



UNIVERSITE D'ANTANANARIVO

ECOLE SUPERIEURE POLYTECHNIQUE

MENTION TELECOMMUNICATION



MEMOIRE

en vue de l'obtention

du DIPLOME de MASTER

Domaine : Sciences de l'Ingénieur

Mention : Télécommunication

Parcours : Ingénierie des Réseaux et Systèmes

Par : RA-HARINANJA Dia

ANALYSE DE PERFORMANCE DE LA VOIX SUR IP

BASEE SUR LA PLATE-FORME RASPBERRY PI

Soutenu le 06 Août 2019 devant la Commission d'Examen composée de :

Président : M. RANDRIAMITANTSOA Andry Auguste, Maître de conférences, Docteur HDR

Examineurs :

M. ANDRIAMIASY Zidora, Maître de conférences

M. RAJAONARISON TIANANDRASANA Roméo, Maître de conférences

M. ANDRIAMANALINA Ando Nirina, Maître de conférences

Directeur de Mémoire :

M. RAVONIMANANTSOA Ndaohialy Manda-Vy, Maître de conférences, Docteur HDR

*« Tsy ny masoandro intsony no ho fanazavana anao nony andro ; ary tsy ny fangarangan'ny volana no hanazava anao ; fa Jehovah no ho fanazavana anao mandrakizay, ary Andriamanitrao no ho voninahitrao » **Isaia 60 : 19***

*« Mifalia amin'ny fanantenana ; miareta amin'ny fahoriana ; mahareta amin'ny fivavahana » **Romana 12 :12***

REMERCIEMENTS

Ayant donné la foi, la force et l'intelligence, je rends d'abord grâce à Dieu Tout Puissant car grâce à sa miséricorde et sa bénédiction divine, j'ai pu parvenir à ce stade de ma vie.

J'exprime ma gratitude envers Monsieur RAVELOMANANA Mamy Raoul, Professeur Titulaire-Agrégé, Président de l'Université d'Antananarivo.

Je remercie Monsieur RAKOTOSAONA Rijalalaina, Professeur d'Enseignement Supérieur, Responsable du domaine Science de l'ingénieur à l'école supérieure polytechnique d'Antananarivo.

Mes remerciements s'adressent aussi à Monsieur RAKOTONDRAINAH Tahiana Ezéchiél, Maître de conférences, Responsable de la mention Télécommunication de m'avoir accueilli au sein de la mention.

J'adresse mes plus profonds et sincères remerciements à Monsieur RAVONIMANANTSOA Ndaohialy Manda-Vy, Maître de conférences, Docteur HDR, qui, en tant que Directeur de ce mémoire, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu me consacrer et sans qui ce mémoire n'aurait jamais vu le jour.

Mes vifs remerciements s'adressent également à Monsieur RANDRIAMITANTSOA Andry Auguste, Maître de conférences, Docteur HDR qui me fait l'honneur de présider le jury de ce mémoire.

Je témoigne toute ma reconnaissance aux autres membres du jury qui ont bien voulu examiner la valeur de ce travail :

- Monsieur ANDRIAMIASY Zidora, Maître de conférences, pour son engagement à juger ce travail malgré ses lourdes responsabilités
- Monsieur RAJAONARISON TIANANDRASANA Roméo, Maître de conférences, qui a accordé son temps précieux pour juger le présent travail
- Monsieur ANDRIAMANALINA Ando Nirina, Maître de conférences, de me faire l'honneur d'accepter de juger ce mémoire.

Je tiens aussi à exprimer ma reconnaissance aux membres de ma famille, de m'avoir soutenu tout au long de mes études.

Enfin, je ne saurais oublier toutes les personnes qui m'ont amplement aidé de près ou de loin dans l'élaboration de ce présent mémoire.

TABLE DES MATIERES

REMERCIEMENTS	i
TABLE DES MATIERES	ii
NOTATIONS	vi
LISTE DES ABREVIATIONS.....	viii
INTRODUCTION GENERALE.....	1
CHAPITRE 1 LA TECHNOLOGIE DE VOIX SUR IP	3
1.1. Introduction	3
1.2. Quelques définitions	3
1.2.1. IP (<i>Internet Protocol</i>)	3
1.2.2. VoIP (<i>Voice Over IP</i>).....	4
1.3. Le principe de la Voix sur IP	5
1.3.1. <i>Traitement de la Voix</i>	5
1.3.2. <i>Principe IP</i>	5
1.3.3. <i>Principe de fonctionnement de VoIP</i>	6
1.4. Architectures de la voix sur IP	8
1.4.1. <i>Architecture PC 2 PC</i>	8
1.4.2. <i>Architecture PC 2 Phone (architecture Phone 2 PC)</i>	8
1.4.3. <i>Architecture Phone 2 Phone</i>	9
1.4.4. <i>Autre type d'architecture VoIP</i>	9
1.5. Les équipements relatifs aux VoIP	10
1.5.1. <i>Les terminaux VOIP</i>	10
1.5.2. <i>Le serveur de communication</i>	10
1.5.3. <i>Le routeur</i>	10
1.5.4. <i>Le Switch</i>	10
1.5.5. <i>Le MCU</i>	10

1.5.6. <i>La passerelle (Gateway)</i>	11
1.5.7. <i>Le Gatekeeper</i>	11
1.5.8. <i>Softphone</i>	12
1.5.9. <i>IP phone</i>	12
1.6. Asterisk	13
1.6.1. <i>Rôle</i>	13
1.6.2. <i>Caractéristiques</i>	14
1.7. Les atouts et points faibles de la VoIP	14
1.7.1. <i>Les atouts</i>	14
1.7.2. <i>Les points faibles du VoIP</i>	15
1.8. Conclusion	16
CHAPITRE 2 LA PLATE-FORME RASPBERRY PI	17
2.1. Introduction	17
2.2. Historique	17
2.3. Définition	18
2.4. Architecture et composant du Raspberry	19
2.4.1. <i>Le SOC</i>	19
2.4.2. <i>Le Port DSI et CSI</i>	20
2.4.3. <i>Le port GPIO</i>	20
2.4.4. <i>L'alimentation</i>	21
2.4.5. <i>Port USB</i>	21
2.4.6. <i>Bluetooth/ Wifi</i>	21
2.5.5. <i>Les spécificités du Raspberry PI 3 model B</i>	25
2.7. Atouts et points faible du Raspberry Pi.....	30
2.7.1. <i>Les atouts</i>	30
2.7.2. <i>Les points faibles</i>	30

2.9. Conclusion	33
CHAPITRE 3 LES METRIQUES DE PERFORMANCE DE LA VOIP.....	34
3.1. Introduction	34
3.2. Les protocoles.....	34
3.2.1. <i>Les protocoles de transports</i>	34
3.2.2. <i>Les Protocoles de signalisation</i>	36
3.2.3. <i>IAX (Inter Asterisk eXchange)</i>	49
3.3. Les codecs	50
3.3.1. <i>Définition</i>	50
3.3.2. <i>Caractéristiques et fonctions</i>	51
3.3.3. <i>Le codec G.711</i>	52
3.3.5. <i>Codec G722</i>	65
3.3.6. <i>Le codec G726</i>	65
3.4. Qualité de service.....	66
3.4.1. <i>Qualité sonore / Fiabilité</i>	66
3.4.2. <i>Génération de bruits de confort</i>	66
3.4.3. <i>Le délai de transit</i>	66
3.4.4. <i>Gigue</i>	67
3.5. Perte de paquets.....	68
3.6. Interopérabilité	68
3.7. Bande passante.....	68
3.8. CPU/RAM	69
3.9. Sécurité	69
3.9.1. <i>Principaux risques</i>	70
3.9.2. <i>Sécurisation protocolaire</i>	71
3.10. Prédiction de la qualité de service.....	72

3.10.1. <i>Score MOS</i>	72
3.10.2. <i>Calcul du facteur R de l'E-model</i>	73
3.11. Conclusion	74
CHAPITRE 4 ANALYSE DE PERFORMANCE	75
4.1. Introduction	75
4.2. Les outils utilisés	75
4.2.1. <i>Asterisk</i>	75
4.2.2. <i>Plate-forme Raspberry Pi</i>	75
4.2.3. <i>Outils Sipp</i>	76
4.2.4. <i>Wireshark</i>	77
4.3. Test de performance du système	77
4.3.1. <i>Scénario avec UAC</i>	77
4.3.2. <i>Génération des appels : UAC avec codec G711u</i>	84
4.3.3. <i>Test et comparaison avec des différents codecs</i>	92
4.4. Conclusion	94
CONCLUSION GENERALE	95
ANNEXES	96
ANNEXE I : PORTION DE UAC .xml	96
ANNEXE II : LES MOTS CLES DANS UN SCENARIO XML	98
ANNEXE III : LES OPTIONS DE SIPP	100
BIBLIOGRAPHIE	104
PAGE DE RENSEIGNEMENT	108
RESUME	109
ABSTRACT	109

NOTATIONS

Minuscules latines

a_i	Coefficients de la prédiction linéaire
$c(n)$	Vecteur d'un dictionnaire fixe
$ew(n)$	Erreur de la pondération perceptuelle
f_s	Fréquence d'échantillonnage
\hat{g}_c	Gain quantifié des dictionnaires fixes
\hat{g}_p	Gain quantifié des dictionnaires adaptatifs
$h(n)$	Réponse impulsionnelle
$u(n)$	Excitation du signal
v_1	Voltage d'entrée
v_2	Voltage de sortie
$v(n)$	Vecteur d'un dictionnaire adaptatif
$x(n)$	Signal filtré
$x_S(t)$	Signal échantillonné

Majuscules latines

A	Constante positive de la loi A
$\hat{A}(z)$	Equation de transfert du LP quantifié
D	Délai de transit
$F(f)$	Transformé de fourrier
G	Gigue de phase
Id	Coefficient de dommage dû au délai de transmission et de transport
Ie	Coefficient de dommage de distorsion causés par les équipements
Is	Coefficient de dommage simultané avec l'émission de la voix
P	Perte de donnée
$R(k)$	Retard de pas d'estimation en boucle ouverte
$R'(t_i)$	Pas en boucle ouverte
RO	Coefficient initial signal / bruit
$S_w(n)$	Signal pondéré
T_{bo}	Meilleur délai en boucle ouverte
T_s	Période d'échantillonnage
$X_S(f)$	Spectre du signal

Minuscules grecques

$\Delta_T(f)$	Transformé de Fourier de δ_T
γ_i	Facteur de pondération
$\delta_T(t)$	Impulsion de Dirac
μ	Constante positive de la loi μ

LISTE DES ABREVIATIONS

ACK	Acknowledgement
ADPCM	Adaptive Differential Pulse Code Modulation
ADSI	Active Directory Service Interfaces
ADSL	Asymmetric Digital Subscriber Line
API	Application Programming Interface
ARM	Acorn Risc Machine
ATA	Adaptateur de Téléphonie Analogique
AWS	Adaptive Wireless System
BBC	British Broadcasting Corporation
BCM	Bit Compression Multiplexer
BLE	Bluetooth Low Energy
BSD	Berkeley Software Distribution
CEC	Consumer Electronics Control
CELP	Code-Excited Linear Prediction
CNAME	Canonical NAME
CNG	Comfort Noise Generation
CPU	Central Processing Unit
CRM	Customer Relationship Management
CS-ACELP	Conjugate Structure Algebraic Code-Excited Linear Prediction
CSI	camera Sensor Interface
DDR	Double Data Rate
DECT	Digital Enhanced Cordless Telephone
DoS	Denial of Service
DOS	Disk Operating System
DSP	Digital Signal Processor
DTX	Discontinuous Transmission
ECN	Electronic Communications Network
EMMC	Embedded Multi Media Card
ETR	Estimated Time to Repair
ETSI	European Telecom Standard Institute
FPGA	Field Programmable Gate Array

GOS	Grade of Service
GPIO	General Purpose Input Output
GPL	General Public Licences
GPS	Global Positioning System
GSM	Global System for Mobile communication
HAT	Hardware Address Translation
HD	High Definition
HDMI	High Definition Media Interface
HTTP	Hyper Text Transport Protocol
I2C	Inter-Integrated Circuit
IAX	Inter-Asterisk-eXchange
ID	Identity/ Intrusion Detection
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IHM	Interface Homme Machine
IoT	Internet of Thing
IP	Internet Protocol
IPBX	Internet Private Branch eXchange
IPS	Intrusion Prevention System
IPsec	Internet Protocol SEcurity protocol
ISDN	Integrated Service Data Network
ISP	Internet Service Provider
ITU	International Telecommunication Union
JTAG	Joint Test Action Group
LAN	Local Area Network
LDLC	Laurent De La Clergerie
LED	Light Emitting Diode
LP	Linear Prediction
LPDDR	Low Power Double Data Rate
LSP	Line Spectral Pair
MC	Multipoint Controller
MCU	Multipoint Control Unit

MGCP	Media Gateway Control Protocole
MIC	Modulation par Impulsion et Codage
MICDA	Modulation par Impulsion et Codage Différentiel Adaptatif
MMC	Multi-Media Card
MOS	Mean Opinion Score
MP	Multipoint Processor
NAT	Network Address Translation
ONG	Organisation Non Gouvernementale
OS	Operating System
OSI	Open Systems Interconnection
PABX	Private Automatic Branch eXchange
PBX	Private Branch eXchange
PC	Personal Computer
PCAP	Ethereal Ethernet Packet sniffer file
PCM	Pulse Code Modulation
POTS	Plain Old Telephone Service
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RAM	Random Access Memory
RAS	Registration, Admission and Status
RasPi	Raspberry Pi
RCA	Radio Corporation of America
RNIS	Réseau Numérique à Intégration de Services
RSVP	Resource reSerVation Protocol
RTC	Réseau Téléphonique Commuté
RTCP	Real Time Control Protocol
RTP	Real Time Protocol
RTPC	Réseau Téléphonique Public Commuté
SD	Serial Disc
SDHC	Secure Digital High Capacity
SDIO	Secure Digital Input Output
SDK	Software Development Kit

SDP	Session Description Protocol
SDRAM	Synchronous Dynamic Random Access Memory
SIP	Session Initiation Protocol
Sipp	Simple Internet Protocol Plus
SMSC	Short Message Service Center
SoC	Système On a Chip
SPI	Serial Peripheral Interface
SSH	Secure Shell
SSRC	Synchronisation SouRCe identifier
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
ToIP	Telephony over Internet Protocol
UAC	User Agent Client
UART	Universal Asynchronous Receiver Transmitter
UAS	User Agent Server
UDP	User Datagram Protocol
UIT	Union Internationale des Télécommunications
UNIX	Uniplexed Information and Computing Service
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
VAD	Voice Activity Detection
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
WIFI	Wireless Fidelity

INTRODUCTION GENERALE

En raison de la croissance exponentielle de la pénétration de l'Internet et de la réduction des prix, les entreprises et les utilisateurs finaux utilisent de plus en plus la technologie IP pour satisfaire leurs besoins en communication. L'une de ces exigences est la communication vocale par le biais du service de téléphonie, qui était l'un des premiers et plus populaires services fournis sur des réseaux d'information. Ainsi, les solutions de voix sur IP ne sont pas seulement mise en œuvre dans les grandes entreprises mais également dans les petites entreprises. En effet, les systèmes de téléphonie IP peuvent désormais être installés dans des environnements domestiques en raison des coûts plus bas d'accès à la technologie.

Raspberry Pi est un ordinateur minuscule, de taille réduite et très bon marché. Il est une plate-forme utilisée pour déployer des applications de différentes sortes. Sa petite taille permet une consommation d'énergie grandement réduite et qui abaisse encore plus le coût global d'une solution basée sur Raspberry Pi. Cet appareil, basé sur l'architecture ARM, fonctionne avec les systèmes d'exploitation basé Linux. Tant de produits logiciels open source peuvent être compilés pour fonctionner les services réseaux, tels que la téléphonie IP. Bien que la plate-forme Raspberry Pi soit testée avec de nombreux projets intéressants, on ne sait pas encore comment les applications auraient un impact sur la performance globale de cette périphérique matériel. Les services de téléphonie IP impliquent un certain processus exigeant en terme de processeur, tels que la numérisation, la codification ou transcodage de la voix.

Ce mémoire intitulé « ANALYSE DE PERFORMANCE DE LA VOIX SUR IP BASEE SUR LA PLATEFORME RASPBERRY PI » consiste à montrer comment un petit matériel combiné à l'open source logiciel peut fournir des services vocaux aux entreprises. Par conséquent, cette étude prétend évaluer la performance de la voix sur IP déployée sur une plate-forme Raspberry Pi lorsqu'on fournit des services de téléphonie sur le réseau d'information. La méthodologie que nous avons utilisée inclut la création d'un réseau simple pour connecter la Plate-forme Raspberry Pi avec une instance de terminal vocal. Les charges des tests sont effectuées de l'instance de terminal au téléphone IP afin de déterminer les limites auxquelles le service est surchargé. Sipp a été utilisé pour générer dynamiquement

des appels via un service de téléphonie IP basé sur Asterisk qui était installé et configuré dans la plate-forme Raspberry Pi.

Pour mieux cerner le sujet, l'ouvrage s'articule en quatre chapitre : le premier chapitre concerne la technologie de Voix sur IP. Le second chapitre est consacré à une explication de la technologie Raspberry Pi. Pour le troisième chapitre, elle parlera des différentes métriques permettant d'évaluer la performance de la VoIP et la capacité de la Raspberry Pi.

Et pour terminer, pour le quatrième chapitre on montrera les résultats des analyses de performance.

CHAPITRE 1

LA TECHNOLOGIE DE VOIX SUR IP

1.1. Introduction

La Voix sur IP (en anglais, *Voice over IP* ou VoIP) est une nouvelle technologie de télécommunication vocale qui transforme la téléphonie. Cette technologie marque un tournant dans le monde de la communication en permettant de transmettre de la voix sur un réseau numérique et sur internet. C'est en 1996 que naquit la première version Voix sur IP, appelée H323. Depuis, la technologie Voix sur IP a progressé au fur et à mesure que les entreprises découvraient ses avantages pour accroître la productivité et l'efficacité de leurs réseaux. L'objectif de la Voix sur IP est d'appliquer à la voix le même traitement que les autres types de données circulant sur internet. Grâce au protocole IP, des paquets de données, constitués de la voix numérisée, y sont transportés. En effet, à force de transférer des fichiers d'information en temps de plus en plus réel, les utilisateurs d'internet en vinrent à transférer de la voix, en temps suffisamment réel pour faire compétition au téléphone. Dans cette banalisation des données voix, deux contraintes majeures sont présentes : transmettre ces paquets dans le bon ordre et le faire dans un délai raisonnable [1].

La téléphonie IP et la téléphonie mobile, deux technologies appelées à se généraliser au cours des prochaines années, auront un impact majeur sur la façon dont les gens communiquent, au bureau comme à la maison [2]. Ce chapitre a pour objectifs de fournir des renseignements permettant de mieux connaître cette nouvelle technologie, d'identifier ses atouts, ses inconvénients, et d'évaluer si elle constitue une voie d'avenir pour les centres de relations clientèles des grands organismes.

1.2. Quelques définitions

1.2.1. IP (*Internet Protocol*)

Internet Protocol, généralement abrégé IP, est un protocole de communication de réseau informatique, il correspond à un protocole de niveau 3 dans le modèle OSI et du modèle TCP/IP permettant un service d'adressage unique pour l'ensemble des terminaux connectés. C'est un protocole sans connexion qui, en tant qu'élément central de la famille des protocoles Internet (un ensemble d'environ 500 protocoles réseau), est responsable de l'adressage et de la fragmentation des paquets de données dans les réseaux numériques.

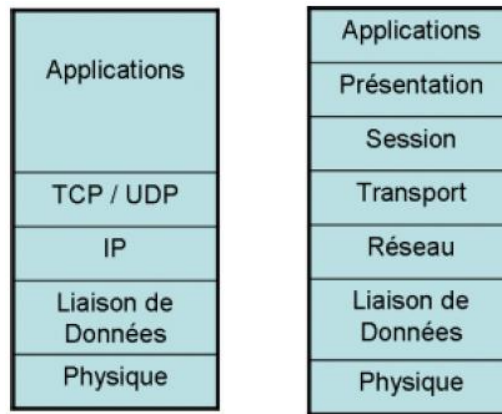


Figure 1.01 : Modèle OSI et TCP/IP

1.2.2. VoIP (Voice Over IP)

La voix sur IP (VoIP) regroupe l'ensemble des techniques permettant de faire transiter de la voix sur un réseau informatique. La voix sur IP comprend ainsi les communications de PC à PC. Pour ce type de communication, chaque utilisateur doit disposer d'un logiciel approprié. Si la connexion passe par le réseau Internet, on parle alors de VoIP, la téléphonie par Internet. Deuxième catégorie de voix sur IP, les communications de PC à téléphone (PC to Phone). Dans les deux cas, le PC communicant est appelé Softphone, terme qui insiste sur l'émulation du PC en téléphone grâce à un logiciel [1] [3].

La ToIP s'inscrit dans la troisième catégorie de communication en voix sur IP, les échanges de téléphone à téléphone. Les postes sont alors baptisés IP-Phone pour les distinguer de leurs homologues standards. Un téléphone IP doit en effet être alimenté par courant au contraire des téléphones classiques. Il est capable de numériser la voix pour la transmettre sur des réseaux IP et peut, à l'inverse, rassembler les paquets entrants pour interpréter la voix reçue. La téléphonie sur IP circule sur des réseaux privés LAN (Local Area Network), VPN (Virtual Private Network) ou publics [2].

Dans la littérature il arrive souvent de tomber sur le terme « ToIP » qui désigne les mêmes technologies que la VoIP, ToIP signifie « Telephony Over IP » ou, plus prosaïquement les techniques employées pour faire circuler la voix sur un réseau informatique dont la couche transport est prise en charge par le protocole IP. La ToIP et la VoIP sont 2 solutions professionnelles de téléphonie ne fixe pas toujours bien distinguées. Les 2 fonctionnent grâce au protocole internet IP, c'est au niveau de leur champ d'action que les différences sont à rechercher.

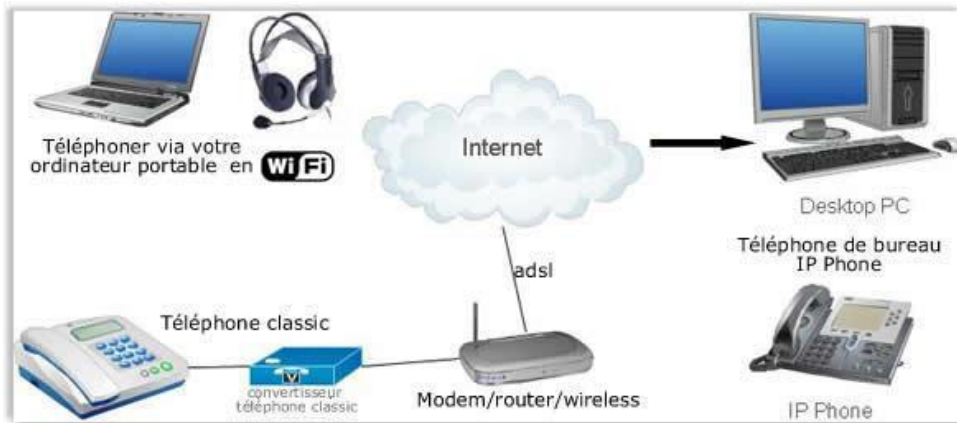


Figure 1.02 : Equipements utilisés pour VoIP

1.3. Le principe de la Voix sur IP

1.3.1. Traitement de la Voix

La numérisation et la compression de la voix sont assurées par le codec audio de l'émetteur. Les données numériques vont être acheminées vers le destinataire dans des paquets IP après la suppression de silence et l'ajout des en-têtes [4]. Ce processus est représenté par la figure suivante :

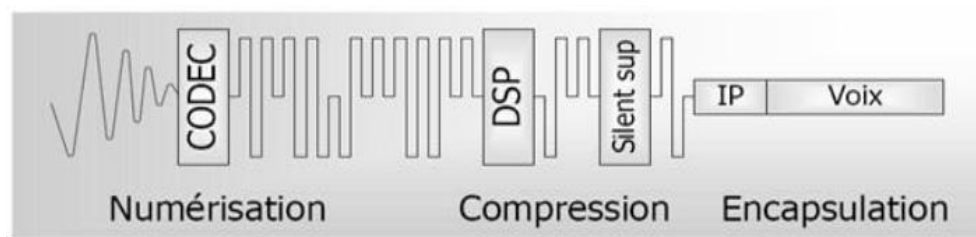


Figure 1.03 : Processus de traitement de la voix

1.3.2. Principe IP

Le protocole IP permet aux paquets de se déplacer sur Internet, indépendamment les uns des autres, sans liaison dédiée. Chacun d'entre eux, envoyé sur le réseau, se voit attribuer une adresse IP. Cette dernière est un en-tête accolé à chaque paquet et contenant certaines informations, notamment, l'adresse source, l'adresse destinataire, son temps de vie, le type de service, etc [5]. IP fournit un service de transport de paquet non fiable, sans connexion (datagrammes) :

- Non fiable : les paquets peuvent être perdus, dupliqués, retardés ou arrivés hors séquence.
- Sans connexion : pas de circuit virtuel. Des paquets successifs peuvent emprunter des chemins différents.

Le protocole IP ne garantit ni l'ordre d'arrivée des paquets à la destination, ni leur temps de transfert, ni l'échange d'information en temps réel. Ces propriétés sont très importantes pour la qualité de la voix et la qualité de la conversation. Il a donc fallu définir un nouveau protocole fournissant plusieurs fonctionnalités :

- Allouer un numéro de séquence pour la remise en ordre des paquets,
- Détecter la perte de paquets pour informer la source dans des délais compatibles avec le service
- Identifier le contenu des données et permettre leur transmission,
- Echanger l'information en temps réel.
- Intégrer des solutions pour traverser des passerelles de certains réseaux locaux.

En effet le protocole qui assure les fonctionnalités ci-dessous est le RTP (Real-Time Transport Protocol) qui se complète par un protocole de contrôle qui transmet des rapports de réception RTCP (Real-Time Transport Control Protocol) [3] [5].

1.3.3.Principe de fonctionnement de VoIP

La figure ci-dessous illustre de façon générale la topologie d'un réseau VoIP. 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 [6].

L'intelligence du réseau est aussi déportée soit sur les terminaux, soit sur les passerelles/Gatekeeper (contrôleur de commutation). On retrouve les éléments communs suivants :

- Le routeur : Il permet d'aiguiller les données et le routage des paquets entre deux réseaux. Certains routeurs, comme les Cisco 2600, permettent de simuler un gatekeeper grâce à l'ajout des cartes spécialisées supportant les protocoles VoIP.
- La passerelle : il s'agit d'une interface entre le réseau commuté et le réseau IP.
- Le PABX : c'est le commutateur du réseau téléphonique classique. Il permet de faire le lien entre la passerelle ou le routeur et le réseau RTC. Une mise à jour du PABX est aussi nécessaire. Si tout le réseau devient IP, on n'a plus besoin de ce matériel.
- Les Terminaux : des PC ou des téléphones VoIP.

On trouve aussi :

- Le point de présence du fournisseur d'accès (ISP) qui effectue l'adressage, l'acheminement et la tarification.

- Le centre de gestion qui apporte le soutien nécessaire à l'utilisateur et au réseau. Il contrôle l'exploitation des accès.
- Le nœud de service qui vérifie les messages d'authentification, effectue le tunneling, l'enregistrement des noms, le contrôle d'accès, celui de la QoS, l'acheminement, la gestion des ressources, les applications demandées par l'abonné distant, etc.

Pour transmettre les paquets, on utilise RTP, standardisé en 1996. C'est un protocole adapté aux applications présentant des propriétés temps réel. Il permet ainsi de reconstituer la base de temps des flux (horodatage des paquets : possibilité de re-synchronisation des flux par le récepteur), de détecter les pertes de paquets et en informer la source, et d'identifier le contenu des paquets pour leur associer un transport sécurisé [6] [7]. En revanche, ce n'est pas "la solution" qui permettrait d'obtenir des transmissions temps réel sur IP. En effet, il ne procure pas de réservation de ressources sur le réseau (pas d'action sur le réseau de type RSVP, Diffserv), de fiabilisation des échanges (pas de retransmission automatique, pas de régulation automatique du débit) et de garantie dans le délai de livraison (seules les couches de niveau inférieur le peuvent) et dans la continuité du flux temps réel. Bien qu'autonome, RTP peut être complété par RTCP. Ce dernier apporte un retour d'informations sur la transmission et sur les éléments destinataires. Ce protocole de contrôle permet de renvoyer à la source des informations sur les récepteurs et ainsi lui permettre, par exemple, d'adapter un type de codage ou encore de modifier le débit des données.

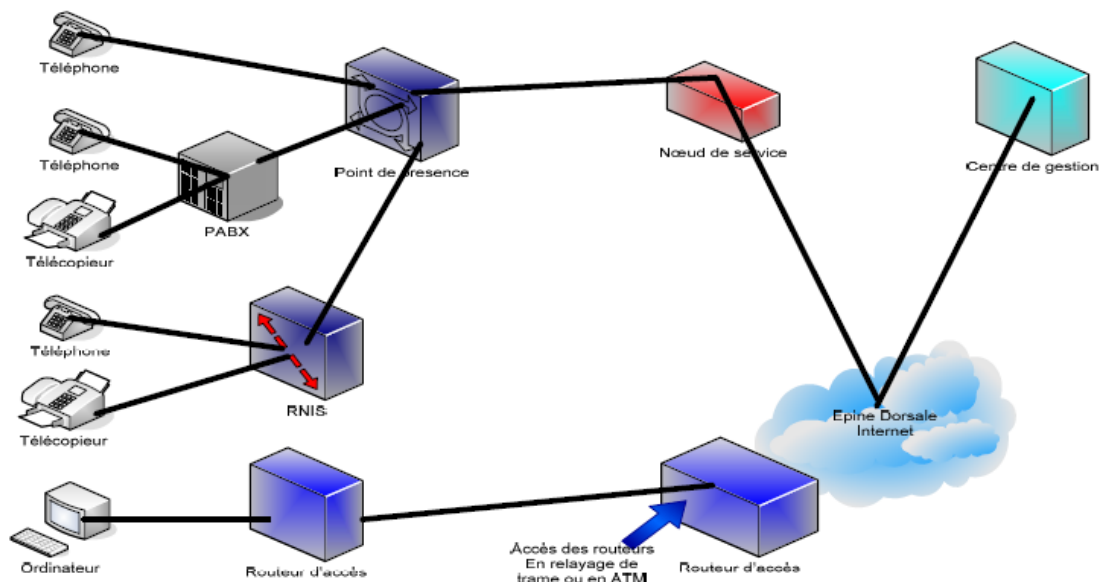


Figure 1.04 : Fonctionnement de la VoIP

1.4. Architectures de la voix sur IP

Il existe généralement 3 architectures utilisées pour la VoIP : La VoIP de PC 2 PC, PC 2 phone et finalement phone 2 téléphone.

1.4.1. Architecture PC 2 PC

Cette architecture nécessite deux ordinateurs équipés chacun d'un soft phone (logiciel de communication) et rattaché au réseau Internet par l'intermédiaire d'un fournisseur d'accès Internet qui assure le support physique de transmission des paquets ou le cas d'un réseau local ils sont rattaché à un PBX (logiciel Asterisk) [3] [8].

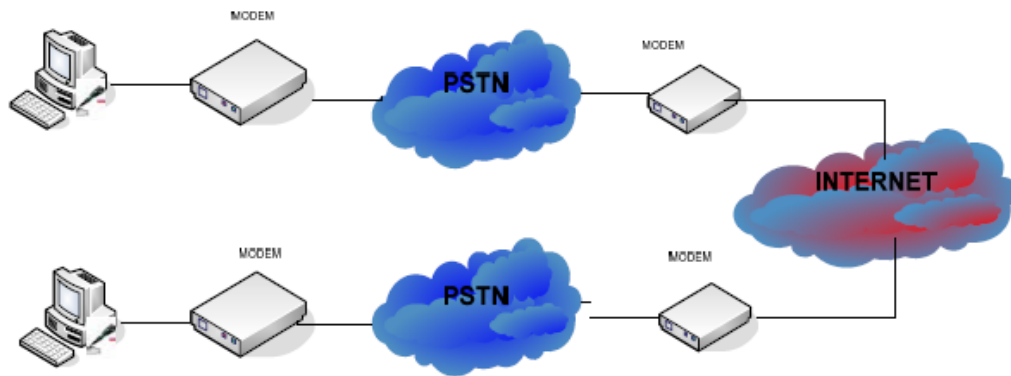


Figure 1.05 : Scenario PC 2 PC

1.4.2. Architecture PC 2 Phone (architecture Phone 2 PC)

Cette architecture assure l'interconnexion de deux types de réseaux différents où un participant utilise un PC rattaché au réseau Internet suivant un fournisseur d'accès Internet et l'autre membre utilise un téléphone classique rattaché au réseau téléphonique commuté. En effet le passage d'un réseau à un autre nécessite la présence des passerelles dédiées pour la conversion des paquets entre les réseaux.

