ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"ESTUDIO DEL SERVICIO DE TELEFONÍA IP SOBRE REDES WIFI EMPLEANDO VIRTUALIZACIÓN DE REDES INALÁMBRICAS"

TRABAJO DE TITULACIÓN

Previo a la obtención del Título de:

MAGISTER EN TELECOMUNICACIONES

JOSÉ JOAQUÍN MOREIRA QUIROZ

GUAYAQUIL – ECUADOR

AÑO: 2017

AGRADECIMIENTO

Gracias a Dios, por sus permanentes bendiciones, siendo la culminación de este trabajo, una muestra de ellas.

Dejo constancia de mi profundo agradecimiento al PhD. Álvaro Suárez Sarmiento, por su valiosa contribución intelectual, durante la dirección del presente trabajo.

De igual manera, mi agradecimiento, a mi Alma Máter, la Escuela Superior Politécnica del Litoral (ESPOL), la cual, a través de la Facultad de Ingeniería en Electricidad y Computación (FIEC), viene promoviendo acertadamente, el perfeccionamiento de profesionales en el área de las Telecomunicaciones, mediante el programa de Maestría en Telecomunicaciones, MET-ESPOL.

DEDICATORIA

En memoria de mi padre Ing. Joaquín Moreira (+), cuya fortaleza hasta el último de sus días, inspiró la culminación de este trabajo.

Dedicado a Dios, a mi esposa, a mi madre, mis hermanos, hermana y sobrinas, seres a quienes amo profundamente.

TRIBUNAL DE EVALUACIÓN

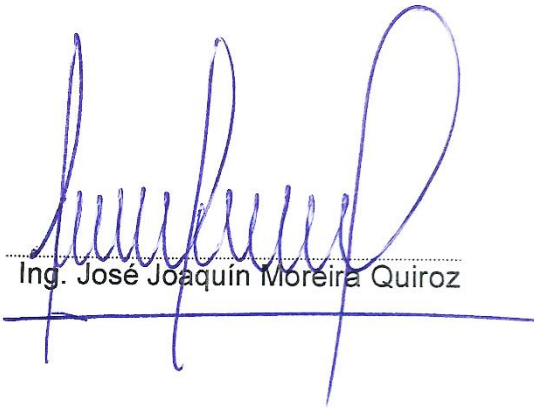
PhD. César Martín Moreno
SUBDECANO DE LA FIEC

PhD. Álvaro Suárez Sarmiento
DIRECTOR DEL TRABAJO DE TITULACIÓN

Mgs. Miguel Molina Villacis
MIEMBRO PRINCIPAL DEL TRIBUNAL

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, me corresponde exclusivamente; y doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



RESUMEN

La Virtualización de Redes Inalámbricas, es un paradigma que ofrece varias características que pueden ser aprovechadas por las tecnologías de comunicación inalámbrica existentes y emergentes, para su despliegue y operación.

A pesar de estas características, existe una ausencia de literatura en el medio, que aborde este tema, por lo que, se ha desarrollado el presente trabajo, a fin de contribuir a llenar esa ausencia de información y de conocer, además, el comportamiento de este paradigma cuando opera sobre una red inalámbrica de tipo WiFi, sobre la cual se transmiten comunicaciones de telefonía IP, para lo cual se implementa un demostrador que utiliza virtualización de redes inalámbricas.

El concepto de *virtualización* ha sido utilizado en el ámbito de las ciencias de la computación desde hace ya algunos años, con el fin de re - utilizar recursos tales como procesamiento, almacenamiento y memoria. De la misma manera, en el campo de las telecomunicaciones, se puede lograr un mayor aprovechamiento de diversos elementos de la red, lo que conlleva en primera instancia a una disminución de costos de operación y mantenimiento. La virtualización puede operar, sobre diferentes componentes de una red inalámbrica, siendo posible incluso aplicar conceptos de virtualización sobre el espectro radioeléctrico.

Por otra parte, diversas tecnologías, algunas ya existentes como la Telefonía IP y en otros casos, tecnologías emergentes como el Internet de las Cosas o la Telefonía Móvil 5G, podrían encontrar en la *virtualización de redes inalámbricas* un mecanismo para facilitar e impulsar su despliegue. Cabe destacar que organismos como la Unión Internacional de las Telecomunicaciones, recientemente han creado una recomendación para empezar a estandarizar el despliegue y uso de la *virtualización de redes inalámbricas*, hecho que, desde nuestro punto de vista, justifica su estudio.

