entreprises que particuliers.

Toutefois, le système de téléphonie IP est confronté à des contraintes liées essentiellement à la sécurité et la qualité de service, qui nuisent au bon fonctionnement de ses services. En termes de sécurité, le système de téléphonie sur IP est ouvert à une variété d'attaques allant de déni de service jusqu'au vol d'identité. Plusieurs solutions et outils de sécurité existent, et d'autres mécanismes assurant la défense contre les vulnérabilités. L'enjeu sera de savoir bien combiner ces mécanismes ensemble afin de dégager une politique de sécurité assez robuste et fiable.

De ce fait, l'objectif principal de notre travail est de sécuriser un réseau VoIP, tout en mettant en place d'une solution sécurisée pour la transmission de la voix sur un réseau IP.

À travers ce projet de recherche, au premier lieu, nous avons présenté les concepts généraux de la voix sur IP, notamment ses protocoles de communication.

Au deuxième lieu, nous avons passé revue sur l'ensemble de points faibles de cette technologie contre les différentes attaques possibles sur les différents niveaux.

Au troisième lieu, nous avons d'abord fait une petite présentation sur Asterisk avec ses fonctionnalités et architectures, ensuite nous avons passé à l'installation, configuration et compilation d'Asterisk.

Pour conclure, une grande partie a été réservée aux bonnes techniques couramment utilisées pour assurer la sécurité de la voix sur IP. D'où, nous avons testé des attaques pour la sécuriser.

Ce projet a été une expérience fructueuse, qui nous a permis de mieux s'approcher du milieu professionnel, notamment les administrations et les entreprises, comme elle nous a permis de savoir comment mettre en place des outils performants pour mieux exploiter avec sécurité le réseau IP dans l'entreprise pour la transmission de la voix

Conclusion générale

Comme perspective de recherche, le test de notre solution proposée sur d'autres types d'attaques est possible pour montrer son efficacité. De plus, l'implémentation d'autres techniques de sécurité de la voix sur IP afin d'augmenter le degré de protection de la transmission de la voix via le réseau IP peut être parmi des travaux futurs.

Résumé

La VoIP (Voix sur IP) constitue actuellement l'évolution la plus importante du domaine de la télécommunication. En effet, la technologie VoIP commence à intéresser les entreprises, surtout celles de services comme les centres d'appels. Cependant, Certaines attaques sur les réseaux VoIP, comme les attaques de déni de service, peuvent causer des pertes catastrophiques et énormes pour les entreprises. Pour cela, la sécurité du réseau VoIP n'est pas seulement une nécessité mais plutôt une obligation, avec laquelle on peut réduire, au maximum, le risque d'attaques sur les réseaux VoIP. Dans ce travail, nous avons mis en place deux mécanismes de sécurité pour la voix IP à savoir : chiffrement des appels dans lequel nous avons réalisé TLS (Transport Layer Security) à l'aide du protocole SIP (Session Initiation Protocol : l'extension du protocole H.323) et SRTP (Secure Real-time Transport Protocol) à l'aide du protocole RTP (Real-time Transport Protocol) ; et implémentation d'un Pare-Feu simplifié (UFW: Uncomplicates FireWall). Les résultats de notre configuration ont montré que les mécanismes proposés garantissent une bonne voie et une signalisation, ainsi qu'une minimisation des attaques contre la VoIP.

Mots clés : VoIP, Protocoles de sécurité (SIP, RTP, TLS, SRTP, H.323), Signalisation, Voix, Protocole d'internet (IP), UFW.

Abstract

VoIP (Voice over Internet Protocol) is currently the most important development in the field of telecommunications. Indeed, VoIP technology is starting to interest businesses, especially those providing services such as call centers. However, some attacks on VoIP networks, such as denial of service attacks, can cause catastrophic and hugs losses for businesses. For this, the security of VoIP network is not only a necessity but rather an obligation, with which we can reduce, as much as possible the risk of attacks on VoIP networks, we have set up two security mechanisms for IP voice namely: call encryption in which we have carried out TLS (Transport Layer Security) using SIP protocol (Session Initialization Protocol, extension of the H.323), and SRTP (Secure Real-time Transport Protocol) using RTP protocol (Real-Time Transport Protocol); and Implementation of a Firewall (UFW: Uncomplicates Firewall or Firewall-Simplified). The results of our configuration showed that the proposed mechanisms guaranteeing a good voice and signage, as well as a minimization of attacks against VoIP.

Key words: VoIP, Security protocols (SIP, RTP, TLS, SRTP, H.323), Voice, Signage, Internet Protocol (IP), UFW.

Résumé