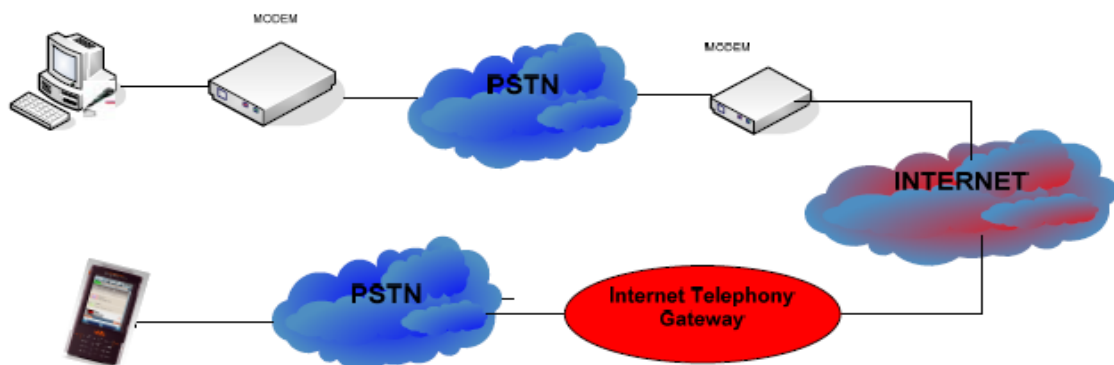


Figure 1.06 : Scenario Phone 2 PC

1.4.3. Architecture Phone 2 Phone

La configuration ne se borne pas à l'interconnexion de PBX, tous les éléments peuvent se contacter. Le réseau IP doit pouvoir supporter un certain niveau de qualité de service. L'utilisation d'une passerelle est primordiale de chaque côté entre le réseau téléphonique et le réseau des paquets IP afin de convertir la voix IP en voix et vice versa [8].

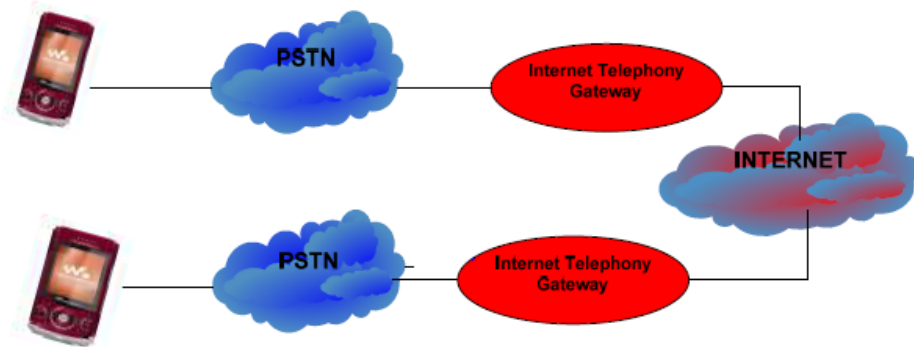


Figure 1.07 : Architecture Phone 2 Phone

1.4.4. Autre type d'architecture VoIP

Dans ce type d'architecture, le réseau IP, le réseau téléphonique et le RNIS sont présents en même temps. Dans ce cas, une passerelle est nécessaire pour relier les trois réseaux. Le réseau IP a besoin d'un contrôleur de communication (H323, SIP, MGCP). Les terminaux H320 sont connectés au RNIS [8]. On peut adjoindre des terminaux H324 et V70 au réseau téléphonique. Sur le réseau IP, il peut y avoir des terminaux VoIP et des téléphones VoIP.

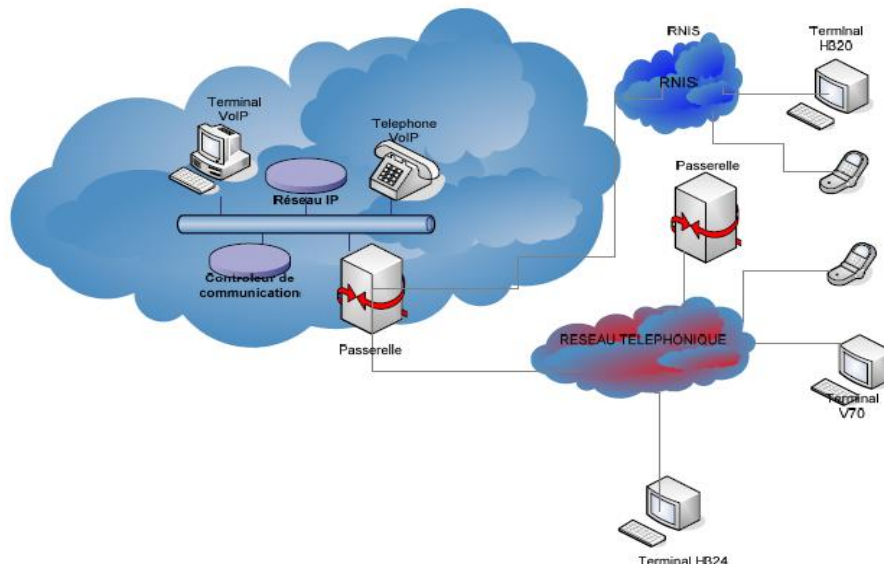


Figure 1.08 : Architecture incluant trois réseaux

1.5. Les équipements relatifs aux VoIP

Pour l'établissement des services utiliser VOIP on doit avoir les infrastructures, le logiciel, et les systèmes nécessaires. La figure ci-dessous décrit, de façon générale, la topologie d'un réseau de téléphonie IP [9].

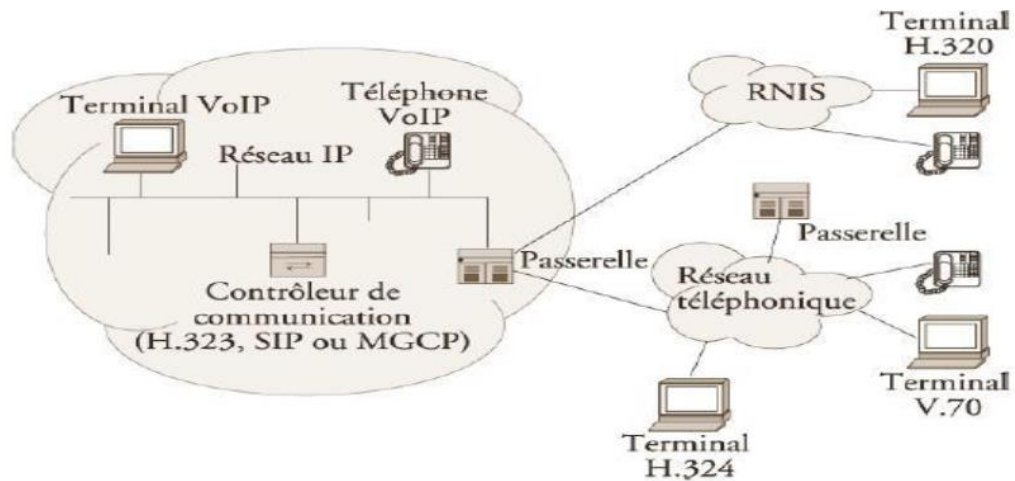


Figure 1.09 : Les équipements relatifs aux VoIP

1.5.1. Les terminaux VOIP

Ils sont des ordinateurs installant les logiciels supportant VOIP, des téléphones IP. Ils établissent, terminent des appels

1.5.2. Le serveur de communication

Il gère les autorisations d'appels entre les terminaux IP ou soft phones et les différentes signalisations du réseau. Il peut posséder des interfaces réseaux opérateurs (RTC-PSTN ou RNIS), sinon les appels externes passeront par la passerelle dédiée à cela (Gateway). Exemple : (Call Manager de Cisco)

1.5.3. Le routeur

Il assure la commutation des paquets d'un réseau vers un autre réseau.

1.5.4. Le Switch

Il assure la distribution et commutation de dizaines de port Ethernet à 10/100 voire 1000 Mbits/s. Suivant les modèles, il peut intégrer la télé alimentation des ports Ethernet à la norme 802.3af pour l'alimentation des IP-phones ou des bornes WIFI en 48V [1] [9].

1.5.5. Le MCU

Le MCU (multi conferences unit) est un élément optionnel et gère les conférences audio vidéo.

1.5.6. La passerelle (Gateway)

Un gateway permet aux terminaux d'opérer en environnements hétérogènes. Ces environnements peuvent être très différents, utilisant diverses technologies telles que le RNIS, la téléphonie commutée ou la téléphonie IP. Les gateways doivent aussi être compatibles avec les terminaux téléphoniques analogiques. Il fournit la possibilité d'établir une connexion entre un terminal analogique et un terminal multimédia (un PC en général). Beaucoup de sociétés fournissent des passerelles mais cela ne signifie pas que ces dernières fournissent le même service. Le gateway (partie physique) et le gatekeeper (partie logicielle) font l'objet de deux sections séparées pour bien cerner la différence. Certaines sociétés vendent un produit " gateway ", mais en réalité, elles incorporent un autre gateway du marché avec leur gatekeeper pour proposer une solution commerciale. La plus-value ne se fait pas sur la gateway mais sur le gatekeeper car c'est sur celui-ci qu'on peut faire la différence [9].

1.5.7. Le Gatekeeper

Il effectue les translations d'adresses (identifiant H323 et adresse IP du référencement du terminal) et gère la bande passante et les droits d'accès. C'est le point de passage obligé pour tous les équipements de sa zone d'action.

Il est aussi responsable de la sécurité. Quand un client veut émettre un appel, il doit le faire au travers du gatekeeper. C'est alors que celui-ci fournit une résolution d'adresse du client de destination. Dans le cas où il y a plusieurs gateways sur le réseau, il peut rediriger l'appel vers un autre couple gateway/gatekeeper qui essaiera à son tour de router l'appel. Pendant la résolution d'adresse, le gatekeeper peut aussi attribuer une certaine quantité de bande passante pour l'appel. Il peut agir comme un administrateur de la bande passante disponible sur le réseau. Le gatekeeper répond aux aspects suivants de la téléphonie IP :

- Le routage des appels : en effet, le gatekeeper est responsable de la fonction de routage. Non seulement, il doit tester si l'appel est permis et faire la résolution d'adresse mais il doit aussi rediriger l'appel vers le bon client ou la bonne passerelle.
- Administration de la bande passante : le gatekeeper alloue une certaine quantité de bande passante pour un appel et sélectionne les codecs à utiliser. Il agit en tant que régulateur de la bande passante pour prémunir le réseau contre les goulots d'étranglement.
- Tolérance aux fautes, sécurité : le gatekeeper est aussi responsable de la sécurité dans un réseau de téléphonie IP. Il doit gérer les redondances des passerelles afin de faire

aboutir tout appel. Il connaît à tout moment l'état de chaque passerelle et route les appels vers les passerelles accessibles et qui ont des ports libres.

- Gestion des différentes gateways : dans un réseau de téléphonie IP, il peut y avoir beaucoup de gateways. Le gatekeeper, de par ses fonctionnalités de routage et de sécurité, doit gérer ces gateways pour faire en sorte que tout appel atteigne sa destination avec la meilleure QoS possible.

Ainsi, le gatekeeper peut remplacer le PABX classique. En effet, il est capable de router les appels entrants et de les rediriger vers leur destination ou une autre passerelle. Mais il peut gérer bien d'autres fonctions telles que la conférence ou le double appel. Il n'existe pas les mêmes contraintes avec un gatekeeper qu'avec un PABX. En effet, le premier est constitué de logiciel et l'opérateur peut implémenter autant de services qu'il le désire [7]. D'ailleurs, l'amélioration des services d'un réseau de téléphonie n'a pas de limite. Alors qu'avec un PABX, l'évolutivité est limitée par le matériel propriétaire de chaque constructeur. En effet, chaque PABX utilise son propre protocole pour communiquer avec les postes clients, ce qui entraîne un surcoût. Avec le couple gateway/gatekeeper, ce problème n'existe pas.

1.5.8.Softphone

Un softphone est un type de logiciel utilisé pour faire de la téléphonie par Internet depuis un ordinateur plutôt qu'un téléphone. Les communications peuvent se faire au moyen d'un microphone et d'un casque ou d'hautparleur reliés à la carte son, mais il existe aussi un type de périphérique dédié à cette tâche, semblable à un téléphone et se branchant sur un port USB. Les interfaces des softphones sont souvent intuitives et de la forme d'un téléphone. Les fonctionnalités des softphones sont les mêmes que celles des téléphones classiques [7] [8]. En plus, les softphones incorporent souvent des services supplémentaires comme la vidéo sur IP, la présence, permettant de connaître la disponibilité de ses contacts et de nombreux autres services. Par ailleurs, l'application peut également être intégrée avec d'autres applications installées sur l'ordinateur : avec l'application e-mail par exemple afin de pouvoir appeler directement un numéro de téléphone dans un e-mail (click-to-call) ou d'accéder au gestionnaire de contacts pour générer des appels.

1.5.9.IP phone

Un téléphone IP utilise la technologie VoIP permettant des appels téléphoniques pour être sur un réseau IP comme Internet au lieu du système PSTN ordinaire. Les appels peuvent parcourir Internet, ou un réseau IP privé, tel que celui d'une entreprise. Les téléphones

utilisent des protocoles de contrôle tels que SIP (Session Initiation Protocol). Les téléphones IP logiciels peuvent être simples à base de Softphones. Les téléphones IP matériels ressemblent à un téléphone ordinaire ou à un téléphone sans fil. Les téléphones RTPC ordinaires sont utilisés comme des téléphones IP avec des Adaptateurs de Téléphonie Analogique (ATA).



Figure 1.10 : *L'application IP phone*

1.6. Asterisk

Le nom Asterisk fait référence au symbole "*" qui signifie "wildcard" en ligne de commande Unix et DOS. Ce choix a été fait car Asterisk est conçu dans le but d'offrir une très grande souplesse dans des réseaux de voix. Asterisk est un autocommutateur téléphonique privée (PABX) open source pour les systèmes d'exploitation UNIX, il est publié sous licence GPL. Le PBX open source Asterisk a vu le jour quand Mark Spencer, a voulu acquérir un PBX traditionnel pour sa société. Le créateur d'Asterisk, trouvant que le prix d'acquisition d'un PBX traditionnel était démesuré, initia un projet open source. Il a donc commencé à développer Asterisk.

1.6.1. Rôle

Asterisk est un IP-PBX qui transforme un ordinateur en "central téléphonique" ou "PABX" (Private Automatic Branch eXchange), qui est un autocommutateur téléphonique privé.

Ce PBX est un commutateur qui relie dans une entreprise les appels d'un poste quelconque vers un autre (appels internes) ou avec un réseau téléphonique public (appels externes). Asterisk a le rôle d'un middleware entre les technologies de téléphonie VOIP (TDM, SIP ...) et les applications (conférence, messagerie vocale, ...).

Ce PBX est basé sur le protocole IP. Donc les communications et les paquets échangés sont transportés sous forme plusieurs protocoles de la voix qu'on veut (SIP, IAX, H.323, ADSI, MGCP) [5].

1.6.2.Caractéristiques

Asterisk offre toutes les fonctions d'un PBX et ses services associés :

- La conférence téléphonique.
- Les répondeurs interactifs.
- La mise en attente d'appels.
- La distribution des appels.
- Les mails vocaux.
- La musique d'attente.
- La génération d'enregistrement d'appels pour l'intégration avec des systèmes de facturation.

Asterisk fonctionne sur les principaux systèmes d'exploitation (Linux, BSD, Windows, Mac OS X).

1.7. Les atouts et points faibles de la VoIP

1.7.1.Les atouts

Les solutions de switching dédiées à la VoIP sont en train de gagner la bataille commerciale par rapport aux solutions de téléphonie conventionnelle car elles offrent de nombreux avantages par rapport aux POTS classiques (« Plain Old Telephone Service »). Les avantages ne peuvent être égaux sont :

1.7.1.1. Le prix.

Les solutions VoIP sont moins coûteuses que les solutions classiques. Les services en ligne comme Skype permettent aux utilisateurs de téléphoner gratuitement à l'étranger. La VoIP destinée à des fins commerciales n'est pas gratuite, mais elle est nettement moins onéreuse que les solutions de téléphonie classique [3] [7].

1.7.1.2. La Qualité de service (QoS)

Auparavant le faible coût de la VoIP allait de pair avec la réduction de la QoS. Aujourd'hui, la technologie VoIP fournit une qualité de voix égale voir supérieure à la qualité des voix des services classiques.

1.7.1.3. Moins de complexité

Le passage à la VoIP réduit généralement la complexité liée aux fournisseurs de téléphonie notamment par rapport à la politique de prix.

1.7.1.4. Les caractéristiques

La VoIP possède maintenant les mêmes caractéristiques que la téléphonie classique allant des fonctions de bases, comme la messagerie vocale, à la détection de présence.

1.7.1.5. Intégration

De plus en plus d'applications d'entreprise comme les CRM et les applications pour le centre d'appel sont basées sur les normes techniques liées au web. La technologie VoIP peut être intégrée à ces applications beaucoup plus facilement que la technologie analogique de la téléphonie classique.

1.7.1.6. Facilité de gestion

L'ajout, la suppression ou la modification des numéros internes sont des processus simples grâce à une interface Web et aux fonctionnalités de découverte automatique. L'activation d'un nouveau téléphone s'effectue de manière quasi automatique [4].

1.7.1.7. Avantages pour le déploiement

La VoIP peut être intégrée à une infrastructure réseau existante comprenant le câblage physique. Cela implique un coût de déploiement et maintenance moindre (pas de nouveau câblage).

1.7.1.8. Pas de risque d'obsolescence

La VoIP est une technologie d'avenir. Les entreprises qui adoptent la VoIP ne risquent de se retrouver avec un système de téléphonie obsolète dans un monde de plus en plus axé vers le numérique.

1.7.2. Les points faibles du VoIP

Les points faibles de la voix sur IP sont :

1.7.2.1. 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 problématique.

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 [2] [7].

1.7.2.2. 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 des 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 efficacité ou à encourir les conséquences.

1.7.2.3. Vol

Les attaquants qui parviennent à accéder à un serveur VoIP peuvent également accéder aux messages vocaux stockés et même au service téléphonique pour écouter des conversations ou effectuer des appels gratuits aux noms d'autres comptes.

1.7.2.4. 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.

1.8. Conclusion

La VoIP est une technologie révolutionnaire qui défie les règles édictées par la téléphonie RTC. Elle est plus souple, conviviale, ne nécessite pas un investissement lourd, coûte moins chère, propose de nouveaux services et beaucoup d'autres avantages, si bien que toute entreprise qui se veut compétitive et moderne aujourd'hui, jette son dévolu sur la téléphonie sur IP pour gérer ses communications tant internes qu'externes. Elle vise principalement à améliorer le cadre de travail des employés de l'entreprise en libérant l'utilisateur du lieu d'implantation du poste téléphonique.

CHAPITRE 2

LA PLATE-FORME RASPBERRY PI

2.1. Introduction

Le concept de RasPi a été dévoilé autour de 2006 par Eben Upton et ses collègues de laboratoire d'informatique de l'université de Cambridge, en Angleterre. Ils étaient préoccupés par la baisse de niveau de connaissance et de compétence des nouveaux étudiants en informatique en comparaison de leurs aînés. Eben Upton a alors décidé de créer un ordinateur à faible prix, car il supposait que les parents craignaient de laisser leurs enfants jouer avec des PC modernes relativement coûteux. C'est de cette idée qu'est parti le développement du RasPi très bon marché. Cet ordinateur devait donner aux jeunes l'opportunité d'apprendre et de pratiquer la programmation, sans que leurs parents ne s'inquiètent des dégâts éventuels sur la machine [10].

Eben Upton a réuni plusieurs personnes pour constituer la fondation Raspberry Pi. Cette association caritative, enregistrée au Royaume-Uni, avait pour objectif de promouvoir l'esprit informatique et l'intérêt pour cette discipline, en particulier chez les plus jeunes, en utilisant le RasPi comme plateforme de départ. Elle semble avoir atteint ce but louable, puisque les ventes du RasPi initialement estimées à 10 000 unités ont été largement dépassées ; elles approchent aujourd'hui le million d'exemplaires.

2.2. Historique

Contrairement à la plupart des technologies disponibles sur le marché, le Raspberry Pi n'est pas le fait d'une multinationale. On doit cet ordinateur à la Fondation Raspberry Pi. Cette organisation caritative, vise à promouvoir l'apprentissage de la programmation dans les écoles.

Elle est très rapidement soutenue dans son action par l'université de Cambridge et par le fabricant de semi-conducteur Broadcom. Il faut dire que parmi les 6 administrateurs, Eben Upton, l'un des fondateurs de Raspberry Pi n'est autre qu'un ancien employé de la firme, architecte de système sur puces (SoC) du fondeur britannique. Par ailleurs, la visibilité de l'ONG a grandement bénéficié de la participation de David Braben. Ce développeur s'est fait connaître dès 1984 comme le deuxième homme avec Ian Bell derrière Elite, un jeu vidéo de simulation spatiale, l'un des plus innovants de l'histoire de ce médium. Il est toujours le

dirigeant de l'entreprise Frontier Developments, dont les derniers succès en date Elite : Dangerous et Planet Coaster.

En 2009, les fondateurs de la fondation Raspberry Pi travaillent déjà le développement de la monocarte, elle est le moyen de faciliter l'accessibilité de la programmation aux personnes au budget très limité. Dès 2006, les premiers prototypes du Raspberry Pi sont en développement. Les concepteurs disent « s'inspirer de l'architecture du BBC Micro », un ordinateur conçu par Acorn Computer en 1981. Dans cette phase de conception, le Raspberry Pi repose sur des microcontrôleurs 8 Bit ATmega644.

Dès la phase de prototypage en 2011, les concepteurs du Raspberry Pi s'approprient l'architecture ARM. À ce moment du projet, la monocarte prend la forme d'une clé USB avec un port de ce type à une extrémité et un port HDMI de l'autre. Il est rapidement prévu de commercialiser deux modèles aux performances légèrement différentes : A et B. Le premier est estimé à 25 dollars, tandis que le second est annoncé à 35 dollars. Dès août 2011, les développeurs mettent au point 50 versions Alpha du Raspberry Pi Model B. Liz Upton, directeur de la communication de l'ONG, publie alors des vidéos sur sa chaîne YouTube pour montrer les possibilités de la "bête". Il y montre la carte sur laquelle est installé un système d'exploitation Debian [10] [11] .

2.3. Définition

Raspberry Pi est un petit ordinateur sous le système d'exploitation Linux sur carte SD destiné à des applications d'informatique embarquée. Le cœur de l'ordinateur est un FPGA (Broadcom 2835) intégrant un processeur ARM11 cadencé à 700MHz et de nombreux périphériques.

Raspberry Pi peut être directement connecté à une IHM classique, souris/clavier/écran HDMI ou vidéo composite, cependant comme tout ordinateur Linux, Raspberry Pi peut intégrer ses propres outils de développement et une interface homme-machine reposant sur SSH contrôlable depuis un autre ordinateur par Ethernet ou WIFI.

Le connecteur d'extension supporte les entrées/sorties parallèles ainsi que la plupart des bus de communication. C'est un support particulièrement économique et puissant qui peut être facilement mis en œuvre dans de petits systèmes nécessitant un accès au monde physique par des capteurs/actionneurs disposants d'interfaces numériques [12].

2.4. Architecture et composant du Raspberry

Dans cette partie, on parlera tous les composants de cette technologie et on va expliquer respectivement ses rôles [11].

La figure 2.01 donne une aperçue de la Raspberry Pi 3 modèle B.

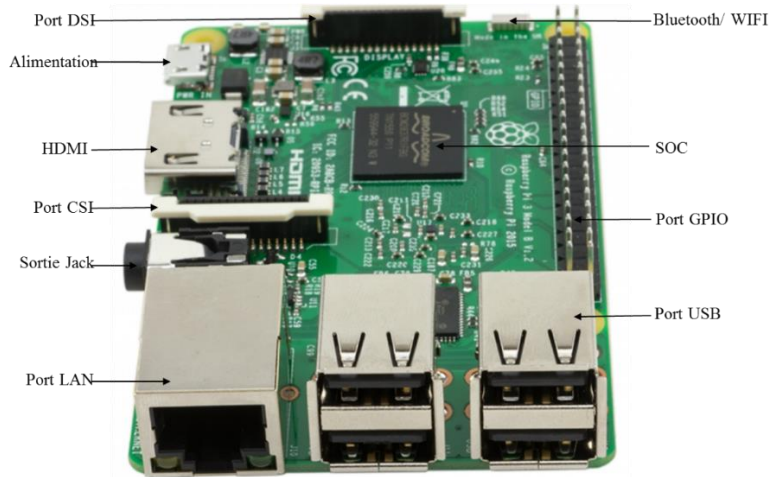


Figure 2.01 : Composants du Raspberry Pi

2.4.1. Le SOC

La pièce centrale du Raspberry Pi est le SOC (Système On a Chip). Ce circuit contient tous les composants nécessaires au fonctionnement d'un ordinateur à savoir :

- Le processeur
- La mémoire
- Le processeur graphique



Figure 2.02 : Le SOC du Raspberry Pi

Si on regarde de plus près, on s'apercevra que le SOC du Raspberry Pi contient plusieurs « couches ». Entre chaque « couche », on visualise des points de soudure. En réalité, le SOC est ici un empilement de circuits électroniques [11].

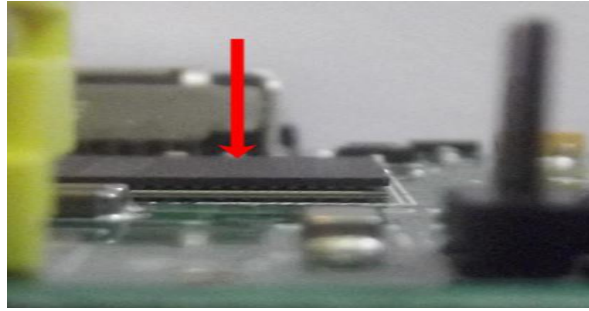


Figure 2.03 : *Empilement sur le SOC*

Construit spécialement pour le nouveau Pi 3, le système sur puce Broadcom BCM2837 comprend :

- Quatre cœurs de traitement ARM Cortex-A53 hautes performances fonctionnant à 1,2 GHz avec 32 Ko
- Mémoire cache de niveau 1 et 512 Ko
- Processeur graphique VideoCore IV, et est liée à un module de mémoire LPDDR2 de 1 Go à l'arrière de la carte.

2.4.2. Le Port DSI et CSI

Le Port DSI permet de connecter un écran tandis que le port CSI permet de brancher une webcam. Il s'agit ici d'une entrée/sortie vidéo.

2.4.3. Le port GPIO

Le port GPIO permet de faire communiquer le Raspberry Pi avec le monde extérieur. Il est, par exemple, possible d'utiliser les broches (ou pin) GPIO pour alimenter un ventilateur pour refroidir le SOC. On peut également y connecter des capteurs ou tout type d'actionneur.

Le Raspberry Pi 3 dispose du même en-tête d'entrée-sortie (GPIO) à 40 broches que tous les Pis remontant au modèle B + et au modèle A +. Tout matériel GPIO existant fonctionnera sans modification ; le seul changement est un commutateur auquel UART est exposé sur les broches, mais cela est géré en interne par le système d'exploitation [12] [13].



Figure 2.04 : *Le port GPIO*

2.4.4.L'alimentation

L'alimentation de la carte se fait par micro-USB. L'adaptateur doit être en 5V, 1A. Pour une utilisation minimale, un adaptateur 5V, 700mA peut suffire. Il est possible de trouver des adaptateurs de téléphone portable ayant ces caractéristiques. L'adaptateur doit supporter une intensité de 1A mais un adaptateur 5V, 1.2A conviendra tout aussi bien.

2.4.5.Port USB

Les quatre ports USB intégrés du Raspberry Pi 3 offrent une connectivité suffisante pour une souris, un clavier ou tout autre élément dont le RPi a besoin, mais si vous souhaitez en ajouter davantage, vous pouvez toujours utiliser un concentrateur USB. N'oubliez pas qu'il est recommandé d'utiliser un concentrateur alimenté afin de ne pas surcharger le régulateur de tension intégré. Il est facile d'alimenter le Raspberry Pi 3, il suffit de brancher toute source d'alimentation USB sur le port micro-USB. Comme il n'y a pas de bouton d'alimentation, le Pi commencera à démarrer dès que celui-ci sera mis sous tension. Pour le désactiver, il suffit de le mettre hors tension. Les quatre ports USB intégrés peuvent même produire jusqu'à 1,2 A, ce qui vous permet de connecter davantage de périphériques USB gourmands en énergie (cela nécessite une alimentation micro USB 2Amp) [14].

2.4.6.Bluetooth/ Wifi

Il n'est pas nécessaire de connecter une antenne externe au Raspberry Pi 3. Ses radios sont connectées à cette antenne puce soudée directement à la carte, afin de conserver la taille de l'appareil au minimum. Malgré sa petite taille, cette antenne devrait être plus que capable de capter les signaux LAN sans fil et Bluetooth, même à travers les murs.

2.5. Les modèles de raspberry Pi

Dès le commencement de la franchise Raspberry Pi, les concepteurs l'ont pensé sous le mode de la déclinaison. Il ne s'agit pas de commercialiser et mettre à jour une unique plateforme technologique comme le robot Nao, mais de décliner les modèles pour répondre aux besoins des utilisateurs et aux différents cas d'usage qui se présente [14] [15]. Entre 2012 et février 2017, huit versions du Raspberry Pi ont été commercialisées. Les voici :

- Le Raspberry Pi 1B à la fin février 2012 à 35 dollars
- Le Raspberry Pi A en février 2013 à 25 dollars
- Le Raspberry Pi 1B+ en juillet 2014 à 35 dollars
- Le Raspberry Pi A+ en novembre 2014 à 20 dollars

- Le Raspberry Pi 2B en février 2015 à 35 dollars
- Le Raspberry Pi Zero en novembre 2015 à 5 dollars
- Le Raspberry Pi 3B en février 2016 à 35 dollars
- Le Raspberry Pi Zero-W en février 2017 à 10 dollars.

Les premiers modèles B bénéficient de 512 Mo de mémoire vive, tout comme les modèles zéro, tandis que les modèles A et A+ dispose de 256 Mo. Les modèles 2B et 3B bénéficient eux d'un 1 Go de mémoire vive et d'un processeur quadri coeur ARM.

Précisons que certains modèles sont obsolètes comme le 1B, le A et A+. Les concepteurs préfèrent les plateformes 2B, 3B et Zero-W. De par son prix et ses capacités le modèle Zero commercialisé 5 dollars était particulièrement apprécié. Il n'est plus disponible. Finalement les Raspberry Pi les plus utilisés sont le modèle 3B et Zero-W. Le premier est disponible à 35 dollars aux États-Unis et à environ 40 euros en Europe. Le second coûte 10 dollars aux États-Unis, soit 11 euros en Europe. Le Raspberry Pi 3 est taillé pour la plupart des applications. Sa taille (85,60 mm × 53,98 mm × 17 mm), son poids (45 grammes) et ses performances en font un allié idéal dans une phase de prototypage. Le modèle Zero W se veut plus léger. Sa taille réduite et sa connectivité en font un composant conçu pour développer de petits produits comme des wearables, des portes-clés connectés, des caméras de surveillance sans fil, etc.

