Étude et Mise en place d'une Solution VOIP Sécurisée

Réalisé par :

Mlle. Rebha Bouzaida

MEMOIRE DE PROJET DE FIN D'ÉTUDES

Pour l'obtention du Master Professionnel
En Nouvelles Technologies
Des Télécommunications et Réseaux

Encadré par :

Mr. Kamel Kedhiri

Année Universitaire : 2010 – 2011

La vie n'est qu'un éclair,

Et un jour de réussite est un jour très cher.

A mon cher père,

et ma chère mère.

Pour l'éducation et le grand amour dont ils m'ont entouré depuis ma naissance.

Et pour leurs patiences et leurs sacrifices.

A mon cher frère ;

A tous mes proches ;

A tous ceux qui m'aiment ;

A tout mes ami(e)s ;

A tous ceux que j'aime.

Je dédie ce mémoire.

Rebha Bouzaida✍

Remerciements

Au terme de ce projet de fin d'études, j'adresse mes sincères remerciements à Monsieur Kamel Kedhiri, mon encadreur de l'UVT, pour m'avoir proposé ce projet et pour son encadrement.

Je tiens à remercier également Monsieur Mohamed Louhichi, gérant du centre d'appel New Call, pour son suivi et ses remarques qui m'ont permis de mener à bien ce travail.

Mes remerciements s'adressent également à l'administration et aux professeurs de l'UVT pour les moyens qu'ils ont mis à notre disposition afin d'élaborer ce travail.

Je souhaite exprimer enfin ma gratitude et mes vifs remerciements à ma famille et mes amis pour leurs soutiens.

Pour finir, je remercie les membres du jury qui ont accepté d'évaluer mon projet. Je leurs présentons toute mes gratitudes et mes profonds respects.

Sommaire

Liste des figures

Introduction générale

Depuis quelques années, la technologie VoIP commence à intéresser les entreprises, surtout celles de service comme les centres d'appels. La migration des entreprises vers ce genre de technologie n'est pas pour rien. Le but est principalement est de : minimiser le coût des communications ; utiliser le même réseau pour offrir des services de données, de voix, et d'images ; et simplifier les coûts de configuration et d'assistance.

Plusieurs fournisseurs offrent certaines solutions qui permettent aux entreprises de migrer vers le monde IP. Des constructeurs de PABX tels que Nortel, Siemens, et Alcatel préfèrent la solution de l'intégration progressive de la VoIP en ajoutant des cartes extensions IP. Cette approche facilite l'adoption du téléphone IP surtout dans les grandes sociétés possédant une plateforme classique et voulant bénéficier de la voix sur IP. Mais elle ne permet pas de bénéficier de tous les services et la bonne intégration vers le monde des données.

Le développement des PABXs software, est la solution proposée par des fournisseurs tels que Cisco et Asterisk. Cette approche permet de bénéficier d'une grande flexibilité, d'une très bonne intégration au monde des données et de voix, et surtout d'un prix beaucoup plus intéressant.

Cette solution, qui est totalement basée sur la technologie IP, est donc affectée par 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. Certaines attaques sur les réseaux VoIP, comme les attaques de déni de service, et les vols d'identité, 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.

La sécurité d'une solution de VoIP doit couvrir toute l'infrastructure réseau, incluant les outils et les équipements de gestion des communications et des utilisateurs, le système d'exploitation sur lesquels sont installés ces outils, et les protocoles de signalisation et de transport de données. Il faut même se protéger contre les personnes malveillantes. Mieux on sécurise, moins il y a de risques.

Ce travail a pour objectif : l'étude des protocoles de VoIP et des architectures proposées ; l'étude des vulnérabilités et des attaques de sécurités surs les divers composants d'une infrastructure VoIP dans des réseaux LAN ; et la mise en place une solution de VoIP sécurisée basée sur des outils open source, précisément le serveur Asterisk et le client X-Lite.

Les entreprises, bénéficiant de notre solution, seront capables de mettre en place une plateforme de VoIP assez flexible, peu couteux, et protégée contre les attaques de sécurité de l'intérieur du réseau comme de l'extérieur aussi.

Ce rapport se compose de quatre chapitres. Le premier chapitre introduit la voix sur IP et ces éléments, décrit et explique son architecture et ces protocoles, et énumère les majeurs points forts de cette technologie ainsi que ses faiblesses.

Le deuxième chapitre s'intéresse à la sécurité des infrastructures de Voix sur IP. Il détaille les différents types de vulnérabilités de sécurité partagées en trois classes: vulnérabilités liées aux protocoles, vulnérabilités liées aux infrastructures, et vulnérabilités liées aux systèmes. Les bonnes pratiques et solutions de sécurités à mettre en places pour remédier à ces vulnérabilités, sont aussi définies.

Le troisième chapitre, s'intéresse à la mise en place d'une solution de VoIP pour les entreprises basée sur le serveur Asterisk et le client X-Lite. Les différents pré-requis et les librairies nécessaires sont installés, et les paramètres essentiels sont définis et configurés.

Le dernier chapitre du rapport s'intéresse aux tests et réalisations de quelques attaques sur l'infrastructure de VoIP déployée dans le troisième chapitre. Une implémentation des différentes solutions et mesures nécessaires à la protection contre ces attaques, est réalisée.

Chapitre 1

Etude générale de la voix sur IP

Introduction

La voix sur IP constitue actuellement l'évolution la plus importante du domaine des Télécommunications. 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 à la transmission numérique (TDM). La transmission de la voix sur les réseaux informatiques à commutation de paquets IP constitue aujourd'hui une nouvelle évolution majeure comparable aux précédentes.

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 de 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 (en particulier la voix) 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.

1.2. 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. 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 égale à égale avec l'intelligence répartie à la périphérie. Chacune ayant ses avantages et ses inconvénients.

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/ contrôleur de commutation, appelées Gatekeeper. On retrouve les éléments communs suivants :

- Le routeur : permet d'aiguiller les données et le routage des paquets entre deux réseaux. Certains routeurs permettent de simuler un Gatekeeper grâce à l'ajout de cartes spécialisées supportant les protocoles VoIP.
- La passerelle : permet d'interfacer le réseau commuté et le réseau IP.
- Le PABX : est le commutateur du réseau téléphonique classique. Il permet de faire le lien entre la passerelle ou le routeur, et le réseau téléphonique commuté (RTC). Toutefois, si tout le réseau devient IP, ce matériel devient obsolète.
- 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 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. Les appels peuvent parcourir par le réseau internet comme par un réseau privé. Un terminal utilise des protocoles comme le SIP (Session Initiation Protocol) ou l'un des protocoles propriétaire tel que celui utilisée par Skype.

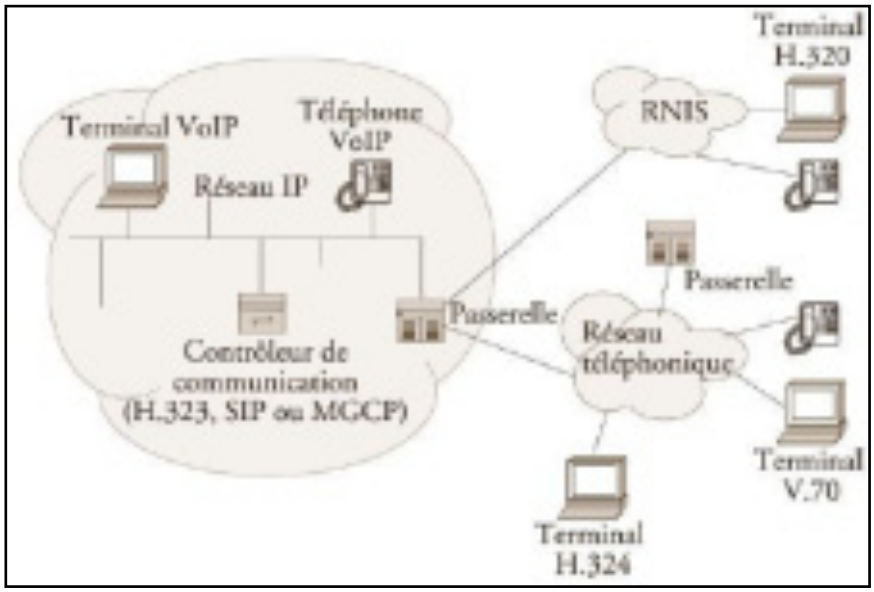

Figure 1 : Architecture générale de la voix sur IP

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 (convertisseur numérique-analogique).

La VoIP fonctionne par numérisation de la voix, 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 en un nouveau format meilleur. Le signal numérique est plus tolérant au bruit que l'analogique.

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

Il existe plusieurs protocoles qui peuvent supporter la voix sur IP tel que le H.323, SIP et MGCP. Les deux protocoles les plus utilisées actuellement dans les solutions VoIP présentes sur le marché sont le H.323 et le SIP.

2. Protocole H.323

2.1 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 (International Telecommunications Union) pour les réseaux qui ne garantissent pas une qualité de service (QoS), tels qu'IP IPX 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 le LAN et les autres réseaux.

Le protocole H.323 fait partie de la série H.32x qui traite de la vidéoconférence au travers différents réseaux. Il inclue H.320 et H.324 liés aux réseaux ISDN (Integrated Service Data Network) et PSTN (Public Switched Telephone Network).

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.

- Les messages de signalisation sont ceux envoyés pour demander la mise en relation de deux clients, qui indique que la ligne est occupée ou que le téléphone sonne, etc. 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.

- 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 du codec le moins gourmand en bande passante ou de celui qui offre la meilleure qualité. Il serait aussi préférable d'avoir plusieurs alternatives de langages. Le protocole utilisé pour la négociation de codec est le H.245

- 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 renégociation des codecs si, par exemple, la bande passante diminue.

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 (Ressource reservation Protocol), l'établissement de la communication 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.

2.2 Rôle des composants

L'infrastructure H.323 repose sur quatre composants principaux : les terminaux, les Gateways, les Gatekeepers, et les MCU (Multipoint Control Units).

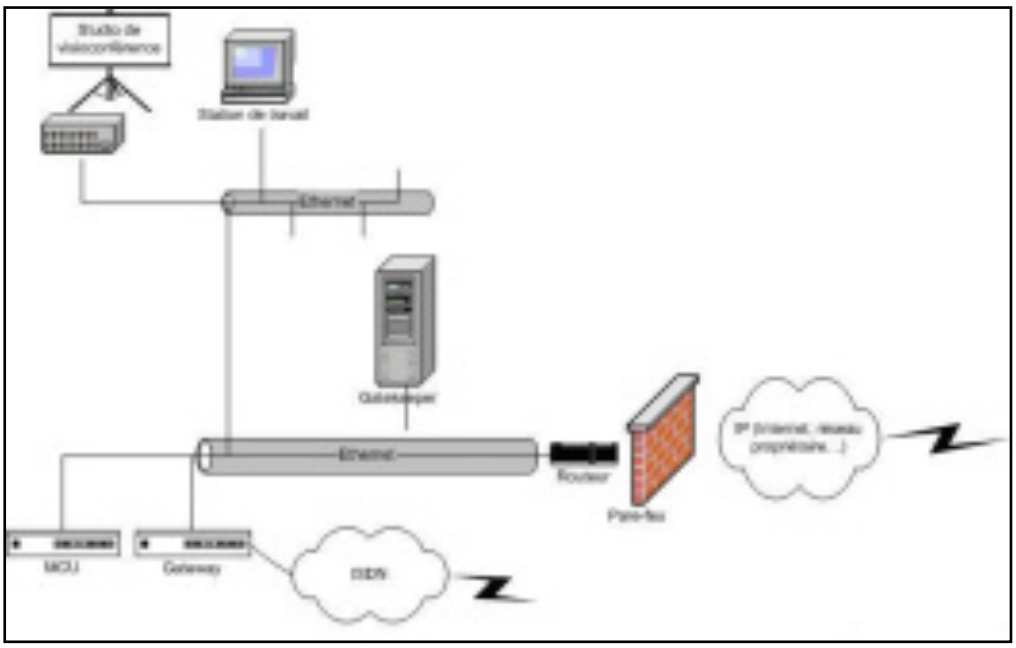

Figure 2 : Les composants de l'architecture H.323

- **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 de petites largeurs de bandes (28,8/33,6 kbit/s – G.723.1 et H.263).

- **Gateway ou les passerelles vers des réseaux classiques (RTC, RNIS, etc.)**

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 des débits, transcodage audio).

- **Gatekeeper ou les portiers**

Dans la norme H323, 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 figure 3 ci-dessous), 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 du Gatekeeper dès l'activation de celui-ci, ce qui leur permet de retrouver n'importe quel autre utilisateur à travers son identifiant fixe obtenu auprès de son Gatekeeper de rattachement.

Le Gatekeeper a pour fonction :

- ✓ La translation des 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 gêner les applications critiques sur le LAN et le support des conférences multipoint adhoc.

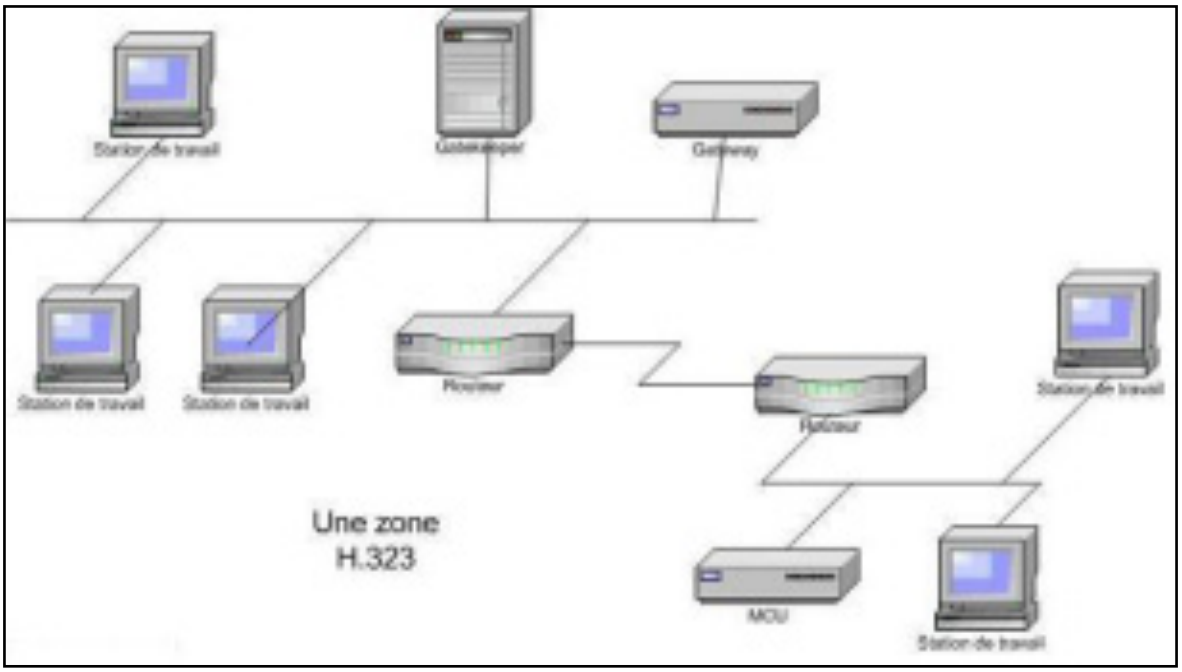

Figure 3 : La zone H.323

- **Les MCU**

Les contrôleurs multipoint appelés MCU (Multipoint Control Unit) offrent aux utilisateurs la possibilité de faire des visioconférences à trois terminaux et plus en « présence continue » ou en « activation à la voix ». Une MCU consiste en un Contrôleur Multipoint (MC), auquel est rajouté un ou plusieurs Processeurs Multipoints (MP). 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 contrôler plusieurs MP distribués sur le réseau et faisant partie d'autres MCU.