Por lo expuesto, a través del presente trabajo, se pretende explorar la tecnología de la virtualización de redes inalámbricas a través de los diferentes estudios realizados sobre la materia, y cómo esta se comporta cuando es utilizada en combinación con otra tecnología, ampliamente utilizada, como lo es la Telefonía IP, utilizando una red inalámbrica basada en WiFi.

ÍNDICE GENERAL

AGRADECIMIENTO ii
DEDICATORIA iii
TRIBUNAL DE EVALUACIÓN iv
DECLARACIÓN EXPRESA v
RESUMEN vi
CAPÍTULO 1 1
1. ANTECEDENTES 1
1.1. Descripción del Problema 1
1.2. Justificación 4
1.3. Objetivos 5
1.4. Metodología 5
1.5. Resultados Esperados 6
1.6. Elementos Diferenciadores o Innovadores 7
CAPÍTULO 2 8
2. CONCEPTOS DE VIRTUALIZACIÓN DE REDES DE COMUNICACIÓN Y TELEFONÍA IP 8
2.1. Ideas básicas de la Virtualización de Redes 8
2.1.1. Definición 8
2.1.2. Importancia de la Virtualización de Redes 9
2.1.3. Consideraciones de Diseño 10
2.1.4. Metas del Diseño 11
2.2. Tipos de Virtualización de Redes 12
2.2.1. Virtualización basada en host 13
2.2.2. Virtualización basada en protocolo 13
2.2.3. Virtualización basada en redes sobre puestas 13
2.2.4. Virtualización basada en redes activas y programables 13
2.3. Virtualización de Funciones de Red 14
2.3.1. Infraestructura NFV 14
2.3.2. Funciones de Red Virtual y Servicios 14
2.3.3. Modelo de Negocio en la NFV 15
2.3.4. Consideraciones de Diseño 16

2.4. Redes Definidas por Software 17
2.4.1. Definición 17
2.4.2. Objetivos 18
2.4.3. Arquitectura SDN 18
2.4.4. Protocolo OpenFlow 21
2.5. Conceptos sobre Telefonía IP 25
2.5.1. CODEC, Paquetizador y Buffer de Reproducción 26
2.5.2. Real Time Protocol 27
2.5.3. Session Initiation Protocol 33
2.6. Telefonía IP sobre WiFi 47
2.6.1. Generalidades y nivel de acceso al medio de WiFi con infraestructura 47
2.6.2. Características de WiFi y movimiento de terminales 53
2.6.3. Gestión de tráfico de telefonía IP en WiFi con herramientas software 57
2.6.4. Parámetros de Calidad de Servicio y Experiencia de usuario para Telefonía IP en WiFi 63
CAPÍTULO 3 71
3. ESTADO DEL ARTE EN VIRTUALIZACIÓN DE WiFi CON INFRAESTRUCTURA 71
3.1. Estratégias para la virtualización 71
3.1.1. Virtualización de redes inalámbricas basada en flujo 71
3.1.2. Virtualización de redes inalámbricas basada en protocolo 72
3.1.3. Virtualización de redes inalámbricas basada en Espectro e Interfaz RF 74
3.2. Virtualización del Punto de Acceso 75
3.2.1. Concepto de Punto de Acceso Virtual 75
3.2.2. Virtualización de Punto de Acceso mediante MAC Compartida 76
3.2.3. Virtualización de Punto de Acceso mediante Hipervisor ... 78
3.2.4. Utilización de Puntos de Acceso Virtuales 79
3.3. Virtualización de la Infraestructura 80
3.3.1. Virtualización de Interfaces Inalámbricas IEEE 802.11 80
3.3.2. Visión completa de WiFi con infraestructura Virtualizada .. 82