2.5.1. Modèles A

2.5.1.1. Raspberry Modèle 1 A

- Processeur : ARM1176JZF-S (ARMv6) 700 MHz Broadcom 2835 (dispose d'un décodeur Broadcom VideoCore IV, permettant le décodage H.264 FullHD 1080P et d'un VFPv2 pour le calcul des opérations à virgule) ;
- RAM : 256 Mo ;
- 2 Sorties vidéo : Composite et HDMI ;
- 1 Sortie audio stéréo Jack 3,5 mm (sortie son 5.1 sur la prise HDMI) ;
- Unité de lecture-écriture de carte mémoire : SDHC / MMC / SDIO ;
- 1 Port USB 2.0 ;
- Prise pour alimentation Micro-USB (consommation : 400 mA + périphériques) ;
- Des entrées / sorties supplémentaires sont accessibles directement sur la carte mère via des pins 3v3 : GPIO, S2C, I2C, SPI ;

- API logicielle vidéo : OpenGL : version embarquée OpenGL ES 2.0 ;
- Décodage vidéo : 1080p30 H.264 high-profile.



Figure 2.05 : *Raspberry Modèle 1 A*

2.5.1.2. *Raspberry Modèle 1 A+*

- Les différences avec le modèle A sont :
- Plus petit : 65 mm de long (contre 86 mm)
- Lecteur de carte microSD en lieu et place du lecteur SD
- GPIO 40 broches
- Nouveau chipset audio
- Consommation électrique moindre

2.5.2. *Modèles B*

2.5.2.1. *Modèle 1 B*

Il existe plusieurs révisions du modèle B

Spécificités du modèle B rev1 :

- 2 ports USB 2.0 au lieu de l'unique port du modèle A, mais sur un seul bus, via le composant SMSC LAN9512 ;
- 1 port réseau Fast Ethernet (10/100 Mbit/s) via le même composant SMSC.

Le circuit LAN9512 qui gère les deux ports USB2 et le port réseau, est connecté au CPU via un unique port USB2 ; la bande passante est donc partagée entre ces trois ports[15].

Spécificités du modèle Rev1 + ECN0003 :

- Suppression des fusibles protégeant les sorties USB
- Suppression de la diode D14, qui pouvait provoquer des interférences avec des périphériques possédant une broche CEC, lorsque le Raspberry restait connecté sans être alimenté.

Spécificités du Raspberry Pi B Rev2 :

- Implantation du reset (en reliant les broches 1 et 2 de P6)
- Support JTAG (deux broches GPIO interchangeables)
- Support I2C (canaux primaire et secondaire inversés)
- Suppression de quatre signaux GPIO utilisés pour l'identification de version, et réaffectation à d'autres rôles
- SMSC +1V8
- Deux trous de fixation
- Correction du marquage des LED sur la platine

2.5.2.2. *Modèle 1 B+*

Ce modèle est annoncé en juillet 2014. Les différences par rapport au modèle initial sont cités ci-dessous :

- GPIO 40 broches
- 4 ports USB 2.0 et meilleur comportement en cas de surcharge
- Micro SD
- Réduction de consommation de 3,5 W à 3 W
- Meilleur circuit audio
- Suppression de la prise RCA au profit d'une prise mini-jack 4 points, comprenant une sortie sonore et vidéo

2.5.2.3. *Modèle 2 B (Raspberry Pi 2)*

Le 2 février 2015, la fondation Raspberry Pi annonce la sortie du Raspberry Pi 2, plus puissant, il est équipé d'un processeur Broadcom BCM2836, quatre cœurs ARMv7 à 900 MHz, accompagné de 1 Go de RAM.



Figure 2.06 : *Modèle 2 B (Raspberry Pi 2)*

Il possède les mêmes dimensions et la même connectique que le modèle B +

2.5.3. Modèle 3 B (Raspberry Pi 3B)

- Carte mère : Broadcom BCM2837
- Processeur : ARM Cortex-A53 Quadricœur 1,2 GHz
- Mémoire Vive : 1 Go
- Carte graphique : Broadcom VideoCore IV (Affichage : 1080p)
- Ethernet, Bluetooth 4.1 et WiFi 802.11n
- Slot : Carte micro SD



Figure 2.07 : Modèle 3 B (Raspberry Pi 3B)

2.5.4. Modèle Zero

2.5.4.1. Modèle Zero W (Raspberry Pi Zero W)

- Carte mère : Broadcom BCM2835
- Processeur : ARM1176JZF – S core 1 GHz
- Mémoire Vive : 512 Mo
- Carte graphique : Broadcom VideoCore IV (Affichage : 1080p 30 sec)
- Ethernet, Bluetooth 4.1 et WiFi 802.11n
- Slot : Carte micro SD

2.5.4.2. Modèle Zero WH (Raspberry Pi Zero WH)

Le 15 janvier 2018, Raspberry Pi sort un nouveau modèle Zero, le Raspberry pi Zero WH. La seule différence avec le précédent modèle est la présence de ports GPIO soudés sur la Raspberry Pi.

2.5.5. Les spécificités du Raspberry PI 3 model B

Les principales différences sont le processeur 64 bits quadricœur et les technologies Wi-Fi et Bluetooth intégrées [17]. La RAM reste 1 Go et il n'y a aucun changement sur les ports

USB ou Ethernet. Cependant, la gestion de l'énergie améliorée devrait permettre au Pi 3 d'utiliser davantage de périphériques USB gourmands en énergie.

Pour Raspberry Pi 3, Broadcom nous ont pris en charge avec un nouveau SoC, BCM2837. Cela conserve la même architecture de base que ses prédécesseurs BCM2835 et BCM2836, de sorte que tous les projets et didacticiels qui reposent sur les détails précis du matériel Raspberry Pi continueront de fonctionner. Le complexe de processeurs ARM Cortex-A7 quadricœur 32 bits à 900 MHz a été remplacé par un processeur ARM Cortex-A53 à 1,2 GHz 64 bits durci sur mesure.

En termes de taille, il est identique au B + et au Pi 2. Tous les connecteurs et les trous de montage se trouvent au même endroit. Tous les compléments, HAT et boîtiers existants doivent donc être parfaitement ajustés, bien que les voyants d'alimentation et d'activité se soient déplacés pour laisser de la place pour l'antenne WiFi.

Les performances du Pi 3 sont environ 50 à 60% plus rapides que celles du Pi 2, ce qui signifie qu'il est dix fois plus rapide que le Pi d'origine.

Tous les connecteurs sont au même endroit et ont les mêmes fonctionnalités, et la carte peut toujours être exécutée à partir d'un adaptateur d'alimentation micro-USB 5V. Cette fois-ci, nous recommandons un adaptateur 2,5 A si vous souhaitez connecter des périphériques USB gourmands en énergie au Raspberry Pi.

2.6. Les concurrents du Raspberry Pi

Au même moment ou en réaction du succès du Raspberry Pi, bon nombre de constructeurs et de concepteurs ont souhaité proposer des alternatives aux différentes versions de la mono carte. Chacun des produits alternatifs tente de se spécialiser dans un cas d'usage. Voici 5 nano-ordinateurs concurrents au Raspberry Pi [16] [18].

2.6.1.Odroid

Odroid est une alternative à Raspberry Pi développée par l'entreprise sud-coréenne HardKernel. Le nom du projet correspond à la contraction de Open et Android, pour autant certains éléments de ses mono cartes sont soumis à la propriété intellectuelle. Les cartes Odroid disposent de capacités similaires aux produits de la fondation à la framboise. La grande différence repose sur l'emploi de processeur Exynos, des chipsets conçus par Samsung. Odroid a su couper l'herbe sur le pied de la fameuse ONG en sortant son premier produit en 2009, mais ce n'est qu'en 2012 qu'Odroid a réussi à dépassé techniquement

Raspberry Pi. Depuis la création de la gamme, Odroid a commercialisé 11 versions de sa mono carte.



Figure 2.08 : Odroid

Trois versions sont actuellement en vente : Odroid C1+, Odroid XU4 et Odroid-C2. Cette dernière version est plus puissante que le Raspberry Pi 3B. L'Odroid-C2 dispose également d'une carte mère ARM 64 Bits sur lequel HardKernel a greffé un SoC quatre coeurs cadencés à 1,5 GHz Amlogic S905 (Cortex A53 Armv8 64bits) et 2 Go de mémoire vive DDR3. Les débits du module Ethernet natif atteignent 900 Mbit/s contre 100 Mbit/s. En termes de stockage un port EMMC assure une vitesse de lecture et d'écriture d'au moins 120 Mo/s. La carte vendue 46 dollars est plus chère et ne dispose ni de module WiFi et ni de Bluetooth.

2.6.2. Banana Pi

Le choix du nom de cette monocarte fait penser à une mauvaise copie chinoise. Alors oui, le Banana Pi est un produit conçu à Shenzhen en Chine, mais il s'agit plutôt d'un clone décliné en plus de 14 versions par l'entreprise SINOVOIP. Commercialisé depuis 2014, les Banana Pi sont majoritairement équipés de SoC fabriqué par le semi-conducteur AllWinner. Le modèle Banana Pi M2 Zero est un clone du Pi Zero-W avec une puissance légèrement supérieure grâce à son processeur Allwinner H2+ ARM Cortex A7, un quadricoeur cadencé à 1GHz. Pour le reste, la carte dispose de capacité identique pour un prix supérieur : 13 euros contre 11 euros, sans les frais de port [10].

Le concurrent de la Raspberry Pi 3B chez Banana Pi se nomme Banana Pi BPI 64. Cette monocarte 64 Bits s'équipe d'un processeur quatre coeurs Allwinner Cortex A-53 1,2 GHz, de 2Go de mémoire vive LPDDR3, d'un support MicroSD jusqu'à 256 Gigaoctets et jusqu'à 64 Gigaoctets de stockage eMMC (8Go de base). Ici, Réseau Ethernet Gigabit, Wifi, et Bluetooth 4.0 sont présents pour assurer la connectivité du nano-ordinateur. En revanche, il

le célèbre adversaire. En revanche, le produit de FriendlyARM n'est compatible qu'avec deux systèmes d'exploitation : Ubuntu Core 16.04 et une version de Debian. Deuxième inconvénient, ce petit format demande bien prévoir la dissipation de la chaleur. Il faudra probablement ajouter des dissipateurs pour obtenir les meilleures performances.

2.6.4. *Asus Tinker Board*

Bien connue des PCistes et des mobinautes, la firme taïwanaise Asus a elle aussi mis un pied dans l'univers des nanocartes de développement. Comme les autres acteurs asiatiques, elle vient concurrencer Raspberry Pi avec un mini-ordinateur sobrement nommé Asus Tinker Board. Ce dernier dispose d'un plateau 40 broches, de quatre ports USB 2.0, du WiFi, du Bluetooth 4.0 d'un port HDMI et d'une prise audio mini-jack. Jusque-là, rien ne le sépare du modèle 3B. Le Tinker Board se différencie par sa puissance puisqu'il est équipé du Rockchip RK3288 basé sur une architecture ARM 32 Bits. Il s'agit d'un processeur quatre cœurs Cortex A17 cadencés à 1,8GHz épaulé par 2Go de mémoire vive LDDR3 et une puce graphique Mali T764 capable de décoder des fichiers 4K. Compatible avec les distributions Linux et Windows, l'Asus Tinker Board est vendu 65 dollars, soit 30 dollars de plus que la Raspberry Pi 3B. Elle doit également faire ses preuves face à la communauté réunie autour de la célèbre fondation [10].

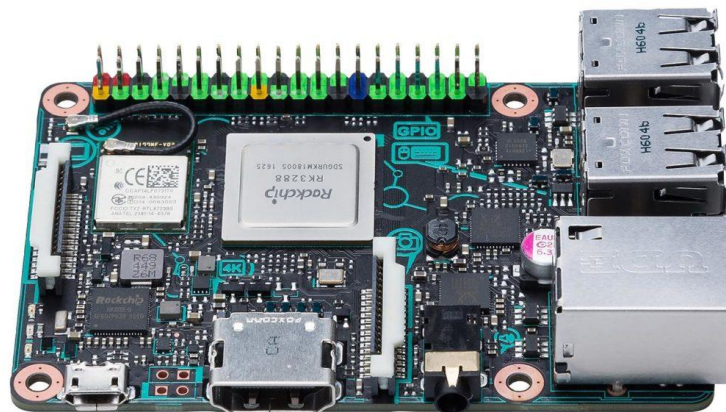


Figure 2.11 : *Asus Tinker Board*

2.6.5. *Le cas particulier d'Arduino*

Arduino est une marque bien connue des concepteurs. Ce projet italien, proche de la fondation à la framboise, a donné naissance à plusieurs variantes de microcontrôleurs plus ou moins puissants. Le plus célèbre d'entre eux n'est autre que le Arduino Uno. Souvent comparé au Raspberry Pi, un Arduino peut avoir les mêmes fonctions, mais n'a pas du tout

la même puissance. Le Uno dispose d'un microcontrôleur ATmega328P cadencé à 16MHz contre 700 MHz pour le premier Pi, et d'une mémoire flash de 32Ko. Son "adversaire" le surpasse haut la main en ces termes. De même un Arduino ne dispose pas de ports audio et vidéo ni de système d'exploitation. En revanche, il possède 14 entrées/sorties ce qui permet d'y raccorder de nombreux capteurs. Arduino est souvent utilisé en complément du Pi, ce dernier étant le cerveau des opérations une fois connecté à cette carte d'origine italienne.



Figure 2.12 : *Arduino Uno*

2.7. Atouts et points faible du Raspberry Pi

2.7.1. Les atouts

Plusieurs sont l'avantages de l'utilisation de la carte Raspberry Pi, comme cité ci-dessous :

- C'est une carte autonome pour le développement
- La compilation croisée peut permettre de gagner du temps et préserver la SD Card
- Elle est du très bon marché environ 40 dollars
- On a un grand nombre d'utilisateurs développent et partagent sur le Raspberry Pi

2.7.2. Les points faibles

Néanmoins, la carte représente des points faibles.

Le choix d'un prix réduit a amené à ne pas choisir des structures optimales, ni les dernières sur le marché. Cependant de version en version ces inconvénients sont moins perceptibles. En plus, certaines documentations de composants ou logicielles ne sont pas du domaine public

2.8. Domaines d'utilisation du Raspberry Pi

Actuellement, on compte de nombreux projets conçus à partir du Raspberry Pi : voitures télécommandées, thermostats connectés, système GPS de suivi de véhicule, émulateur de

jeux vidéo, miroir connecté, niche connectée... Les seules limites sont l'imagination et les capacités techniques de ces dispositifs à bas coût et à faible consommation.

Les projets le plus repandus sont cités ci-dessous :

2.8.1. Un système distribué de surveillance de la qualité de l'air

Si les solutions de suivi de qualité de l'air sont bien connues par les associations comme AirParif qui les embarquent dans des véhicules spécialisés, ce projet proposé par James Puderer, un habitant de Lima au Pérou, consiste à équiper les taxis d'un Raspberry Pi connecté à un GPS, un capteur de particules, une station météorologique et d'une batterie. Car oui, le projet consiste à placer les dispositifs à l'intérieur des signaux lumineux des taxis afin de faire une carte de la pollution de la ville. Cette cartographie des émissions de particules est réalisée à partir de la plateforme IoT de Google : Google Cloud IoT. Cela permet de faciliter la configuration afin d'associer les données sur la pollution à Google Map. À cette occasion, le Raspberry Pi 3B fonctionne sous le système d'exploitation Android Things. Le concepteur a déjà réalisé des tests à Lima et son projet fonctionne parfaitement. Rapidement, cela fait apparaître les zones chaudes d'émission de carbone. Ce projet qui demande environ 16 heures de configuration pour un taxi a gagné un prix au concours "Android Things : Rapid Prototypes to Real Products". James Puderer a laissé l'ensemble des instructions sur la page Hackster du projet et la liste de composants à acheter.

2.8.2. L'automatisation de la maison avec un Raspberry Pi

S'il y a bien un cas d'usage du Raspberry Pi apprécié par les amateurs de domotique, c'est l'automatisation de la maison connectée. Plusieurs pages proposent des solutions basées sur les nano-ordinateurs comme Arduino et le sujet de cet article. Sur Hackster.Io, Christian Kratky fait lui-même sa proposition reposant sur le modèle 2B configuré à l'aide du système d'exploitation Windows 10 IoT Core. À partir de la monocarte, ce système peut automatiser le réglage de la température, l'éclairage, les prises électriques, la ventilation, les ventilateurs, les volets, la porte des animaux domestiques ou encore les désinsectiseurs [11] [18].

Tout comme le projet précédent, le concepteur a mis les différents éléments et instructions nécessaires pour tenter de le reproduire. Contrairement à l'idée précédente, cela demande beaucoup plus de préparation et de travaux. Le temps et les compétences requises ne sont pas à la portée de tous. Finalement cette possibilité donnée par le nano-ordinateur permet surtout aux entreprises d'expérimenter des solutions pour la maison connectée et de proposer ensuite leur propre configuration matérielle.

2.8.3.La culture hydroponique connectée

Voilà une tendance qui a percé en zone urbaine. La culture hydroponique est une technique de croissance des plantes qui consiste en un substrat neutre que l'on alimente d'oligoéléments, de sels minéraux et d'eau. Si elle permet de produire correctement des légumes sur des petites surfaces, cette culture hors-sol demande de "veiller au grain", car il faut régulièrement ajuster le taux d'humidité dans le lieu de culture. Il faut également apporter la bonne quantité d'eau, le PH de la terre, la température et vérifier le taux d'ensoleillement/d'éclairage.

Paul Langdon propose sa version du projet à Base du Raspberry Pi. Dans cette construction, le développeur propose sa version connectée de la culture hydroponique. Dans cette histoire de plante et de croissance, le Raspberry Pi 2B est le cerveau, tandis que les microcontrôleurs Arduino assurent l'automatisation des différents critères cités au-dessus. La monocarte s'avère finalement l'interface de relevé des données en temps réel vers le cloud ou le serveur dédié. C'est aussi le moyen de connexion de l'application compagnon. Elle permet de contrôler la ferme hydroponique depuis sa poche.

2.8.4.Reconnaissance faciale avec un Raspberry Pi

L'importance des modules complémentaires au Raspberry Pi se fait sentir dans la plupart des projets présentés ici. Cette fois-ci l'idée est de transformer la monocarte en outil de reconnaissance faciale. Pour ce faire, il faut connecter un câble module caméra à un Raspberry Pi 3 avec une caméra équipée de la vision nocturne. Le tout fonctionne à l'aide de l'outil freemium AWS Rekognition, un service d'analyse d'image à distance gratuit pendant un an sous condition d'analyser 5000 clichés. Couplé au module applicatif pi-timilo capable de capturer des images de nuit et en mouvement, il faut programmer le service pour détecter des visages et reconnaître les personnes enregistrées afin, par exemple, d'ouvrir une porte. En ajoutant le bon script, le Raspberry Pi 3 peut envoyer des notifications par mail ou encore ouvrir directement la porte protégée par une serrure connectée [11].

2.8.5.Miroir connecté contrôlable à la voix

Tout Geek qui se respecte a rêvé un jour que son miroir soit capable de lui renvoyer autre chose que son propre visage. Avec des technologies comme le Raspberry Pi, ce qui pourrait sembler coûteux devient abordable à condition de mettre "les mains dans le cambouis". De tous les projets en provenance d'Hackster présenté ici, ce dernier est l'un des plus techniques, car il demande de démonter un écran, de fabriquer un cadre en bois pour placer les câbles et

la monocarte, de l'adapter à une vitre réfléchissante préexistante et de programmer le Pi pour faire apparaître le texte derrière le reflet. Ici le concepteur a installé le module applicatif MagicMirror qui permet d'afficher des widgets de météo, un calendrier, le trafic routier, etc. Il a ajouté un microphone pour pouvoir faire fonctionner Alexa depuis le Kit SDK fourni par Amazon. Ainsi le miroir se transforme en véritable assistant vocal.

2.9. Conclusion

Ainsi, Raspberry Pi est une merveille de miniaturisation, qui intègre une puissance de calcul considérable sur une surface pas plus grande qu'une carte de crédit. Considéré comme un nano ordinateur, il est capable de réaliser des prouesses. Parmi les autres technologies de nano ordinateur, le Raspberry est le plus intéressant en terme de performance technique et aussi financière.

CHAPITRE 3

LES METRIQUES DE PERFORMANCE DE LA VOIP

3.1. Introduction

La raison pour laquelle VoIP est une application intéressante est sa facilité de déploiement, ce qui fait qu'elle peut se répandre très rapidement. Avec un minimum d'effort et de dépenses, on peut utiliser l'infrastructure réseau existante pour faire des appels téléphoniques essentiellement gratuits. Par contre, afin que cette alternative au système téléphonique traditionnel soit largement acceptée de la part des utilisateurs, il est indispensable que la qualité de service offerte soit équivalente.

La problématique de qualité de la voix sur IP est particulière car la voix attend de son transporteur autre chose que les données. La transmission des données classiques (fichiers, messages, transactions ...) ne supporte aucune perte en ligne sous peine des graves conséquences pour l'interprétation et l'utilisation de ces données par l'équipement récepteur, mais elle supporte en revanche une dérive importante en termes de durée d'acheminement. Peu importe qu'un paquet arrive avec 100 ms de retard. Le comportement attendu pour la voix est exactement inverse : 1% ou 2% de perte des données de voix en ligne ne sont pas trop gênants pour la qualité du service de VoIP, mais en revanche une variation fréquente de 100 ms sur le délai de transit est catastrophique et rend le service inutilisable.

3.2. Les protocoles

3.2.1. Les protocoles de transports

De nombreux protocoles de couches inférieures sont concernés par le transport de la VoIP. Parmi lesquels sont le TCP (Transmission Control Protocol), UDP (User Datagramme Protocol), RTP (Real Time Protocol) et RTCP (Real Time Protocol Control Protocol) [19].

3.2.1.1. RTP (Real Time Transport Protocol)

a. Présentation

RTP 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 garantit 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.

b. Rôles et objectifs

RTP ne se contente pas de pallier l'absence de TCP.

C'est un protocole entièrement nouveau qui fournit un service temps réel aux applications audio et vidéo, en effet il nous permet de :

- Reconstituer la base de temps des flux (horodatage des paquets : possibilité de resynchronisation des flux par le récepteur)
- Mettre en place un séquençement 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 nombre. Cependant il est très important de savoir quel est le paquet qui a été perdu afin de pouvoir pallier à cette perte. Et ce par le remplacement par un paquet qui se compose d'une synthèse des paquets précédant et suivant [19].
- Identifier le contenu des données pour leurs associer un transport sécurisé.
- 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 dépaquetisation afin que l'application soit décodée correctement.

3.2.1.2. RTCP (Real time Transport Control Protocol)

a. Présentation

Le protocole RTCP est basé sur des transmissions périodiques de paquets de contrôle par tous les participants dans la session.

C'est un protocole de contrôle des flux RTP, permettant de véhiculer des informations basiques sur les participants d'une session, et sur la qualité de service [20].

b. Rôles et objectifs

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, Realtime Transport Control Protocol, qui transporte les informations supplémentaires suivantes pour la gestion de la session. Ainsi le RTCP assure les fonctions suivantes :

- Fournir des informations sur la qualité de la session : information en retour pour une source (feedback), permet à une source de changer de politique met en évidence des défauts de distribution individuels, collectifs
- Garder une trace de tous les participants à une session CNAME (Canonical Name) : identifiant unique et permanent pour un participant SSRC (Synchronisation Source Identifier)
- Contrôler le débit auquel les participants à une session RTP transmettent leurs paquets
- RTCP. Plus il y a des participants, moins la fréquence d'envoi de paquets RTCP par un participant est grande. Il faut garder le trafic RTCP en dessous de 5% du trafic de la session
- Transmettre des informations de contrôle sur la session (optionnel) exemple : identifier un participant sur les écrans des 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. 204 : application spécifique

3.2.2.Les Protocoles de signalisation

3.2.2.1. Le protocole H.323

a. Description

H.323 est un protocole de communication englobant un ensemble de normes utilisées pour l'envoi des données audio et vidéo sur Internet. Il existe depuis 1996 et a été initié par l'UIT (Union Internationale Communication), un groupe international de téléphonie qui développe des standards 26 de communication. Concrètement, il est utilisé dans des programmes tels que Microsoft Netmeeting, ou encore dans des équipements tels que les routeurs Cisco. Il existe un projet OpenH.323 qui développe un client H.323 en logiciel libre afin que les

utilisateurs et les petites entreprises puissent avoir accès à ce protocole sans avoir à déboursier beaucoup d'argent [19] [20].

Le protocole H.323 est utilisé pour l'interactivité en temps réel, notamment la visioconférence (signalisation, enregistrement, contrôle d'admission, transport et encodage). Il 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).

Une communication H.323 se déroule en cinq phases :

- Établissement d'appel ;
- Échange de capacité et réservation éventuelle de la bande passante à travers le protocole RSVP (Resource reSerVation Protocol) ;
- Établissement de la communication audio-visuelle ;
- Invocation éventuelle de services en phase d'appel (par exemple, transfert d'appel, changement de bande passante,) ;
- Libération de l'appel

b. Architecture et composants

La norme H.323 définit quatre composants principaux : Terminal, Gatekeeper, Passerelle, MCU. La figure suivante représente les composants de l'architecture H.323.

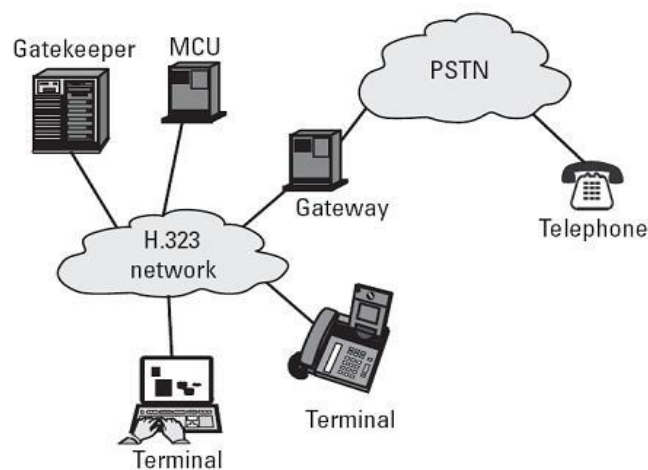


Figure 3.01 : Les composants de l'architecture H.323

Terminal H.323

Il est nœud du réseau de la VOIP, il peut connecter sur le réseau pour :

- Faire un appel avec un autre terminal de la VOIP ou d'autre réseau.
- Accepter un appel de d'autre terminal.

- Terminer un appel.

Gatekeeper

Il est un composant de l'architecture de la VOIP qui gère la registration, L'admission et la statue des terminaux ou des passerelles. Il peut faire la gestion de zone, traiter d'appel et signer d'appel [5] .

- Traduire des adresses.
- Registrer les terminaux.
- Confirmer.
- Contrôler l'acceptation de canal d'information.

MCU : Multipoint Control Unit

Il est noeud du réseau de la VOIP. Le MCU est une station sur le réseau qui fournit les possibilités pour trois terminaux ou plus et passages pour participer à une conférence multipoints. Le MCU gère les ressources de la conférence, négocie avec les terminaux pour déterminer les codecs audio et vidéo à utiliser et gère les flux de données. Le MCU se compose d'un contrôleur multipoints obligatoire (MC) et des processeurs multipoints optionnels (MP). MC détermine les possibilités communes de terminaux en utilisant H.245 mais il n'exécute pas le multiplexage d'audio, de la vidéo et des données. Le multiplexage de médias est manipulé par le MP sous la commande de MC.

Passerelle (gateways)

Les passerelles H.323 assurent l'interconnexion avec les autres réseaux, ex:((H.320/RNIS), les modems H.324, les 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)

c. Famille de protocole H.323

H.323 se dessine en 3 grandes parties. En effet, pour établir une communication audio ou vidéo sur IP, le signal doit être encodé en utilisant des codecs normalisés définis dans la norme H.323, qui normalise aussi la signalisation à utiliser pour l'établissement d'une communication. La voix ou la vidéo est transmise en utilisant le protocole UDP, associé aux protocoles RTP et RTCP pour le transfert des données en temps réel.

- Parmi les codecs possibles figurent G.711, G.723 et G.729 pour les signaux audio, H.261 et H.263 pour les signaux vidéo.

- La signalisation pour l'établissement des appels est mise en œuvre à l'aide de trois protocoles :

H.225 RAS (Registration, Admission and Status)

La signalisation RAS est utilisée entre les end-points (Terminal) et le Gatekeeper qui les contrôle. RAS permet donc au Gatekeeper de contrôler les end-points présents dans sa zone. Autrement dit la gestion du trafic entre le client et le serveur de communication.

H.225 Call signaling

Cette signalisation permet d'établir et de libérer des connexions entre end-points H.323.

H.245

Lorsque l'appelé décroche, le protocole H.245 permet l'établissement de canaux RTP/RTCP permettant le transfert de données multimédia et le contrôle de ce transfert.

Les protocoles temps réel sur IP utilisés sont RTP et RTCP. RTP fournit un transport de bout en bout sur un réseau pour les applications transmettant des données en temps réel, telles que la voix ou la vidéo, en unicast et en multicast. RTP ne se préoccupe pas de la réservation de ressources et ne garantit pas la qualité de service des transferts de données en temps réel. Le transport des données bénéficie aussi du protocole de contrôle RTCP qui fournit un contrôle minimal et des fonctions d'identification particulièrement utiles dans le cas de réseaux multicast. RTP et RTCP sont conçus pour être indépendants des réseaux sous-jacents [5] [20].

Transmission	Protocoles utilisés
Contrôle et la signalisation	H.225, H.245, RTCP
Voix	G.711, G.722, G.723, G.726, G.728, G.729
Vidéo	H.261, H.263, H.263+, H.264
Données	T.123, T.124, T.125

Tableau 3.01 : Différents protocoles utilisés par H323

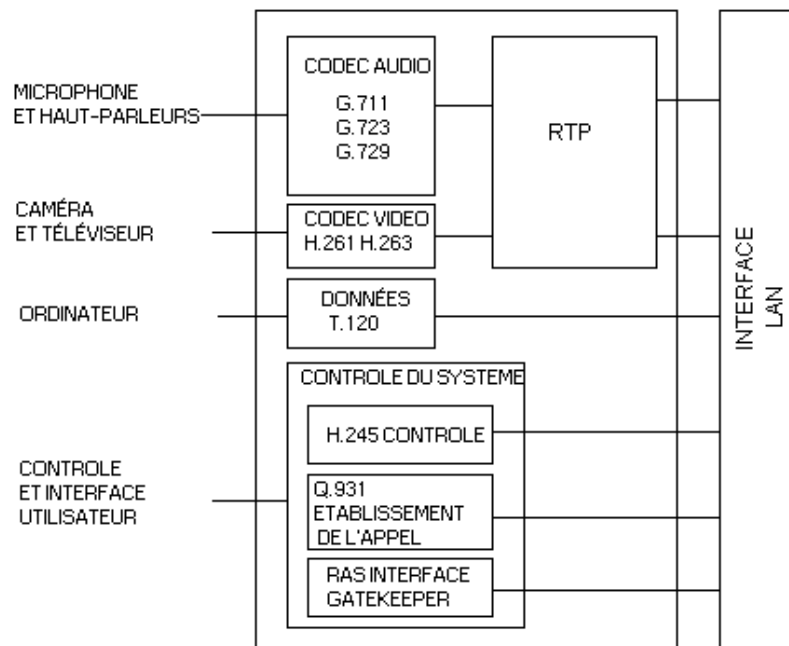


Figure 3.02 : Empilement de protocole H323

d. Fonctionnement de la norme

Son architecture est souple, il est donc possible de l'implémenter de différentes manières. Tous les messages ne sont pas absolument nécessaires dans tous les cas et les différents protocoles qui composent le H323 ne sont pas absolument solidaires, ils peuvent emprunter des chemins différents. On peut citer trois types de communications dans l'utilisation de la norme H323 tels que la communication point à point, la communication auprès d'un Gatekeeper et la communication avec plusieurs clients.

Deux clients simples en point à point

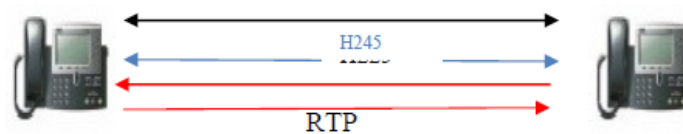


Figure 3.03 : Communication point à point

- L'appelant entre l'adresse IP du destinataire dans le champ du logiciel réservé à cet effet.
- Les protocoles de signalisation proposent au logiciel du destinataire d'établir la communication et transmet son ID H323.

Le logiciel du destinataire répond soit « occupé » soit « libre ».