2.3 Avantages et inconvénients de la technologie H323

La technologie H.323 possède des avantages et des inconvénients. Parmi les avantages, nous citons :

- ***Gestion de la bande passante*** : 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 Multipoint*** : H.323 permet de faire des conférences multipoint via une structure centralisée de type MCU (Multipoint Control Unit) ou en mode ad-hoc.

- ***Support Multicast*** : H.323 permet également de faire des transmissions en multicast.

- ***Interopérabilité :*** H.323 permet aux utilisateurs de ne pas se préoccuper de la manière dont se font les communications, les paramètres (les codecs, le débit…) sont négociés de manière transparente.

- ***Flexibilité :*** une conférence H.323 peut inclure des terminaux hétérogènes (studio de visioconférence, PC, téléphones…) qui peuvent partager selon le cas, de la voix de la vidéo et même des données grâce aux spécifications T.120.

Les inconvénients de la technologie H.323 sont :

- 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 façon différentes par les constructeurs et donc de poser des problèmes d'interopérabilité.

3. Protocole SIP

3.1 Description générale du protocole SIP

Le protocole SIP (Session Initiation Protocol) est un protocole normalisé et standardisé par l'IETF (décrit par le RFC 3261 qui rend obsolète le RFC 2543, et complété par le 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 de média utilisables par les différents participants en encapsulant des messages SDP (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 (Real-time Transport Protocol) assure le plus souvent les sessions audio et vidéo. SIP remplace progressivement H323.

SIP est le standard ouvert de VoIP, interopérable, le plus étendu et vise à devenir le standard des télécommunications multimédia (son, image, etc.). Skype par exemple, qui utilise un format propriétaire, ne permet pas l'interopérabilité avec un autre réseau de voix sur IP 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.

3.2 Principe de fonctionnement

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

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 voix sur IP 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).

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).

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.

- ***Mode Point à point*** : on parle dans ce cas là d'«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 – Multipoint Control Unit).
- ***Combinatoire :*** combine les deux modes précédents. Plusieurs utilisateurs interconnectés en multicast via un réseau à maillage complet de connexion.

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 mis en attente (cette particularité rejoint les fonctionnalités d'un PABX par exemple, où l'appelant peut être transféré vers un numéro donné ou être mis en attente).

Négociation des médias supportés

Cela 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.

Adressage

Les utilisateurs disposant d'un numéro (compte) SIP disposent d'une adresse ressemblant à une adresse mail (sip:numéro@serveursip.com). Le numéro SIP est unique pour chaque utilisateur.

Modèle d'échange

Le protocole SIP repose sur un modèle Requête/Réponse. Les échanges entre un terminal appelant et un terminal appelé se font par l'intermédiaire de requêtes. La liste des requêtes échangées est la suivante :

- ***Invite*** : cette requête indique que l'application (ou utilisateur) correspondante à l'url SIP spécifié est invité à participer à une session. Le corps du message décrit cette session (par ex : média supportés par l'appelant). En cas de réponse favorable, l'invité doit spécifier les médias qu'il supporte.

- ***Ack :*** cette requête permet de confirmer que le terminal appelant a bien reçu une réponse définitive à une requête Invite.

- ***Options :*** un proxy server en mesure de contacter l'UAS (terminal) 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'appelé à fin de signaler qu'il souhaite mettre un terme à la session.

- ***Cancel :*** cette requête est envoyée par un terminal ou un proxy server à fin 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.

- ***Register :*** cette méthode est utilisée par le client pour enregistrer l'adresse listée dans l'URL TO par le serveur auquel il est relié.

Codes d'erreurs

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 du 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 de codes d'état, représentées par le premier digit :

- 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 réussi à traiter une requête apparemment correcte.
- 6xx = Echec général - La requête ne peut être traitée par aucun serveur.

3.3 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** (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.

L'**U.A.C** (User Agent Client) représente l'agent de la partie appelante. C'est une application de type client qui initie les requêtes.

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 (figure 4).

Les **URI SIP** sont très similaires dans leur forme à des adresses email : sip:utilisateur@domaine.com. Généralement, des mécanismes d'authentification permettent d'éviter que quiconque puisse s'enregistrer avec n'importe quelle URI.

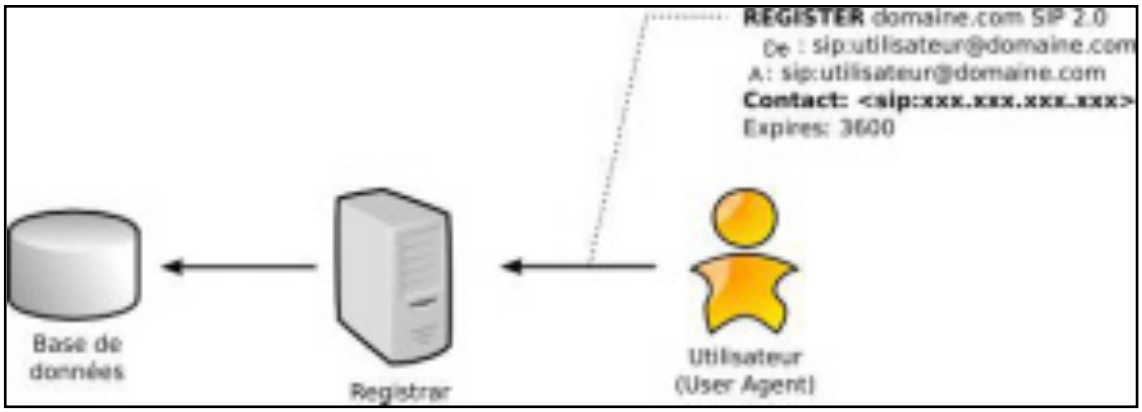

Figure 4 : Enregistrement d'un utilisateur

Un **Proxy SIP** sert d'être l'intermédiaire entre deux User Agents qui ne connaissent pas leurs emplacements respectifs (adresse IP). En effet, l'association URI-Adresse IP a été stockée préalablement dans une base de données par un Registrar. Le Proxy peut donc interroger cette base de données pour diriger les messages vers le destinataire. La figure 5 montre les étapes de l'interrogation du proxy la base de données

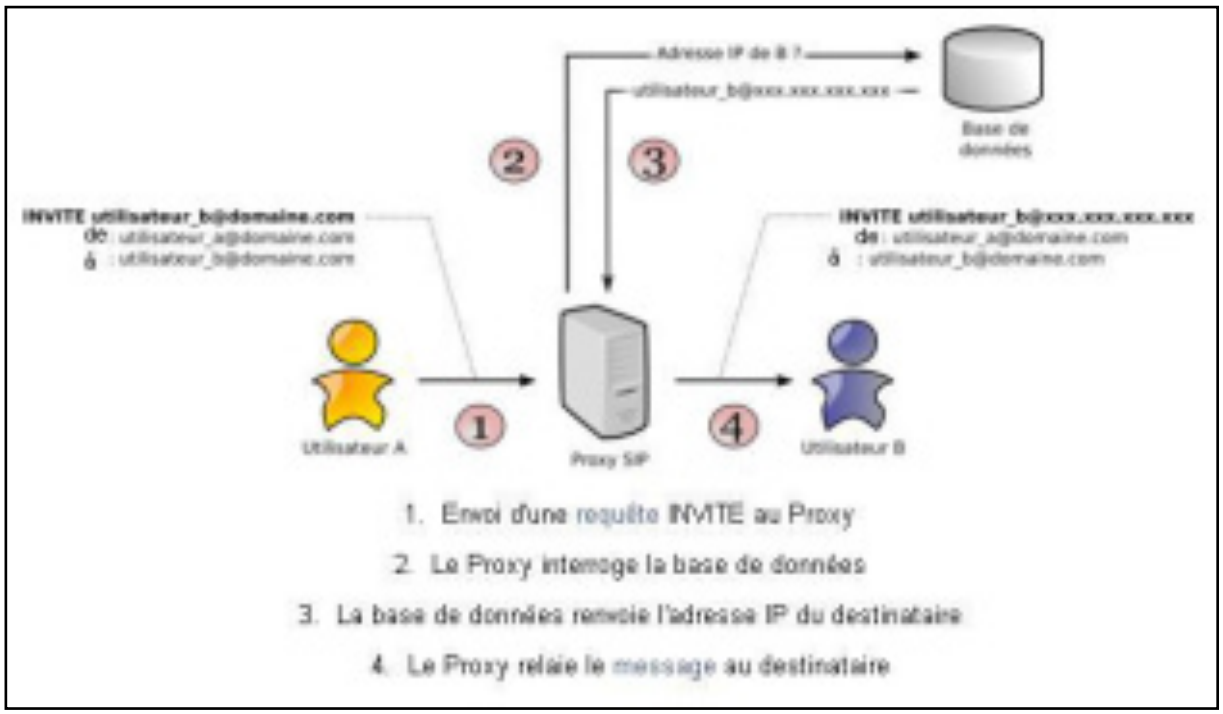

Figure 5 : Principe du protocole SIP

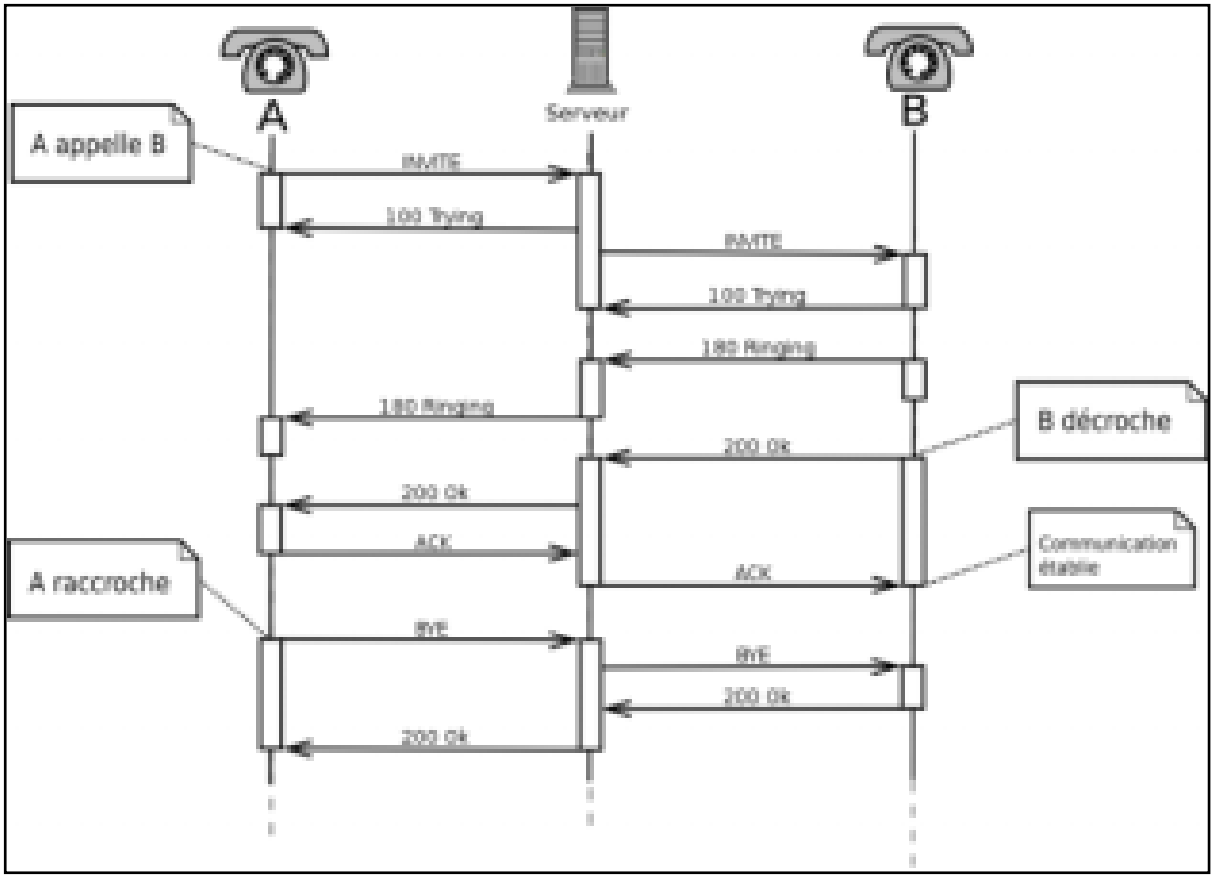

Figure 6 : Session SIP à travers un proxy

Le Proxy se contente de relayer uniquement les messages SIP pour établir, contrôler et terminer la session (voir figure 6). 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.

3.4 Avantages et inconvénients

Ouvert, standard, simple et flexible sont les principales atouts du protocole SIP, voilà en détails ces différents avantages :

- Ouvert : les protocoles et documents officiels sont détaillés et accessibles à tous en téléchargement.
- Standard : l'IETF a 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, etc.).
- Téléphonie sur réseaux publics : il existe de nombreuses passerelles (services payants) vers le réseau public de téléphonie (RTC, GSM, etc.) permettant d'émettre ou de recevoir des appels vocaux.
- Points communs avec H323 : l'utilisation du protocole RTP et quelques codecs son et vidéo 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 et utilisé par le grand public, n'ayant pas atteint une masse critique, il ne bénéficie pas de l'effet réseau.

4. Protocoles de transport

Nous décrivons deux autres protocoles de transport utilisés dans la voix sur IP à savoir l'RTP et le RTCP

4.1 Le protocole RTP

4.1.1 Description générale de RTP

RTP (Real time Transport Protocol), standardisé en 1996, est un protocole qui a été développé par l'IETF afin de faciliter le transport temps réel de bout en bout des flots données audio et vidéo sur les réseaux IP, c'est à dire sur les réseaux de paquets. RTP est un protocole qui se situe au niveau de l'application et qui utilise les protocoles sous-jacents de transport TCP ou UDP. Mais l'utilisation 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 ou la visioconférence constitue un véritable problème pour Internet. Qui dit application temps réel, dit présence d'une certaine qualité de service (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.

4.1.2 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 reformer 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 :

- 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 leurs 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.

4.1.3 Avantages et inconvénients

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

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

4.2 Le protocole RTCP

4.2.1 Description générale de RTCP

Le protocole RTCP est fondé sur la transmission périodique de paquets 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, Real-time Transport Control Protocol, qui transporte les informations supplémentaires suivantes pour la gestion de la session.

Les récepteurs utilisent RTCP pour renvoyer 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 appelé 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.

Parmi les principales fonctions qu'offre le protocole RTCP sont les suivants :

- 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 gérées et suivre des chemins différents.

- L'identification des participants à une session : en effet, les paquets RTCP contiennent des informations d'adresses, comme l'adresse d'un message électronique, un numéro de téléphone ou le nom d'un participant à une conférence téléphonique.

- Le contrôle de la session : en effet le protocole RTCP permet aux participants d'indiquer leur 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 RTP 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:

- 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.
- RR (Receiver Report) : Ensemble de statistiques portant sur la communication entre les participants. Ces rapports sont issus des récepteurs d'une session.
- SDES (Source Description) : Carte de visite de la source (nom, e-mail, localisation).
- BYE : Message de fin de participation à une session.
- APP : Fonctions spécifiques à une application.

4.2.2 Point fort et limite du protocole RTCP

Le protocole de 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 ".