3.3.3. Casos de Uso 84
CAPÍTULO 4 88
4. DISEÑO DEL DEMOSTRADOR, PRUEBAS A REALIZAR Y RECURSOS UTILIZADOS 88
4.1. Consideraciones Iniciales 88
4.1.1. Descripción del escenario propuesto 88
4.1.2. Selección de variables a estudiar y pruebas a realizar 89
4.1.3. Condiciones y limitaciones del demostrador propuesto 90
4.2. Recursos utilizados en la implementación 91
4.2.1. Metodología a utilizar en la recolección de datos 91
4.2.2. Selección de equipos utilizados en la implementación 92
4.2.3. Hardware y software empleados en la implementación 93
4.2.4. Herramientas utilizadas para medición de parámetros de red 94
CAPÍTULO 5 95
5. IMPLEMENTACIÓN Y RESULTADOS DE LAS PRUEBAS EXPERIMENTALES CON Y SIN WiFi VIRTUALIZADO PARA TELEFONÍA IP 95
5.1. Implementación de la red Inalámbrica Virtualizada 95
5.1.1. Virtualización de Interfaz Inalámbrica WiFi: creación de VPA 95
5.1.2. Instalación y configuración de Servidores de Telefonía IP virtuales 99
5.1.3. Segmentación de la Red mediante Conmutador Virtual y VLANs 109
5.1.4. Instalación y configuración de Servidor y Clientes de Telefonía IP 113
5.2. Resultados de parámetros de Calidad de Servicio 121
5.2.1. Throughput y latencia 121
5.2.2. Paquetes perdidos e influencia de buffers 127
5.2.3. Comparación entre WiFi con y sin virtualización 128
5.3. Resultados de Calidad de experiencia de usuario 129
5.3.1. Medición del MOS 130
5.3.2. Comparación entre WiFi con y sin virtualización 139

5.3.3. Correlación entre parámetros de Calidad de Servicio y Calidad de Experiencia de usuario 141
5.4. Estudio Económico y de Viabilidad del Demostrador Propuesto 143
5.4.1. Cálculo de los costos de implementación 144
5.4.2. Cálculo del Valor Actual Neto del Demostrador 144
5.4.3. Análisis de resultados de Estudio Económico 146
CONCLUSIONES Y RECOMENDACIONES 147
BIBLIOGRAFÍA 149
ANEXOS 155

CAPÍTULO 1

1. ANTECEDENTES

En este capítulo se exponen las consideraciones que han dado origen al presente trabajo. Se empieza describiendo el problema a investigar, pasando por la justificación para su realización junto con los objetivos buscados y la metodología empleada. Se incluyen también, los resultados esperados además de los elementos diferenciadores e innovadores que se han introducido en esta investigación.

1.1. Descripción del Problema

En la actualidad existe un interés elevado por desarrollar y redefinir las tecnologías de Red para implantar redes apoyándose en tecnologías como la Virtualización de Redes Inalámbricas (*WNV*, del inglés *Wireless Network Virtualization*) [1] [2] [3] [4] [5]. Las redes basadas en el estándar IEEE 802.11 del Instituto de Ingenieros Eléctricos y Electrónicos (*IEEE,* del inglés *Institute of Electrical and Electronics Engineers*) (estandarizado como *Wireless Fidelity* (*WiFi*) por la Empresa) también son objeto de re implantación teniendo en cuenta la WNV [6]. Existen varios trabajos científico - técnicos dedicados al estudio de la aplicación del concepto de Virtualización y de la técnica *Software Defined Networking* (SDN).

En [7] se presenta el concepto SDN, el por qué es necesaria esta tecnología, su arquitectura, haciendo además una presentación del protocolo utilizado en SDN llamado *OpenFlow* e implementaciones de redes basadas en este protocolo. La descripción del Marco de Referencia Y.3300 de la Unión Internacional de Telecomunicaciones sector Estandarización (*ITU-T,* del inglés *International Union of Telecommunications - Standardization*) (ITU-T Y.3300) para crear SDN y se especifican sus elementos indispensables, objetivos, capacidades, requisitos y arquitectura, se presenta en [8].

La virtualización y la tecnología SDN se ha aplicado a WiFi. En [9] se estudian las SDN y la virtualización de red inalámbrica. Se proporciona además, los desafíos, espectativas y otros aspectos que proporcionan las *Redes Inalámbricas*

Definidas por Software (*SDWN*) y WNV. En [10] se presenta el concepto SDN, Virtualización de Red y Virtualización de Funciones de Red (*NFV*, del inglés *Network Function Virtualization*) y se realiza diferenciaciones entre dichos términos.

La virtualización de *Puntos de Acceso* (*PA*) WiFi ha sido también objeto de muchos estudios. En [11] se analiza el desempeño de los *Virtual PA* (*VPA*) sobre IEEE 802.11. Además, se presentan varios escenarios los cuales aprovechan de mejor manera los beneficios de los VPA junto con las recomendaciones para implementar esos escenarios. En [12] se realiza una exposición de los factores que benefician la adopción de la Virtualización de PA así como las técnicas de virtualización que existen, presentando los inconvenientes principales a la hora de implementar la virtualización de redes inalámbricas. En [13] se proponen nuevas ideas sobre PA WiFi virtuales.