- Si « libre », l'appelant énumère ses possibilités de codecs audio et vidéo (si disponibles)
- Le destinataire énumère les codecs compatibles à l'appelant pour accord
- Si accord, d'autres ports TCP et UDP sont négociés pour l'audio (UDP), la vidéo (UDP) et les données (TCP).
- Chaque flux est ensuite transmis indépendamment des autres.
- A la fermeture d'une session, les ports sont libérés et les transmissions de contrôle stoppées [20].

Deux clients enregistrés auprès d'un Gatekeeper

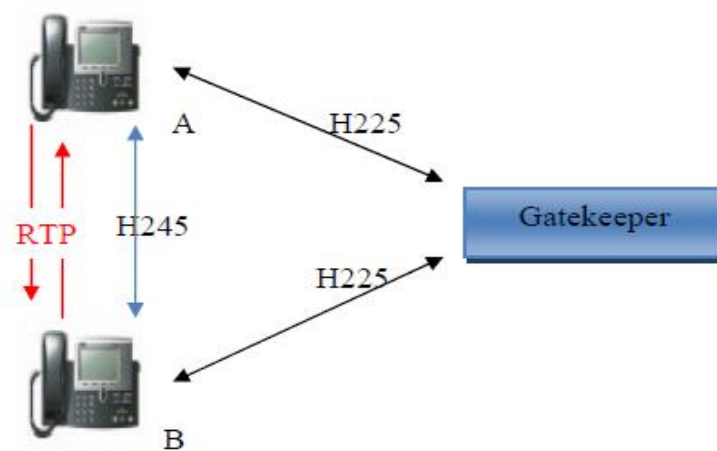


Figure 3.04 : *Communication auprès d'un Gatekeeper*

- Le client A demande l'autorisation au Gatekeeper pour se connecter au client B.
- C'est le Gatekeeper qui informe le client B qu'une communication va avoir lieu.
- Quand la communication est établie, tous les flux sont transmis indépendamment les uns des autres sans passer par le Gatekeeper mais directement entre les clients.
- A la fermeture d'une session, le Gatekeeper est informé de la fin de connexion, les ports sont libérés et les transmissions de contrôle stoppées.

Plusieurs clients en multipoints

Le MCU s'annonce auprès de Gatekeeper et lui annonce ses possibilités :

- Nombre de clients possibles
- Débits (en octets/seconde) possibles par client ou débit total maximal
- Id H323 de connexion

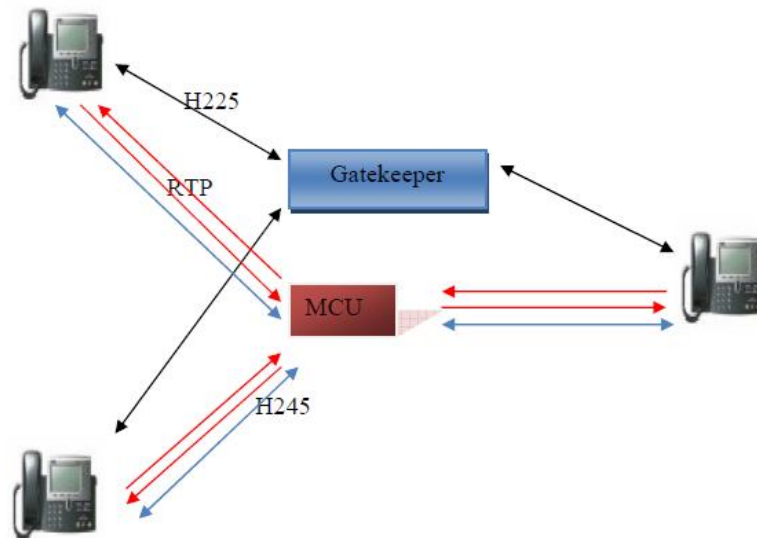


Figure 3.05 : *Communication avec plusieurs clients*

e. Les avantages et inconvénients du protocole H.323

Avantages

Voici les principaux bénéfices qu'apporte la norme H.323

Codecs standards : H.323 établit des standards pour la compression et la décompression des flux audio et vidéo. Ceci assure que des équipements provenant de fabricants différents ont une base commune de dialogue.

- Interopérabilité

Les utilisateurs peuvent dialoguer sans avoir à se soucier de la compatibilité du terminal destinataire. En plus d'assurer que le destinataire est en mesure de décompresser l'information, H.323 établit des méthodes communes d'établissement et de contrôle d'appel. Indépendance vis à vis du réseau : H.323 est conçu pour fonctionner sur tout type d'architecture réseau. Comme les technologies évoluent et les techniques de gestion de la bande passante s'améliorent, les solutions basées sur H.323 seront capables de bénéficier de ces améliorations futures [21].

- Indépendance vis à vis des plates-formes et des applications :

H.323 n'est lié à aucun équipement ou système d'exploitation.

- Support multipoint :

H.323 supporte des conférences entre trois terminaux ou plus sans nécessiter la présence d'une unité de contrôle spécialisée.

- Gestion de la bande passante

Le trafic audio et vidéo est un grand consommateur de ressource réseau. Afin d'éviter que ces flux ne congestionnent le réseau, H.323 permet une gestion de la bande passante à disposition. En particulier, le gestionnaire du réseau peut limiter le nombre simultané de connexions H.323 sur son réseau ou limiter la largeur de bande à disposition de chaque connexion. De telles limites permettent de garantir que le trafic important ne soit pas interrompu.

- Support multicast :

H.323 supporte le multicast dans les conférences multipoint. Multicast, c'est le fait d'envoyer un paquet vers un sous ensemble de destinataires sans réplication, permet une utilisation optimale du réseau.

Indispensable pour permettre un minimum d'interopérabilité entre équipements de fournisseurs différents, ce standard présente toutefois les inconvénients suivants :

Inconvénients

H.323 est un protocole complexe, créé initialement pour les conférences multimédia et qui incorpore des mécanismes superflus dans un contexte purement téléphonique [22] [23]. Ceci a notamment des incidences au niveau des terminaux H.323 (téléphones IP, par exemple) qui nécessitent de ce fait une capacité mémoire et de traitement non sans incidence au niveau de leur coût.

Il comprend de nombreuses options susceptibles d'être implémentées de façon différentes par les constructeurs et donc de poser des problèmes d'interopérabilité ou de plus petit dénominateur commun (dans le choix du codec, par exemple) ; D'autre part, comme le seul codec obligatoire est le codec G.711 (64 Kbps) et que le support des autres codecs plus efficaces est optionnel, l'interopérabilité entre produits provenant des constructeurs différents ne signifie pas qu'ils feront un usage optimal de la bande passante. En effet, dans le cas où les codecs à bas débits sont différents, le transport de la voix se fera à 64 Kbps, ce qui, en termes de bande passante, ne présente guère d'avantages par rapport à un système téléphonique classique.

Le protocole H.323 est une des normes envisageables pour la voix sur IP à cause de son développement inspiré de la téléphonie. Cependant, il est pour l'instant employé par des programmes propriétaires (Microsoft, etc.). La documentation est difficile d'accès car l'ITU fait payer les droits d'accès aux derniers développements de cette technologie, en dehors des efforts faits par le projet Open H.323 pour rendre cette technologie accessible à tous. Ainsi

son adaptation au réseau IP est assez lourde. C'est pourquoi au fil des recherches est né le protocole SIP [24].

3.2.2.2. Protocole SIP

a. Description

SIP (Session Initiation Protocol), est le standard IETF (Internet Engineering Task Force) pour la signalisation de communications multimédias interactives, A savoir : établissement, redirection, relaying, terminaison des sessions vidéo ou audio entre plusieurs terminaux, sur un réseau à commutation de paquet. Il se base entre autre sur le protocole HTTP, la structure de l'entête est semblable et la transmission se fait également en mode texte [22].

La première version de SIP, SIPv1, a été soumise à l'IETF comme une ébauche d'internet en février 1996. SIPv1 a employé le protocole SDP (Session Description Protocol) pour décrire les sessions et UDP comme protocole de transport. SIPv1 a été basé sur du texte.

b. Les messages SIP : requêtes et réponses

Les requêtes SIP constituent les messages qualifiés par un performatif (invitation, demande d'information, souscription et publication d'information, etc.) appelé méthode, définissant la qualité de la transaction.

INVITE sip :user@domain.com
Via :
Call-ID :
From :
To :
Call-ID :
Cseq :
Subject :
Content-Type :
Content-Length :
Ligne vide
Données SDP

Figure 3.06 : *Allure d'une requête SIP*

Afin de mieux saisir les possibilités offertes par SIP, voici une liste des principales méthodes définies par ce protocole et utilisés pour qualifier les requêtes :

Requêtes	Rôle
INVITE	invitation d'un client à une session ou pour modification de celle-ci
REGISTER	enregistrement auprès d'un proxy SIP et donc dans un domaine SIP
ACK	acquiescement d'une requête
BYE	terminaison d'une session ou une transaction
REFER	transfert d'appel
OPTIONS	test des capacités ou la disponibilité d'un serveur SIP
UPDATE	mise à jour d'une session établie

Tableau 3.02 : *Les principales requêtes du protocole SIP*

Les réponses constituent les informations renvoyées par le serveur au client, et concernent autant l'évolution de la transaction que les erreurs pouvant survenir (transport, serveur, client, etc). On distingue les réponses provisionnelles, qui donnent une information optionnelle, et les réponses finales qui clôturent une transition [25] [26] .

SIP/2.0 302 Moved temporarily
From :
To :
Call-ID :
Localization :
Expires :
Cseq :
Ligne vide :
Données de réponse (SDP vide, SDP chiffré, etc)

Figure 3.07 : *Allure d'une réponse SIP*

Les réponses reprennent les codes définis dans la norme HTTP. Les codes sont constitués de trois chiffres dont le premier caractérise la classe de réponse.

Réponses	Types		
1xx	100 Trying (tentative)	180 Ringing (sonnerie)	181 Call Is Being Forwarded (renvoi d'appel)
	182 Queued (en attente)	183 Session progress (session en cours)	
2xx	200 Ok		
3xx	300 Multiple choices	301 Moved Permanently	302 Moved Temporarily
	305 Use Proxy	380 Alternative Service	
4xx	400 Bad Request	401 Unauthorized	402 Payment Required
	403 Forbidden	404 Not Found (non trouvé)	405 Method Not Allowed
	406 Not Acceptable	407 Proxy Authentication Required	408 Request Timeout
	409 Conflicts	410 Gone	411 Length Required
	413 Request Entity Too Large	414 Request-URI Too Long	415 Unsupported Media Type
	420 Bad Extension	480 Temporarily Unavailable	481 Call Leg/Transaction Does Not Exist
	482 Loop Detected	483 Too Many Hops	484 Address Incomplete
	485 Ambiguous	486 Busy Here (occupé)	
5xx	500 Server Internal Error	501 Not Implemented	502 Bad Gateway
	503 Service Unavailable	504 Gateway Time-out	505 Version Not Supported
6xx	600 Busy Everywhere	603 Decline	604 Does Not Exist Anywhere
	606 Not Acceptable		

Tableau 3.03 : Liste complète des codes de réponse SIP

c. Les entités de SIP

Un réseau de SIP se compose de quatre types d'entités. Chaque entité a des fonctions spécifiques et participe à la communication de SIP comme le client (demandes d'initiés), le serveur (répond aux demandes), ou tous les deux. Un « dispositif physique » peut avoir plus de fonctionnalités qu'une entité logique de SIP [26]. Par exemple, un serveur de réseau fonctionne comme proxy serveur peut également fonctionner comme serveur d'enregistrement en même temps. Voici les quatre types d'entités logiques de SIP :

- Agents d'utilisateurs,
- Proxy serveur,
- Serveur de redirection,
- Serveur d'enregistrement.

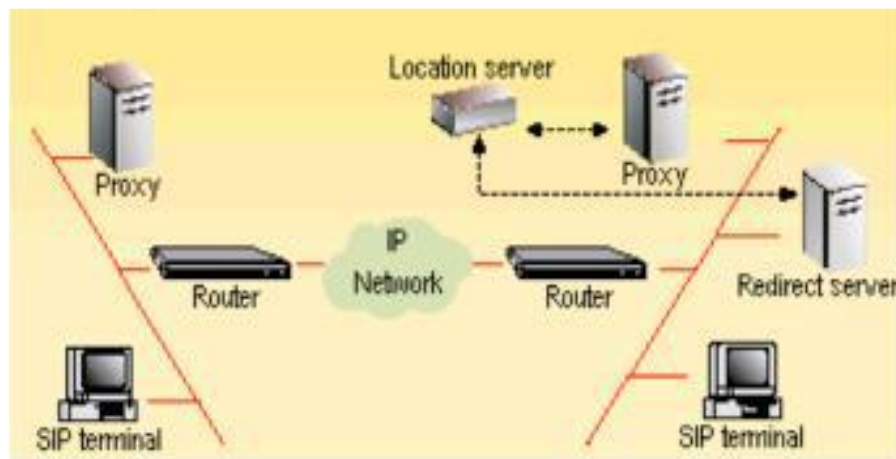


Figure 3.08 : Architecture SIP

Agents d'utilisateur

Les agents d'utilisateur représentent le téléphone (client d'agent d'utilisateur) et le serveur (serveur d'agent d'utilisateur). Le client d'agent d'utilisateur (UAC) lance des appels de médias. Le serveur d'agent d'utilisateur (UAS) répond à ces demandes d'installation au nom de l'UAC. L'UAS est également responsable de trouver la destination UAC ou UAS intermédiaire.

Proxy serveur

Il contient des caractéristiques du client et du serveur. Le proxy server peut recevoir des messages de demandes et des messages de réponse. Il peut ajuster l'information d'en-tête avant d'expédier la demande sur le prochain proxy server ou de rentrer au client d'utilisateur.

Serveur de redirection

Un serveur de redirection est un serveur qui accepte une demande de SIP, projette l'adresse de SIP de la partie appelée en zéro (s'il n'y a aucune adresse connue) ou des nouvelles adresses et les renvoie au client. Différence des proxys serveurs, les serveurs de redirections ne transmettent pas la demande à d'autres serveurs.

Serveur d'enregistrement

Un serveur d'enregistrement est un serveur qui accepte des demandes de REGISTRE afin de mettre à jour une base de données d'endroit avec l'information de contact de l'utilisateur indiqué dans la demande [26].

d. Enregistrement SIP et établissement de sessions

La première étape à réaliser dans un réseau VoIP pour établir une session est d'accéder à ce réseau. Le client SIP doit s'enregistrer dans son domaine VoIP auprès d'un serveur d'enregistrement. Il envoie un message REGISTER contenant son adresse SIP URI et son adresse de contact au registrar du domaine 1. Ce dernier enregistre cette correspondance dans un serveur de localisation.

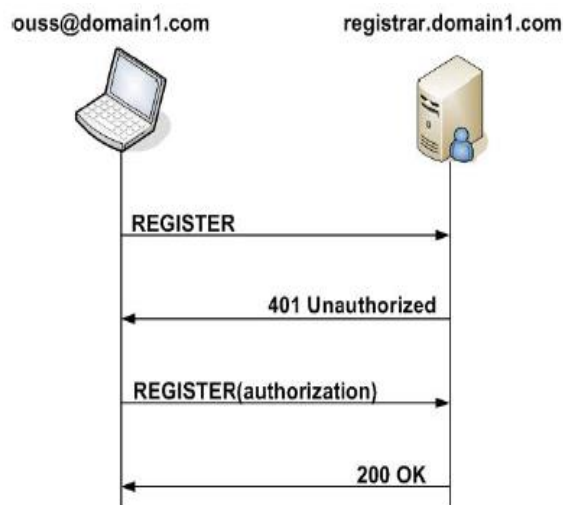


Figure 3.09 : Enregistrement SIP

L'établissement d'une session VoIP se fait entre deux agents client et plus précisément entre l'UAC du premier client SIP et l'UAS du deuxième client. Le champ From dans ce (SIP INVITE) message contient son adresse SIP URI, le champ To contient l'adresse logique du destinataire et le champ Via contient l'adresse du contact du serveur proxy SIP du domaine VoIP.

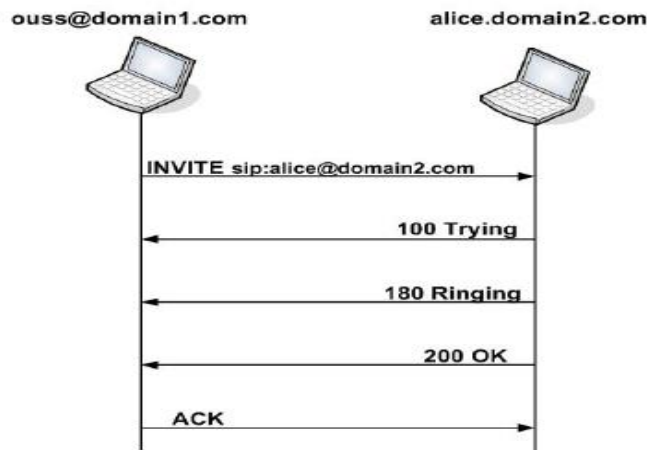


Figure 3.10 : *Etablissement d'une session SIP*

e. Les avantages et inconvénients SIP

Les avantages

L'implémentation de la VoIP avec le protocole de signalisation SIP (Session Initiation Protocol) fournit un service efficace, rapide et simple d'utilisation. SIP est un protocole rapide et léger. Les utilisateurs s'adressent à ces serveurs Proxy pour s'enregistrer ou demander l'établissement de communications. On peut s'enregistrer sur le Proxy de son choix indépendamment de sa situation géographique. L'utilisateur n'est plus "attaché" à son autocommutateur. Une entreprise avec plusieurs centaines d'implantations physiques différentes n'a besoin que d'un serveur Proxy quelque part sur l'Internet pour établir "son" réseau de téléphonie "gratuit" sur l'Internet un peu à la manière de l'email [25] [26].

Les inconvénients

L'une des conséquences de cette convergence est que le trafic de voix et ses systèmes associés sont devenus aussi vulnérables aux menaces de sécurité que n'importe quelle autre donnée véhiculée par le réseau.

En effet SIP est un protocole d'échange de message basé sur HTTP. C'est pourquoi SIP est très vulnérable face à des attaques de types DoS (dénis de service), détournement d'appel, trafic de taxation, etc.

3.2.3.IAX (Inter Asterisk eXchange)

C'est un protocole de signalisation de voix/ vidéo sur IP. IAX (Inter Asterisk eXchange), est utilisé par Asterisk (Asterisk est un PABX ou Private Automatic Branch eXchange et surtout un IPBX (Internet Protocole Branche eXchange)).

Ce protocole fonctionne sur le port 4569 en UDP et transporte à la fois les données (voix) et la signalisation. L'intérêt principal de ce protocole est d'être fait pour traverser le NAT (Network Address Translation un routeur fait la traduction d'adresse réseau) et qu'il est possible de créer des trunks IAX (appelés également Canaux (définissant le nombre d'appels simultanés entre un opérateur IP et un IPBX)), entre les serveurs dans lesquels les communications RTP sont multiplexées ainsi on économise les surcharges d'entêtes IP.

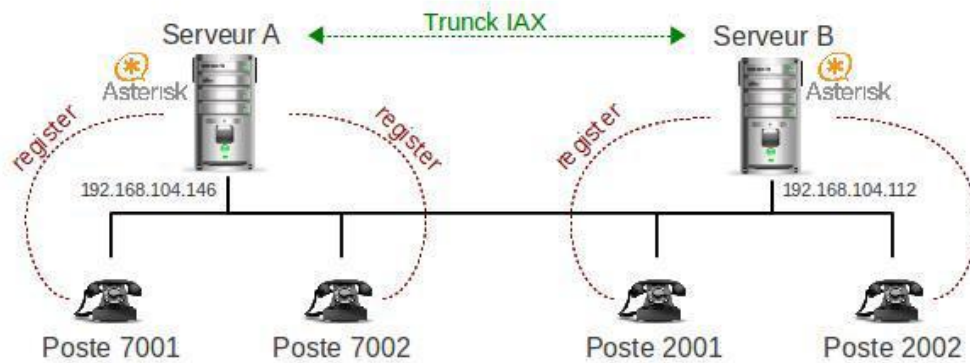


Figure 3.11 : Rôle du protocole IAX

L'avantage principal d'IAX est qu'il permet à plusieurs appels d'être rassemblés dans un seul ensemble de paquets IP, transportant des informations concernant plusieurs appels en cours. Et ça réduit la consommation de bande passante pour un ensemble d'appels (par l'utilisation du trunking).

3.3. Les codecs

A cause de la commutation par paquets des réseaux informatiques, les données audio doivent être codées et divisées en paquets avant la transmission. Le codage, ainsi que le décodage, est fait par des codeurs qui transforment les échantillons de voix en une représentation spécifique au niveau du réseau ou vice versa. La plupart des codeurs est définie par des standards d'UIT-T. Chacun a des propriétés différentes concernant la bande passante nécessaire, mais aussi par rapport à la qualité du signal vocal obtenu [27].

3.3.1. Définition

Codec est une abréviation pour Codeur/Décodeur. Un codec est basé sur un algorithme qui permet la compression des données qu'on lui donne. Il s'agit d'un procédé permettant de compresser et de décompresser un signal, de l'audio ou de la vidéo, le plus souvent en temps réel, permet une réduction de la taille du fichier original. Le codec numérise (dans le cas où les signaux téléphoniques à transmettre sont sous forme analogique, ces derniers doivent d'abord être convertis sous forme numérique) et compresse la voix de l'émetteur, ainsi les

données numériques sont encapsulées dans des paquets IP et acheminées vers le destinataire. A l'arrivée au destinataire, ce dernier grâce au même codec décompresse et restitue le son.

3.3.2.Caractéristiques et fonctions

L'objectif d'un codec est d'obtenir une bonne qualité de voix avec un débit et un délai de compression le plus faible possible.

Les codecs les plus souvent mise en œuvre dans les solutions VoIP sont G.711, G.729 ; G.723.1 et GSM et ils sont caractérisés par :

3.3.2.1. Le Débit

- Compris entre 1.2 Kbps et 64 Kbps
- Compromis entre la qualité de la voix et la bande utilisée.
- Réseau Téléphonique Commuté : 64 Kbps.
- On peut atteindre aujourd'hui une bonne qualité avec 5 Kbps.

3.3.2.2. Compression de silence

Il n'est pas nécessaire de transmettre des données pendant les périodes de silence.

- Taux d'activité effectif de la voix dans un sens.
- VAD (Voice Activity Detection) : détection d'activité de la voix.
- DTX (Discontinuous Transmission) : transmission discontinue ; le codec stoppe la transmission d'informations lorsque le VAD est activé.
- CNG (Comfort Noise Generation) : génération du bruit de confort ; transmission de faible quantité d'information destinée à reproduire chez le récepteur l'ambiance sonore de l'émetteur.

3.3.2.3. Robustesse aux pertes

- Causes des pertes : congestion dans les routeurs, délais trop importants.
- La répétition des trames est impossible (délais trop importants).
- Une solution possible : ajouter de la redondance (Forward Error Correction).
- Une autre solution : remplir les trous avec des trames répétées ou interpolées (Packet Loss Concealment).
- Un critère pour le codec : dégradation de la qualité perçue en fonction du taux d'erreur paquets/trames.

3.3.2.4. Délai de paquetsisation

Le codec crée périodiquement des trames de voix ; il lui faut un certain temps pour traiter le signal analogique et créer la trame.

Certains codecs améliorent la compression en observant plus que le contenu effectif d'une trame (look ahead) [27].

Exemple :

G723.1 doit mémoriser 37.5 ms d'échantillons voix avant de créer la première trame

3.3.2.5. Bande étroite ou large bande

Les codecs bande étroite (ex : G711) échantillonnent la voix sur une bande d'environ 4 KHz (300 Hz – 3400 Hz).

Les codecs large bande permettent de transmettre des fréquences audio jusqu'à 7 KHz (fréquence d'échantillonnage 16 KHz) ; il en résulte une bien meilleure qualité de voix (clarté).

Remarque :

L'oreille humaine peut capter des sons jusqu'à des fréquences de 16 voire 20 KHz.

3.3.3. Le codec G.711

Ce codec est le premier à avoir été utilisé dans la VoIP. Même s'il existe maintenant des codecs nettement plus intéressants, celui-ci continue d'être implémenté dans les équipements à des fins de compatibilité entre marques d'équipements différentes.

Le codec G.711 est basé sur le principe de la modulation par impulsion codées ou Pulse Modulation Code (PCM) [27] [28].

La modulation PCM se distingue en trois étapes pour convertir le signal analogique en signal numérique avant la transmission :

- L'échantillonnage,
- La quantification,
- La codification.

3.3.3.1. L'échantillonnage

Soit un signal $x(t)$ avec énergie définie. Le signal $x(t)$ est échantillonné instantanément à un taux régulier à chaque instant T_s secondes. On appelle T_s la période d'échantillonnage .

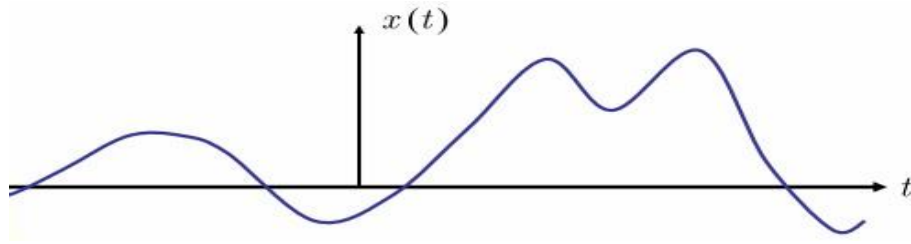


Figure 3.12 : *Signal analogique $x(t)$*

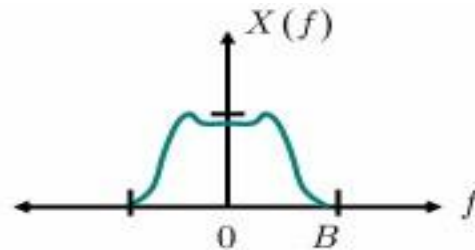


Figure 3.13 : *Spectre du signal $x(t)$*

Selon le théorème de Nyquist :

Soit $x(t)$ un signal de largeur de bande B . ce signal $x(t)$ peut être parfaitement reconstitué à partir de ses échantillons, pris à des intervalles uniformes $\{nT_s\}$ si la fréquence d'échantillonnage $f_s = 1/T_s$ est au moins le double de la largeur de bande B du signal $x(t)$.

Pour échantillonner un signal, on intervient par l'impulsion de Dirac :

$$\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \quad (3.01)$$

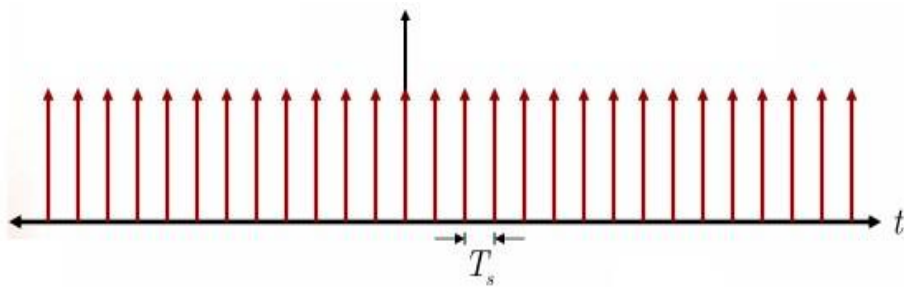


Figure 3.14 : *Impulsion Dirac ou peigne de Dirac*

Donc, soit l'expression du signal échantillonné après la combinaison avec l'impulsion de Dirac :

$$x_s(t) = x(t)\delta_T(t) \quad (3.02)$$

D'où,

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(t) \delta(t - nT_s) \quad (3.03)$$

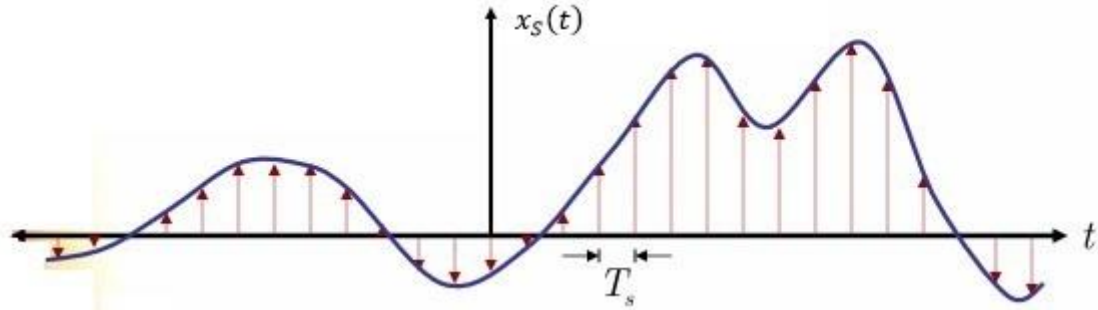


Figure 3.15 : Un signal $x_s(t)$ échantillonné

Pour calculer le spectre du signal $x(t)$, on calcul sa transformé de Fourier :

$$X_s(f) = F[x(t)\delta_T(t)] \quad (3.04)$$

$$X_s(f) = F[x(t)] \otimes [\delta_T(t)] \quad (3.05)$$

$$X_s(f) = X(f) \otimes \Delta_T(f) \quad (3.06)$$

Avec

$$\Delta_T(f) = F[\delta_T(t)] = F[\sum_{n=-\infty}^{\infty} \delta(t - nT_s)] \quad (3.07)$$

Or $\delta_T(t)$ est un signal périodique,

Soit le développement en série de Fourier :

$\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s)$ de période T_s

$$C_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} \delta(t) e^{-j2\pi n f_s t} dt = \frac{1}{T_s} e^{-j2\pi n f_s t} |_{t=0} = \frac{1}{T_s} \quad (3.08)$$

Donc,

$$\Delta_T(f) = F[\delta_T(t)] = F\left[\sum_{n=-\infty}^{\infty} C_n e^{-j2\pi n f_s t}\right] = \sum_{n=-\infty}^{\infty} F[C_n e^{-j2\pi n f_s t}] \quad (3.09)$$

$$\Delta_T(f) = \sum_{n=-\infty}^{\infty} \delta(f - n f_s) \quad (3.10)$$

On va mettre cette équation 3.10 à l'équation 3.06, et on obtient le spectre du signal échantillonné :

$$X_S(f) = X(f) \otimes \Delta_T(f)$$

$$X_S(f) = X(f) \otimes \left[\frac{1}{T_S} \sum_{n=-\infty}^{\infty} \delta(f - nf_s) \right] \quad (3.11)$$

D'où,

$$X_S(f) = \frac{1}{T_S} \sum_{n=-\infty}^{\infty} X(f - nf_s) \quad (3.12)$$

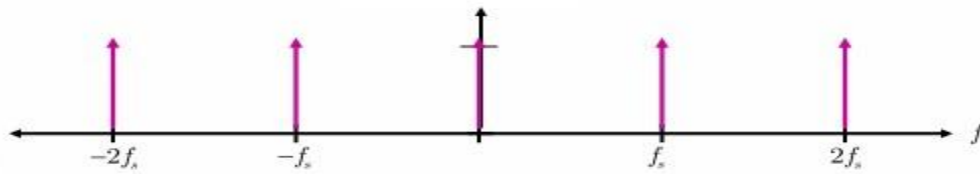


Figure 3.16 : Spectre du peigne de Dirac

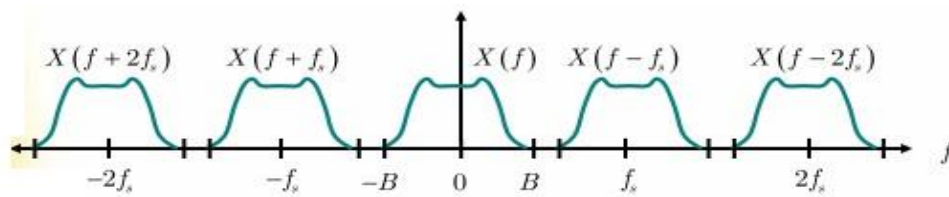


Figure 3.17 : Spectre du signal échantillonné

La voix est un phénomène analogique. La conversion en numérique est requise pour la transporter dans un réseau numérique.

La fréquence la plus haute de la voix humaine en téléphonie est 4000Hz. D'après le Théorème de Shannon, la fréquence d'échantillonnage est au moins deux fois plus que la fréquence du signal continu à échantillonner. Alors, le taux d'échantillonnage sur cette voix est donc $2 * 4000$ c'est-à-dire 8000 échantillons/seconde [19] [28].