5. 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 :

- ***Réduction des coûts :*** En effet le trafic véhiculé à travers le réseau RTC est plus couteux que sur un réseau IP. Réductions importantes pour des communications internationales en utilisant le VoIP, ces réductions deviennent encore plus intéressantes dans la mutualisation voix/données du réseau IP intersites (WAN). Dans ce dernier cas, le gain est directement proportionnel au nombre de sites distants.
- ***Standards ouverts :*** La VoIP n'est plus uniquement H323, mais un usage multi-protocoles selon les besoins de services nécessaires. Par exemple, H323 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) :*** Grace 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 :*** Les 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 :

- ***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 tels la qualité de la reproduction de la voix du correspondant ainsi que le délai entre le moment où l'un des interlocuteurs parle et le moment où 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 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. 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 communication, 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 représente t'il un risque ou une opportunité pour les utilisateurs et les opérateurs téléphoniques ?

Chapitre 2

Attaques contre la VoIP et bonnes pratiques de sécurisation

Introduction

L'opportunité de migrer de la téléphonie classique vers la téléphonie IP, a offert plusieurs avantages pour les entreprises, et les a permirent 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 celle qui participe 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 le protocole

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

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

1.1. Sniffing

Un reniflage (Sniffing) 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.

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.

1.3. Injection de paquet RTP

Cette attaque se fait au niveau du réseau LAN/VPN. Elle cible le serveur registrar, et a pour but de perturber une communication en cours.

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.

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:

Call Spam : 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 (User Agent Client) 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) **Spam** : 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. Ce pourraient être des requêtes INVITE avec un entête « Subject » très grand, ou des requêtes INVITE avec un corps en format texte ou HTML. 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 Spam : 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 communications. 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.

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 à l'IP Spoofing, il est impossible, pour le destinataire, de connaître l'adresse source exacte des paquets IP. De ce fait, à moins que le destinataire ne limite ses échanges avec certaines stations, il lui est impossible de contrer ce type d'attaques.

- 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.

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 peut vite troubler et saturer le trafic transitant sur le réseau et donc de perturbe le plus la bande passante. Presque touts 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.

- 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 :

    1. Le client envoie un paquet SYN au serveur.
    2. Le serveur répond avec un paquet SYN-ACK.
    3. Le client envoie un paquet ACK au serveur.

L'attaque consiste en 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 là le cœur de l'attaque parce que les ACK final 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.

Couche applications :

- 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 :

- ✓ 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.

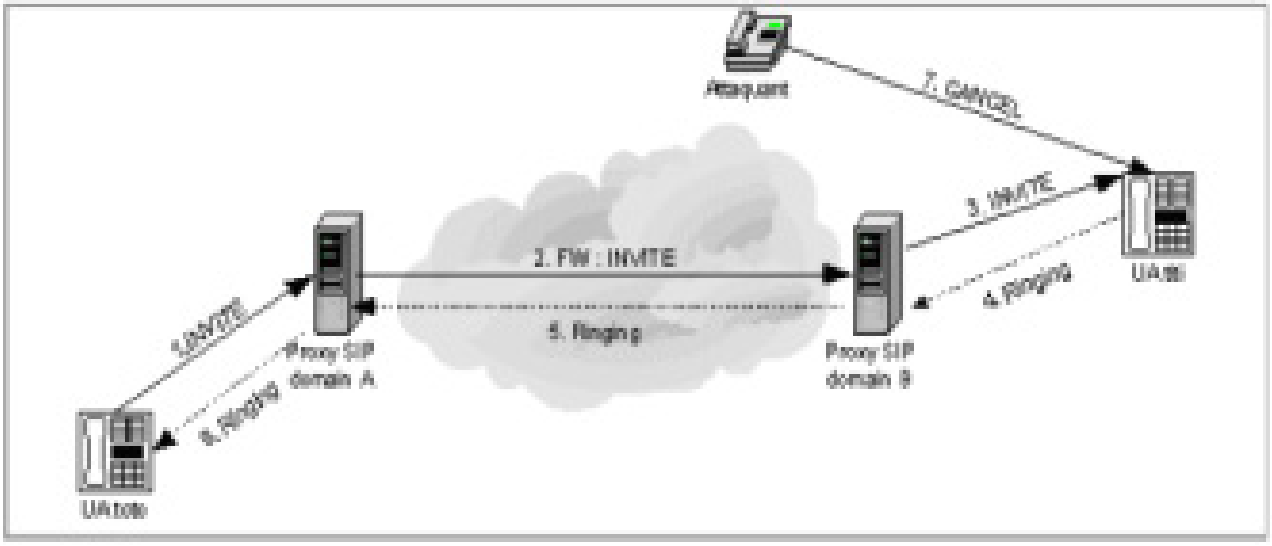

Figure 7 : Attaque DoS via une requête CANCEL

La figure suivante montre un scénario d'attaque DoS CANCEL, 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 lieu.

- ✓ REGISTER

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.

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. A 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.

1.7. L'écoute clandestine

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 (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.

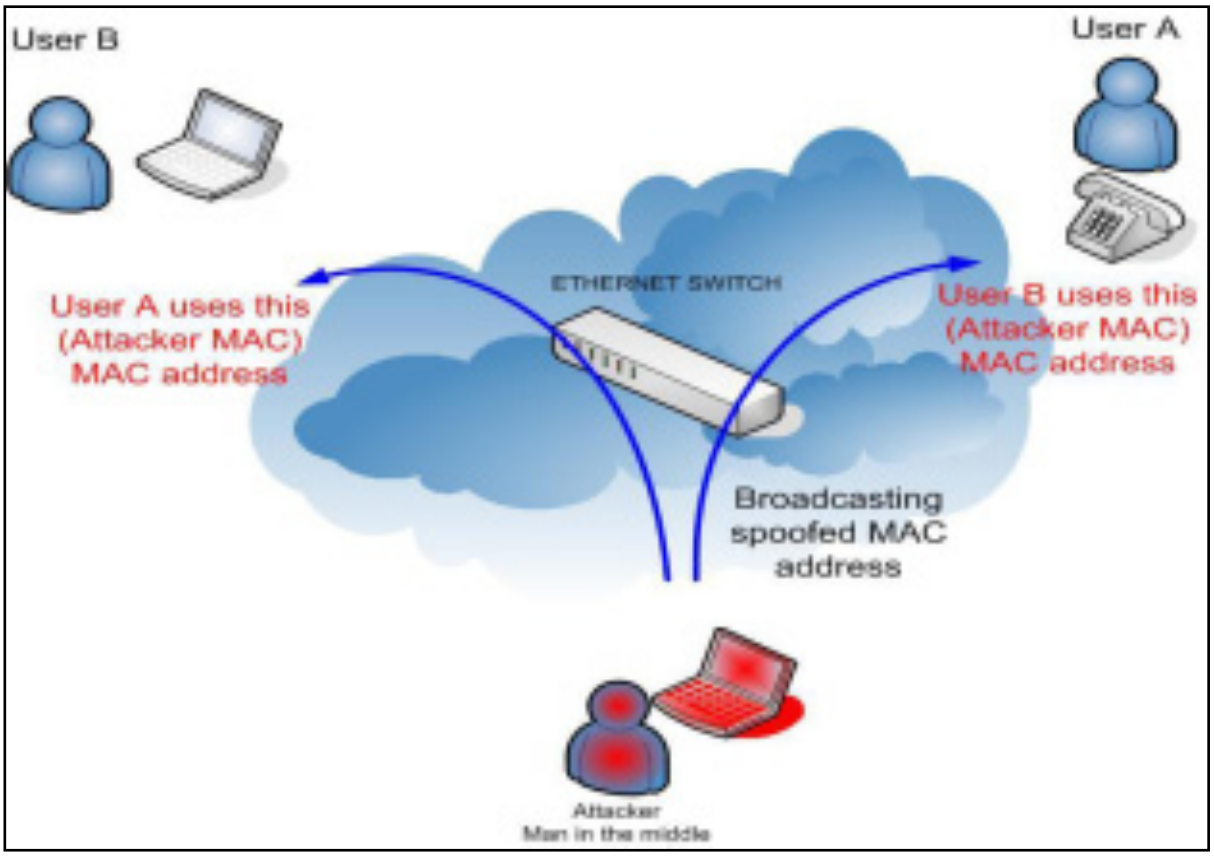

Figure 8 : Exemple de détournement d'appel " Man in the middle"

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

1. déterminer les adresses MAC des victimes (client serveur) par l'attaquant
2. Envoi d'une requête ARP non sollicités au client, pour l'informer du changement de l'adresse MAC du serveur VoIP à l'adresse MAC.
3. Envoi d'une requête ARP non sollicités au serveur, pour l'informer du changement de l'adresse MAC du client à l'adresse MAC.
4. 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

Une infrastructure VoIP est composée de téléphones IP, Gateway, serveurs (proxy, register, 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.

2.4 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 (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 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.

2.5 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.

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.

- A 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.

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.

2.6 Les serveurs

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.

2.7 Les vulnérabilités du système d'exploitation

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.

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.

Peu importe comment, 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 bonne pratiques

On déjà vu que les vulnérabilités existe 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 interconnecté.

IPsec peut être utilisé pour réaliser deux objectifs. 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 ». Combiné avec un pare-feu, IPsec fait 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 celle la plus approprié vu qu'elle offre le cryptage des données grâces a des mécanismes de cryptages, puisqu'elle permet d'offrir l'intégrité des paquets VoIP.

Cryptage aux points terminaux

Vu que notre objectif est d'assurer la confidentialité et l'intégrité des clients, le mode choisie est donc le mode tunnel. Puisqu'il sécurise le paquet comme un tout (contrairement en 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 choisi le protocole ESP qui lui a son tour va assurer le cryptage des données et donc la confidentialité contrairement au protocole AH qui lui 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 appliqué uniquement sur les points de terminaison à la voix IP, c'est-a-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 aussi est bâti sur le protocole temps réel RTP (Real Time Transport Protocol). 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 (Session Initiation Protocol), ainsi que le protocole de diffusion de contenu multimédia en temps réel RTSP (Real Time Streaming Protocol). Mais, surtout, il s'adjoint 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 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 sécurisé avec des mécanismes cryptographiques.

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

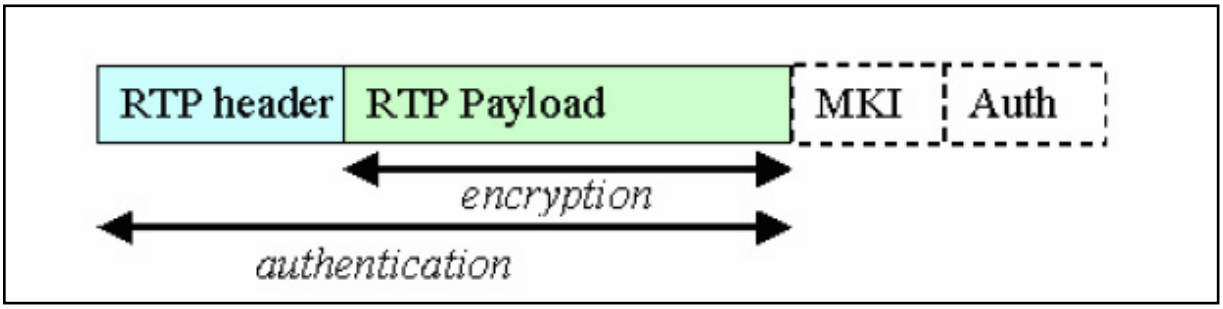

Figure 9 : Format d'un paquet SRTP

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

- SRTP MKI (SRTP Master Key identifier) : sert à re-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.

- Authentication 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.

3.2 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 :

- 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 déterminé et selon le besoin bien sûr. Ces paramètres protègent généralement contre le dénie de service et ces différentes variantes. Il est conseiller d'utiliser les paramètres qui 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.

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

- utiliser un système d'exploitation stable. Les nouvelles versions toujours contiennent 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 simple et robuste. 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 former 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 : 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 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 limiter l'accès à des personnes bien déterminé 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 analyse de l'entête des paquets IP qui s'échange entre les machines.

Le firewall peut être implémenté avec un 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.

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 pare feu ou les routeurs, mais aussi ils existent dans les systèmes d'exploitation.

Conclusion

La voix sur IP devient jour après jour plus ciblé. Il existe plusieurs autres attaques qui menace la sécurité du VoIP, les attaques cité dans ce chapitre sont les plus fameux et courant dans les réseaux VoIP.

Mais en suivant certaines bonnes pratiques parmi les cités, on peut créer un réseau bien sécurisé.

Chapitre 3

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

Introduction

Asterisk est un autocommutateur téléphonique privée (PABX) open source pour les systèmes d'exploitation UNIX, il est publié sous licence GPL.

Asterisk comprend un nombre très élevé de fonctions, tel que les appels téléphoniques, la messagerie vocale, les fils d'attentes, les conférences, etc. Il implémente plusieurs protocoles H.320, H.323, SIP et IAX.

Durant ce chapitre, on montrera les étapes d'installation et de configuration d'Asterisk sous le système d'exploitation Linux, ainsi que l'installation et la configuration de X-Lite qui est un téléphone VoIP softphone, freeware.

1. Architecture du réseau VoIP déployé

La figure 10 montre l'architecture adoptée au cours de la configuration de la solution de VoIP à base d'Asterisk

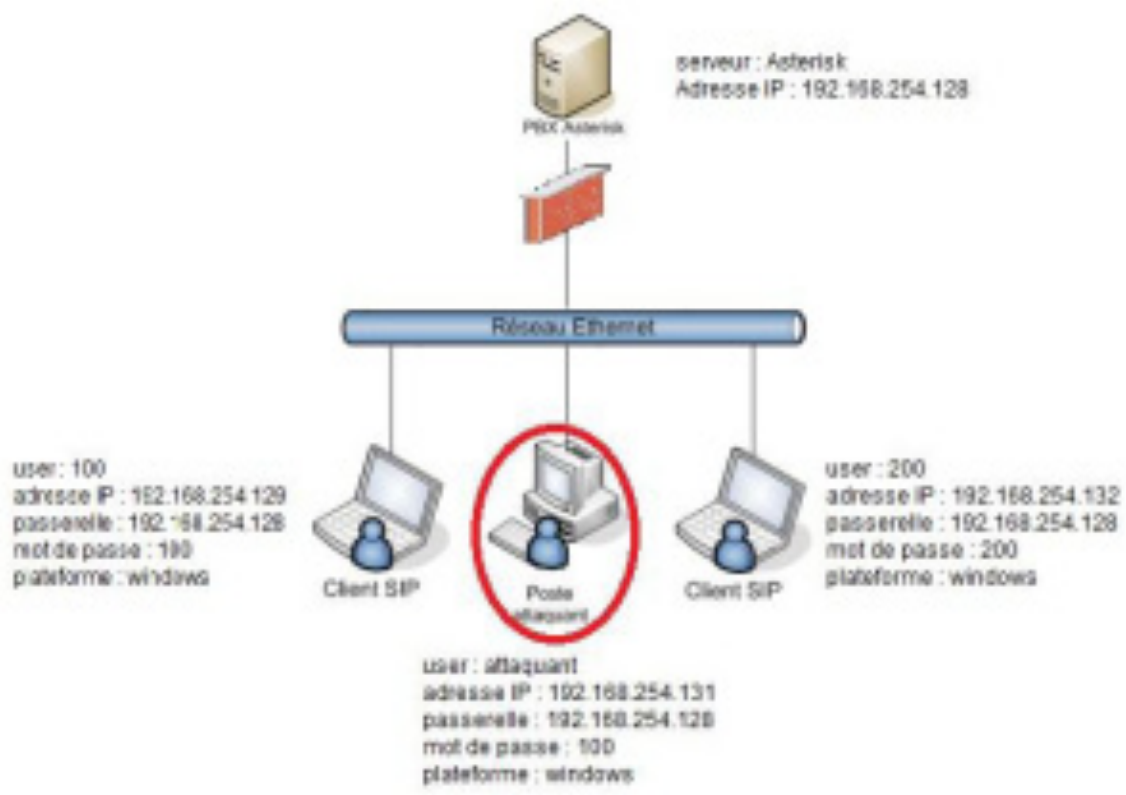

Figure 10 : Architecture du réseau VoIP à réaliser

- **Les deux clients SIP** : Sont des machines sur lesquelles installé le système d'exploitation linux et un client X-Lite.
- **Poste attaquant** : Dans le quel on a installé Windows XP pour réaliser les attaques.
- **Machine serveur** : Sur laquelle installé un système d'exploitation Linux Centos , et le serveur de VoIP, Asterisk.
- **Firewall** : Un firewall software est installé dans la machine serveur pour limiter l'accès.