Los PA virtuales permiten definir redes WiFi virtuales para los que se han publicado varios trabajos. En [14] se presenta dos métodos que permite virtualizar la *Interfaz de Red Inalámbricas* (*WNIC*). Adicionalmente, presenta varias simulaciones con la intención de mostrar como mediante la virtualización de WNIC IEEE 802.11 se posibilita el uso eficiente de energía en redes en malla inalámbricas y en el *Soft Handover*. En [15] se muestra el diseño e implementación de Virtual WiFi, en la cual se emplean máquinas virtuales para soportar las funciones inalámbricas.

Con esta propuesta se pueden lograr múltiples conexiones inalámbrica de manera separada, realizadas mediante una interfaz de red inalámbrica física. En [16] se presenta la posibilidad de realizar conexiones de múltiples redes inalámbricas por medio de una sola red física. Un elemento importante es la existencia de herramientas de virtualización de software abierto para definición y almacenamiento de redes inalámbricas virtuales sobre una infraestructura física compartida [17]. La gestión dinámica de tráfico en WiFi puede ser implantada de manera eficiente mediante SDN [18].

Por otro lado, la Telefonía sobre Protocolo Internet (*Telefonía IP*) (*IP,* del inglés *Internet Protocol*) [19], es ampliamente usada en todas las empresas del Mundo.

Un problema muy importante que se deriva del estudio preliminar del estado del Arte del despliegue de la Telefonía IP sobre redes inalámbricas WiFi virtualizadas, es que no hay estudios que demuestren su viabilidad para la Empresa.

En el presente trabajo, se presenta un estudio del Estado del Arte del uso de la virtualización de redes inalámbricas, en redes WiFi y posteriormente se analiza un caso de uso de ese tipo de redes para soportar Telefonía IP en una Empresa hipotética del Ecuador que se abstrae mediante un escenario hipotético (pero que puede ser muy común en la práctica). En la Figura 1.1 se muestra dicho escenario. El escenario hipotético propuesto que presenta el problema cuenta con 3 áreas: *Call Center*, *Bodega* y *Transportación*. Dichas áreas cuentan cada una con un servidor de telefonía IP, un encaminador y un conmutador Ethernet. Se desea proporcionar Telefonía IP mediante WiFi a los usuarios de la Bodega y Transportación. Por ello se instala en dichas áreas AP WiFi.