3.3.3.2. La quantification

C'est le processus de valeurs numériques à l'amplitude de chaque échantillon. Il existe deux types de quantification :

- Quantification linéaire,
- Quantification non linéaire.

a. La quantification linéaire

La quantification est donc basée sur une prise de décision de l'assignation de valeurs finies, c'est à-dire elle donne une valeur approximative à l'échantillon. Ce qui introduit l'erreur de quantification.

Par exemple, les échantillons de signaux échantillonnés deviennent la séquence numérique : [0, 0.8, 1, 0.2, -0.8, -1].

Noter que les valeurs réelles sont [0, 0.84, 0.91, 0.15, -0.75, -0.96].

Soient les valeurs quantifiées : [0, 0.8, 1, 0.2, -0.8, -1].

Sur cet exemple, il existe donc une différence pour chaque échantillon :

[0, 0.04, -0.09, -0.05, 0.05, 0.04].

Cet exemple applique la quantification linéaire, c'est-à-dire que les niveaux de quantification sont espacés de degrés égaux.

b. La quantification non linéaire

Les signaux analogiques, par exemple la voix, ont typiquement plus de chance d'avoir de faibles valeurs que des valeurs près du maximum permis. Le bruit de grenaille devient alors important. Une solution est de réduire le pas de quantification pour des valeurs du signal proche de 0 : c'est le principe de la quantification non linéaire [19] [21].

3.3.3.3. La codification

Selon la qualité de service ou Grade of Service (GOS) requiert 2000 échelons de quantifications pour assurer une qualité téléphonique. Il faudrait donc 12 bits pour présenter toutes les possibilités. C'est-à-dire $2^{12} = 2048$. Et 12 bits x 8000 échantillons/sec donnent 96 kbps par canal.

Cette largeur de bande est très élevée. Donc, pour réduire cette largeur de bande, on utilise la méthode de compression-extension qui est un processus de quantification en appliquant l'encodage non linéaire afin d'offrir plus d'échelons dans les bas volumes et moins dans les hauts.

Sur cette méthode, un meilleur GOS peut être obtenu avec seulement 256 niveaux au lieu de 2000. Soit 256 niveaux est équivalent de 2^8 , on a 8 bits/échantillon. Et la largeur de bande réduit de 64 kbps.

La compression-extension demande qu'une loi logarithmique soit appliquée sur l'étendue des valeurs de quantification linéaires :

- La loi-A utilisée pour les signaux ITU-T E1,
- La loi μ utilisée pour les signaux T1 nord-américain.

a. La loi μ

La loi μ est représentée par l'équation suivante pour $\mu=255$

$$|v_2| = \frac{\ln(1 + \mu|v_1|)}{\ln(1 + \mu)} \text{ avec } |v_1| \leq 1 \quad (3.13)$$

μ : une constante positive, lorsque $\mu=0$, c'est la quantification uniforme

v_1 et v_2 : voltage d'entrée et voltage de sortie

Au Canada et au Japon, on utilise $\mu=255$

Voici la quantification linéaire en utilisant la loi μ

	Segment	unité d'amplitude	échelon		Segment	unité d'amplitude	échelon
Chaque échelon comprend 256 unités d'amplitude. (Il y a 16 échelons dans ce segment, il y a 4096 unités d'amplitude dans ce segment.)	8	8159	128			0	1
		.	.	1	.	-31	16
	7	4063	112	2	.	-95	32
		.	.	3	.	-223	48
	6	2015	96	4	.	-479	64
		.	.	5	.	-991	80
	5	991	80	6	.	-2015	96
		.	.	7	.	-4063	112
Chaque échelon comprend 2 unités d'amplitude. (Il y a 16 échelons dans ce segment, il y a 32 unités d'amplitude dans ce segment.)	4	479	64	8	.	-8159	128
		.	.				
	3	223	48				
		.	.				
	2	95	32				
		.	.				
	1	31	16				
		.	.				
		0	0				
Amplitudes positives				amplitudes négatives			

Figure 3.18 : La quantification linéaire avec la loi μ

b. La loi A

La loi A est représentée par l'équation suivante pour A=255 :

$$|v_2| = \begin{cases} \frac{A|v_1|}{1 + \ln A} & \text{si } 0 \leq |v_1| \leq \frac{1}{A} \\ \frac{1 + \ln(A|v_1|)}{1 + \ln A} & \text{si } \frac{1}{A} \leq |v_1| \leq 1 \end{cases} \quad (3.14)$$

A : constante positive, sa valeur typique est 87.6

Voici la quantification linéaire en utilisant la loi A,

segment	unités d'amplitude	écheleon	segment	unités d'amplitude	écheleon
	4096	128		0	1
7	.	.	1	.	.
6	2048	112	2	-64	32
5	.	.	3	-128	48
4	1024	96	4	-256	64
3	.	.	5	-512	80
2	512	80	6	-1024	96
1	256	64	7	-2048	112
	128	48		-4096	128
	64	32			
	0	0			
amplitudes positives			amplitudes négatives		

Chaque écheleon comprend 128 unités d'amplitude. (Il y a 16 écheleons dans ce segment. Il y a 2048 unités d'amplitude dans ce segment.)

Chaque écheleon comprend 2 unités d'amplitude. (Il y a 32 écheleons dans ce segment. Il y a 64 unités d'amplitude dans ce segment.)

Figure 3.19 : La quantification linéaire avec la loi A

c. La codification

Un échantillon PCM est codé sur un mot de 8 bits [29].

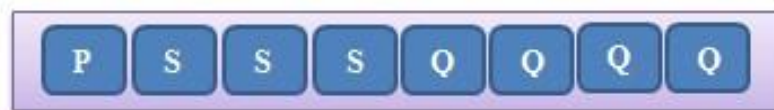


Figure 3.20 : Codage PCM

P : 1 bit utilisé pour indiquer la polarité de l'amplitude de l'échantillon

S : 3 bits utilisés pour définir le segment de compression

Q : 4 bits utilisés pour définir l'écheleon de quantification dans le segment de compression

Exemple

Pour bien comprendre le codage du G.711, on va prendre un petit exemple.

Soit 3.5 volts un échantillon reçu et supposons que le niveau maximal de bit est à 5 volts.

On va chercher alors le codage de ce 3.5 volts en G.711 en utilisant la loi μ .

Notons que les 8 niveaux d'amplitudes notés de 0 à 7 représentent 5 volts.

Soit, $8159 \text{ amplitudes} / 5 \text{ volts} = 1631.8 \text{ amplitudes par volt}$.

Donc, à 3.5 volts le niveau est : $1631.8 \times 3.5 = 5711.3^{\text{ème}}$ amplitude.

Ceci est situé dans le niveau 8 « 111 » où chaque échelon comprend 256 amplitudes.

Le niveau 8 commence à l'amplitude 4063, donc $5711.3 - 4063 = 1648.3$ amplitudes au niveau 8.

Il y a 256 amplitudes par échelon au niveau 8. Donc $1648.3/256 = 6.44^{\text{ème}}$ échelon du niveau 8 ou approximativement 6 « 0101 ».

D'où le codage du 3.5 volt est « 1 111 0101 ». Le premier 1 indique la polarité

3.3.4.Le codec G.729

G729 est un codec audio qui comprime les données en paquets avec la durée de 10 ms. Il est basé sur l'algorithme de CS-ACELP (Conjugate Structure ACELP).

3.3.4.1. La description d'algorithme CS-ACELP du codeur G.729

L'algorithme de CS-ACELP est la dérivée de la modèle d'une prédiction linéaire à excitation par code qui est conçu pour fonctionner avec un taux d'échantillonnage de 8000 échantillons par seconde. Les échantillons d'entrées et de sortie sont représentés par le 16 bits du signal PCM. Son principe est représenté sur la figure 3.21.

Après le prétraitement, un filtre LP de dixième ordre est effectué et ses paramètres sont quantifiés dans le domaine de pair de ligne spectrale de 18 bits. Pour avoir une meilleure analyse des paramètres comme le pas et le gain de trame en particulier avec la recherche réduite du dictionnaire de code, la trame d'entrée est divisée en deux sous trames de 5ms chacun. Et l'excitation de chaque sous trames est indiquée par la contribution de dictionnaire adaptative et de dictionnaire fixe. Ces paramètres sont transmis et analysés dans chaque sous trames [29] [30].

L'excitation de dictionnaire adaptatif est notée par un retard d'entier et un retard fractionnaire avec une résolution fixe. Le retard de pas en boucle ouverte est créé en fonctionnement de trame qui est basé sur la pondération du signal perceptuel. La recherche de pas en boucle fermée est faite par l'approximation du retard de pas en boucle ouverte, et

le signal ciblé est calculé par une combinaison du filtre LP résiduel à la synthèse de filtre de pondération.

L'indice de dictionnaire adaptatif est codé avec 8 bits dans la première sous-trame et de 5 bits dans la seconde sous-trame. La recherche de dictionnaire fixe est effectuée en utilisant un nouveau signal ciblé de sous trame toutes les 5 ms avec la mise à niveau de ce signal aussi. Le dictionnaire fixe est un dictionnaire algébrique de 17 bits.

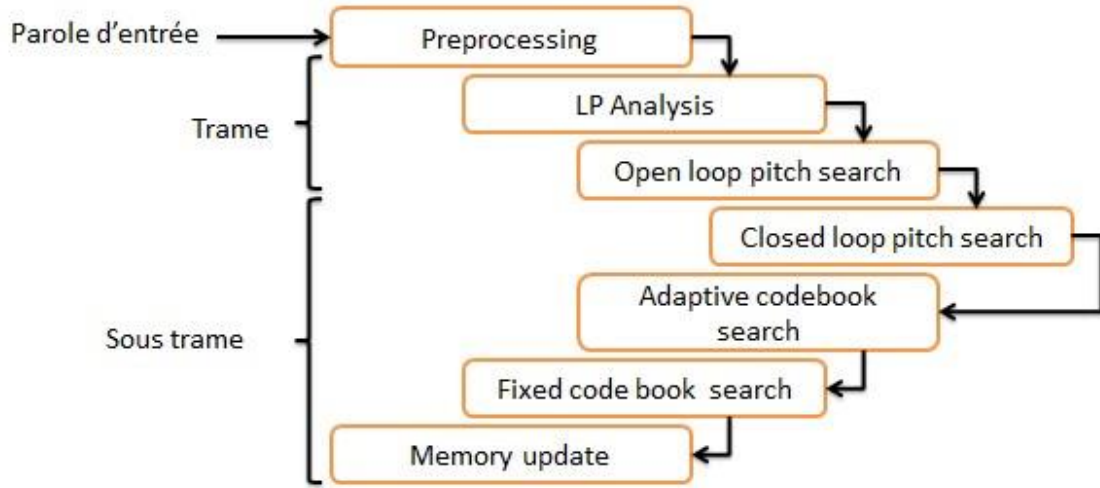


Figure 3.21 : Bloc du codeur G.729

a. Preprocessing ou pré-traitement

C'est un filtre passe haut qui sert à éliminer la basse fréquence ou DC du composant indésirable.

Et après le filtrage, le signal $s[n]$ est mis en échelle.

b. Analyse et quantification du LP

La prédiction linéaire est déroulée deux fois par trame utilisant une approche d'autocorrélation, avec des fenêtres non symétriques. Les coefficients de prédiction linéaire sont cherchés en utilisant l'algorithme de Levinson-Durbin. Ils sont ensuite transformés en paire de lignes spectrales LSP, puis quantifiés et interpolés. Les coefficients interpolés, quantifiés et non quantifiés seront reconvertis en coefficient de prédiction linéaire, afin de reconstruire les filtres de synthèse pour chaque sous trame [30].

Le filtre de prédiction linéaire (prédiction à court terme) utilisé est d'ordre $m=10$, donné par :

$$\frac{1}{\hat{A}(z)} = \frac{1}{1 + \sum_{i=1}^m \hat{a}_i z^{-i}} \quad (3.15)$$

Où $a_i, i= 1, \dots, 10$ sont les coefficients de la prédiction linéaire après la quantification.

c. La recherche de pas en boucle ouverte

Le retard de pas d'estimation en boucle ouverte utilise le signal pondéré $S_w(n)$:

$$R(k) = \sum_{n=0}^{79} S_w(n) S_w(n - k) \quad (3.16)$$

Les trois maxima de la corrélation sont trouvés sur la suite de trois rangs : (20 :39), (40 :79), (80 :143). Le pas en boucle ouverte est obtenu en utilisant le maxima des trois rangs sur la fonction d'autocorrélation normalisé.

$$R'(t_i) = \frac{R(t_i)}{\sqrt{\sum_n S_w^2(n - t_i)}} \text{ avec } i = 1, \dots, 3 \quad (3.17)$$

Le gagnant parmi les trois corrélations normalisées est sélectionné en favorisant le retard avec les valeurs de la gamme inférieure. Ceci est réalisé en augmentant les corrélations normalisées correspondant à la plage de retard plus faible si les retards sont des sous-multiples des retards dans la plage de retard plus élevé. Les meilleurs délais T_{bo} en boucle ouverte sont définis comme suit :

$$\text{If } |t_2 * 2 - t_3| < 5$$

$$R'(t_2) = R'(t_2) + 0.25 * R'(t_3)$$

$$\text{If } |t_2 * 3 - t_3| < 7$$

$$R'(t_2) = R'(t_2) + 0.25 * R'(t_3)$$

$$\text{If } |t_1 * 2 - t_2| < 5$$

$$R'(t_1) = R'(t_1) + 0.20 * R'(t_2)$$

$$\text{If } |t_1 * 3 - t_2| < 7$$

$$R'(t_1) = R'(t_1) + 0.20 * R'(t_2)$$

$$T_{bo} = t_1$$

$$R'(T_{bo}) = R'(t_1)$$

$$\text{If } R'(t_1) \geq R'(T_{bo})$$

$$R'(T_{bo}) = R'(t_2)$$

$$T_{bo} = t_2$$

End

$$\text{If } R'(t_3) \geq R'(T_{bo})$$

$$R'(T_{bo}) = R'(t_3)$$

$$T_{bo} = t_3$$

End

On note seulement que la moitié du nombre d'échantillons est utilisée dans le calcul des corrélations dans l'équation (3.16). En outre, dans la troisième plage de retard (80 :143), seule les corrélations de même retard sont calculées dans le premier passage ; les retards à ± 1 du retard sélectionné sont testés. Sur la base de tests subjectifs informels, la simplification des analyses en boucle ouverte n'a pas introduit une dégradation significative de la performance du codeur [29] [30].

d. La recherche de pas en boucle fermée

Le pas en boucle fermée ou ouverte dans un système est essentiel pour une bonne performance de l'algorithme CELP sur le débit intermédiaire. Le pas en boucle fermé peut être interprété comme un dictionnaire adaptatif du chevauchement des vecteurs. La procédure de correction de point d'extrémité ou de récurrence d'énergie peut être appliquée au pas en boucle fermée, depuis que les procédures prennent l'avantage de la nature de chevauchement du dictionnaire et ne sont pas affecté par le caractère dynamique. L'analyse de pas d'un système en boucle fermée est alors effectuée (pour trouver le délai et le gain de dictionnaire adaptatif), dépend sur le signal filtré $x(n)$ et de réponse impulsionnelle $h(n)$, en effectuant une recherche d'une valeur de pas en boucle ouverte. Le retard de pas est codé sur 8 bits dans la première sous trame et différenciellement codé avec 5 bits dans la seconde sous trame.

e. La recherche de dictionnaire fixe

En général, le dictionnaire fixe prend de 17 bits. Dans le tableau ci-dessous, nous allons décrire la recherche de dictionnaire fixe. Les positions d'impulsions des deux premières impulsions sont codées à 3 bits chacun, et la troisième impulsion est codée en 4 bits. Les signes de trois impulsions sont codés en 1 bit. Les deux premières impulsions sont fixées à l'amplitude +1, et la dernière est fixée sur -1.

Impulsion	Signe	Positions
i_0	$s_0 = \pm 1$	$m_0: 0,5,10,15,20,25,30,35$
i_1	$s_1 = \pm 1$	$m_1: 1,6,11,16,21,26,31,36$
i_2	$s_2 = \pm 1$	$m_2: 2,7,12,17,22,27,32,37$
i_3	$s_3 = \pm 1$	$m_3: 3,8,13,18,23,28,33,38$ 4,9,14,19,24,29,34,39

Tableau 3.04: Structure de la recherche de dictionnaire de code

Le vecteur d'un dictionnaire $c(n)$ est en procédant un vecteur nul de dimension 40, et mettant les unités de quatre impulsions à l'emplacement trouvé, en multipliant par les signes correspondants :

$$c_n = s_0\delta(n - m_0) + s_1\delta(n - m_1) + s_2\delta(n - m_2) + s_3\delta(n - m_3) \quad n = 0, \dots, 39 \quad (3.18)$$

Où $\delta(0)$ est l'unité d'impulsions.

f. La mise à jour de la mémoire

La mise à jour des états d'un filtre de la synthèse et la pondération d'un filtre est nécessaire pour le calcul du signal cible dans la sous trame suivante [28]. Après que les deux gains sont quantifiés, l'excitation du signal $u(n)$, dans le sous trame courante est obtenu par :

$$u(n) = \hat{g}_p v(n) + \hat{g}_c c(n) \quad n = 0, \dots, 39 \quad (3.19)$$

Où \hat{g}_p et \hat{g}_c sont respectivement les gains quantifiés des dictionnaires adaptatifs et fixes, $v(n)$ est le vecteur de dictionnaire adaptatif, et $c(n)$ est le vecteur de dictionnaire fixe.

Les états du filtre peuvent mettre à jour en filtrant le signal , $r(n) - u(n)$ (différence entre le signal résiduel et l'excitation) à travers le filtre $\frac{1}{A(z)}$ et $\frac{A(\frac{z}{\gamma_1})}{A(\frac{z}{\gamma_2})}$ et pour les 40 échantillons du sous trame et la sauvegarde des états du filtre. Il faudrait pour cela trois opérations de filtrage. Pour une simple approche, qui ne nécessite qu'une seule opération de filtrage, est le suivant. Le signal reconstruit est calculé en filtrant le signal d'excitation par l'intermédiaire de $\frac{1}{A(z)}$. La sortie du filtre par

l'entrée $r(n) - u(n)$ est équivalente $e(n) = s(n) - \hat{s}(n)$. Ainsi, les états du filtre de synthèse $\frac{1}{A(z)}$ sont donnés par $e(n), n = 30 \dots 39$. . Mettre à jour les états du filtre $\frac{A(\frac{z}{\gamma_1})}{A(\frac{z}{\gamma_2})}$ peut être donné par le filtrage de signal d'erreur à travers du filtre pour trouver l'erreur de la pondération perceptuelle $ew(n)$. Toutefois, le signal d'erreur est donné par :

$$ew(n) = x(n) - \hat{g}_p y(n) - \hat{g}_c z(n) \quad (3.20)$$

Avec $z(n) = \sum_0^n c(i)h(n-i) \quad n = 0, \dots 39$

Où $z(n)$: vecteur de convolution de dictionnaire fixe

Pour que les signaux $x(n), y(n)$ et $z(n)$ soient disponibles, les états du filtre de pondération sont mis à jour en calculant $ew(n)$ dans l'équation (3.19) pour $n = 30 \dots 39$. Cela permet d'économiser deux opérations de filtrage.

3.3.4.2. Le principe de la décodeur G.729

Le principe du décodeur est représenté sur la figure ci-dessous. D'abord, les indices des paramètres sont extraits du flux binaire reçu. Ces indices sont décodés pour obtenir les paramètres du codeur correspondant à la trame de 10 ms. Ces paramètres sont les coefficients LSP, les retards de pas fractionnaire, les deux vecteurs de dictionnaire fixe, et les deux ensembles de dictionnaire adaptatif et de guide de codification du gain fixe. Les LSP sont interpolés et convertis en coefficient LP pour chaque sous trame. Pour toutes les 5 ms de sous trame, les opérations suivantes sont répétées. Tout d'abord, l'excitation est construite en ajoutant l'adaptation du vecteur de dictionnaire fixe et la mise en échelle de leurs gains respectifs. Ensuite, le signal de la parole est reconstruit en filtrant le signal d'excitation construit à travers le filtre de synthèse LP. Enfin, le signal reconstruit est passé à travers une

étape de post-traitement, qui comprend un post-filtre adaptatif basé sur des filtres de synthèse à long terme, suivi d'un filtre passe-haut et l'exploitation d'échelle.

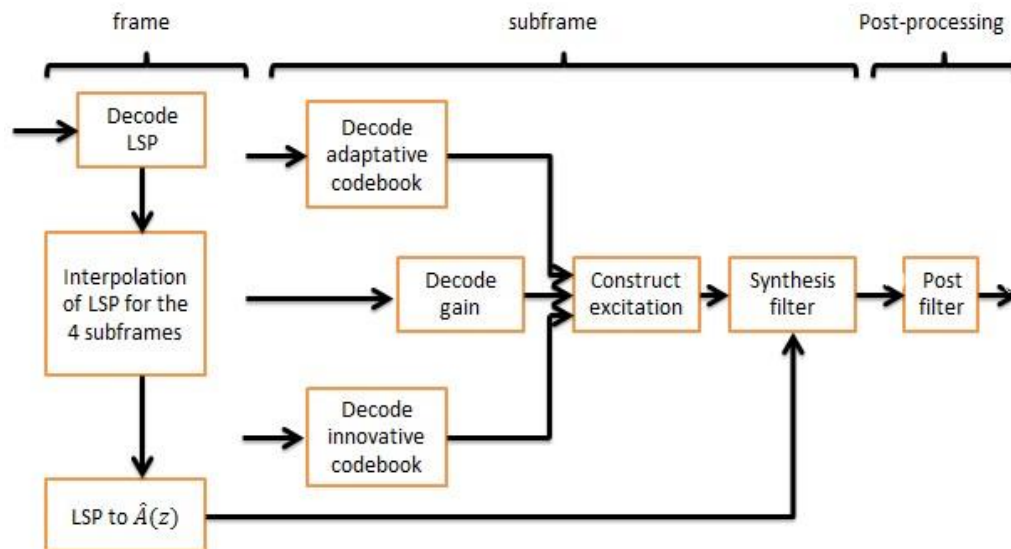


Figure 3.22 : Principe de décodeur du G.729

3.3.5.Codec G722

La norme de codage mondiale G722 permet d'obtenir en VOIP une qualité de voix « haute définition » dite téléphonie large bande.

Cette qualité est obtenue par doublement de la bande de fréquence codée (50-7000Hz) par rapport à la qualité téléphonique usuelle dite bande étroite (300-3400hz).

La G722 utilise un algorithme de codage PCM différentiel adaptatif dit codage MICDA (Modulation par Impulsion et Codage Différentiel Adaptatif) OU ADPCM en anglais.

Comme application, Le format de codage UIT-T G722 a été choisi en 2007 par l'organisme de normalisation européen ETSI comme format obligatoire pour les terminaux DECT de nouvelle génération et haute qualité audio. Certains opérateurs comme orange ont pu lancer avec succès des services de téléphonies IP en voix haute definition utilisant le codec G722.

3.3.6.Le codec G726

Avec plusieurs débits mais le plus populaire est 32 Kbps, elle utilise le même algorithme que G722 le MICDA.

Au lieu d'envoyer le niveau de quantification, il ne code que la différence de niveau avec l'échantillon précédent.

3.4. Qualité de service

3.4.1. Qualité sonore / Fiabilité

Dans le cas de la VoIP, la nécessité d'assurer une qualité de communication de haut niveau est critique. En téléphonie IP on remarque certaines problématiques principalement liées à la qualité sonore : l'écho, les sifflements (causés par les logiciels visant à corriger l'écho) ou un temps de réaction lent. De plus, on observe des problèmes liés à la perte de paquets, à la transmission des télécopies et à la gestion d'appel d'urgence (l'utilisation d'adresses IP complexifie le routage des appels) [24] [25].

Pour des raisons techniques, le phénomène d'écho est, à degré variable, omniprésent dans ce type de communication. Les logiciels qui compensent cet effet sont souvent propriétaires et lorsqu'ils sont différents entre deux communicants, ils risquent de traiter l'écho de façon légèrement différente, ce qui amène des effets de bord non contrôlés tels des sifflements pendant la communication. Finalement, la latence variable du réseau Internet fait que les données voyagent plus ou moins vite. Alors que cette variabilité est acceptable pour des données, elle ne l'est pas pour la voix, phénomène physique qui demande une certaine continuité pour que les gens puissent se comprendre.

Il y a deux manières de juger la qualité de la voix :

- Les méthodes objectives fondées sur des métriques objectives (distorsion du signal, rapport signal à bruit, taux de perte des paquets, etc) ;
- Les méthodes subjectives.

3.4.2. Génération de bruits de confort

Pendant une conversation où les silences sont effacés, l'absence de bruit chez le récepteur peut vite se révéler inconfortable. Dans cette optique, les codecs disposent d'un générateur de bruits de confort visant à simuler des bruits de fond pour améliorer le confort des utilisateurs.

3.4.3. Le délai de transit

Le délai de transit (ou end-to-end delay dans la dénomination anglo-saxonne) est un des paramètres critiques influençant fortement la QoS d'un service de voix sur IP. C'est le temps que va mettre en moyenne un paquet IP contenant un échantillon de voix pour traverser l'infrastructure entre deux interlocuteurs. Ce temps de transit comporte quatre composantes :

- Le délai d'échantillonnage
- Le délai de propagation
- Le délai de transport
- Le délai des buffers de gigue

Le délai d'échantillonnage est la durée de numérisation de la voix à l'émission puis de conversion en signal voix à la réception. Ce temps dépend du type de codec choisi et varie de quelques millisecondes avec le codec G.711 (échantillonnage 64 kbps) à plus de 50 ms en G.723 (échantillonnage 6,3 ou 5,3 kbps). C'est une des raisons pour laquelle le choix du codec impacte le score MOS d'appréciation de la clarté de la voix, indépendamment des autres caractéristiques de l'infrastructure.

Le délai de propagation est la durée de transmission en ligne des données numérisées. Cette durée est normalement très faible par rapport aux autres composantes du délai de transit, de l'ordre de quelques millisecondes.

Le délai de transport est la durée passée à traverser les routeurs, les commutateurs et les autres composants du réseau et de l'infrastructure de téléphonie IP. L'ordre de grandeur est de plusieurs dizaines de millisecondes, voir centaines de millisecondes.

Le délai des buffers de gigue est le retard introduit à la réception en vue de lisser la variation de temps de transit, et donc de réduire la gigue de phase. L'ordre de grandeur est de 50 ms. Les éléments d'infrastructure, notamment les routeurs, peuvent également mettre en œuvre des buffers de gigue [28].

3.4.4. Gigue

La gigue est la variance statistique du délai de transmission. En d'autres termes, elle mesure la variation temporelle entre le moment où deux paquets auraient dû arriver et le moment de leur arrivée effective.

Pour compenser la gigue, on utilise généralement des mémoires tampon (buffer de gigue) qui permettent de lisser l'irrégularité des paquets. Malheureusement ces paquets présentent l'inconvénient de rallonger d'autant le temps de traversé global du système. Leur taille doit donc être soigneusement définie, et si possible adaptée de manière dynamique aux conditions du réseau.

La dégradation de la qualité de service due à la présence de gigue, se traduit en fait, par une combinaison des deux facteurs cités précédemment : le délai et la perte de paquet ; puisque d'une part on introduit un délai supplémentaire de traitement (buffer de gigue) lorsque l'on

décide d'attendre les paquets qui arrivent en retard, et que d'autre part on finit tout de même par perdre certains paquets lorsque ceux-ci ont un retard qui dépasse le délai maximum autorisé par le buffer.

3.5. Perte de paquets

Lorsque les buffers (mémoires) des différents éléments du réseau IP sont congestionnés, ils « libèrent » automatiquement de la bande passante en se débarrassant d'une certaine proportion des paquets entrants, en fonction de seuils prédéfinis.

Si aucun mécanisme performant de récupération des paquets perdus n'est mis en place (cas le plus fréquent dans les équipements actuels), alors la perte des paquets IP se traduit par des ruptures de la conversation et une impression de coupure de la parole.

Le tableau 3.05 présente les seuils de valeurs pour les paramètres critiques et les conséquences constatées pour le niveau de service de VoIP en codec G.711 64 kbps :

	Bon	Moyen	Mauvais
Délai de transit	$D < 150\text{ms}$	$150\text{ms} < D < 400\text{ms}$	$400\text{ms} < D$
Gigue de phase	$G < 20\text{ms}$	$20\text{ms} < G < 50\text{ms}$	$50\text{ms} < G$
Perte de données	$P < 1\%$	$1\% < P < 3\%$	$3\% < P$

Tableau 3.05: Valeurs seuils pour les paramètres critiques

3.6. Interopérabilité

L'interopérabilité consiste à transmettre des paquets VoIP à travers n'importe quel réseau compatible VoIP, lesquels peuvent être acheminés par des chemins différents. Un des défis importants de la téléphonie IP est de s'assurer que le trafic VoIP est supporté à travers tout le réseau afin de mettre en place un système universel équivalent à la téléphonie traditionnelle [27] [29].

3.7. Bande passante

Le choix du codec est un compromis entre la qualité de service souhaitée et la capacité de l'infrastructure IP à délivrer une bande passante et des paramètres de QoS qui vont impacter cette qualité. Le paramètre le plus déterminant auquel on s'intéresse pour commencer est la bande passante que l'on met en regard du nombre de communications simultanées à écouler. La figure 3.23 permet d'effectuer rapidement le bilan de bande passante en fonction du codec choisi :

Echantillonnage (codec)				Calcul de bande passante nécessaire									
Codec	Débit (kbps)	Interv. échant. (ms)	Délai échant. (ms)	Volume de données de voix par échant. de codec (octet)	Volume des données de voix dans RTP (octets)	Durée des données de voix dans RTP (ms)	Nb de paq. par sec.	Bande passante IP/UDP/R TP (kbps)	Bande passante ethernet avec IP/UDP/R TP (kbps)	Bande passante IP/UDP/R TP (kbps)	Bande passante ethernet avec IP/UDP/R TP (kbps)	Bande passante RTP/IP pour 10 canaux (kbps)	Bande passante RTP/IP pour 32 canaux (kbps)
G.711	64	20	1	160	160	20	50	80,0	87,2	65,6	72,8	800	2560
G.726	32	20	1	80	80	20	50	48,0	55,2	33,6	40,8	480	1536
G.726	24	20	1	60	60	20	50	40,0	47,2	25,6	32,8	400	1280
G.728	16	20	25	40	60	30	33	26,7	31,5	17,1	21,9	267	853
G.729	8	20	25	20	20	20	50	24,0	31,2	9,6	16,8	240	768
G.723.1	6,3	30	67.5	24	24	30	33	17,1	21,9	7,5	12,3	171	546
G.723.1	5,3	30	67.5	20	20	30	33	16,0	20,8	6,4	11,2	160	512

Figure 3.23 : Bande passante en fonction du codec

3.8. CPU/RAM

Du côté logiciel et matériel, nombreux paramètres peuvent être mesurés afin d'évaluer les performances d'un système téléphonique qui en Voip est un système informatique. Ce système est connecté à un réseau avec les terminaux et certainement à internet. Ce pourquoi, dans le but de fournir des services vocaux il faut tenir compte des capacités du système téléphonique à savoir la capacité de la connexion internet, le temps que prend le processeur pour répondre à une requête etc.

D'une part, parmi les tâches exécutées par les systèmes téléphoniques ; les codages et les décodages sont les processus les plus gourmands en ressources CPU qui maintient la capacité d'une plate-forme de téléphonie IP limitée. Les codecs utilisés déterminent efficacement la capacité de CPU à utiliser vu qu'ils sont responsables du codage et décodage de la voix. Lorsque le transcodage est terminé, la charge de la CPU sur un système augmente considérablement.

D'une autre part, les temps de réponse augmentent en même temps que la charge du système, en fin de compte, les appels commencent à échouer lorsque le système est surchargé d'où il est clair que la capacité du processeur doit être considérée comme raison de la performance d'un service VOIP.