2. Installation d'Asterisk 1.4

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 un tableau ci-dessous :

Tableau 1 : Liste de paquetages nécessaires pour compiler asterisk et libpri

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

Le meilleur chemin pour obtenir le code source d'Asterisk et ces paquetages est de les télécharger à partir du site web www.asterisk.org. On peut aussi les télécharger directement du serveur de Digium.

2.2 Téléchargement des codes sources

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

```
# cd /usr/src/

# wget http://downloads.digium.com/pub/asterisk/asterisk-1.4-current.tar.gz
```

```
# wget http://downloads.digium.com/pub/libpri/libpri-1.4-current.tar.gz

# wget http://downloads.digium.com/pub/zaptel/zaptel-1.4-current.tar.gz
```

2.3 Extraction des paquetages

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

```
# cd /usr/src/

# tar zxvf zaptel-1.4-current.tar.gz

# tar zxvf libpri-1.4-current.tar.gz

# tar zxvf asterisk-1.4-current.tar.gz
```

2.4 Compilation et installation:

Le Zaptel est un noyau chargeable qui présente une couche d'abstraction entre le matériel et les pilotes de Zapata dans le module Asterisk.

```
# cd /usr/src/zaptel-1.4        =accès au dossier du Zaptel

# make clean                    =supprime les fichiers inutiles après installation.

#./configure                    =construction d'un nouveau fichier makefile.

#make menuselect                =execution de la partie menuselect dans le fichier make file

#make                           =compilation du code source

# make install                  =execution de la partie install dans makefile.
```

Makefile est un fichier qui contient les instructions à exécuter à partir des commandes, ./configure, make, make install, make config, etc. chacune de ces commandes exécute le code approprié à elle dans ce fichier.

Libpri est utilisé par les décideurs du multiplexage temporel (TDM) des appareils VoIP, mais même s'il n'y a pas le matériel installé, il est conseillé de compiler et installer cette bibliothèque. Elle doit être compilé et installé avant Asterisk, car elle sera détecté et utilisé lorsqu'Asterisk est compilé.

```
# cd /usr/src/libpri-1.4

# make clean

# make

# make install
```

Asterisk est un serveur de téléphonie open-source permettant de disposer sur un simple PC les fonctions réservées aux PABX professionnel.

```
# cd /usr/src/asterisk-1.4

# make clean

#. /configure

# make menuselect

# make install

# make samples
```

Dans le cas où on voudrait bien lancer zaptel et le serveur asterisk au démarrage du système, il faut exécuter après la compilation et l'installation des paquetages la commande suivante :

```
# make config          cette commande charge le serveur Asterisk au démarrage du système
```

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

```
# asterisk –r
```

3. Configuration d'Asterisk

3.1 Identification des fichiers de configuration

Une fois l'installation d'Asterisk est effectuée, plusieurs fichiers sont créés :

- /usr/sbin/ : Contient le fichier binaire d'Asterisk (programme principal).
- /usr/lib/asterisk/ : Contient les fichiers binaires qu'Asterisk utilise pour fonctionner.
- /usr/lib/asterisk/modules/ : Contient les modules pour les applications, les codecs, et les drivers.
- /var/lib/asterisk/sounds/ : Contient les fichiers audio utilisés par Asterisk, par exemple pour les invites de la boite vocale.
- /var/run/asterisk.pid : Fichier contenant le numéro du processus Asterisk en cours.
- /var/spool/asterisk/outgoing/: Continent les appels sortants d'Asterisk.
- /etc/asterisk/ : Contient tous les fichiers de configuration.

Le dernier dossier nous intéresse vu qu'il contient les fichiers de configuration du serveur Asterisk, parmi ces fichiers on trouve :

- agents.conf: Contient la configuration de l'utilisation des agents, comme dans le cas d'un centre d'appel. Ceci nous permet de définir les agents et de leurs assigner des ID et des mots de passe.
- asterisk.conf: Définit certaines variables pour l'utilisation d'Asterisk. Il sert essentiellement à indiquer à Asterisk où chercher certains fichiers et certains programmes exécutables.
- extensions.conf: Configure le comportement d'Asterisk. C'est le fichier qui nous intéresse le plus dans ce travail.
- iax.conf: Configure les conversations VoIP en utilisant le protocole Inter-Asterisk Exchange (IAX).

- rtp.conf: Ce fichier de configuration définit les ports à utiliser pour le protocole RTP (Real-Time Protocol). Il faut noter que les numéros listés sont des ports UDP.

- sip.conf: Définit les utilisateurs du protocole SIP et leurs options. On peut aussi définir d'autres options globales pour SIP telles que, quels ports utiliser et les timeout qu'on va imposer. Nous focalisons sur ce fichier puisque notre solution est basée sur le protocole SIP.

- zapata.conf: Configure les paramètres de l'interface téléphonique Zapata.

3.2 Configuration des comptes users

Les deux fichiers à configurer sont sip.conf et extensions.conf. Dans le fichier sip.conf, on créera des utilisateurs utilisant le protocole sip pour l'établissement de la connexion, voilà les deux clients que nous avons crée au niveau du fichier :

[100]	
type=friend	(spécifie le type d'utilisateur)
secret=100	(mot de passé)
host=dynamic	(spécifie une adresse IP par laquelle l'utilisateur peut acceder à son compte)
defaultip=192.168.254.29	(adresse IP du client)
dtmfmode=rfc2833	
context=sip	(spécifie le type de routage à utiliser)
callerid="100"<1111>	(identifiant d'utilisateur)
[200]	
type=friend	(spécifie le type d'utilisateur)
secret=200	(mot de passe)
host=dynamic	(spécifie une adresse IP par laquelle l'utilisateur peut accéder à son compte)
defaultip=192.168.254.132	(adresse IP du client)

```
dtmfmode=rfc2833
context=sip                    (spécifie le type de routage à utiliser)
callerid=”200”<1113>           (identifiant d’utilisateur)
```

Passant maintenant à la configuration du fichier extensions.conf

3.3 Configuration des extensions

```
[sip]                          (il faut saisir le nom du context entre crochet)
exten=>1111,1,Dial (SIP/100,20,tr) (20 est la durée en seconde de l’attente avant la décrochage si pas de réponse)
exten=>1113,1,Dial (SIP/200,20,tr)
```

Si l’appelant compose le numéro 1111, il est mit en relation avec le poste dont le numéro est 1111 qui utilise le protocole SIP.

Il existe d’autres options qu’on peut ajouter dans le fichier extensions.conf, telles que la boîte vocale et le renvoi d’appel. La syntaxe du fichier est sous le format suivant :

Exten= extension, priorité, commande (paramètre)

- Extension : C’est généralement le numéro de téléphone ou le nom du client.
- Priorité : C’est un numéro qui indique la priorité de la commande, le serveur prend en considération la priorité de la commande en utilisant le numéro inscrit dans la syntaxe.
- Commande : C’est la commande qui peut exister, comme la commande dial (appel), voicemail (boîte vocale), etc.

On peut utiliser plusieurs options pour un seul numéro d’appel, on peut mettre par exemple un transfert d’appel vers un autre numéro ou vers la boite vocale selon des priorités.

```
exten => 123,1,Answer
exten => 123,2,Playback(répondeur)
exten => 123,3,Voicemail(9)          (9 est le numéro de la boîte vocale)
```

```
exten => 123,4,Hangup
```

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

```
Localhost*CLI> sip reload

Localhost*CLI> dialplan reload

Localhost*CLI> reload
```

4. Installation et configuration de X-Lite

X-Lite est un freeware, son utilisation est simple, il est disponible pour les différents systèmes d'exploitation Windows, Mac and Linux sur le site de l'éditeur CounterPath[1].

4.1 Installation de X-Lite

L'installation sous Windows est classique, mais pour linux, il faut décompresser le fichier, accéder au répertoire et lancer l'exécutable, comme indiquer ci-dessous :

```
# tar -zxvf X-Lite_Install.tar.gz      (décompression du dossier X-Lite avec la commande tar)

# cd xten-xlite                        (accès au dossier)

# chmod +x xtensoftphone

# ./xtensoftphone                      (lancement du logiciel X-Lite)
```

4.2 Configuration de X-lite

Pour configurer le client X-Lite l'utilisateur « 100 » et aussi « 200 » doivent accéder au menu « Sip Account Setting » puis de ce menu vers le sous menu « Sip Account». Dans la fenêtre qui s'ouvre, il suffit de remplir les champs illustré suivant des deux utilisateurs :

L'utilisateur 100 :

- Identifiant affiché pour l'utilisateur (Display Name) : 100

[1] http://www.counterpath.com/index.php?menu=download

- Identifient servant a loguer l'utilisateur (User Name) : 100
- Mot de passe associé (User Name) : 100
- Nom sous lequel l'autorisation d'accès est possible (Authorization user) : 100
- Nom de domaine (Domain) : 192.168.254.129

Figure 11 : Configuration du compte du client « 100 »

Il est à noter qu'afin que l'authentification soit possible, ces valeurs doivent être conformes à celles saisies dans le fichier sip.conf du serveur Asterisk.

Une fois la configuration est achevée, le softphone se connectera automatiquement au serveur et s'enregistrera. Un message « Logged in » s'affichera, indiquant que les communications sont désormais possibles. Sinon, un message d'erreur explique le motif qui a fait échouer le processus.

Conclusion

Ouvert à tous, gratuit et simple d'utilisation. Asterisk a de quoi s'imposer. Ces vrais concurrents sont plutôt les PBX Hardware. Qui sont chers mais performant et fiable. Les solutions libres peuvent fournir les outils les plus performants et les mieux documentés sans procurer un même service relationnel.

Chapitre 4

Sécurisation de la solution mise en place

Introduction

Après avoir étudié les protocoles de la VoIP, identifié les attaques qui menacent les systèmes de VoIP et les bonnes solutions afin de sécuriser le serveur Asterisk. Nous allons nous intéresser dans ce chapitre aux techniques, mécanismes et configurations à mettre en place dans le but de sécuriser la solution VoIP basée sur le serveur Asterisk.

Ce chapitre se compose de deux grandes parties. Dans la première, nous utiliserons deux logiciels d'attaques, Wireshark et SiVus, et nous expliquerons comment ils fonctionnent. Nous présenterons des scénarios d'attaques réalisés par ces deux logiciels. Dans la deuxième partie, nous montrerons les solutions implémentées pour sécuriser la solution déployée.

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 printing. Une empreinte englobe les informations sur la cible qui déploie le serveur VoIP et ces paramètres de sécurité.

Il existe plusieurs méthodes pour la collecte des informations en voici quelques unes des plus utilisées :

1.1. Utilisation des serveurs Whois

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 :

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. Utilisation des aspirateurs de sites

Si la cible a 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

1.3. Utilisation 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 des 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 une variété de façons 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:

- 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 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 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.

Dans le cadre de notre projet on va se limiter seulement au sniffing des paquets au niveau du serveur et des clients VoIP. Pour cela on aura besoin des outils adéquats pour effectuer cette opération. Parmi les logiciels les plus connues en matière de sniffing dans un réseau basé sur la VoIP sont Wireshark Vomit et SiVus.

2. Les logiciels d'attaques

2.1 Wireshark

Wireshark est un logiciel libre d'analyse de protocole, utilisé dans le dépannage et l'analyse de réseaux informatiques, le développement de protocoles, l'éducation et la rétro-ingénierie, mais aussi 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, dénie de service, etc.). Ainsi que nous pouvons écouter une communication entre deux clients en décodant les paquets RTP (écoute clandestine).

2.1.1 Captures de trames

Nous avons placé Wireshark dans une 3éme machine qui va jouer le rôle de l'attaquant. Elle va sniffer tous 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, « 200 » ayant comme adresse IP 192.168.254.132 et « 100 » ayant comme adresse IP 192.168.254.129. Voilà le résultat de la capture :

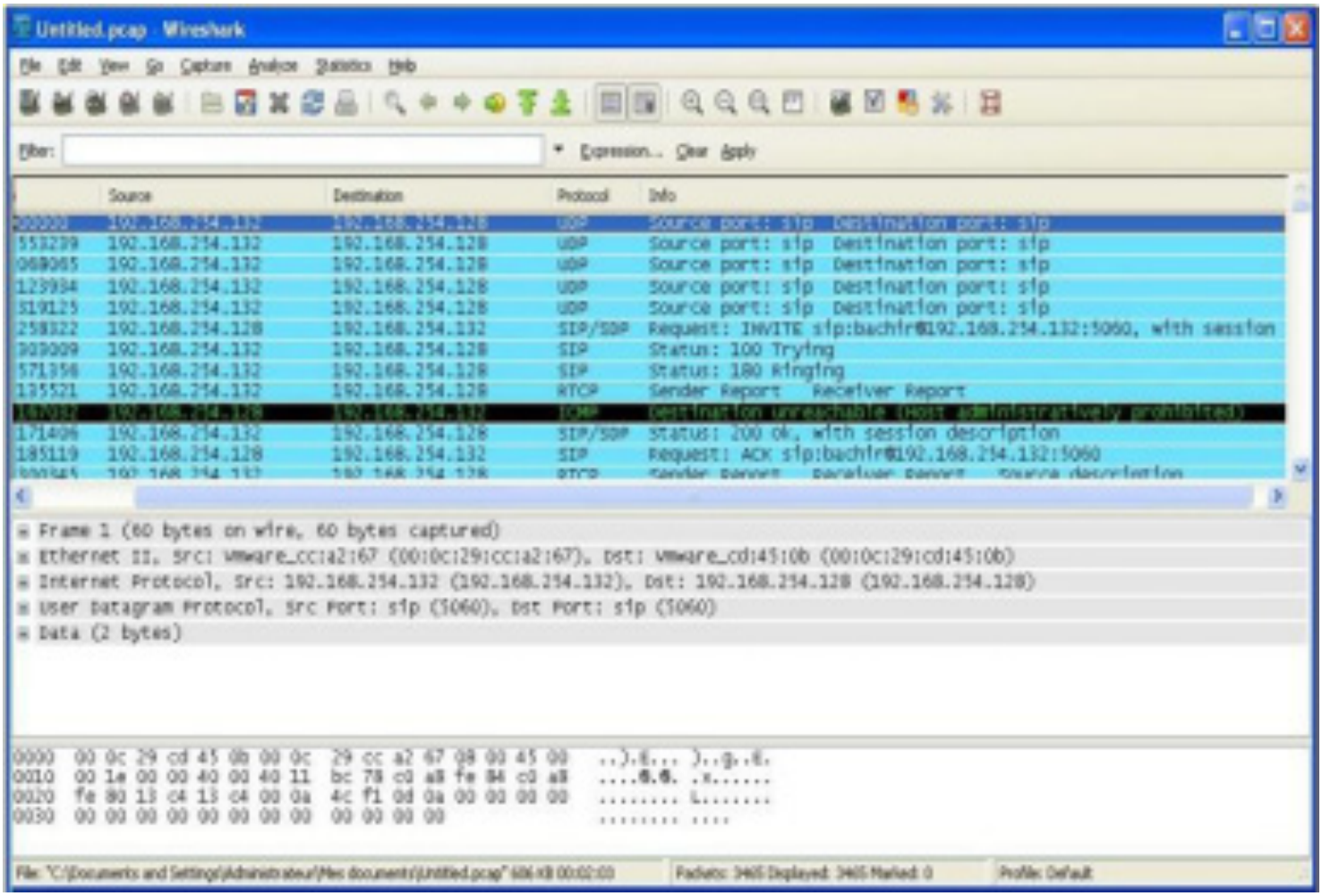

Figure 12 : Ecran de capture Wireshark

Comme nous pouvons le voir dans la figure 12, la conversation entre ces deux hôtes a 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, celle la plus intéressante, nous pouvons lire le contenue des paquets et donc collecter des informations très indispensables pour effectuer une bonne attaque.