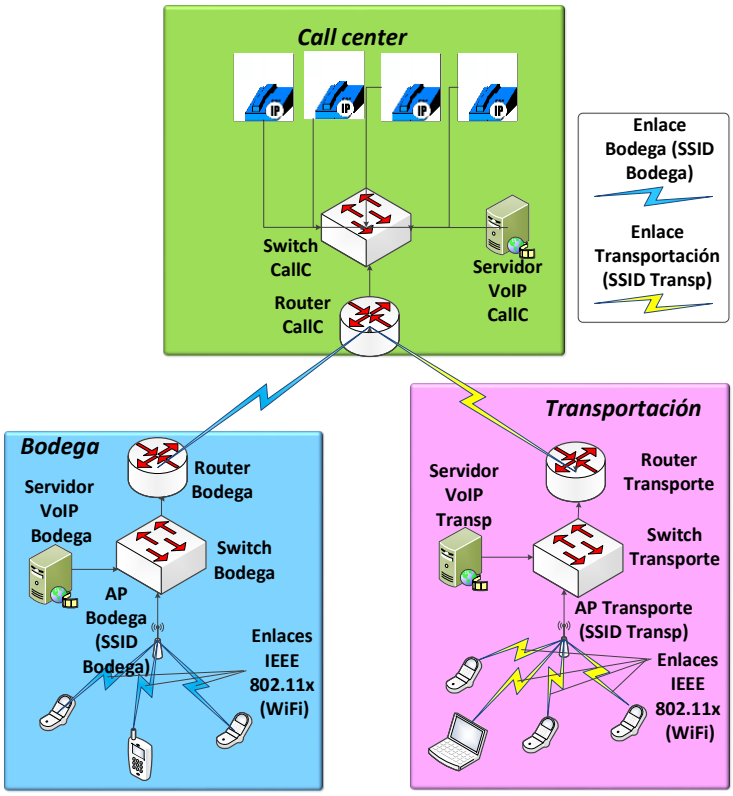


Figura 1.1: Escenario propuesto para demostrar el despliegue y operación

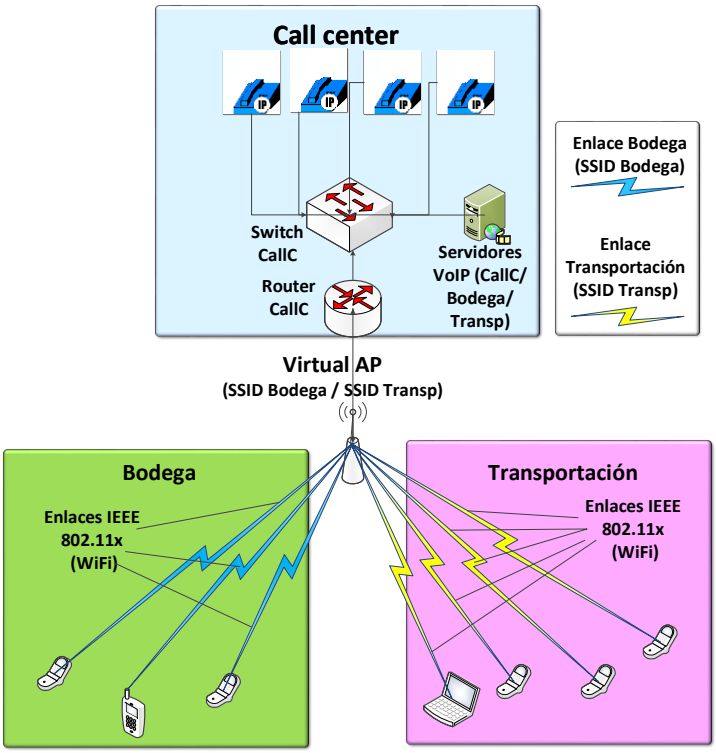


Figura 1.2: Red Inalámbrica Virtualizada

En el escenario de la Figura 1.1 sustituimos los dos PA por un solo PA con capacidad para cubrir inalámbricamente (físicamente) a la Bodega y la Transportación. Además, se eliminan los dos servidores de Telefonía IP de estas áreas, los dos encaminadores y los dos conmutadores, con el consiguiente ahorro económico y gastos de administración de recursos de red. Para mantener las dos redes WiFi originales se virtualizan dos redes (mediante SDN) WiFi que se encaminan por dos redes de área local virtual (*VLAN,* del inglés *Virtual Local Area Network*) en el encaminador del *Call Center*. En la Figura 1.2 se esboza un esquema del escenario a desplegar.

La intención es demostrar que se obtienen niveles de *Calidad de Servicio* (*QoS*, del inglés *Quality of Service*) y de Experiencia de usuario (*QoE*, del inglés *Quality of Experience*) similares a los del escenario original en presencia de diversos tipos de tráfico en la red.

1.2. Justificación

El trabajo por realizar se justifica, a distintos niveles, en la medida en que:

a) *Técnico*: de todos los trabajos científico-técnicos analizados y publicados recientemente, no se ha encontrado ninguno que combine WLAN, WNV y Telefonía IP.

b) *Económico*: una solución del tipo anterior puede reportar beneficios económicos a la Empresa al disponer de una red de muy bajo coste y de un servicio de Telefonía IP también de bajo coste.

c) *Académico*: un avance en el estudio de un caso de uso de WNV con redes WiFi para Telefonía IP es de mucho interés puesto que pondría de manifiesto la viabilidad y futuro de estas tecnologías para un servicio de Telecomunicación muy usado en la actualidad.

1.3. Objetivos

- Estudiar la evolución, y el estado de desarrollo en que se encuentra el estado del arte de la aplicación de la Virtualización de Redes Inalámbricas sobre redes WiFi con topología en infraestructura.

- Identificar carencias o necesidades relacionadas con la literatura, en la aplicación de la Virtualización de Redes Inalámbricas sobre redes WiFi en modo infraestructura para el soporte de Telefonía IP.

- Identificar las ventajas, problemas y desafíos tanto técnicos como económicos que puede reportar el caso de uso a analizar.

1.4. Metodología

De la revisión preliminar efectuada, el presente trabajo se trata de un *Estudio de Caso*, en el cual la Investigación se lleva a cabo aplicando un Diseño Transaccional Descriptivo, por cuanto no se manipula de manera intencional ni se realizan asignaciones al azar, del objeto estudiado (aplicación de la Virtualización de Redes Inalámbricas sobre WiFi para el soporte de VoIP y análisis de un Caso de Uso).

Por una parte, se realiza una revisión de Literatura, para tratar de establecer la situación actual del Arte de la Virtualización de Redes Inalámbricas para su

utilización en comunicaciones de Telefonía IP y de esta manera llenar la ausencia de literatura en torno al tema.