3.9. Sécurité

La sécurité est l'un d'un critère de performance qu'on ne peut pas négliger. La VoIP, actuellement en plein développement, pouvant faire sujet d'attaque des pirates informatiques, au niveau spécifique des réseaux IP ou la VoIP proprement dite. Celle-ci

présente un nombre de vulnérabilités en terme : de protocole, de logiciel, de système d'exploitation, d'infrastructure physique et d'erreur humaine. Il convient d'étudier avec précaution ses vulnérabilités dans le but d'établir une protection efficace contre les attaques. Pour faire face à ces attaques, la sécurité du réseau VoIP doit s'appuier sur deux types de sécurités :

- La sécurité traditionnelle des réseaux informatiques (Firewall, IPS, IDS, Antivirus,)
- La sécurité spécifique VoIP

3.9.1.Principaux risques

3.9.1.1. 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.

3.9.1.2. Suivre des appels

Appelé aussi Call tracking, cette attaque 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.

3.9.1.3. Injection de paquet RTP

Cette attaque 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.

3.9.1.4. 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.

3.9.1.5. Les vulnérabilités de l'infrastructure

Une infrastructure VoIP est composée de téléphones IP, Gateway, serveurs. Ces derniers tournant sur un système d'exploitation, est accessible via le réseau comme n'importe quel ordinateur et comportent un processeur qui exécute des logiciels qui peuvent être attaqués ou employés en tant que point de lancement d'une attaque plus profonde.

3.9.2.Sécurisation protocolaire

La prévalence et la facilité de sniffer des paquets et d'autres techniques pour la capture des paquets IP sur un réseau pour la voix sur IP fait que le cryptage soit une nécessité. La sécurisation de la VoIP est à la protection des personnes qui sont interconnectées. IPsec peut être utilisé pour réaliser deux objectifs. Garantir l'identité des deux points terminaux et protéger la voix. 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 [30].

3.9.2.1. 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 appropriée vu qu'elle offre le cryptage des données grâce à 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, la nécessité de concevoir des mécanismes d'authentification et de chiffrement pour IP.

3.9.2.2. L'authentification

L'une des méthodes les plus importantes pour anticiper une attaque sur un système de téléphonie est de déterminer clairement l'identité des périphériques ou des personnes participant à la conversation.

Plusieurs solutions simples sont mises en œuvre pour cela, il est recommandé d'utiliser des mots de passe complexes lors de la configuration des clients SIP ; en effet, il faut savoir que

certain hackers développent des robots en charge de sonder les réseaux informatiques et dès que l'un d'eux répond au protocole SIP, un algorithme sophistiqué est engagé et teste toutes les combinaisons possibles de mots de passe. Ainsi, il faut éviter

- Les mots de passe trop courts
- Les suites numériques (123456) ou alphabétiques (abcd)
- Les suites logiques tels prénoms ou dates
- Un mot de passe unique pour toutes les extensions SIP

3.9.2.3. 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 une utilisation d'une version stable, Il est bien connu que toute application non stable contient sûrement des erreurs et des vulnérabilités. Pour minimiser les risques, il est impératif d'utiliser une version stable [19] [21].

3.10. Prédiction de la qualité de service

3.10.1.Score MOS

La qualité d'un codec est mesurée de façon subjective en laboratoire. Ces dernières écoutent tout un ensemble de conversations compressées selon les différents codecs à tester et les évaluent qualitativement selon la table 3.05 :

Qualité de la parole	Score
Excellente	5
Bonne	4
Correcte	3
Pauvre	2
Insuffisante	1

Tableau 3.05: Valeurs seuils pour les paramètres critiques

Sur la base des données numériques des appréciations, une opinion moyenne de la qualité d'écoute (Mean Opinion Score : MOS, pour « note moyenne d'appréciation ») est ensuite calculée pour chaque codec.

La méthode MOS est une mesure subjective de la QoS voix par des opérateurs humains. Elle n'a pas pour objet de fournir des données objectives mais d'obtenir une appréciation de

la clarté de la voix reçue grâce à une enquête auprès d'un panel d'utilisateurs ou d'opérateurs représentatifs. Cette méthode est définie par la spécification ITU P800. Son application est coûteuse et délicate, mais elle reste le « juge de paix » en dernier recours lors des problèmes de plainte des utilisateurs du service de téléphonie. Elle s'applique aussi bien à la téléphonie traditionnelle qu'à la ToIP.

3.10.2. Calcul du facteur R de l'E-model

L'ETSI a développé un modèle de calcul de la qualité de transport de la voix de bout en bout, de la bouche de l'émetteur à l'oreille du récepteur, connu sous le nom d'E-model (référence ETSI : ETR 250). Ce modèle a été standardisé par l'ITU sous la référence G.107. Le principe de l'E-model consiste à calculer une grandeur unique R. La formule simplifiée du calcul de R est la suivante :

$$R = R0 - Is - Id - Ie + A \quad (3.21)$$

Le principe de la formule est de partir d'un certain capital de QoS, égal à R0, et de lui imputer les dommages causés par les différents aspects de la transmission.

- R0 : coefficient initial signal / bruit, « capital initial de QoS », égal à 94,3 en VoIP.
- Is : coefficient de dommages simultanés avec l'émission de la voix (bruit de fond ...)
- Id : coefficient de dommages dus au délai de transmission et de transport
- Ie : coefficient de dommage de distorsion causés par les équipements
- A : coefficient d'amélioration

Le facteur R ainsi calculé de 0 à 100 permet de déduire directement un coefficient MOS de 0 à 5. Dans la réalité les valeurs de R oscillent entre 50 et 93,2, soit la limite basse acceptable pour le récepteur, et la limite haute liée aux possibilités techniques de transformation de la voix humaine en signal [27]. La moyenne de la valeur R sur les services en production se situe dans la fourchette 70 à 80.

Valeurs de R	0-60	60-70	70-80	80-90	90-100
MOS	0-3.10	3.10-3.60	3.60-4.03	4.03-4.34	4.34-5
Qualité	Très mauvaise	Mauvaise	Moyenne	Bonne	Excellente

Tableau 3.06: Evaluation des valeurs de R et du score MOS

Les résultats obtenus pour les principaux codecs, les plus utilisés avec leurs vitesses d'échantillonnage sont résumés dans le tableau 3.07 :

Codec VoIP	Débit (Kbps)	Score MOS
G.711 (PCM)	64	4.1
G.726	32	3.85
G.729	8	3.92
G.723.1 MPMLQ	6.4	3.9
G.723.1 ACELP	5.3	3.65
GSM	13	3.5
Ilbc	13.3-15.2	3.9

Tableau 3.07: *Débit et score MOS en fonction du codec*

3.11. Conclusion

En guise de conclusion, afin de mesurer la performance de la voix sur IP, plusieurs critères sont à prendre en compte comme les codecs, la qualité de service, la sécurisation. On constate que ses performances sont basées sur les choix d'un codec et du protocole utilisé.

Les codecs prennent une place très importante parmi les autres paramètres car le codage et le décodage sont des étapes très important lors de la transmission de donnée c'est pourquoi comme métriques de performance notre choix s'est penché sur les codecs et les protocoles.

CHAPITRE 4

ANALYSE DE PERFORMANCE

4.1. Introduction

La mise en place d'une infrastructure de VoIP représente quelque difficulté à savoir le coût d'installation des matériels ainsi que la garantie de la qualité de service offerte. Afin de remédier à cela, notre choix s'est penché sur l'utilisation d'une technologie moderne « la technologie de Raspberry ». Cette partie, comme but de notre travail, va montrer les résultats d'une analyse de performance de la Voix sur IP sur le Raspberry Pi 3 Model B .

4.2. Les outils utilisés

Pour tester la capacité du système de téléphonie IP, nous avons utilisé la carte Raspberry Pi modèle B, et nous l'avons connectée à un ordinateur portable à partir duquel nous avons lancé le test de résistance automatisé. On a utilisé Raspbx pour le système de téléphonie IP dans la carte Raspberry Pi, et Sipp qu'on a installé sur l'ordinateur portable en tant qu'instance client générant un certain nombre d'appels pour solliciter le système téléphonique [31].

4.2.1. Asterisk

Asterisk est un service de commutation téléphonique open source (basé sur Linux), qui implémente de nombreuses fonctionnalités d'un système PBX IP. Comme il est très populaire parmi les enthousiastes de la téléphonie IP, certaines distributions Linux basées sur Asterisk se sont apparues, orientées de manière à fournir un ensemble complet de service de téléphonie IP. Raspbx est l'une de ces distributions qui intègre une interface graphique Web permettant à l'utilisateur de configurer facilement le démon Asterisk intégré. Raspbx intègre à la fois Asterisk et FreePBX et c'est pourquoi notre choix s'est penché sur cette distribution [32].

4.2.2. Plate-forme Raspberry Pi

Raspberry Pi est une plate-forme informatique minuscule, de taille réduite et très bon marché, créée pour aider les étudiants à apprendre l'informatique. Il comprend un processeur, un processeur graphique, une RAM et des sockets SD ou micro SD pour un stockage persistant. Une plate-forme Raspberry Pi propose également des ports USB, une entrée / sortie audio / vidéo, une interface Ethernet et des capacités de périphérique de bas niveau.

Actuellement, la puissance de son processeur est comparée à celle de vieux processeurs vendus il y a quelques années [33] [34]. Compte tenu de sa très petite taille et sa faible consommation d'énergie, elle offre un excellent rapport coût / avantages. C'est la raison pour laquelle il devient une alternative intéressante pour déployer des services de réseau de petite entreprise à un coût très bas. Les caractéristiques techniques de la Raspi que nous avons utilisées sont données par le tableau 4.01 :

Paramètre	Valeur
Cadencement	1,2 GHz
Puce (SoC)	Broadcom BCM2837
Processeur	ARM Cortex-A53 64 bits quatre cœurs
Processeur graphique	Broadcom VideoCore IV double coeur (OpenGL ES 2.0, H.264 Full HD à 30 ips)
Mémoire (SDRAM)	1GB LPDDR2
Nombre de ports USB 2.0	4
Port extension	GPIO 40 pin
Sorties vidéos	HDMI et RCA, plus 1 connecteur de caméra CSI
Sorties audio	Stéréo Jack 3,5mm ou HDMI
Sauvegarde des données	Carte MicroSD
Connexion réseau	10/100 Ethernet, WiFi 802.11n et Bluetooth 4.1 (BLE - Low Energy)
Périphériques	17 × GPIO
Alimentation	5V 2.5A via micro-USB

Tableau 4.01: *Caractéristiques techniques de la Raspberry Pi utilisée*

4.2.3.Outils Sipp

Sipp est un outil de test de charge pour les systèmes de téléphonie IP. Il ne fonctionne qu'avec le protocole SIP et est capable de générer des messages SIP pour lancer et libérer un appel vocal via un PBX IP. De plus, le Sipp permet l'envoi des médias (voix ou vidéo) lorsqu'un appel est établi. Il est très flexible lors de l'exécution des tests de performance automatisés et fournit des statistiques détaillées pendant et après ces tests [35].

4.2.4. Wireshark

Conçu en 1997-1998 par Gerald Combs sous le nom historique de "Ethereal", il est repris en 2006 sous le nom moderne de "Wireshark".

Wireshark est un analyseur de paquet open source (GNU) populaire. Ses "dissectors" ou décodeurs de protocoles permettent d'interpréter le trafic du réseau.

4.3. Test de performance du système

4.3.1. Scénario avec UAC

4.3.1.1. Description du scénario

Comme décrit précédemment, SIPp permet de générer un ou plusieurs appels SIP vers un système distant. Cet outil est lancé depuis la ligne de commande. Pour cette analyse, elle est faite en utilisant un scénario prédéfini dans SIPp.

Le call flow est comme suit :

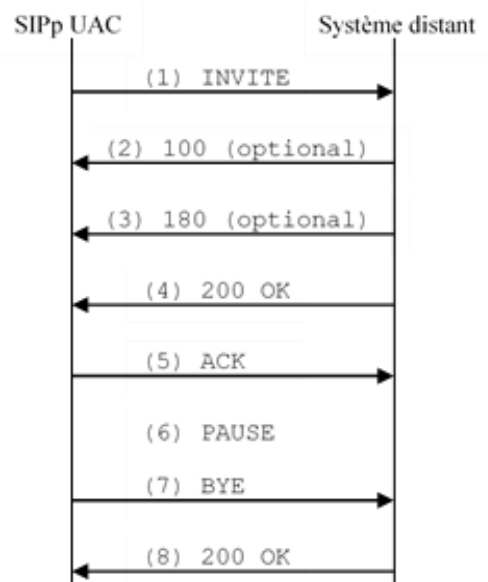


Figure 4.01: Call flow du scénario UAC

Le client envoie des requêtes au serveur, qui lui renvoie une réponse. Les méthodes de bases (RFC 3261) comprises dans ces requêtes sont :

- INVITE : permet à un client de demander une nouvelle session,
- ACK : confirme l'établissement de la session,
- CANCEL : annule un INVITE en suspens,
- BYE : termine une session en cours,

- OPTIONS : permet de récupérer les capacités de gestion des usagers, sans ouvrir de session,
- REGISTER : permet de s'enregistrer auprès d'un serveur d'enregistrement.

Ainsi, on lance le test avec la commande :

sipp 192.168.0.10:5160 -sn uac

192.168.0.10 : c'est l'adresse IP du serveur ou on envoie l'appel ou test

5160 : c'est le port utilisé

-sn : utilise un scénario par défaut intégré dans l'exécutable sipp.

uac : le scénario par défaut.

4.3.1.2. Résultat obtenue du test

Lors du démarrage du test, on a la figure 4.02 :

```

dia@raharinanja: ~
Fichier Édition Affichage Rechercher Terminal Aide
----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length) Port Total-time Total-calls Remote-host
10.0(0 ms)/1.000s 5060 32.13 s 321 192.168.0.10:5160(UDP)

10 new calls during 1.004 s period 1 ms scheduler resolution
0 calls (limit 30) Peak was 1 calls, after 0 s
0 Running, 324 Paused, 23 Woken up
0 dead call msg (discarded) 0 out-of-call msg (discarded)
3 open sockets

Messages Retrans Timeout Unexpected-Msg
INVITE -----> 321 0 0
100 <----- 321 0 0
180 <----- 0 0 0
183 <----- 0 0 0
200 <----- E-RTD1 321 0 0
ACK -----> 321 0
Pause [ 0ms] 321
BYE -----> 321 0 0
200 <----- 321 0 0

----- [+|-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----

```

Figure 4.02 : Démarrage du test avec UAC

Pour notre test, le taux d'appel est de 10appel/ seconde. On constate que tous les appels sont bien transmis. Le test se fait normalement sans problème.

On peut ainsi conclure que le Raspberry Pi supporte donc un taux d'appel de 10 appels/seconde. La figure 4.03 ci-dessous montre les résultats récapitulatifs :

```
dia@raharinanja: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Last Reset Time      | 2019-06-04   13:30:08.671507  1559644208.671507
Current Time         | 2019-06-04   13:30:09.674506  1559644209.674506

-----+-----+-----
Counter Name         | Periodic value          | Cumulative value
-----+-----+-----
Elapsed Time         | 00:00:01:002000         | 00:00:52:211000
Call Rate            | 9.980 cps                | 9.998 cps
-----+-----+-----
Incoming call created | 0                         | 0
OutGoing call created | 10                       | 522
Total Call created   |                           | 522
Current Call         | 1                         |
-----+-----+-----
Successful call       | 10                       | 521
Failed call           | 0                        | 0
-----+-----+-----
Response Time 1       | 00:00:00:008000         | 00:00:00:009000
Call Length           | 00:00:00:019000         | 00:00:00:017000
----- [+|-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----
```

Figure 4.03 : Visuel de la récapitulation de la transmission

Pour le test de performance, on a fait augmenter le taux d'appel de pas de 10 appel en partant à un taux de 10 appels/seconde. Le résultat est donné par la figure 4.04.

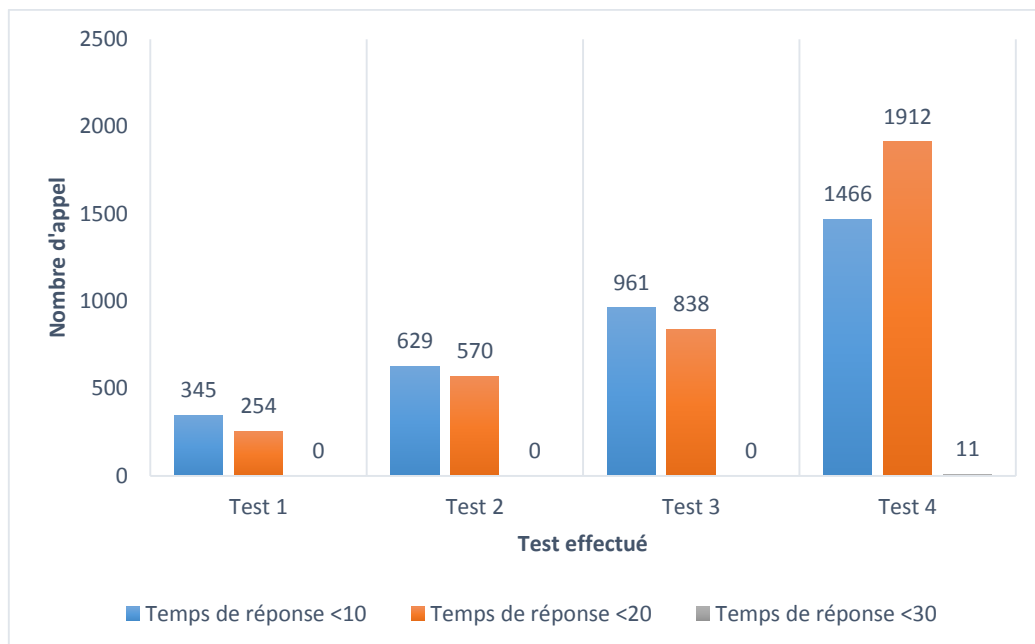


Figure 4.04 : Temps de réponse pour un taux d'appel inférieur à 50

On voit bien sur cette diagramme que le temps de réponse pour chaque appel est inférieur à 20ms qui est acceptable pour un système téléphonique.

Le résultat de ce test nous montre aucune retransmission c'est-à-dire que tous les appels sont tous réussis.

Ensuite, on a augmenté progressivement le taux d'appel de 10 appels/seconde. On arrête le test lorsque celui-ci atteint la valeur de 100 appels/seconde.

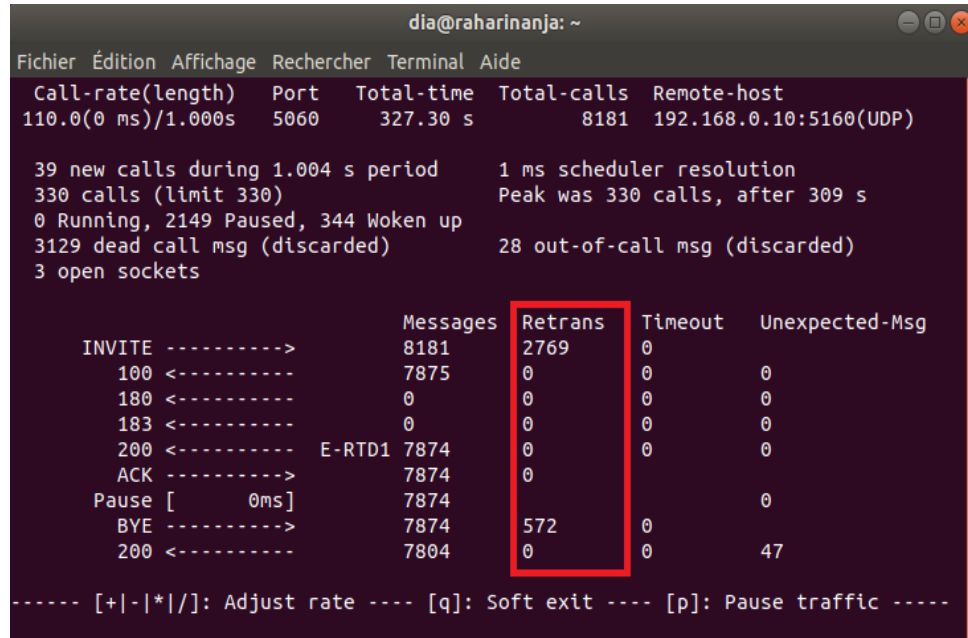


Figure 4.05 : Retransmissions et échecs lors du test

On constate sur la figure que pour un taux d'appel de 50 appels/seconde, le système fait des retransmissions et même certains appels ont commencé à échouer. C'est-à-dire que le système ne supporte pas bien le traitement des appels avec un taux de plus de 50 appels/seconde. La distribution du temps de réponse est donnée par la figure 4.06.

Il est à noter que le test a pour départ de taux de 50 appels/seconde et avec une augmentation de 10 appels/seconde à chaque pas comme auparavant.

Ainsi, on peut dire d'après l'histogramme que à chaque test effectué, la majorité des appels a pour temps de réponse inférieur à 10 ms et 20ms. Quelques appels seulement ont pour temps de réponses supérieur à 20ms, environ une centaine d'appels pour les autres intervalles de temps (entre 20ms et 30 ms, entre 30ms et 40 ms, entre 40ms et 50 ms et supérieur à 50ms).

Le temps de réponse des autres appels est dû à la hausse du nombre d'appels par seconde qui entraîne une surcharge du système.

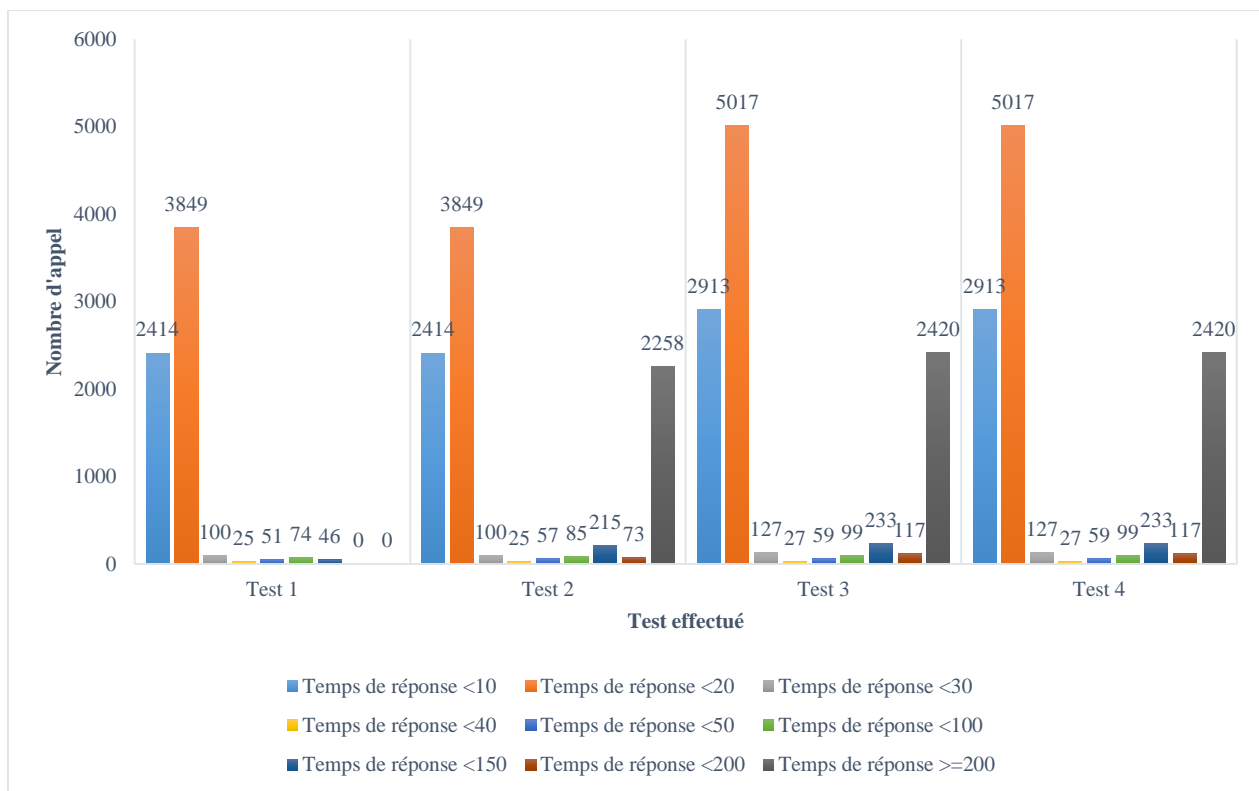


Figure 4.06 : Temps de réponse pour les taux supérieurs à 40appels/seconde

Le nombre d'appel réussi et échoué lors de la transmission est représenté par l'histogramme ci-dessous :

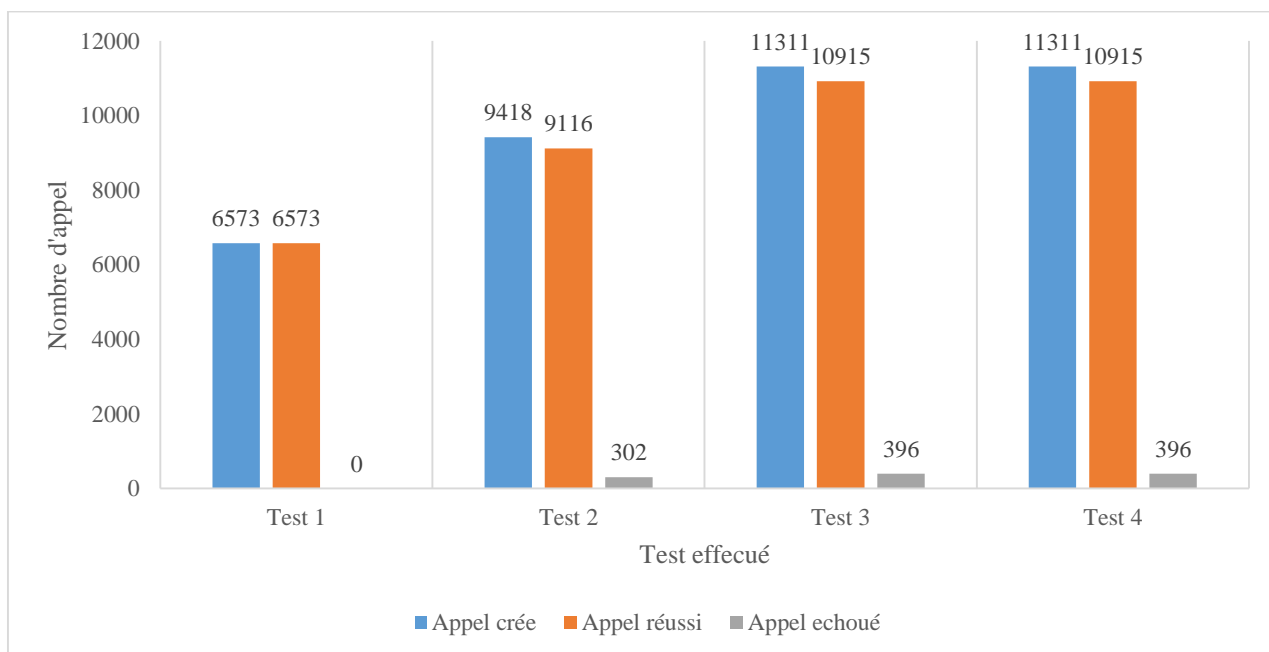


Figure 4.07 : Appels réussis et échoués pour les taux supérieurs à 40appels/seconde

On en conclue que le nombre d'appel échoué est négligeable par rapport au nombre d'appel réussi. Même si le système est surchargé, quelques appels seulement sont mise en retransmission et d'autre échouent.

Pour l'utilisation du processeur, l'outil htop nous a permis de visualiser l'utilisation du processeur de la Raspberry Pi.

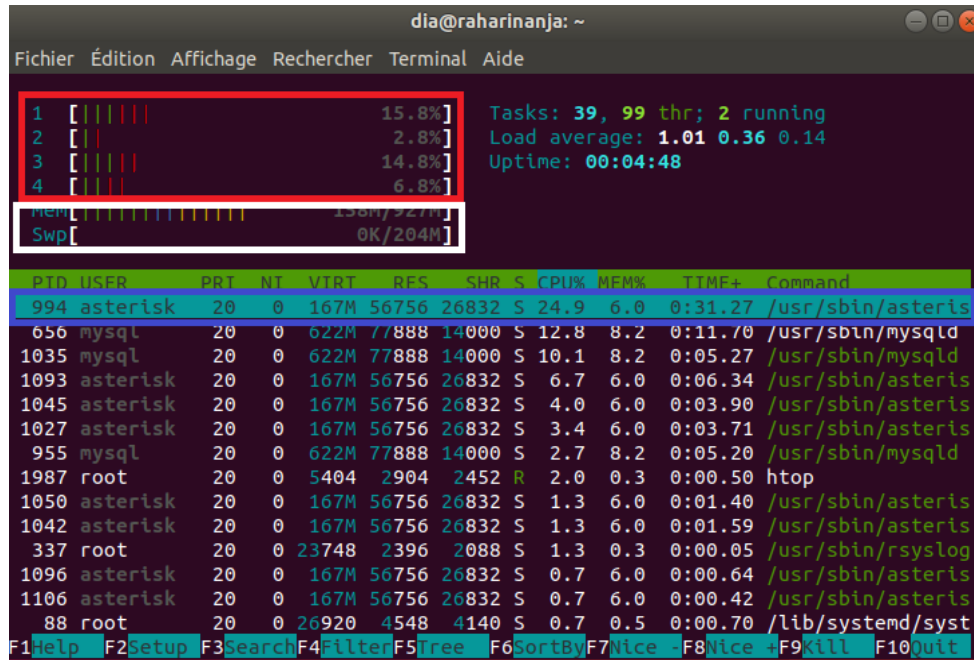


Figure 4.08 : Utilisation du processeur au démarrage du test

Dans cette figure, le rectangle rouge montre tous les processeurs de la Raspberry Pi au début du lancement du test de performance automatisé. A noter que le Raspberry Pi possède 4 cœurs. On voit sur la figure l'utilisation de chacun de ces 4 processeurs. Ils fonctionnent respectivement à 15,8%, 2,8%, 14,8% et 6,8%. En moyenne, l'utilisation du processeur est mentionnée dans le rectangle en bleu. On voit que le processeur travail à 24,9%.

L'utilisation du RAM est mentionnée dans le rectangle en blanc.

Lorsque le système commence à être saturés, c'est-à-dire on augmente le nombre d'appel par seconde, le processeur de la Raspberry devient de plus en plus surchargé.

A un taux inférieur à 50 appels/seconde, le processeur n'est pas encore chargé car l'utilisation ne dépasse pas le 80%.

Au-dessus de cette valeur, il commence à être chargé et atteint un taux d'utilisation presque 100%.

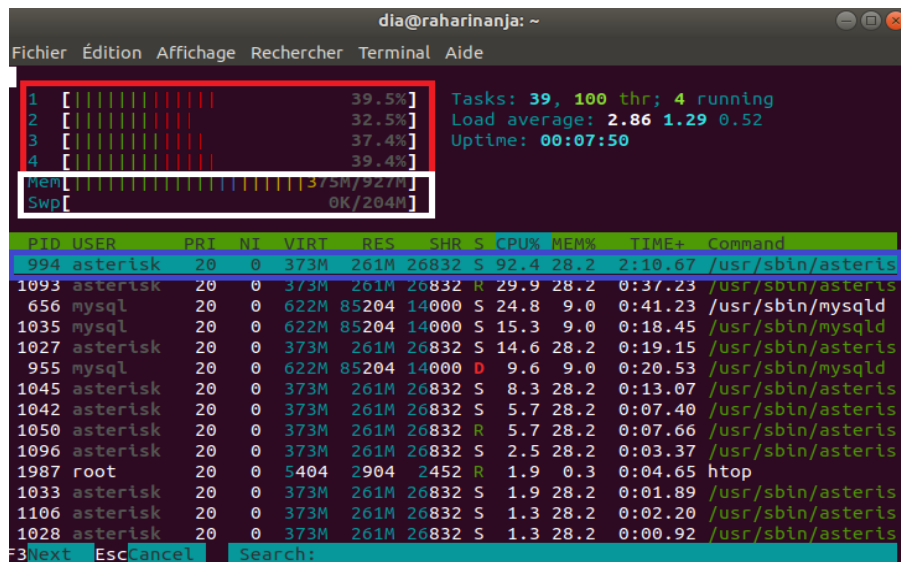


Figure 4.09 : Utilisation du processeur à un taux seuil

Le capture est ici prise lorsque le taux d'appel atteint une valeur de 100appels/seconde, grâce à celui-ci on peut tirer une valeur limite pour ne pas saturer le test.