Figure 13 : Exemple de paquet qui contient une requête INVITE

Voilà en plus grand, dans la figure 13, le paquet que nous avons choisi pour examiner. Celui-ci est un paquet utilisant le protocole SIP contenant une requête INVITE. Cette requête contient des informations indispensables dans le cas où nous voulons effectuer une attaque basée sur le protocole SIP. Par exemple dans le cas où nous voulons exécuter une attaque de type DoS en utilisant le protocole SIP, nous aurions besoin de connaitre le user agent. Dans cet exemple il

n’est autre que le serveur Asterisk, l’adresse SIP de notre victime, son identité et d’autres paramètres.

2.1.2 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 circulant entre deux machines effectuant une conversation VoIP, et décoder par la suite les paquets afin d’écouter la conversation effectuée.

Le principe est le suivant. Un client nommé 100 ayant comme adresse IP 192.168.254.128 va appeler le client nommé 200 ayant comme adresse 192.168.254.132. Il faut savoir que ces deux clients utilisent un serveur Asterisk qui a été préalablement configuré pour effectuer leurs appels. Avant cet appel, il faut tout d'abord activer Wireshark afin de sniffer le trafic. Il est installé sur une troisième machine qui n'est pas autorisé à passer des appels à travers le serveur Asterisk puisqu'elle n'est pas configurée dans les fichiers de ce dernier. Toutes ces machines sont installées sous le même réseau. Durant la capture nous pouvons voir les différentes phases d'appel, la signalisation et le transport des paquets.

A la fin de l'appel, nous aurions sniffé tous les paquets dont nous aurions besoin pour l'écoute clandestine, les paquets les plus importants sont ceux basés sur le protocole RTP vu qu'ils contiennent les conversations audio entre les deux clients comme l'indique la figure 14 :

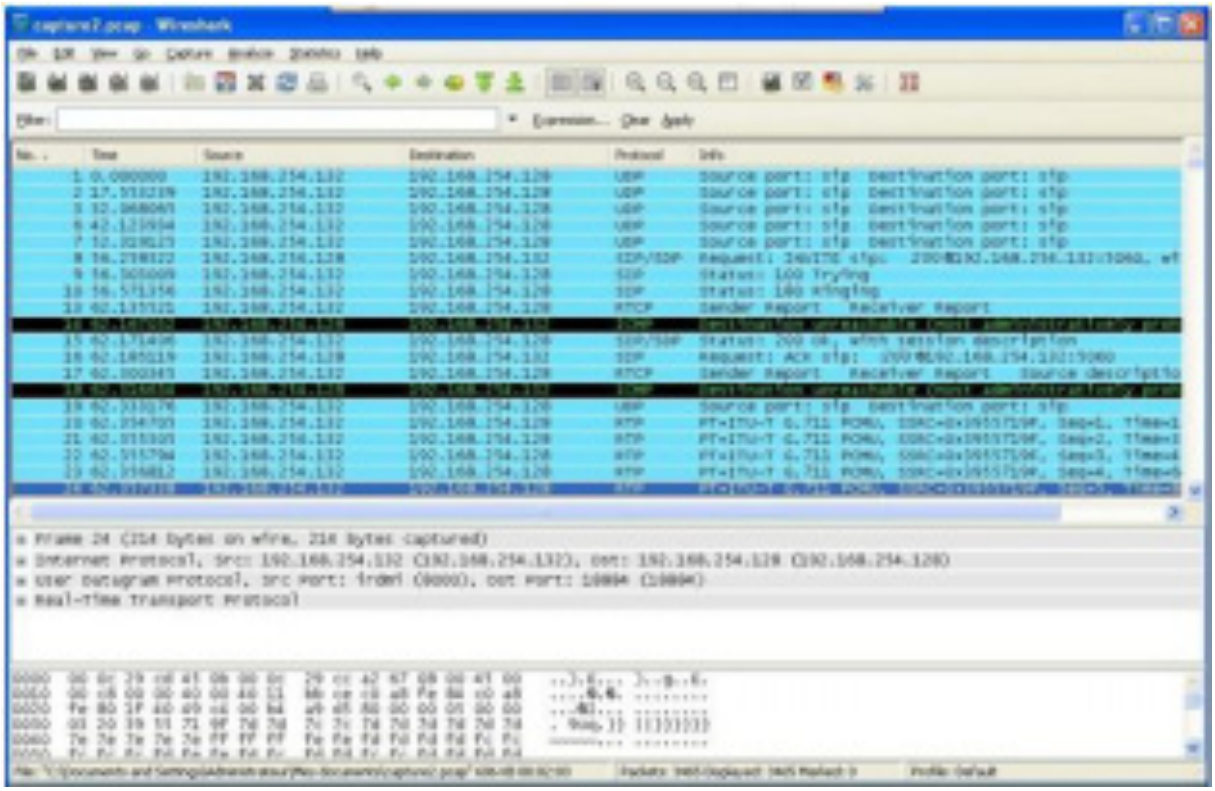

Figure 14 : Capture d'une communication téléphonique

Après avoir identifié les paquets RTP, nous allons maintenant procéder au décodage de l'appel. Dans le menu de Wireshark, nous cliquons sur le bouton « Statistics », puis ensuite le bouton « VoIP Calls » comme l'indique la figure 15.

Figure 15 : Décodage: Bouton VoIP Calls

Une deuxième fenêtre s'ouvre (voir figure 16) contenant les communications dans les deux sens, du client 192.168.254.128 vers 192.168.254.132, et inversement.

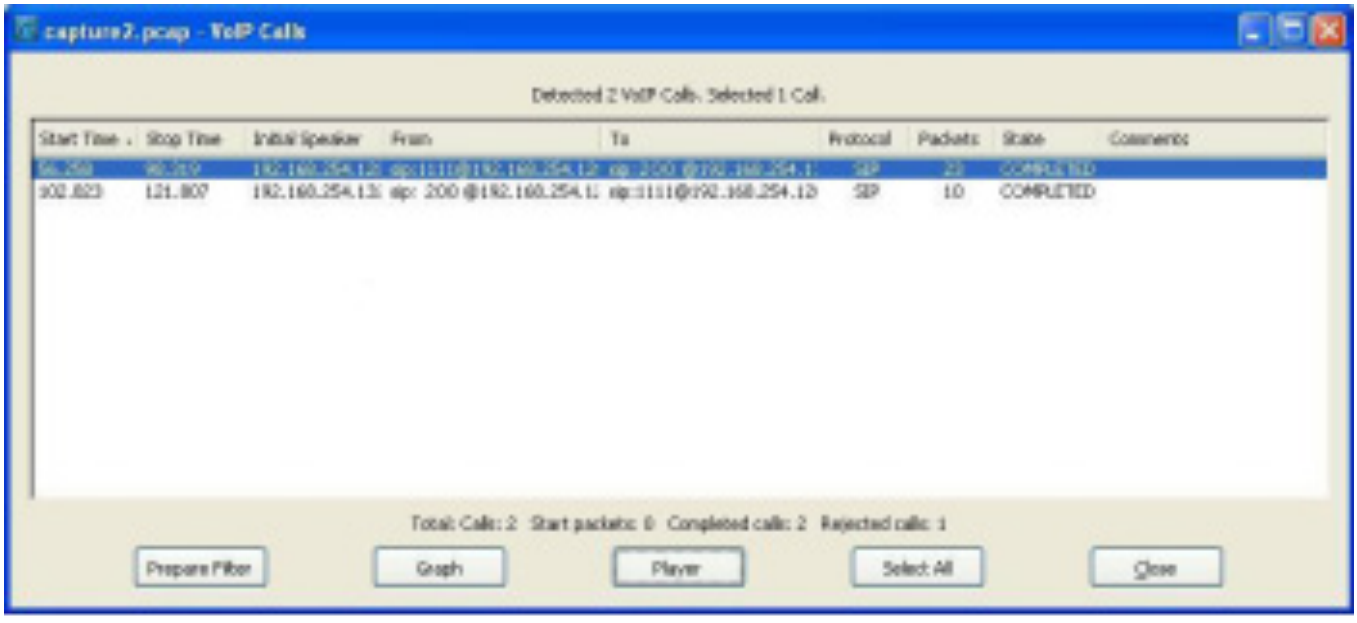

Figure 16 : Communication téléphonique détectés

Nous choisissons une des communications détectées et nous cliquons sur le bouton « Player », une fenêtre « RTP Player » s'ouvre pour le décodage (figure 17). nous cliquons sur « Decode » pour que l'opération commence.

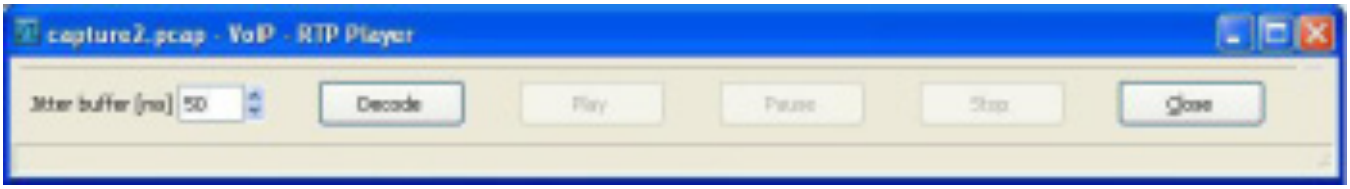

Figure 17 : Fenêtre RTP Player

Maintenant que le décodage a abouti nous pouvons aussi voir sur la figure 18 que le son est décodé et qu'il est prêt à être écouté.

Pour l'écoute, il faut choisir le parcours de la communication, plus précisément, il faut choisir la direction de la communication. Nous avons choisi ici d'écouter la communication qui se dirige de l'adresse IP192.168.254.128 vers l'adresse IP 192.168.254.132. La communication est de durée 17,26 s.

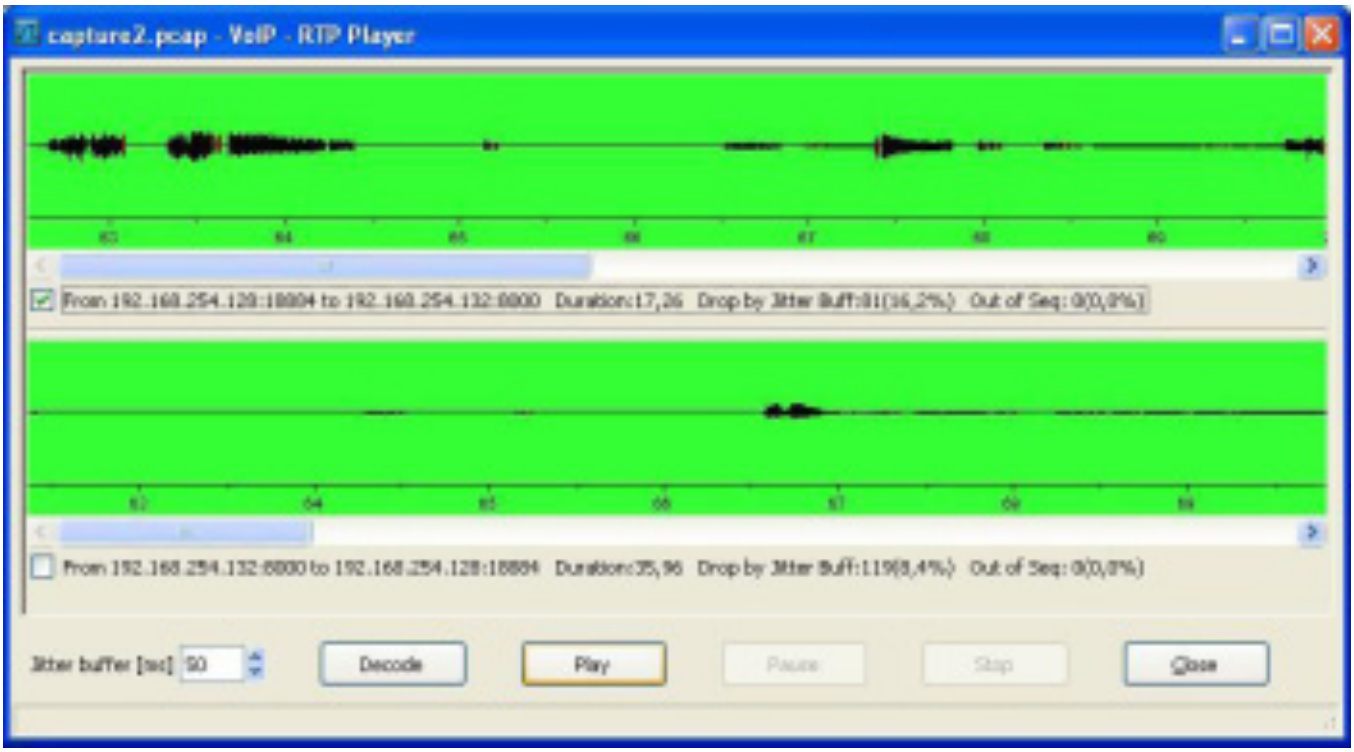

Figure 18 : Communication téléphonique décodé

2.2 Le logiciel SiVus

SiVus, est un scanner de vulnérabilités pour les réseaux VoIP utilisant le protocole SIP mais aussi un logiciel de capture de trames. Les administrateurs et les installeurs de réseaux VoIP l'utilisent pour tester l'efficacité du réseau. C'est un logiciel tournant sur les deux systèmes d'exploitation, Linux et Windows.

Pour tester la sécurité du serveur Asterisk, il faut installer SiVus avec Wireshark, qui aidera à récolter des informations pour générer des requêtes de différents types (INVITE, BYE, CANCEL…). SiVus, est un générateur d'attaques de dénie de service.

Pour pouvoir télécharger SiVus, il faut s'enregistrer au site Vopsecurity[2] et y accéder. Plusieurs versions existent sur le site dont la meilleure est la version 1.09 étant donné sa stabilité.

L'installation est simple mais pour que SiVus fonctionne, il faut disposer du java runtime Enviroment (version supérieure à 1.4).

2.2.1 Utilisation de SiVus

SiVus se compose de 4 principaux sous menus dans le menu SIP:

- **Component Discovery:** sert à découvrir les machines qui se trouvent dans un réseau VoIP, et déterminer sur quel port elles sont connectées et quel protocole elles utilisent.

- **SIP Scanner:** contient lui aussi des sous menus. Le premier sert à créer une configuration au choix (les ports à scanner, les protocoles utilisés, les adresses IP ou la plage d'adresse IP. Le deuxième sert à lancer la configuration crée et enregistrée, et voir quelles sont les vulnérabilités existantes, afin qu'un attaquant puisse l'utiliser.