Por otro lado, aprovechando los conceptos encontrados y estudiados en el transcurso de la presente investigación, se despliega un demostrador como se muestra en la Figura 1.3, que simula el escenario de una Red Inalámbrica Virtualizada, basado en el escenario de la Figura 1.2 y se analiza el desempeño de comunicaciones basadas en VoIP, sobre este demostrador.

Este despliegue, se realiza en un ambiente basado en un Sistema Operativo de Código Libre (Linux Ubuntu), utilizando los paquetes y herramientas que proporcione el Sistema para este tipo de implementaciones.

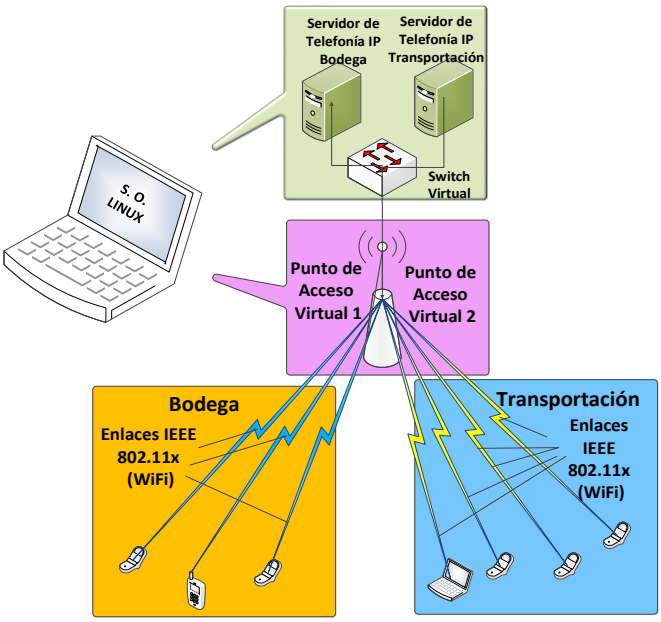


Figura 1.3: Demostrador de aplicación de WiFi con topología de infraestructura

Además, se analizan posibles efectos adversos que podrían presentarse al tratar de virtualizar redes WiFi que sean incompatibles para soportar Telefonía IP.

1.5. Resultados Esperados

- Conocer los trabajos de investigación que han sido realizados, relacionados con la WiFi aplicando Virtualización de Redes Inalámbricas para el soporte de Telefonía IP.

- Definir cuáles son los recursos necesarios para la implementación de una red WiFi con infraestructura que aplique Virtualización de Redes Inalámbricas que soporte a la Telefonía IP.

- Poder establecer ventajas y desventajas de realizar una comunicación inalámbrica de Telefonía IP en un escenario concreto con una red WiFi con infraestructura aplicando Virtualización de Redes Inalámbricas.

- Presentar los resultados del proyecto propuesto, en publicaciones internacionales que verifiquen la importancia del mismo.

1.6. Elementos Diferenciadores o Innovadores

El presente trabajo, combina la utilización de conocimientos adquiridos en varias Materias estudiadas dentro de la Maestría en Telecomunicaciones: Redes de Datos, Telefonía IP e incorpora el concepto de la Virtualización de Redes Inalámbricas sobre comunicaciones WiFi como elemento diferenciador, para establecer los beneficios de Virtualizar una Red Inalámbrica WiFi, para el transporte de Telefonía IP.

Adicionalmente, de la revisión previa efectuada, no se han desarrollado Estados del Arte que analicen los efectos de combinar las tecnologías de Telefonía IP con WiFi aplicando Virtualización de Redes Inalámbricas, por lo que el presente estudio pretende contribuir a llenar ese vacío.

Por otra parte, esta propuesta introduce como elemento innovador el estudio de un caso de uso de red WiFi con infraestructura aplicando Virtualización de Redes Inalámbricas. Para ello se analizará el equipamiento comercial o de libre distribución que exista a nivel internacional para proponer una solución viable a nivel práctico, estudiando su viabilidad tanto técnica como económica.

CAPÍTULO 2

2. CONCEPTOS DE VIRTUALIZACIÓN DE REDES DE COMUNICACIÓN Y TELEFONÍA IP

En este capítulo se abordan las bases en las que se sustenta el presente trabajo, en el área de la virtualización de redes y otras tecnologías que tienen el mismo fin: un mejor aprovechamiento de los recursos de infraestructura de telecomunicaciones.