On peut voir sur cette figure que le processeur travail à 92,4%. La RAM utilisée est de 375Mo sur 1Go. Partiellement, chaque processeur travail respectivement à 39,5%, 32,5%, 37,4% et 39,4% et ces résultats donnent son utilisation totale.

La répartition de la durée d'appel lorsque le système commence à être chargé est aussi présentée sur l'histogramme de la figure 4.10 suivant :

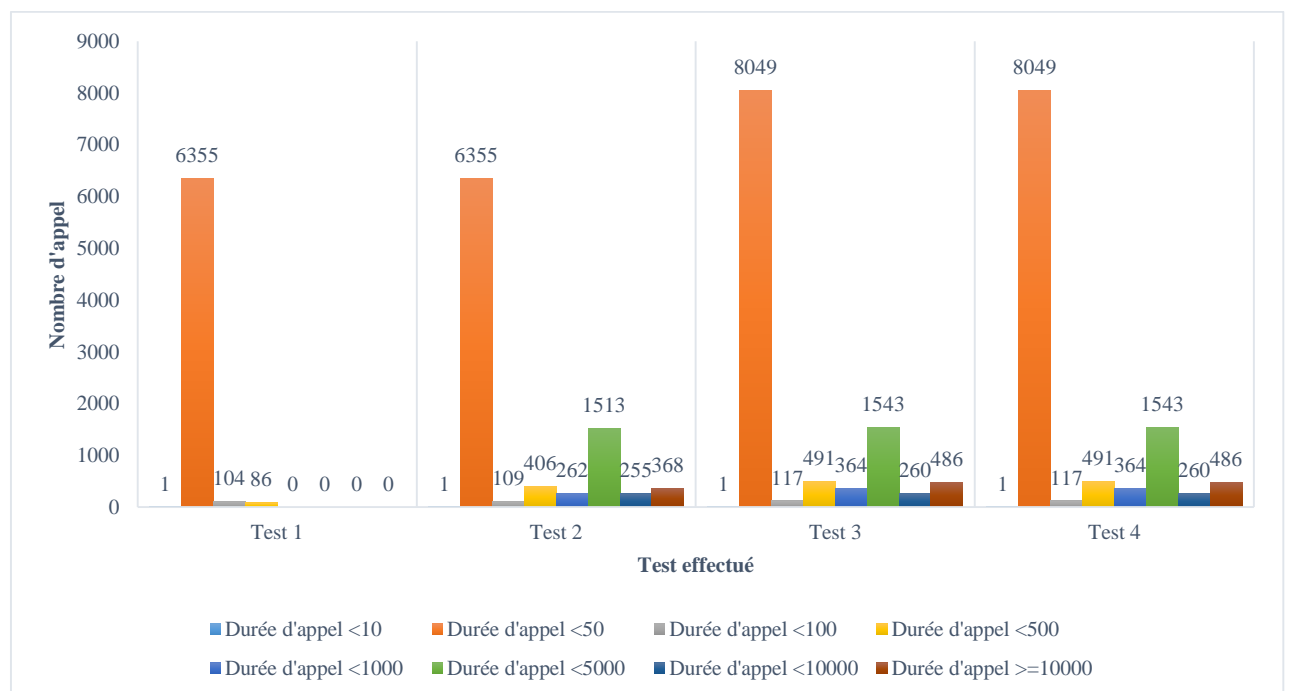


Figure 4.10 : Répartition de la durée d'appel

A propos du taux de paquet transmis, on a fait une analyse avec wireshark. La figure 4.11 montre le taux de transmission de paquet.

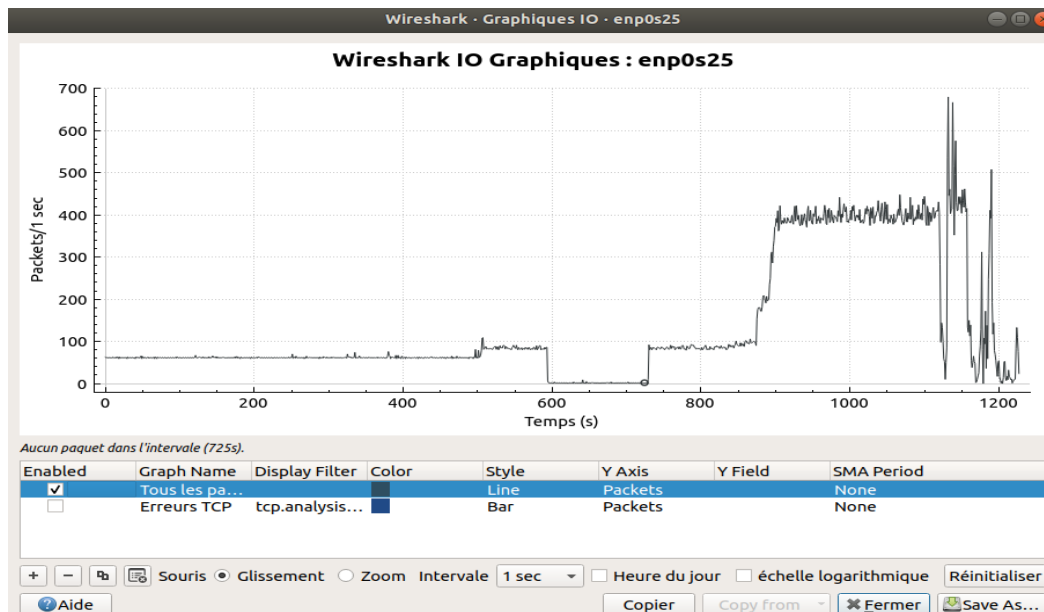


Figure 4.11 : Evolution de la transmission des paquets avec wireshark

Au début de la simulation, le taux est de 65 paquets/seconde. La transmission est stable tant qu'on n'a pas augmenté le taux d'appel transmis par seconde. La figure montre ainsi que plus on augmente le taux d'appel plus le taux de paquet augmente. Elle atteint jusqu'à 650 paquet/seconde.

On constate aussi que la descente du paquet transmis à l'instant $t = 1100$ seconde est due au surcharge du système téléphonique. La transmission de paquet diminue voire nulle après certain temps.

4.3.2. Génération des appels : UAC avec codec G711u

4.3.2.1. Les configurations

Le traitement des appels s'effectue principalement dans le fichier de configuration « *extensions.conf* » d'Asterisk.

[sipp]

exten => 2002,1,Answer()

exten => 2002,2,SetMusicOnHold(default)

exten => 2002,3,WaitMusicOnHold(20)

exten => 2002,4,Hangup()

Pour que le serveur puisse accepter le traitement des appels, on va créer le SIP-peer. Pour cela, les lignes suivantes ont été ajoutées au fichier de configuration « *sip.conf* » :

[sipp]

type=friend

context=sipp

host=dynamic

port=6000

user=sipp

canreinvite=no

disallow=all

allow=ulaw (ou autre codec selon le test)

Les deux dernières lignes permettent de refuser tous les codecs sauf celui (ou ceux) spécifiés dans la ligne allow, ce qui permet d'effectuer des tests en charge pour un codec particulier.

4.3.2.2. Test de performance et résultat

Le script de génération d'appel utilisé avec SIPp est la suivante :

sipp 192.168.0.10:5160 -s 2002 -i 192.168.0.5 -d 100 -l 500 -aa -rtp_echo -nd -r 10

192.168.0.10 serveur Asterisk ou on va envoyer des appels

-s 2002 définit le username de la requête

-i 192.168.0.5 adresse IP de l'interface locale d'écoute

-d 100s définit la durée de l'appel en ms

-l 5000 nombre max d'appels simultanés

-aa réponse automatique 200 OK sur requêtes INFO, UPDATE et NOTIFY.

-mi 192.168.0.5 adresse IP locale pour le flux média

-rtp_echo lance le mode RTP echo :

-nd sipp va s'arrêter les appels en cas de recevoir de message SIP incorrect.

-r 10 nombre d'appels par seconde

Le résultat est montré par la figure suivante :

```

dia@raharinanja: ~
Fichier Édition Affichage Rechercher Terminal Aide
10.0(100 ms)/1.000s 5060 21.08 s 210 192.168.0.10:5160(UDP)

10 new calls during 1.004 s period      1 ms scheduler resolution
1 calls (limit 500)                      Peak was 2 calls, after 0 s
0 Running, 213 Paused, 24 Woken up
0 dead call msg (discarded)              0 out-of-call msg (discarded)
3 open sockets
0 Total echo RTP pkts 1st stream          0.000 last period RTP rate (kB/s)
0 Total echo RTP pkts 2nd stream          0.000 last period RTP rate (kB/s)

Messages  Retrans  Timeout  Unexpected-Msg
INVITE ----->      210      0      0
100 <-----      210      0      0
180 <-----      0      0      0
183 <-----      0      0      0
200 <-----      0      0      0
ACK ----->      210      0      0
Pause [ 100ms]      210      0      0
BYE ----->      209      0      0
200 <-----      209      0      0

----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----

```

Figure 4.12 : *Visuel du démarrage du test sans problème*

La figure nous montre qu'aucun problème n'est survenue car le taux d'appel est de 10 appels/secondes.

Le statistique sur une période et en totalité du temps de simulation sera visualisé sur la figure 4.13. Le test est évalué sur une période, ce qui est dans la colonne « Periodic value » de la figure. Tous les statistiques seront ensuite cumulés dans la colonne « Cumulative value » sur la même figure.

```
dia@raharinanja: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

Last Reset Time      | 2019-05-30   18:11:39.576923  1559229099.576923
Current Time         | 2019-05-30   18:11:40.580246  1559229100.580246

-----+-----+-----
Counter Name         | Periodic value | Cumulative value
-----+-----+-----
Elapsed Time         | 00:00:01:003000 | 00:00:53:211000
Call Rate            | 9.970 cps      | 9.998 cps
-----+-----+-----
Incoming call created | 0              | 0
OutGoing call created | 10             | 532
Total Call created    |                 | 532
Current Call          | 2              |
-----+-----+-----
Successful call       | 10             | 530
Failed call           | 0              | 0
-----+-----+-----
Response Time 1       | 00:00:00:006000 | 00:00:00:007000
Call Length           | 00:00:00:114000 | 00:00:00:115000
-----+-----+-----
----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----
```

Figure 4.13 : *Statistique des appels après le démarrage*

Pour le temps de réponse d'appel, la plupart des appels a pour temps de réponse inférieur à 10 milliseconde. Plus précisément, le temps de réponse se reparti comme suit :

- Inferieur à 10ms
- Entre 10ms et 20 ms
- Entre 20ms et 30 ms
- Entre 30ms et 40 ms
- Entre 40ms et 50 ms
- Supérieur à 50ms

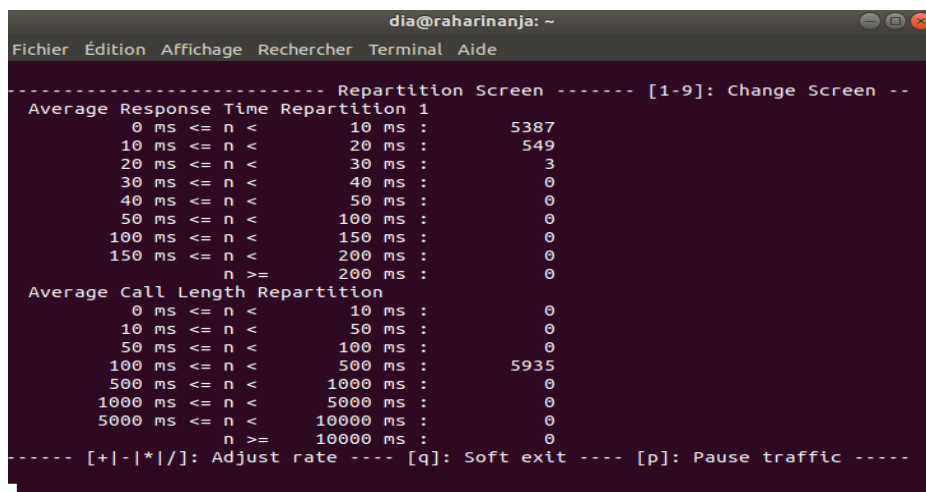


Figure 4.14 : Répartition du temps de réponse après démarrage

Le temps de réponse est en totalité inferieur à 20ms ou plus précisément inferieur à 10ms. Les autres intervalles sont négligeables. On peut ainsi dire que le système a un temps de réponse acceptable.

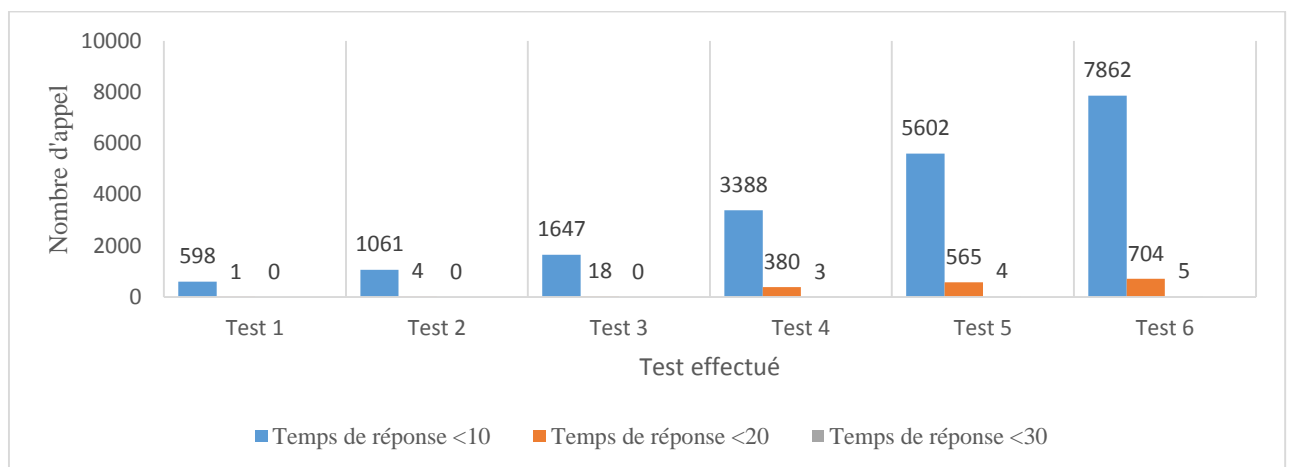


Figure 4.15 : Diagramme de répartition du temps de réponse

Lorsqu'on augmente ainsi le taux d'appel au-dessus de 60 appels/seconde, le système commence à avoir de retransmission. On remarque le début d'une saturation.

```

dia@raharinanja: ~
Fichier Édition Affichage Rechercher Terminal Aide

76 new calls during 1.004 s period      0 ms scheduler resolution
500 calls (limit 500)                   Peak was 500 calls, after 395 s
0 Running, 3556 Paused, 483 Woken up
4391 dead call msg (discarded)          0 out-of-call msg (discarded)
3 open sockets
0 Total echo RTP pkts 1st stream        0.000 last period RTP rate (kB/s)
0 Total echo RTP pkts 2nd stream        0.000 last period RTP rate (kB/s)

      Messages  Retrans  Timeout  Unexpected-Msg
INVITE ----->      13231    4723      0           0
  100 <-----      12871      0        0           0
  180 <-----         0        0        0           0
  183 <-----         0        0        0           0
  200 <----- E-RTD1 12870      0        0           0
    ACK ----->      12870      0
  Pause [ 100ms]      12870
    BYE ----->      12860    1977      0           0
    200 <-----      12731      0        0          2836

----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----
Last Error: Dead call 10156-2186@192.168.0.5 (successful), received 'SIP...

```

Figure 4.16 : *Retransmissions et échecs des appels*

Le temps de réponse devient alors de plus en plus lent, parmi les intervalles du temps de réponse énuméré précédemment, les deux premiers intervalles se cumulent encore mais on remarque qu'il y a commencement des appels dont le temps de réponse n'est pas acceptable. On a dit qu'on a des appels qui sont échoués et retransmis à cause de la surcharge du Raspberry Pi.

Il est important de noter que le nombre d'appel échoué augmente en fonction du nombre d'appel simultané mais de façon non uniforme.

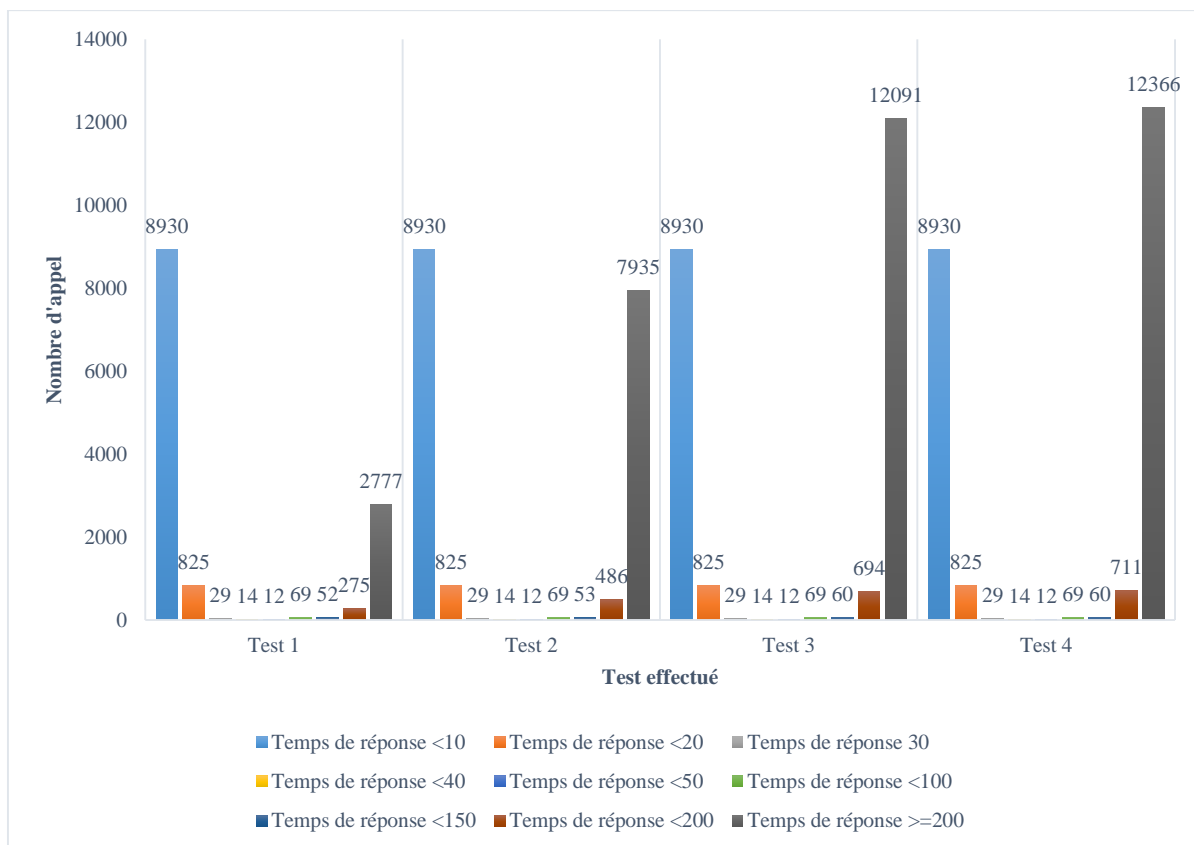


Figure 4.17 : *Temps de réponse pour les taux d'appels supérieurs à 50*

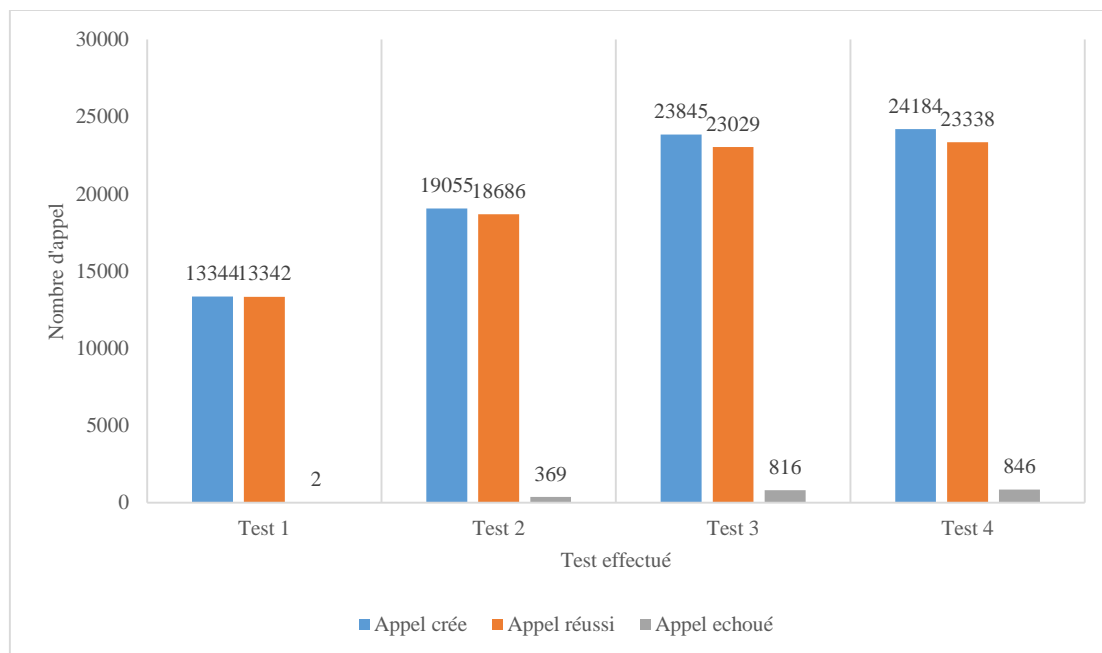


Figure 4.18 : *Appels réussis et appels échoués*

Parmi les nombreux appels fournis, des centaines sont mise en retransmission et même arrêtés. Dans cette histogramme, l'appel échoué regroupe tous les appels qui ne sont pas transmis quel que soit les causes qui peuvent exister : impossible de rétablir l'appel, délai d'attente dépassé, et d'autre. La figure 4.19 donne la répartition de la durée des appels. La durée moyenne des appels est entre 10s et 60s.

On a généré des appels entre cette intervalle de temps.

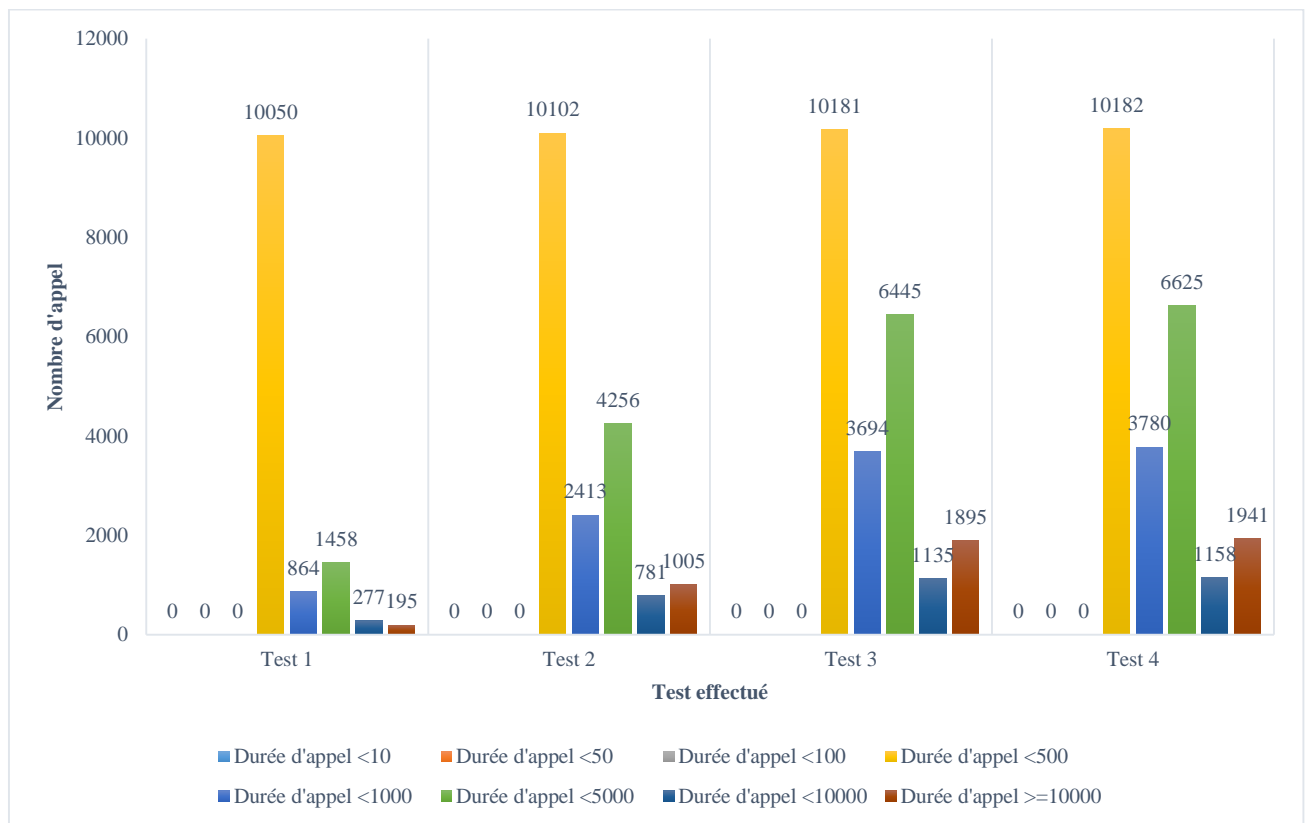


Figure 4.19 : Répartition de la durée d'appel pour taux d'appel supérieur à 50

En ce qui concerne l'utilisation du processeur de la carte Raspberry Pi, il travaille en moyenne à 73%. Cette hausse est due à l'augmentation du nombre des appels traités.

D'après l'outil htop, chacun des processeurs fonctionnent respectivement à 31,4%, 23,2%, 31,6% et 34,0%.

A propos de la RAM, elle est utilisée à 22,1%, le 363 Mo de 1Go de la RAM est utilisé pour le traitement de la voix sur IP implémenté sur la Raspberry Pi.

Ces chiffres sont représentés sur la figure 4.20.

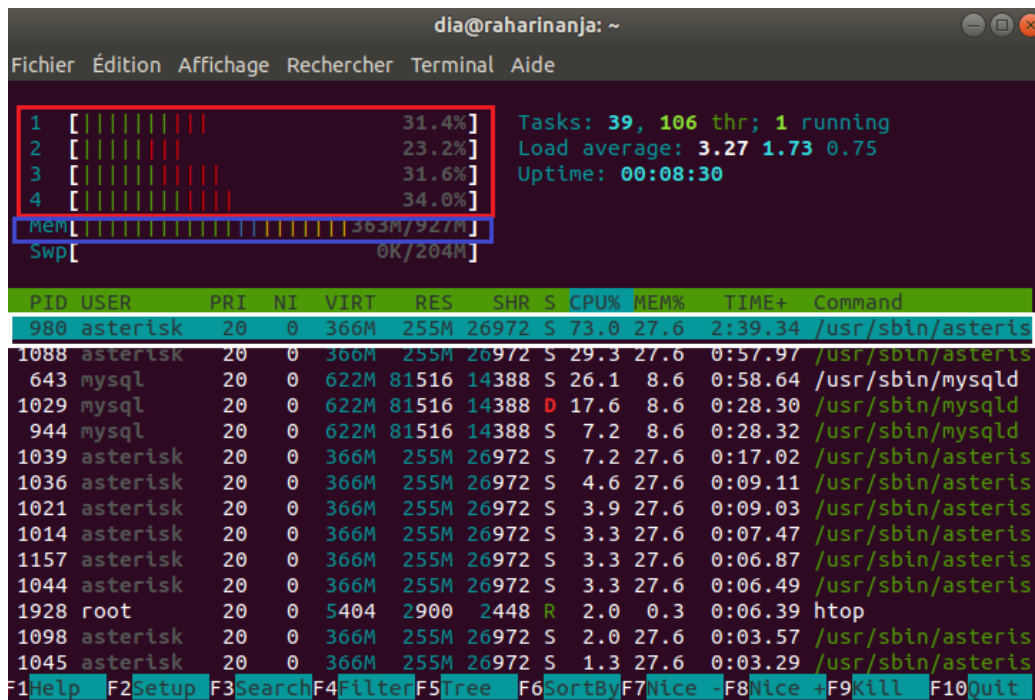


Figure 4.20 : Utilisation du processeur

On peut alors dire pour ce scenario, la voix sur IP est stable car l'utilisation du processeur ne dépasse pas de 80%. Durant le test de performance, cette valeur varie entre 70% et 80% de façon irrégulier lorsque le taux est à 80appels/seconde et plus.

La figure 4.21 donne une évolution de la transmission de paquet pendant notre test.

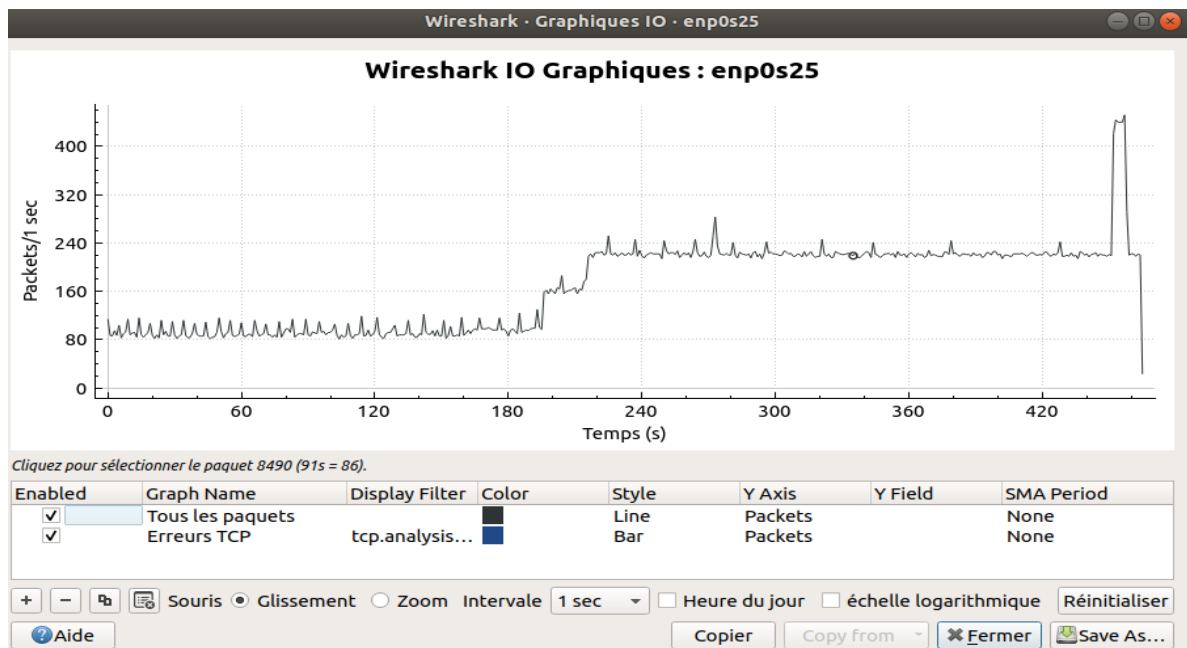


Figure 4.21 : Evolution de la transmission de paquet

En terme de transmission de paquet, elle se fait à 220 paquets/seconde en moyenne. Au début du test, le paquet transmis est environ 100paquets/seconde. Lorsqu'on a augmenté le taux, il augmente aussi et atteint un seuil de 450paquets/seconde mais se tourne sur la valeur de 220. Au moment où le système commence à être saturé, alors certains paquets sont perdus ou retransmis. Cela entraîne ainsi une baisse de la vitesse de transmission et voir même nul lorsqu'il est totalement bloqué.

Bref, le taux de paquet est d'environ 80 paquets/seconde. Plus on augmente le nombre d'appel, plus le nombre de paquet augmente aussi.

4.3.3. Test et comparaison avec des différents codecs

4.3.3.1. Scenario

Dans ce test, le processus de génération d'appel commence par 10 appels simultanés pour atteindre un nombre maximal d'appels simultanés, d'une durée approximative de 10 secondes chacun. Lors de chaque appel, on a envoyé un média à l'aide des codecs G.711, GSM, G.722. Pour chaque codec, nous avons effectué 5 tests, en faisant varier le nombre d'appels simultanés de 10 à 50 par pas de 10.