- **Utilities:** Ce troisième sous menu, contient lui aussi deux sous menus, le premier est un générateur de requêtes. Nous remplissons les champs nécessaires après avoir collecté les informations à l'aide de Wireshark. Nous choisissons la méthode après avoir scanné le réseau et savoir quelles méthodes est vulnérable. Nous lançons le message en appuyant sur « START »

[2] www.vopsecurity.org, portail sur la sécurité

Le deuxième sous menu de Utilities est « authentication analysis », c'est un analyseur d'authentification, il permet de déterminer les mots de passe par exemple.

- **SIP HELP :** c'est le sous menu de l'aide, il contient des exemples de différents messages SIP qui peuvent être envoyé, le RFC de SIP et quelques liens qui expliquent en détail le protocole SIP.

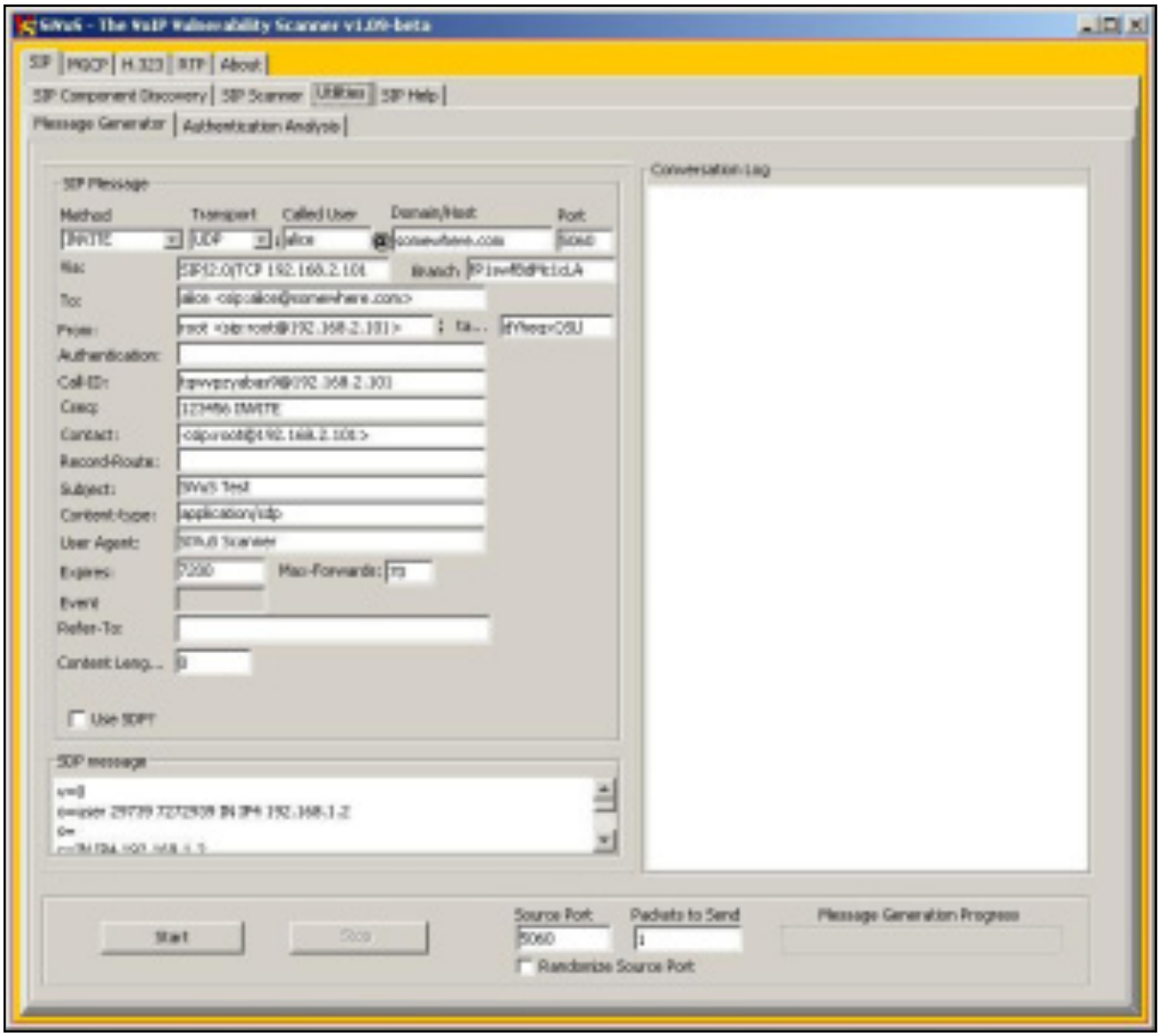

Figure 19 : SiVus : fenêtre de génération de message

La figure 19, montre un exemple de message généré. Pour générer un message, il faut tout d'abord choisir une méthode, il faut aussi changer les paramètres se trouvant dans le corps du message avec les paramètres que l'attaquant à collecté à propos de la victime.

2.2.2 Attaque réalisée par SiVus : DOS de type BYE

Le logiciel SiVus permettra d'effectuer un type d'attaque très fréquent sur les réseaux VoIP. Cette attaque n'est autre que DoS. Un attaquant va essayer de créer un déni de service auprès du serveur Asterisk. Pour cela il utilisera le protocole SIP et plus exactement les requêtes de types BYE, et les enverra successivement de manière fréquente. Ainsi il est possible de monopoliser les ressources du serveur Asterisk. Comme le nombre de connexions est la plupart du temps limité, le serveur n'accepte plus de nouveaux clients. Il est donc en déni de service.

Le principe est le suivant, un serveur Asterisk est implémenté sur une machine dans un réseau local ayant pour adresse IP 192.168.254.128 avec deux autres machines qui sont « 200 » ayant comme adresse IP 192.168.254.132 et « 100 » ayant comme adresse IP 192.168.254.128 et une quatrième machine qui va jouer le rôle de l'attaquant ayant comme adresse IP 192.168.254.131. Tous d'abord, l'attaquant va scanner le réseau et plus particulièrement le protocole SIP afin de déterminer ses failles. Cette étape est possible grâce à l'outil SiVus qui contient un SIP SCANNER. Voici le résultat (figure 20).

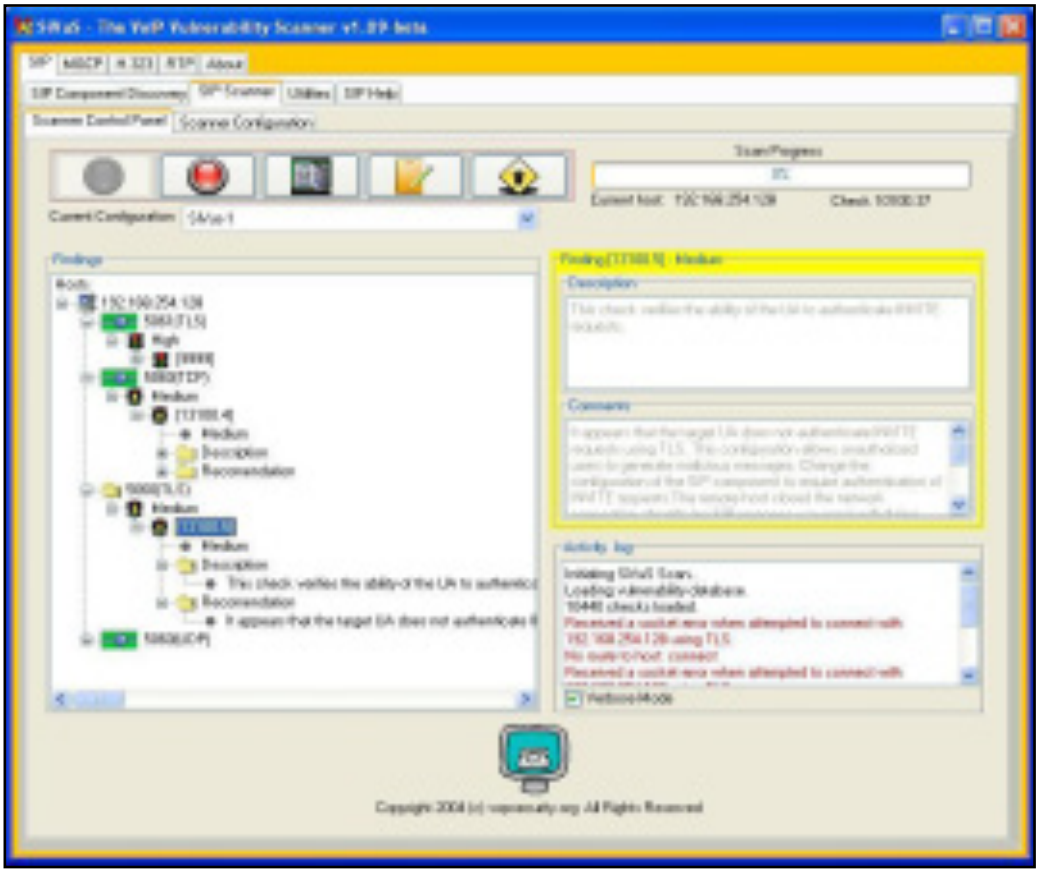

Figure 20 : Scanne de la machine 192.168.254.128

Le scan montre qu'il y a une vulnérabilité au niveau du protocole TLS due à une mauvaise configuration dans le serveur Asterisk (configuration au niveau du fichier sip.conf). Le *scanner* montre d'autres vulnérabilités de différents degrés « medium, High ».

Après avoir élaboré le scan du réseau, et extrait les données nécessaires de Wireshark, nous pouvons générer une requête de type BYE (figure 21).

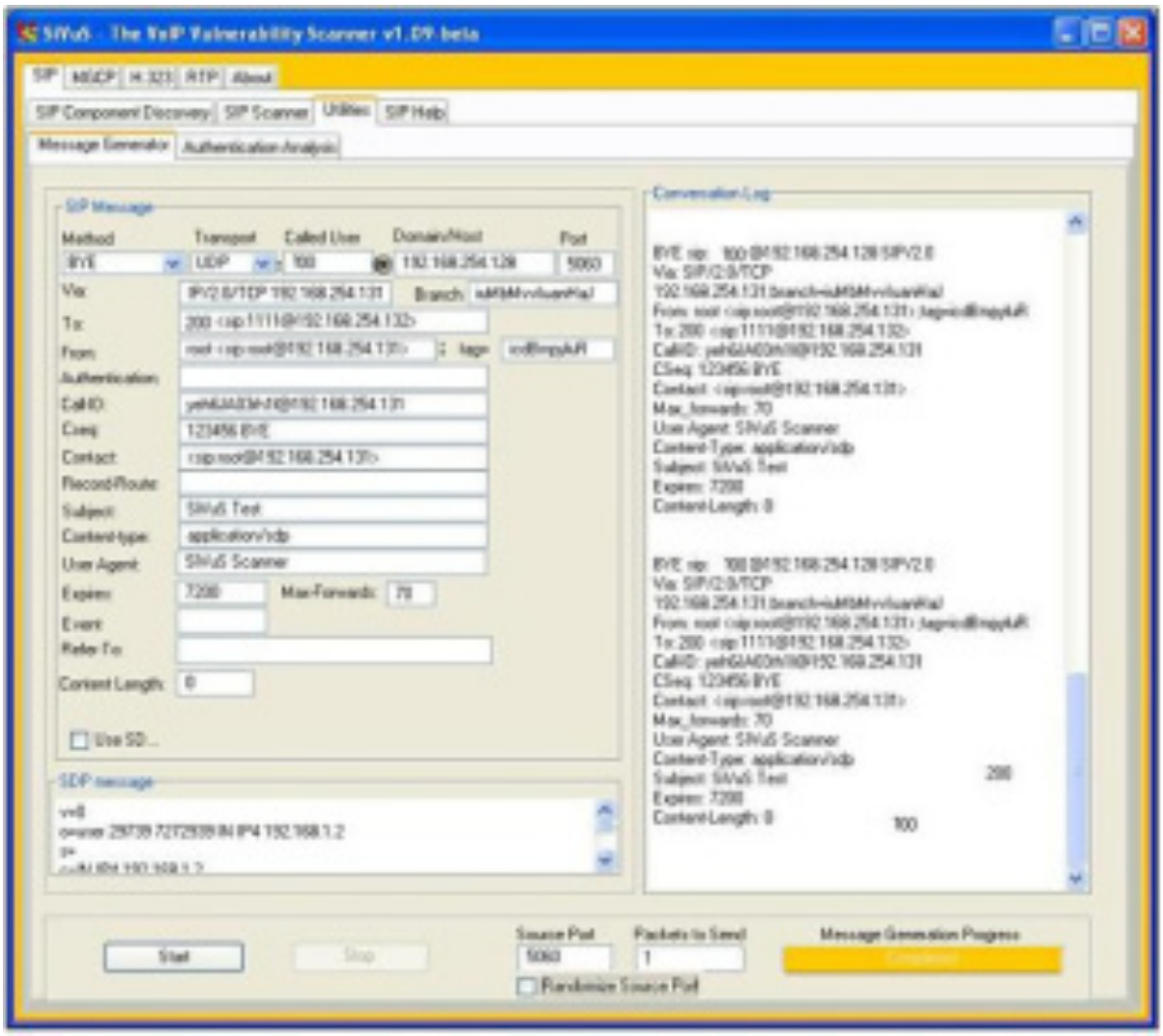

Figure 21 : Génération de message de type BYE

Dans le menu du générateur il faut indiquer le type des requêtes que nous allons utiliser, dans notre cas c'est BYE. Le protocole de transport à utiliser est UDP, vu que SIP est basé sur lui. L'onglet « Called User » va permettre de cacher la vrai identité de l'attaquant, c'est-à-dire que les requêtes que va générer SiVus auront comme identifiant SIP les valeurs avec lesquelles on les a crée. Dans notre cas nous avons choisi le client « 100 » ayant pour adresse IP 192.168.254.128 et enfin le numéro de port 5060 qui représente celui associé au protocole SIP.

Ensuite il faut spécifier la machine à attaquer dans le champ « TO ». C'est la deuxième machine de notre réseau, celle du client nommé « 200 ».

Enfin, en cliquant sur le bouton « START », le message sera alors généré, nous pouvons voir en bas du menu (figure 21) un ticket jaune « Completed ».

Pour vérifier que l'attaque a bien eu lieu, nous pouvons utiliser Wireshark pour visualiser le trafic et voir si oui ou non des requêtes de type BYE ont été générées.

La figure 22 montre que les requêtes de types BYE sortent de la machine de l'attaquant ayant comme adresse IP 192.168.254.131 vers la machine client d'adresse IP 192.168.254.128. Nous pouvons voir aussi que l'identité SIP réelle de l'attaquant a été falsifiée.