Se revisan, además, los componentes básicos, de un sistema de Telefonía IP. Además, se estudia el desempeño de la Telefonía IP sobre WiFi y los parámetros que garantizan una adecuada cooperación entre ambas tecnologías.

2.1. Ideas básicas de la Virtualización de Redes

Previo al estudio de la WNV, es necesario conocer varios aspectos relacionados con la virtualización de redes de comunicación, en la cual se fundamenta la WNV.

2.1.1. Definición

La virtualización como tal, no es un concepto nuevo, ya que se lo ha venido empleando en el campo de la virtualización de computadores, donde recursos físicos como memoria, procesamiento, almacenamiento, tarjeta de red... de un equipo host, pueden ser compartidos entre varios computadores virtuales.

La virtualización de red [3] puede ser definida como: *una tecnología que posibilita la creación de particiones de red aisladas lógicamente sobre redes físicas compartidas*. *Esto incluye la agregación de múltiples recursos de un proveedor como un solo recurso*.

Otros autores [20] definen a la virtualización de red como *la compartición de recursos de red a través de la abstracción y aislamiento de las funcionalidades de red del hardware físico*.

En el caso de la Virtualización de Redes, los recursos en los que se enfoca, son los nodos y enlaces. Un nodo puede ser cualquier equipo de

red, como un conmutador o un encaminador. Un enlace, es una conexión física o lógica entre dos nodos en la red [20].

Con lo expuesto, la virtualización de redes, puede ser visualizada como se muestra en la Figura 2.1.

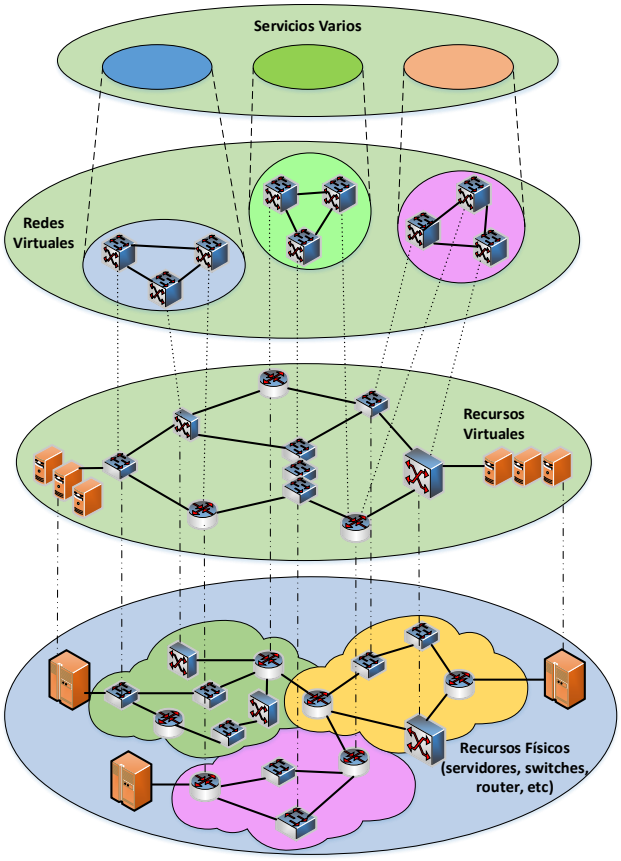


Figura 2.1: Arquitectura de la Virtualización de Redes

2.1.2. Importancia de la Virtualización de Redes

Existen varios aspectos por los cuales la Virtualización de Redes se destaca. Entre ellos se tiene que, esta tecnología puede tener aplicaciones en el campo académico o comercial. En el campo académico, esta tecnología puede ser usada para implementar bancos de prueba, aprovechando la flexibilidad, programabilidad, personalización y aislamiento que se puede lograr cuando se utiliza la virtualización de

redes para probar experimentos sin las dificultades que se presentan en entornos reales [21].

Mientras que, comercialmente, un factor importante a la hora de inclinarse por virtualizar una red, es el hecho de que esta tecnología permite optimizar recursos, esto mediante la reutilización de componentes de infraestructura de red a través de la compartición de los mismos. Esto en consecuencia, conlleva a una disminución en el consumo energético de los componentes virtualizados.

Los recursos que pueden ser virtualizados, van desde enlaces, nodos, interfaces de red e incluso se habla de la virtualización del espectro radioeléctrico.

Adicionalmente, la virtualización de redes, es una de las tecnologías en la cual se fundamenta la Computación en la Nube (Cloud Computing), por cuanto pueden coexistir en una misma Infraestructura Sistemas Heterogéneos a bajo costo [22].