Du côté du système téléphonique, nous avons configuré un flux d'appel simple qui était exécuté à chaque fois qu'un nouvel appel était reçu. En gros, l'appel est pris et une partie de la musique est jouée pendant 10 secondes. Enfin, l'appel est raccroché. Le schéma complet est illustré à la figure 4.22 :

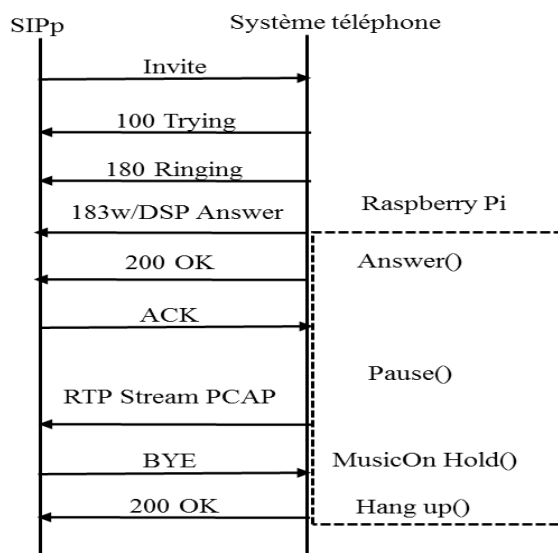


Figure 4.22 : Call flow pour évaluer les codecs

4.3.3.2. Résultat du test

Comme prévu, les ressources matérielles requises pour le traitement des appels ne dépendent pas uniquement du codec utilisé pour compresser la voix, mais surtout de la charge simultanée reçue par le système en termes de nombre d'appels (simultanés). Nous avons mesuré l'utilisation du processeur et de la RAM de la plate-forme Raspberry Pi en utilisant chacun des codecs G.711, GSM, G.722. Les résultats suivants montrent que l'utilisation de la CPU augmente avec le nombre d'appels simultanés, quel que soit le codec utilisé. De plus, en termes d'utilisation de la CPU, le GSM est le codec le moins exigeant, contrairement à G.729, qui nécessitent plus de travail de traitement que les autres.

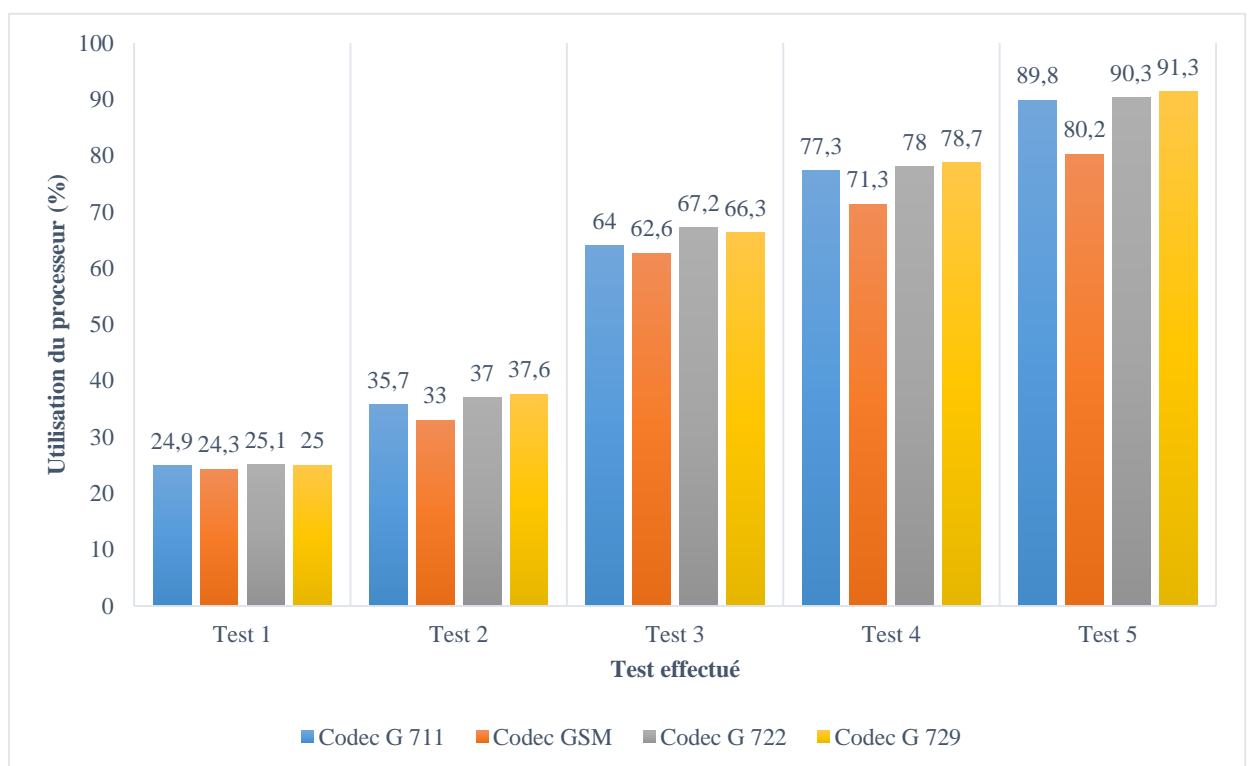


Figure 4.23 : *Utilisation du processeur en fonction du codec*

Cela semble raisonnable étant donné la qualité élevée offerte par ces derniers codecs. Bien que, théoriquement, le G.711 ne soit pas un codec gourmand en ressources processeur, il ne présente pas les meilleures performances en termes d'utilisation du processeur.

Le pourcentage d'utilisation du processeur atteint des valeurs supérieures à 80% lorsque plus de 75 appels sont traités simultanément par la plate-forme Raspberry Pi.

Les résultats obtenus sur l'utilisation du processeur et de la RAM révèlent des comportements intéressants liés aux codecs utilisés.

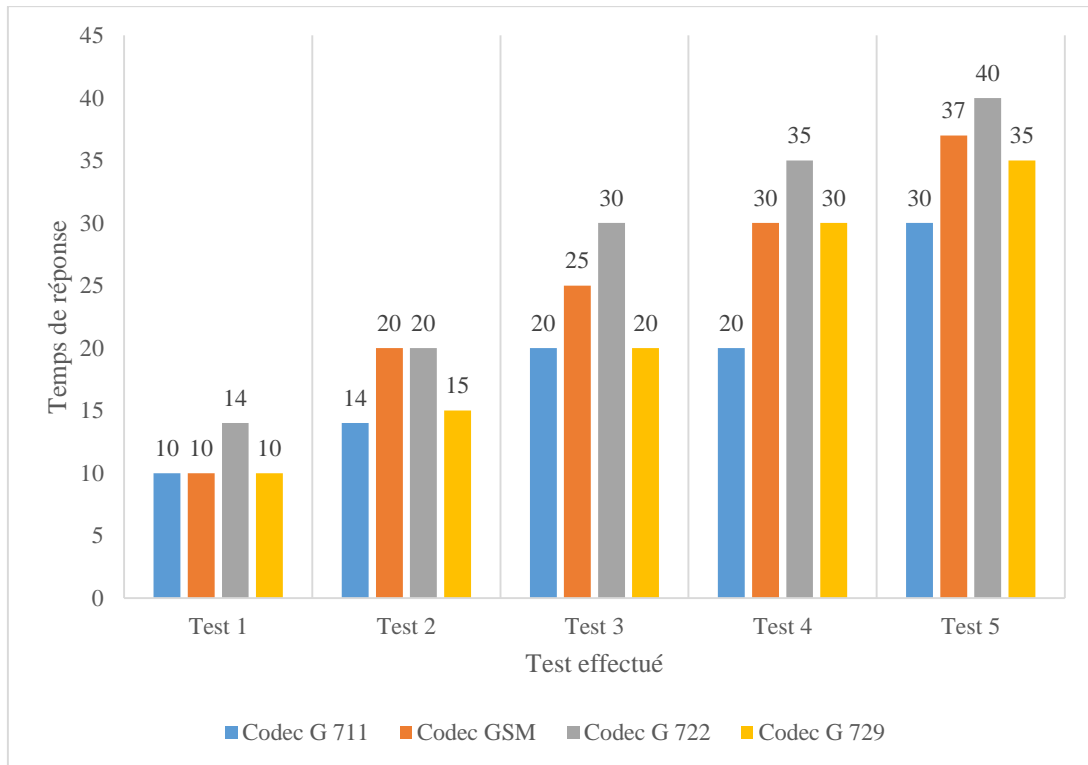


Figure 4.10 : *Temps de réponse en fonction du codec*

Sur la base des statistiques générées, on a pu collecter des informations sur les temps de réponse pris par le système de téléphonie IP pour répondre à chaque appel entrant. Lorsque le système commence à surcharger, ses temps de réponse varient de manière non uniforme car certains appels ne peuvent plus être traités.

4.4. Conclusion

Cette partie a permis d'évaluer et mesurer les performances d'un système de téléphonie IP déployé sur une plate-forme Raspberry Pi.

Les résultats obtenus sont intéressants en paramètre d'utilisation de processeur, en temps de réponse du système téléphonique et finalement le taux de réussite et le taux d'échec des appels générés. L'utilisation du processeur ne dépasse pas de 80% au-dessous d'un taux d'appel de 50 appels/seconde. Pour le temps de réponse, la majorité des appels a pour temps de réponse inférieur à 20ms ce qui est acceptable pour le système.

CONCLUSION GENERALE

La téléphonie est un service très intéressant au sein d'une entreprise. Son intégration à internet a donné naissance à une nouvelle technologie VoIP qui consistait à transmettre la voix sur un réseau numérique. Et grâce au développement open source, la plupart des systèmes informatiques supportent l'installation d'un service PBX. C'est le cas de la plate-forme Raspberry Pi qui est un matériel informatique de taille réduite et très économique, et compatible avec les systèmes d'exploitation basé Linux.

Ce présent mémoire est une approche permettant de mesurer la performance d'un système de téléphonie IP déployé sur une plate-forme Raspberry Pi.

Apparemment, la plate-forme effectue un très bon travail de traitement des appels puisque le pourcentage d'utilisation du processeur n'a pas dépassé 80% même pour 50 appels simultanés. Cependant, lors de l'évaluation des paramètres d'application spécifiques, tels que les temps de réponse et les appels échoués, pour nombre d'appel supérieur à 50, on est confronté au divers problèmes de l'état actuel du système téléphonique.

Les temps de réponse augmentent considérablement jusqu'à 500 ms lorsque plus de 50 appels simultanés sont traités. G.729 fonctionne très bien même mieux que G.711 et GSM en termes de temps de réponse, par contre GSM est le meilleur pour l'évaluation des appels échoués. Le codec G.722 provoque un décrochage ou retransmission lorsque plus 60 appels simultanés sont générés.

Les travaux futurs devraient inclure une évaluation plus approfondie des perceptions à propos du système, en marquant tous les codecs déjà décrit et, peut-être, un environnement sans fil plus complexe.

Les processus d'encodage / décodage étant très gourmands en ressources processeur, il serait intéressant d'évaluer le système également devant un schéma où plusieurs codecs sont utilisés simultanément. À première vue, une carte Raspberry Pi sera convenable dans l'installation d'une plateforme VoIP pour une petite et moyenne entreprise, en terme de communication vocale bien que d'autres services couplés tels que la vidéoconférence doit encore être testé.

ANNEXES

ANNEXE I : PORTION DE UAC.xml

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE scenario SYSTEM "sipp.dtd">

<scenario name="Basic Sipstone UAC">
  <!-- In client mode (sipp placing calls), the Call-ID MUST be -
->
  <!-- generated by sipp. To do so, use [call_id] keyword. -
->
  <send retrans="500">
    <![CDATA[

      INVITE sip:[service]@[remote_ip]:[remote_port] SIP/2.0
      Via: SIP/2.0/[transport] [local_ip]:[local_port];branch=[branch]
      From: sipp <sip:sipp@[local_ip]:[local_port]>;tag=[call_number]
      To: sut <sip:[service]@[remote_ip]:[remote_port]>
      Call-ID: [call_id]
      CSeq: 1 INVITE
      Contact: sip:sipp@[local_ip]:[local_port]
      Max-Forwards: 70
      Subject: Performance Test
      Content-Type: application/sdp
      Content-Length: [len]

      v=0
      o=user1 53655765 2353687637 IN IP[local_ip_type] [local_ip]
      s=-
      c=IN IP[media_ip_type] [media_ip]
      t=0 0
      m=audio [media_port] RTP/AVP 0
      a=rtpmap:0 PCMU/8000

    ]]>
  </send>

  <recv response="100"
    optional="true">
  </recv>

  <recv response="180" optional="true">
  </recv>

  <!-- By adding rrs="true" (Record Route Sets), the route sets -
->
  <!-- are saved and used for following messages sent. Useful to test -
->
  <!-- against stateful SIP proxies/B2BUAs. -
->
  <recv response="200" rtd="true">
  </recv>

  <!-- Packet lost can be simulated in any send/recv message by -
->
```

```

<!-- by adding the 'lost = "10"'. Value can be [1-100] percent. -
->
<send>
  <![CDATA[

    ACK sip:[service]@[remote_ip]:[remote_port] SIP/2.0
    Via: SIP/2.0/[transport] [local_ip]:[local_port];branch=[branch]
    From: sipp <sip:sipp@[local_ip]:[local_port]>;tag=[call_number]
    To: sut <sip:[service]@[remote_ip]:[remote_port]>[peer_tag_param]
    Call-ID: [call_id]
    CSeq: 1 ACK
    Contact: sip:sipp@[local_ip]:[local_port]
    Max-Forwards: 70
    Subject: Performance Test
    Content-Length: 0

  ]]>
</send>

<!-- This delay can be customized by the -d command-line option -
->
<!-- or by adding a 'milliseconds = "value"' option here. -
->
<pause/>

<!-- The 'crlf' option inserts a blank line in the statistics report. -
->
<send retrans="500">
  <![CDATA[

    BYE sip:[service]@[remote_ip]:[remote_port] SIP/2.0
    Via: SIP/2.0/[transport] [local_ip]:[local_port];branch=[branch]
    From: sipp <sip:sipp@[local_ip]:[local_port]>;tag=[call_number]
    To: sut <sip:[service]@[remote_ip]:[remote_port]>[peer_tag_param]
    Call-ID: [call_id]
    CSeq: 2 BYE
    Contact: sip:sipp@[local_ip]:[local_port]
    Max-Forwards: 70
    Subject: Performance Test
    Content-Length: 0

  ]]>
</send>

<recv response="200" crlf="true">
</recv>

<!-- definition of the response time repartition table (unit is ms) -
->
<ResponseTimeRepartition value="10, 20, 30, 40, 50, 100, 150, 200"/>

<!-- definition of the call length repartition table (unit is ms) -
->
<CallLengthRepartition value="10, 50, 100, 500, 1000, 5000, 10000"/>

</scenario>

```

ANNEXE II : LES MOTS CLES DANS UN SCENARIO XML

Un scénario SIPp s'écrit en XML. La liste des mots clés est donnée par le tableau suivant :

Mot clé	Défaut	Description
[service]	service	Champ service, tel qu'il est passé en paramètre dans l'option –s service_name
[remote_ip]	-	Adresse IP distante, telle qu'elle est passée dans la ligne de commande
[remote_port]	5060	Port IP distant. Il est possible d'ajouter un offset calculé [remote_port+3] à cette valeur
[transport]	UDP	Selon la valeur du paramètre –t, prendra la valeur 'UDP' ou 'TCP'
[local_ip]	Adresse de l'hôte	Valeur du paramètre –i
[local_ip_type]	-	Prendra la valeur '4' ou '6' selon le type d'adresse du paramètre –i (IPv4 ou v6)
[local_port]	Random	Valeur du paramètre –p. Il est possible d'y ajouter un offset calculé [local_port+3]
[len]	-	Longueur calculée du body SIP. A utiliser dans le header "Content-Length". Il est possible d'y ajouter un offset [len+3]
[call_number]	-	Index. Le call_number commence à 1 et est incrémenté de 1 à chaque appel
[cseq]	-	Génère automatiquement le numéro de CSeq. La valeur initiale est 1 par défaut. Il est possible de la changer en utilisant l'option –base_cseq dans la ligne de commande
[call_id]	-	Un Call ID identifie un appel et est généré par SIPp pour chaque nouvel appel. En mode client, il est obligatoire d'utiliser la valeur générée par SIPp dans le header "Call-ID". Sinon, SIPp ne reconnaîtra pas la réponse au message envoyé comme faisant partie de l'appel en cours.

Tableau A2.01 : *Les mots clés dans un scenario XML(1)*

Mot clé	Défaut	Description
[media_ip]	-	Valeur du paramètre –mi
[media_ip_type]	-	Prendra la valeur '4' ou '6' selon le type d'adresse du paramètre –mi (IPv4 ou v6). Utile pour construire le SDP indépendamment du type d'adresse IP media
[media_port]	-	Valeur du paramètre –mp. Il est possible d'y ajouter un offset [media_port+3]
[auto_media_port]	-	Seulement pour pcap. Utilisé pour que les ports audio et video commencent à la valeur du paramètre –mp et changent pour chaque appel en utilisant un système périodique, modulo 1000 (ce qui limite à 10000 le nombre de sessions concurrentes pour pcap_play)
[field0-n]	-	Utilisé pour injecter les valeurs d'un fichier CSV externe
[\$n]	-	Utilisé pour injecter la valeur d'un numéro de variable d'appel n
[authentication]	-	Utilisé pour construire le header d'authentification. Ce champ peut avoir plusieurs paramètres, de la forme suivante : [authentication username=myusername password=mypassword]. Si aucun username n'est spécifié, la valeur du paramètre –s est utilisée. Si aucun password n'est spécifié, la valeur du paramètre –ap est utilisée.
[pid]	-	Donne le process ID du principal thread SIPp
[last_*]	-	Remplacé automatiquement par le header spécifié s'il était présent dans le précédent message (sauf s'il s'agissait d'une retransmission).
[routes]	-	Si l'attribut 'rrs' d'une commande recv est à "true" alors le header "Record-Route" du message reçu est stocké et peut être rappelé en utilisant le mot clé [routes]

Tableau A2.02 : *Les mots clés dans un scenario XML (2)*

ANNEXE III : LES OPTIONS DE SIPP

L'usage de la commande SIPP est le suivant :

```
# sipp remote_host[:remote_port] [options]
```

De nombreuses options sont disponibles, afin de spécifier le scénario, les options de contrôle de trafic, et bien d'autre :

Option	Description
-v	Affiche la version et les informations de copyright
-bg	Lance SIPP en mode Background
-p local_port	Définit le numéro du port local. Par défaut c'est un port libre aléatoire choisi par le système
-buff_size buff_size	Définit la taille du buffer d'émission et de réception
-i local_ip	Définit l'adresse IP locale pour les headers 'Contact', 'Via' et 'From'. Par défaut c'est l'adresse IP de l'hôte
-inf file_name	Injecte les valeurs d'un fichier CSV externe dans les scénarios.
-d duration	Contrôle la durée des instructions 'pause' dans le scénario, s'ils n'ont pas de section 'milliseconds'. La valeur par défaut est 0.
-r rate (cps)	Définit le taux d'appels (en calls per second). Cette valeur peut être modifiée dynamiquement pendant le test en pressant '+', '-', '*' ou '/'.
-rp period (ms)	Spécifier la période en millisecondes pour le taux d'appel. La valeur par défaut est 1 seconde. Cela permet d'avoir n appels tous les m millisecondes, en utilisant -r n -rp m.
-rate_increase	Spécifier l'augmentation du taux d'appel toutes les -fd secondes. Cela permet d'augmenter la charge sur plusieurs périodes indépendantes
-rate_max	Si -rate_increase est défini, quitter après que le taux d'appels ait atteint cette valeur.

Tableau A3.01 : Les options de SIPP(1)

Option	Description
-max_sockets max	Définit le nombre maximum de sockets à ouvrir simultanément.
-timer_resol	Définit la résolution du timer en millisecondes. Cette option a un impact sur la précision des timers. De petites valeurs permettent un ordonnancement plus précis mais ont un impact sur l'utilisation du CPU.
-max_recv_loops	Définit le nombre maximum de messages reçus lus par cycle. La valeur par défaut est 1000.
-base_cseq n	Valeur de départ de [cseq] pour chaque appel
-cid_str string	Chaine de caractère du Call ID (par défaut %u-%p@%s) %u=call_number, %s = ip_address, %p = process_number
-auth_uri uri	Forcer la valeur de l'URI pour l'authentification. Par défaut, l'URI est composée de remote_ip:remote_port
-sf filename	Charge un fichier scénario xml alternatif.
-sn name	Utilise un scénario par défaut (intégré dans l'exécutable sipp).
-sd name	Affiche un scénario par défaut (i.e. intégré à l'exécutable sipp)
-t transport_mode	u1 UDP avec un socket (défaut) un UDP avec un socket par appel ui UDP avec un socket par adresse IP. t1 TCP avec un socket tn TCP avec un socket par appel l1 TLS avec un socket ln TLS avec un socket par appel
-trace_msg	Enregistre les messages SIP envoyés et reçus dans <scenario file name>_<pid>_messages.log
-trace_screen	Enregistre les écrans de statistiques dans le fichier <scenario_ name>_<pid>_screens.log en quittant SIPp.

Tableau A3.02 : Les options de SIPp(2)

Option	Description
-trace_timeout	Enregistre les Call IDs des appels avec timeout dans <scenario_name>_<pid>_timeout.log
-trace_stat	Enregistre toutes les statistiques dans le fichier <scenario_name>_<pid>.csv. Utiliser l'option '-h stat' pour une description détaillée du contenu du fichier de statistiques.
-stf file_name	Définir le nom de fichier à utiliser pour enregistrer les statistiques.
-stat_delimiter string	Définit le délimiteur pour le fichier de statistiques
-trace_err	Tracer tous les messages inattendus dans <scenario_name>_<pid>_errors.log.
-trace_logs	Autoriser le traçage des actions <log> dans <scenario_name>_<pid>_logs.log
-trace_rtt	Autoriser le traçage des temps de réponse dans <scenario_name>_<pid>_rtt.csv
-rtt_freq freq	La fréquence est obligatoire. Enregistre les temps de réponse tous les 'freq' appels dans le fichier de log défini par -trace_rtt. La valeur par défaut est 200.
-s service_name	Définir la partie username de la Request URI. La valeur par défaut est 'service'.
-ap password	Définir le mot de passe pour les problèmes d'identification. La valeur par défaut est 'password'.
-f frequency	Définir la fréquence des rapports de statistiques à l'écran, en secondes. La valeur par défaut est 1
-fd frequency	Définir la fréquence de l'enregistrement des statistiques dans les logs, en secondes. La valeur par défaut est 60
-l call limits	Définir le nombre maximum d'appels simultanés. La valeur par défaut est 3 * durée d'un appel * taux d'appels
-m nb_calls	Arrêter le test et quitter SIPp quand 'nb_calls' appels sont traités

Tableau A3.03 : Les options de SIPp (3)

Option	Description
-rtp_echo	Activer l'écho RTP
-mp media_port	Définir le numéro de port local pour l'écho RTP. La valeur par défaut est 6000
-mi local_rtp_ip	Définir l'adresse IP locale pour le media
-mb buff_size	Définir la taille du buffer pour l'écho RTP. La valeur par défaut est 2048
-nr	Désactiver la retransmission des paquets en mode UDP
-max_retrans	Nombre maximum de retransmissions UDP avant que l'appel ne se termine pour cause de timeout. La valeur par défaut est de 5 pour les requêtes INVITE et de 7 pour les autres.
-revc_timeout nb	Timeout de réception en millisecondes. Si le message attendu n'est pas reçu, l'appel est annulé pour cause de timeout
-timeout nb	Timeout global en secondes. Si cette option est définie, SIPp se ferme au bout de nb secondes
-nd	No Default. Désactive tous les comportements par défaut de SIPp, comme si sur un timeout de retransmission UDP, annuler l'appel par un BYE ou un CANCEL
-pause_msg_ign	Ignorer les messages reçus pendant une pause définie dans le scénario
-rsa host[:port]	Définir l'adresse d'envoi distante (Remote Sending Address) comme host:port pour l'envoi de messages
-max_reconnect	Définir le nombre maximum de reconnections
-reconnect_sleep int	Définir le temps de 'sleep' entre la fermeture et la reconnexion
-aa	Activer la réponse 200 OK automatique pour les messages INFO et NOTIFY
-key keyword value	Définir le paramètre générique 'keyword' à la valeur 'value'

Tableau A3.04 : Les options de SIPp (4)

BIBLIOGRAPHIE

- [1] A. Benchikh, K. Mechernene, « *Etude de la sécurité dans la VOIP* » Mémoire de fin d'études, Université Abou Bakr Belkaid– Tlemcen A.U : 2014-2015
- [2] Z. Boukherrou, « *Optimisation du routage P2P pour la VoIP : Application à la signalisation SIP* », Mémoire de Magister, Université A/Mira de Béjaïa, 2005-2006
- [3] P. M. Simon, « *La Qualité de Service de la Voix sur IP Principes et Assurance* », Le Descartes, excellent
- [4] N.A. Rakotoarisoa, « *Evaluation de la QoS de la VoIP dans les réseaux sans fils 802.11* », Mémoire de fin d'études, dép. TCO, A.U : 2009 / 2010
- [5] N. M. V. Ravonimanantsoa, « *Analyse de performance de la voip dans les reseaux IEEE 802.11* », DEA, dép TCO, A. U : 2008 / 2009
- [6] A. M. B. Aboubakr, « *Application mobile de la voIP sur un réseau Wifi* », Mémoire, Université de Tlemcen , Juin 2014
- [7] M. Schaefer, « *Introduction à la téléphonie IP Asterisk sous GNU/Linux Informatique libre* », Avril 2008
- [8] A. Ahmed, « *La téléphonie IP* », INSCAE, 2012-2013
- [9] O. DABBEBI, « *Gestion des risques dans les infrastructures VoIP* », Doctorat, Université de Lorraine, L.L.R.I.A, Juin 2013
- [10] J. Cartailier, R. R. Poliméni, « *Le Raspberry Pi : un nano -ordinateur au service de la science et de l'enseignement INRA* », UMR1145, F - 91300 Massy, Avril 2013
- [11] C. Dupaty, « *Raspberry Pi presentation-ressources* », BTS, Systèmes Numériques Lycée Fourcade, Gardanne Académie d'Aix-Marseille.

- [12] F.Sion, « *Raspberry Pi - Prise en main Documentation* », Version 0.1 , Sept 2017
- [13] R. Fondation, « *welcome to RasPBX–Asterisk for Raspberry Pi* » <http://www.raspberry-asterisk.org>, Fevrier 2016
- [14] L. McNamara, I. Marsh, S. Forslincy, « *CheesePi:A Raspberry Pi based measurement platform* », Mai 2010
- [15] R. Foundation,«*raspberry-pi-3-model-b*»,<https://www.raspberrypi.org/product> ,Mai 2016
- [16] A. Lazin, « *Caractéristiques et spécifications du Raspberry Pi* » <https://raspy-world.com/raspberry-pi-model> , Avril 2017
- [17] A. MICHEL, « *Premiers pas avec un Raspberry Pi 3* », Creative Commons, Juin 2017
- [18] S. Monk, « *Programmez un raspberry pi* », Dunod Paris, 2014
- [19] A. L. Razafindrakoto, « *Analyse des performances des codecs G.711, G.723.1 et G.729 utilises en VOIP dans le réseaux WIMAX mobile* », Mémoire de fin d'études, dép. TCO, A.U : 2012 / 2013
- [20] T. Christiansen, I. Giotis, S. Mathur, « *Performance Evaluation of VoIP in Different Settings* », Juin 2010.
- [21] P. Gomathinayagam, S. Jayanthi, «*Performance Optimization of Codec in VOIP using Raspberry Pi* » Engineering and Manufacturing, 2018
- [22] N.M.V Ravonimanantsoa, M.A. Rakotomalala, « *évaluation de performance de la voip* », LR-TASI ED-STII, ESPA.

- [23] G.Friets ,« G.729 coding of speech at 8 kbps using CS-ACELP» Juin 2011
- [24] R. Molet «G.711 PCM Codec at 64 Kbps (A-Law/ μ -Law Companding) », Sept 2010
- [25] A.A. Ravoavahy, « *Analyse de performance de la voip sur un backbone MPLS avec traffic engineering* » Mémoire de fin d'études, dép. TCO, A.U : 2012 / 2013
- [26] M. Docs, «*Le Protocole SIP : Contexte, protocole, analyses et mise en œuvre* » <https://sip.goffinet.org/sip-01-architecture> ,Mars 2019
- [27] J. Fredh, « *Présentation des codecs : Complexité, prise en charge matérielle, note moyenne d'opinion (MOS) et négociation* », Mars 2013.
- [28] N. Lucien, G. Marcus, M. Leary, « *Hardware implementation of the PCM codec for VoIP telephony* », International Journal of Engineering Science and Technology, 2010
- [29] ITU-T, « *speech-codecs* », www.floreatinc.com/products/speech-codecs, ITU-T Recommendation,
- [30] U. Pettersson, «*Measuring VoIP Quality of Experience using the Raspberry Pis*», Master's Thesis, TRITA, 2011
- [31] J.Mathias, « *utiliser-sipp-pour-tester-la-montee-en-charge-de-votre-serveur-sip* » , <https://www.blog-des-telecoms.com>, Mars 2019
- [32] A.Salys,« *asterisk-freepbx-raspberry-pi-2-voip sipserver* » ,<https://www.algissalys.com>,Avril 2019
- [33] R. Thymio, L. Mindstorms, R. Cozmo ,« *Raspberry Pi 3 modèle B* », [http://Génération Robots/Electronique et robotique/Raspberry Pi/Raspberry Pi 3 modèle B](http://GénérationRobots/Electronique-et-robotique/RaspberryPi/RaspberryPi3modèleB), Juin 2018

- [34] J. Philippe, « *realiser-une-expertise-monitoring-analyse-troubleshooting-voip-sip-h-323* », <https://www.wavetel.fr/solution/reseaux-performances>, Mars 2019
- [35] Tsrit, « *Sip* », <https://tsrit.com/category/sip>, Mars 2019

PAGE DE RENSEIGNEMENT

Nom : RA-HARINANJA

Prénoms : Dia

Adresse : Lot A II 29 Avaratsena Betafo
Betafo 113
MADAGASCAR
Tel : +261 34 37 981 61
e-mail: dia_raharinanja@yahoo.fr



Titre du mémoire :

« ANALYSE DE PERFORMANCE DE LA VOIX SUR IP BASEE SUR LA
PLATE-FORME RASPBERRY PI »

Nombre de pages : 108

Nombre de tableaux : 12

Nombre de figures : 55

Mots clés : VoIP, Raspberry, performance, protocole, codec

Directeur de mémoire :

Nom : RAVONIMANANTSOA

Prénoms : Ndaohialy Manda-Vy

Grade : Maitre de conférences /Docteur HDR

Tel : +261 34 11 358 00

e-mail : ndaohialy@gmail.com

RESUME

Notre travail a pour but d'analyser la performance de la voix sur IP basée sur le Raspberry Pi. Des paramètres objectifs ont été utilisés pour l'analyse tel l'utilisation du processeur et de la mémoire RAM, les temps de réponse aux appels et les appels échoués. En effet, on a constaté après les tests que le choix du codec et le nombre d'appels simultanés déterminent le point de surcharge du processeur de la Raspberry Pi ce qui est justifié par des temps de réponse plus longs et des appels échoués. En tenant compte du cout de la RasPi parce qu'aucune licence de matériel ou logiciel n'est requise, et compte tenu des performances offertes par le RasPi avec des niveaux de charge différents, son choix est très intéressant pour les petites et moyennes entreprises.

Mots clés : VoIP, Raspberry, performance, protocole, codec

ABSTRACT

Our work aims to analyze the performance of VoIP based on the Raspberry Pi. Objective parameters were used for the analysis such as the use of the processor and the RAM memory, the call response times and failed calls. Indeed, it was found after the tests that the choice of the codec and the number of simultaneous calls determine the point of overload of the processor of the Raspberry Pi which is justified by longer response times and calls failed. Taking into account the cost of the RasPi because no hardware or software license is required, and given the performance offered by the RasPi with different load levels, its choice is very interesting for small and medium enterprises.

Keywords: VoIP, Raspberry, performance, protocol, codec