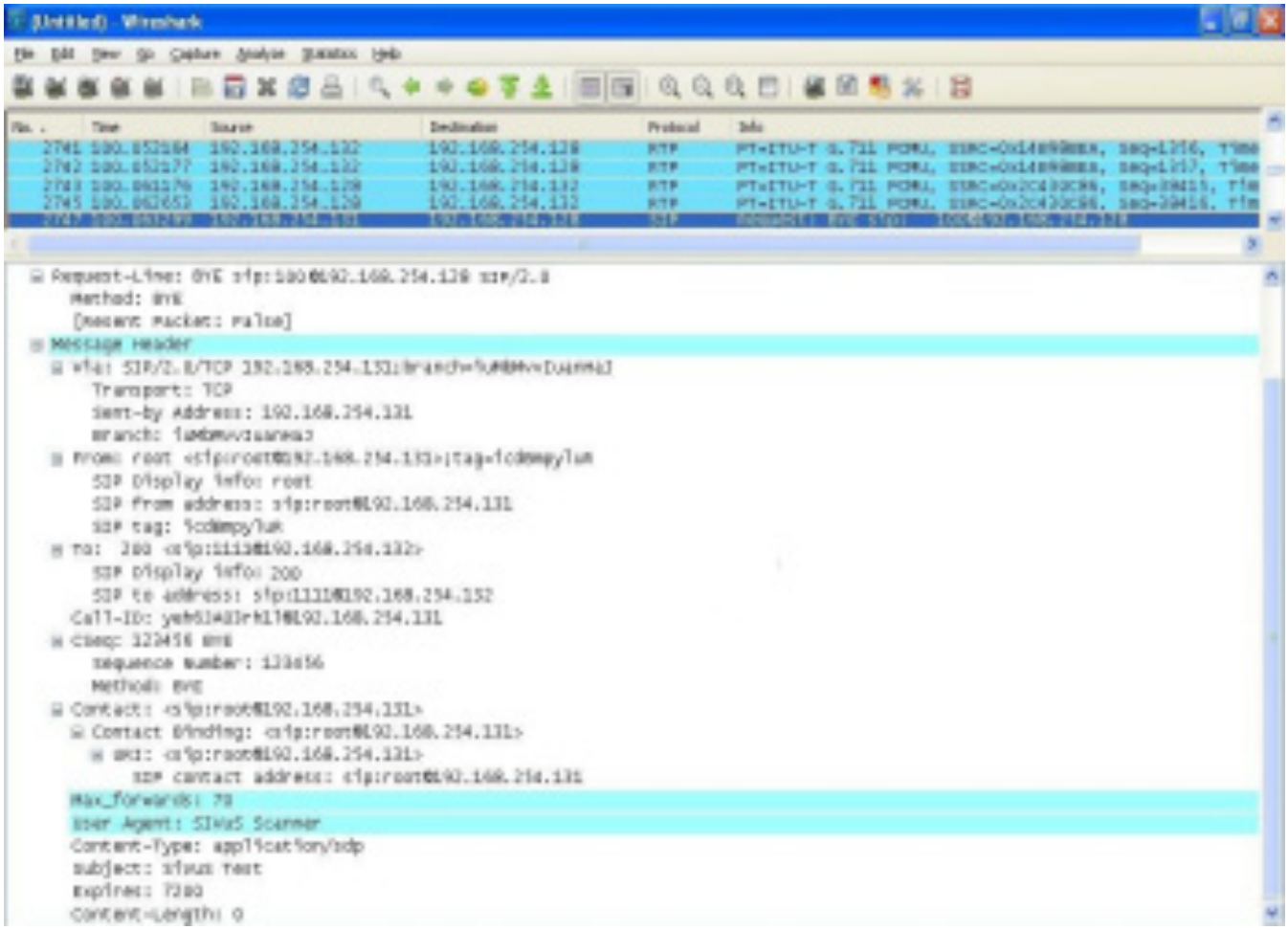

Figure 22 : Message envoyé par SiVus apparait dans Wireshark

Le deuxième écran de Wireshark (figure 22) montre les détails de la requête envoyée par l'attaquant. Celle-ci contient des informations indiquant qui a envoyé cette requête. Nous pouvons voir en bas que le *User Agent* est SiVus Scanner.

3. Choix et implémentation des bonnes pratiques

Pour se protéger contre les attaques réalisées et même d'autres attaques similaires comme les attaques DOS, nous avons choisi un ensemble de solutions qui peuvent aider à minimiser les menaces, nous ne pouvons pas dire que les solutions proposés et implémentés sont efficaces. Parce qu'il existe toujours des problèmes de sécurités.

3.1 Bonne pratique contre l'écoute clandestine

3.1.1 Implémentation du protocole SRTP

Parmi les solutions les plus performantes et les plus faciles à mettre en œuvre, pour contourner l'attaque de l'Eavesdropping ou l'écoute clandestine, est l'implémentation du protocole SRTP sur le serveur Asterisk. En effet, ce protocole permet de chiffrer les données et de les injecter dans le trafic. De cette façon une personne malveillante essayant de décoder les paquets, ne pourra plus user de cette attaque. Il faut savoir que SRTP est une branche d'Asterisk et donc on aura besoin de le récupérer depuis le serveur SVN (subversion). Subversion est un système de gestion de version, c'est-à-dire qu'il permet de gérer la version d'un fichier source ou de garder un historique de toutes ces versions. Voici les étapes à suivre pour la configuration de SRTP sur Asterisk :

Avant de commencer la configuration, il faut compiler et installer la librairie de SRTP **LIBSRTP :**

```
# tar -xzf srtp-tarball                    (décompression des paquetages)

# ./configure --prefix=/usr make    (installation des composants binaires de la librairie dans le répertoire /usr)

# make runtest

# make install
```

Ensuite il faut installer la librairie **MINISIP libraries :**

```
# svn co -r3250 svn://svn.minisip.org/minisip/trunk minisip-trunk
(commande permettant de récupérer une révision ainsi que ses métadonnées depuis le dépôt)

# cd minisip-trunk
```

Une révision est une instance d'un fichier à un moment donné. Une métadonnée est une donnée servant à définir ou décrire une autre donnée

Ensuite il faut installer et compiler **libmutil** :

Mais d'abord il faut lancer le script de démarrage pour générer le script de configuration.

```
# cd libmutil

# libmutil$ ./bootstrap                    (charger la librairie dans le rom)
```

Ensuite il faut compiler le code source de **libmutil** et l'installer:

```
# libmutil$ ./configure --prefix=/usr      (installation des composants binaires de la librairie dans le répertoire /usr)

# libmutil$ make

# libmutil$ make install
```

Aussi il faut compiler et installer les librairies, libmnetutil, libmcrypto, libmikey. La compilation et installation de ces librairies se fait de la même façon que la librairie **libmnetutil:**

```
# cd ../libmnetutil

# libmnetutil$ ./bootstrap                 (charger la librairie dans le rom)

# libmnetutil$ ./configure --prefix=/usr                    (installation des composants binaires de la librairie dans le répertoire /usr)

# libmnetutil$ make

# libmnetutil$ make install
```

Passons maintenant à la configuration d'Asterisk en lui ajoutant les correctifs et les fichiers nécessaires ainsi que l'ajout du module SRTP dans le menu d'Asterisk.

```
# svn checkout -r61760        http://svn.digium.com/svn/asterisk/trunk asterisk-trunk

# cd asterisk-trunk

#wget http://bugs.digium.com/file_download.php?file_id=13837&type=bug
```

(Télécharger un fichier depuis cette url)

patch -p1 < ast_srtp_r61760_mikey_r3250.patch (l'application d'un patch sur le serveur Asterisk)

./bootstrap.sh (compilateur permettant de lancer le compilateur de configuration du menu d'Asterisk)

./configure

make menuselect (vérifier res_srtp dans "resource modules")

make

make install

Ensuite il faut configurer les fichiers **sip.conf** et **extensions.conf**:

- **extensions.conf**

[main]

Le fichier permettant le routage doit être au courant que nous allons utiliser le protocole SRTP pour cela nous devons rajouter cette option dans le fichier. Il faut aussi créer un test sur le port 7 appelé port echo qui permettra de trouver les causes des problèmes liés a la piles TCP / IP dans le cas ou il y'a des problèmes de connexion. Cette configuration s'applique à l'utilisateur ayant pour numéro 1111 (utilisateur « 100 »). Dans cet exemple SRTP est utilisé optionnellement :

exten => 1111,1,Set(_SIPSRTP=optional) (l'utilisation de SRTP est optionnel)

exten => 1111,2,Set(_SIPSRTP_CRYPTO=enable) (activer le cryptage)

exten => 1111,3,Playback(demo-echotest) (création d'un test echo permettant de savoir si oui ou non la connexion a eu lieu)

exten => 1111,4,Echo (faire le test echo)

exten => 1111,5,Playback(demo-echodone) (si le test echo a réussi cette option nous permettra d'écouté le son qu'on a créer pour le test)

exten => 1111,n,hangup

Cette configuration s'applique à l'utilisateur ayant pour numéro 1113, c'est donc l'utilisateur « 200 » :

```
# exten => 1113,1,Set(_SIPSRTP=require) (l'utilisation de SRTP est requise)

# exten => 1113,2,Set(_SIPSRTP_MIKEY=enable)

# exten => 1113,3,Playback(demo-echotest)

# exten => 1113,4,Echo

# exten => 1113,5,Playback(demo-echodone)

# exten => 1113,n,hangup
```

- **sip.conf**

Dans ce fichier l'option à modifier dans les paramètres des clients est la suivante :

```
context=main  (et non plus SIP le main fait référence à la configuration que nous venons de créer dans le fichier extensions.conf et donc le routage choisi)
```

Pour finir voici quelques remarques à prendre en considération dans le cas où nous voulons utiliser SRTP avec le serveur Asterisk :

- ✓ MIKEY ne prend pas en charge le cryptage en option.
- ✓ L'appelé ne peut pas sélectionner la méthode de cryptage

3.1.2 Mise en place de la solution VPN

Autre solution pour crypter le trafic dans notre réseau, est l'implémentation d'un VPN au sein des machines utilisées pour la VoIP. Comme nous l'avons vu précédemment un VPN permet de véhiculer du trafic crypté grâce à des clés de cryptage ce qui rend leur déchiffrage presque impossible par une tierce partie. Un VPN permettra donc de contourner les attaques d'écoute clandestine.

L'outil que nous avons choisi pour la mise en place d'un VPN est OpenVPN. C'est un logiciel « open source » permettant de créer un réseau virtuel basé sur SSL. Il peut être utilisé afin de relier deux réseaux ou plus via un tunnel chiffré à travers l'Internet. Par ailleurs,

OpenVPN n'utilise pas de protocole de communication standard. Il faut donc utiliser un client OpenVPN pour se connecter à un serveur OpenVPN.

- **Installation d'OpenVPN**

L'installation d'OpenVPN se fait grâce à la commande suivante :

```
# yum install –y openvpn (l'option –y permet d'accepter l'installation directement, l'installation avec yum nous permet d'installer les dépendances du logiciel automatiquement)
```

- **Générations des certificats**

Maintenant et après l'installation, il faut créer les certificats et les clés qui vont permettre aux clients et au serveur de s'authentifier mutuellement de telle sorte que personne d'autres ne puisse se connecter au VPN.

Il faut se diriger vers le répertoire où se trouve les fichiers que nous allons configurer

```
# cd /usr/share/openvpn/easy-rsa/
```

Première chose, il faut modifier les valeurs des variables d'environnement afin de ne pas avoir à répéter les renseignements à fournir à la génération des clés comme indiqué dans la figure 23 :

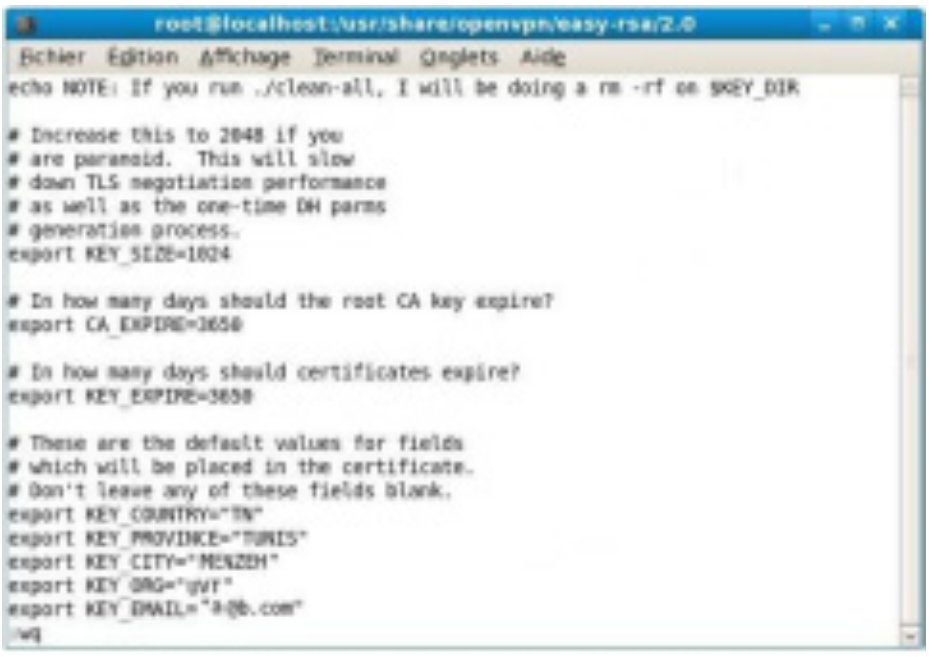

Figure 23 : Modification des valeurs des variables d'environnements

Après la modification du fichier vars, il faut nettoyer le répertoire /keys avant la génération des nouveaux certificats et relancer la prise en charge des nouvelles variables grâce à la commande suivantes :

```
# .. /vars

#./clean-all
```

Nous passons maintenant à la création des certificats, mais tour d'abord il faut commencer à créer l'autorité de certification en tapant la commande suivante :

```
# . /build-ca
```

Voici ce qui devrai ce produire lors de l'exécution de cette commande

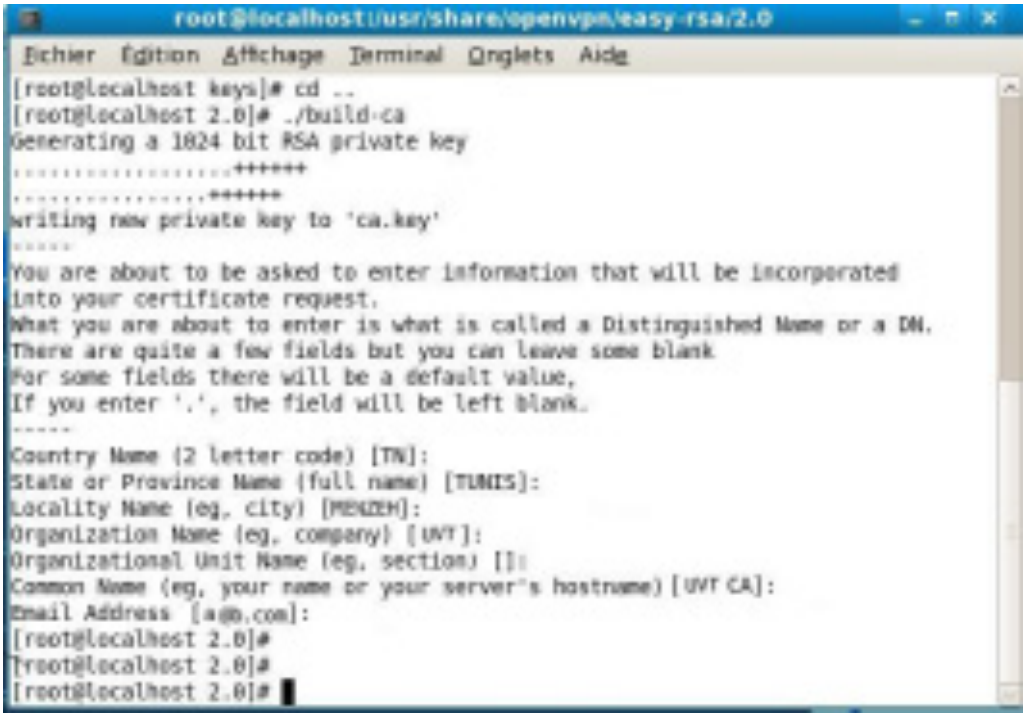

Figure 24 : Création du certificat d'autorité

Ce certificat est le certificat racine qui va ensuite nous permettre de créer le certificat serveur et les certificats clients.

Maintenant nous allons créer le certificat pour le serveur grâce à la commande suivante

```
# ./build-key-server server
```

Le résultat de l'exécution de la commande est le suivant :

Figure 25 : Création d'un certificat pour le serveur

Maintenant il faut créer le certificat pour le client grâce à la commande suivante :

```
# ./build-key client
```

L'exécution de cette commande affiche le résultat suivant :

Figure 26 : Création du certificat client

A présent, il reste à créer les paramètres Diffie-helmann : Diffie-Hellma (D-H) est un algorithme permettant la génération de clés secrètes à travers des canaux non sécurisés. La commande de création est la suivantes :

```
# ./build-dh
```

L'exécution de cette commande donne le résultat suivant (voir figure 27) :

Figure 27 : Création des paramètres Diffie-helmann

Nous avons maintenant l'ensemble des informations cryptographiques dont nous avons besoin pour configurer le VPN. Ensuite il faut copier l'ensemble des informations cryptographiques, que nous venons de générer dans le répertoire keys, dans le répertoire /etc/openvpn crée par défaut à l'installation d'OpenVPN. La commande suivante est utilisée:

```
# cp Keys/* /etc/openvpn
```

- **Création d'un utilisateur OpenVPN**

Nous allons maintenant passer à la création d'un utilisateur ayant des droits restreints qui sera chargé de lancer le service de telle sorte que même si nous nous font pirater la machine, l'attaquant n'aura que les droits de cet utilisateur et pas avec les droits root.

Il faut créer un groupe d'utilisateur dans lequel nous allons affecter l'utilisateur grâce à la commande suivante :

```
# groupeadd openvpn                (le groupe qu'on vient de créer se nomme openvpn)

Ensuite créer l'utilisateur

# useradd –d /dev/null –s /bin/false –g openvpn
```

- **Configuration et lancement du serveur**

Il faut tous d'abord copier le fichier server.conf se trouvant dans le répertoire /usr/share/doc/openvpn-2.1/sample-config-files et le placer dans le répertoire suivant /etc/openvpn.

```
# cp server.conf /etc/openvpn
```

Il faut maintenant éditer ce fichier pour y positionner les variables pour la mise en place du VPN.

```
# Vi server.conf                    (éditer les paramètres du fichier)
```

Les paramètres à modifier sont les suivants :

- ✓ **Dev tun**

Pour pouvoir utiliser OpenVPN en mode tunnel.

- ✓ **Server 10.8.0.0 255.255.255.0**

Nous donnerons cette plage par défaut au serveur. A chaque fois qu'un client se connectera au vpn, le serveur lui attribuera une adresse IP contenue dans cette plage.

- ✓ **Comp-lzo**

Bien vérifier en bas du fichier l'utilisation de la librairie lzo pour la compression des données.

- ✓ **User openvpn / group openvpn**

Utiliser l'utilisateur et son groupe openvpn qu'on a crée pour lancer le serveur.

Ensuite il faut sauvegarder et lancer le service par le script contenu dans /etc/init.d.

Le serveur est prêt a être utilisé. Il faut maintenant passer à la configuration du coté du client.

- **Configuration du client**

La configuration du client est simple. Il faut tout d'abord installer le client OpenVPN sur la machine. Ensuite il faut copier les fichiers suivant (Ca.crt, Client.crt, Client.csr et Client.key) qui se trouvent sous le répertoire /etc/openvpn du coté serveur.

Ensuite il faut configurer le fichier **client.conf** afin qu'il puisse reconnaître le serveur grâce a l'ajout de la ligne suivante dans le fichier :

```
Remote  172.16.64.26 1194
```

C'est l'adresse du serveur et le port sur lequel va s'effectuer la connexion VPN. Voila maintenant le réseau VPN est prêt à être utiliser entre le serveur Asterisk et ces clients.

3.2 Bonne pratique contre le DOS – BYE

Les bonnes pratiques contre les attaques DOS ne permettent pas aussi d'avoir une sécurité totale mais elle limite les attaques et minimise les vulnérabilités.

3.2.1 Implémentation d'un firewall Netfilter

Un firewall doit être imprenable car sinon le réseau entier est compromis. Un firewall efficace doit posséder plusieurs interfaces réseaux pour pouvoir faire un filtrage entre plusieurs zones.

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 objectif 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. C'est pour cela que nous avons choisi de mettre un firewall au niveau du serveur et de cette façon toutes les requêtes en direction du serveur Asterisk passeront automatiquement par le firewall. La plupart des firewalls propre a la VoIP sont a titre commercial et donc non libre. IPtable est la commande permettant de paramétrer le filtre Netfilter du noyau Linux et donc de configurer le Firewall.

Dans l'exemple qui suit nous avons programmé notre firewall pour qu'il puisse laisser passe seulement le trafic VoIP au niveau du serveur Asterisk et de bloquer tous le trafic restant. Voici les commandes exécutées :

```
# iptables -A INPUT -p udp -m udp --dport 5060 -j ACCEPT
```

Cette commande va permettre d'accepter le trafic UDP entrant du port 5060. Ce numéro de port n'est autre que celui du protocole SIP.

```
# iptables -A INPUT -p udp -m udp --dport 10000:20000 -j ACCEPT
```

Cette commande va permettre d'accepter le trafic UDP entrant du protocole RTP.

Ensuite il faut attribuer une règle par défaut pour bloquer tous le trafic restant et qui passe par UDP :

```
# iptables -A INPUT -p UDP -j DROP
```

En ce qui concerne le trafic TCP nous pouvons effectuer les règles suivantes pour ne laisser passer que le trafic de synchronisation en le limitant la réception des requêtes de synchronisation à une requête par secondes. Ainsi nous pouvons éviter les attaques de type SYN flood:

```
# iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT
```

3.2.2 Exécuter Asterisk sous un utilisateur non privilégié :

Parmi les bonnes pratiques pour sécuriser notre serveur Asterisk est de changer l'utilisateur sur lequel Asterisk tourne. Le principal objectif de cette sécurisation est si le serveur Asterisk est compromis au niveau de sa sécurité ceci ne doit en aucun cas affecter toute la machine sur laquelle tourne le serveur. Idéalement, la compromission ne devrait pas permettre d'éditer les fichiers de configuration. Voici les étapes à suivre pour ce changement :

Tous d'abord il faut arrêter le service Asterisk :

```
# /etc/init.d/asterisk stop #> Shutting down asterisk:                    OK
```

Ensuite il faut créer un utilisateur depuis lequel Asterisk va démarrer. Nous avons choisie le nom du groupe Asterisk et comme nom d'utilisateur Asterisk.

```
# /usr/sbin/groupadd asterisk

#/usr/sbin/useradd -d /var/lib/asterisk -g asterisk asterisk
```

Ensuite il faut attribuer les droits d'accès vu qu'Asterisk en a besoin. Les fichiers dans le répertoire /var/spool/asterisk doivent être la propriété de l'utilisateur Asterisk et accessibles en écriture. Les commandes suivantes sont celles qui ont été exécutées pour le répertoire /var/lib /asterisk. Les mêmes commandes sont appliquées pour les répertoires suivant : /var/log/asterisk, /var/run/asterisk, /var/spool/asterisk, /usr/lib/asterisk et le dossier /dev/zap.

```
# chown --recursive asterisk:asterisk /var/lib/asterisk (l'option --recursive permet de modifier les permissions d'un répertoire et de ses sous-répertoires. Ainsi grâce a la commande chown le propriétaire du répertoire /var/lib/asterisk er ses sous-répertoires est devenue asterisk)

# chmod --recursive u=rwX,g=rX, /var/lib/asterisk
```

Asterisk a besoins de lire le répertoire /etc/asterisk et son contenu afin de le modifier.

```
# chown --recursive root:asterisk /etc/asterisk

# chmod --recursive u=rwX,g=rX, /etc/asterisk
```

Ensuite il faut changer le répertoire d'Asterisk afin qu'il puisse démarrer du nouveau chemin crée:

```
# cp /etc/asterisk/asterisk.conf /etc/asterisk/asterisk.conf.org

# vi /etc/asterisk/asterisk.conf
```

Modifier le chemin en changeant la ligne suivante de:

```
astrundir => /var/run          à          astrundir => /var/run/asterisk
```

Ensuite il faut activer le nouveau groupe que nous venons de créer et le nouvel utilisateur aussi:

```
# cp /etc/init.d/asterisk /etc/init.d/asterisk.org

# vi /etc/init.d/asterisk
```

Changer la ligne suivante afin d'informer Asterisk de son nouveau utilisateur:

```
#AST_USER="asterisk"

#AST_GROUP="asterisk"

à

AST_USER="asterisk"

AST_GROUP="asterisk"
```

Maintenant il faut redémarrer Asterisk avec les nouveaux paramètres, à savoir le groupe Asterisk et le nom d'utilisateur Asterisk

```
/etc/init.d/asterisk restart

asterisk -U asterisk -G asterisk
```

Configuration des fichiers sip.conf et extensions.conf

Une autre bonne pratique pour mieux assurer la sécurité du serveur Asterisk est de crypter le mot de passe de l'utilisateur ou client. En effet grâce au cryptage le mot de passe du client devient illisible dans le cas ou une personne malveillante accède au fichier **sip.conf.** La commande suivante permet d'effectuer le cryptage :

```
echo - n "<user>:<realm>:<secret>" | md5sum
```

Voici un exemple pris de notre travail:

```
echo - n "<beshir>:<asterisk>:<beshir>" | md5sum
```

Le résultat de hachage est le suivant:

```
bed1e076ced1aadeba7e151240c7a955
```

Ce résultat est placé à la place de l'ancien mot de passe non crypté dans le fichier **sip.conf**.

Nous pouvons aussi assurer la sécurité du serveur en attribuent des privilèges et des limites d'accès au utilisateur et cela s'effectue au niveau du fichier sip.conf. En effet nous pouvons limiter les attaques de types DoS en limitant les requêtes d'invitation vers le serveur. Voici un exemple de configuration d'un utilisateur :

```
[general]

allowguest=no
```

Cette option permet de donner le droit ou non à des utilisateurs non authentifiés de s'enregistrer et de faire des communications. Dans notre cas on a interdit l'accès à toutes personnes non authentifié.

```
allowtransfer=no
```

Cette option permet d'activer ou de désactiver le transfert d'appel.

Il existe d'autres options qu'on peut activer au niveau des comptes des utilisateurs. Voici un exemple de l'utilisateur nommé « 100 » :

```
[100]

type=friend

md5secret=bed1e076ced1aadeba7e151240c7a955

host=dynamic

defaultip=192.168.254.128

canreinvite=no
```

Cette option permet de limiter les attaques de types DoS. En effet, un utilisateur légitime ne peut effectuer qu'une seule invitation durant un appel et donc il est impossible pour une personne malveillante de mettre en déni de service le serveur Asterisk avec les requêtes INVITE.

```
insecure=no
```

Cette option permet de modifier le degré de sécurité d'authentification. Pour avoir un maximum de sécurité, il faut toujours la mettre en NO. Dans ce cas, le serveur interrogera toujours pour l'authentification à chaque nouvelle connexion du client vers le serveur.

```
call-limit=1
```

Cette option permet de limiter le nombre d'appels sortant ou rentrant a un seul ce qui permet de contourner les attaques de types DoS visant le serveur Asterisk et aussi les utilisateurs.

```
dtmfmode=rfc2833

context=sip

callerid="100"<1111>
```

Conclusion

Tout au long de ce chapitre, nous avons pu voir les différentes attaques effectués au sein du réseau VoIP et les mesures de sécurités à prendre, afin de les éviter. 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

L'objectif de ce projet, après avoir établi des études sur voix sur IP et des études de la sécurité, est de sécuriser un réseau VoIP mis en place. L'étude consiste à effectuer des scénarios d'attaques sur le réseau et voir quelles sont les vulnérabilités existantes afin de sécuriser le réseau VoIP.

Dans une première étape, nous nous sommes intéressés à l'étude de cette technologie avec ses différents protocoles et standards. Dans une deuxième étape, nous avons étudié les problèmes de sécurité de la voix sur IP, les attaques, les vulnérabilités sur différents niveaux et les bonnes pratiques possibles pour les attaques cités. Comme troisième étapes, nous avons installé et configuré une solution de VoIP utilisant le serveur Asterisk et de deux clients x-lite. En dernière étape, nous avons testé des attaques de sécurité contre la solution installée, et nous avons proposé et implémenté des mécanismes et des protocoles pour la sécuriser.

Ce projet a été une expérience fructueuse qui nous a permis de mieux s'approcher du milieu professionnel. Cette expérience nous a permis de savoir comment gérer et optimiser le temps dans le but d'en profiter au maximum.

Acronyme

ACE = Access Control Entry

ACL = Access Control List

AH = Authentification Header

ARP = Address Resolution Protocol

CAN = Convertisseur analogique numérique

CLI = Command Line Interface

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

ESP = Encapsulated Security Payload

FTP = File Transfer Protocol

GSM = Global System for Mobile Communications

HTTP = HyperText Transfer Protocol

IAX = Inter-Asterisk eXchange

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

ISDN = Integrated Service Data Network

ITU = International Telecommunications Union

LAN = Local Area Network

MD5 = Message Digest 5

MIKEY = Multimedia Internet KEYing

MKI = Master Key identifier

NAT = Network Address Translation

OS = Operating System

PABX = Private Automatic Branch eXchange

PBX = Private Branch eXchange

PSTN = Public Switched Telephone Network

QoS = Quality of Service

RFC = Requests For Comment

RNIS = Réseau Numérique à Intégration de Service

RTC = Réseau Téléphonique de Commuté

RTCP = Real-time Transport Control Protocol

RTP = Real-Time Transport Protocol

RTSP = Real Time Streaming Protocol

SIP = Session Initiation Protocol

SNMP = Simple Network Management Protocol

SRTP = Secure Real-time Transport Protocol

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

UDP = User Datagram Protocol

URL = Uniform Resource Locator

VoIP = Voice over Internet Protocol

VPN = Virtual Private Network

WAN = World Area Network

Étude et Mise en place d'une Solution VOIP Sécurisée

Réalisés par : Rebha BOUZAIDA

Encadreur : Kamel KEDHIRI

Mots clés : VoIP, SIP, RTP, Asterisk, sécurité VoIP

Résumé : Dans le cadre de mon projet, je me suis intéressée à la protection des solutions de VoIP contre les attaques de sécurité.

Ce travail a pour objectif : l'étude des protocoles de VoIP et des architectures proposées ; l'étude des vulnérabilités et des attaques de sécurités sur les divers composants d'une infrastructure VoIP dans des réseaux LAN ; et la mise en place une solution de VoIP sécurisée basée sur des outils open source, précisément le serveur Asterisk et le client X-Lite.

Bibliographie

[1] Hacking Exposed VoIP: Voice Over IP Security Secrets & Solutions par David Endler et Mark Collier (McGraw-Hill/Osborne © 2007)

[2] VoIP Hacks Tips and Tools for Internet Telephony par Theodore Wallingford (O'Reilly © 2005)

[3] Security Considerations for Voice Over IP Systems par D. Richard Kuhn, Thomas J. Walsh, Steffen Fries (US National Institute of Standards and Technology © 2005)

[4] Asterisk The future of telephony par Jim Van Megglen, Leif Madsen & Jared Smith (O'Reilly © 2005)

[5] Securing VoIP networks threats, vulnerabilities, and counter measures par Peter Thermos and Ari Takanen (Addison-Wesley © 2007)

[6] http://www.frameip.com/voip/, Voix sur IP - VoIP, par SebF (date de dernier accès : 19/01/2008)

[7] http://ftp.traduc.org/doc-vf/gazette-linux/html/2003/097/lg97-C.html Configure, make, make install Linux Gazette n°97 — (date de dernier accès : 19/01/2008)

[8] http://www.securityfocus.com/infocus/1862 two attacks against VoIP par Peter Thermos. (Date de dernier accès: 19/01/2008)