La recuperación de desastres, también es un campo de aplicación en el que la Virtualización de Redes podría ser utilizada, ya que se podrían desarrollar Servicios de Administración de Emergencia para Recuperación de Desastres, utilizando redes heterogéneas [4].

2.1.3. Consideraciones de Diseño

Existen varios principios que se deben seguir en la virtualización de redes según [23], los cuales son:

- *Coexistencia*: se refiere al hecho de que múltiples redes virtuales de diferentes proveedores de servicio pueden coexistir juntas, extendiéndose sobre parte o la totalidad de las redes físicas subyacentes

- *Recursión*: esta existe, cuando una o más redes virtuales son engendradas desde otra red virtual creando una jerarquía de redes virtuales con relaciones padre – hijo. A este principio también se lo conoce como Anidamiento de Redes Virtuales.

- *Sucesión*: un *Hijo* de una red virtual en un ambiente de virtualización de red puede heredar atributos de arquitectura desde sus padres, lo cual también significa que las restricciones sobre la red virtual *Padre* automáticamente se trasladan a restricciones similares sobre los hijos. La sucesión permite a un proveedor de servicios añadir valor a los Hijos de las redes virtuales antes de re venderlos a otros proveedores de servicios.

- *Revisión*: permite a un nodo físico alojar múltiples nodos virtuales de una sola red virtual. El uso de múltiples encaminadores lógicos para manipular diversas funcionalidades en una gran red compleja permite a un proveedor de servicios re arreglar lógicamente la estructura de red y simplificar la administración de una red virtual. La Revisación puede ser útil también para crear bancos de prueba.

2.1.4. Metas del Diseño

Para este caso, en [3] se establecen varias características que deben poseer las particiones lógicas que se crean al utilizar virtualización de redes:

- *Particionamiento*: cada partición de red aislada lógicamente consiste de un conjunto de recursos virtuales que son una partición administrable independientemente de recursos físicos. Pueden existir múltiples particiones de red aisladas lógicamente sobre una red física.

- *Abstracción*: un recurso virtual dado no necesita corresponder directamente a su recurso físico. La información detallada del recurso físico se puede abstraer para que otros sistemas, aplicaciones o usuarios accedan a las capacidades del recurso virtual usando interfaces abstraídas. Esas interfaces se pueden usar para garantizar compatibilidad para acceder el recurso virtual y proveer su control eficiente. También, es posible extender las interfaces para proveer mayor capacidad. El recurso virtual puede ser manipulado a través de interfaces bien definidas y extensibles, y asignadas para crear, modificar, reclamar y liberar particiones de red aisladas lógicamente.

- *Aislamiento*: los recursos virtuales que forman una partición de red aislada lógicamente, están aislados de las otras particiones, para que no puedan interferirse entre ellos, en términos de desempeño, seguridad y espacio de nombres y para que cualquier partición individual no pueda causar disrupciones a otra partición o red física. Los datos en una partición no se escapan a través de las particiones sin autorización y las aplicaciones se pueden comunicar solamente sobre conexiones de red configuradas. Los accesos no autorizados a otra partición se prohíben.

- *Flexibilidad* (*Elasticidad*): los recursos virtuales para construir una partición de red aislada lógicamente son asignados flexiblemente, reclamados y liberados sobre demanda para maximizar la acomodación de múltiples particiones sobre recursos físicos, para optimizar el uso de recursos físicos tanto temporal como espacialmente, y también permitir el uso instantáneo y explosivo además de uso continuo de recursos físicos.

- *Programabilidad*: los recursos virtuales para construir una partición de red aislada lógicamente pueden ser programados para desarrollar, desplegar y experimentar con nuevos protocolos de comunicación para la difusión de datos innovadores y para facilitar el procesamiento eficiente de datos para ser habilitado dentro de la partición de red aislada lógicamente.

- *Autenticación*, *Autorización y Contabilización*: el uso de recursos virtuales para crear una partición de red aislada lógicamente debe ser autenticado y autorizado para que pueda lograr operaciones seguras de las particiones de red aisladas lógicamente, previniendo el abuso de los recursos virtuales y ataques maliciosos sobre ellos. Es necesario tener en cuenta los recursos virtuales asignados en las redes físicas para que la integridad de los recursos virtuales pueda ser examinada y monitoreada y el uso de los recursos virtuales pueda ser optimizado.

2.2. Tipos de Virtualización de Redes

De acuerdo con [24], existen varios tipos de virtualización de redes. Se podría inferir, que esta depende del ambiente en el que se realiza la Virtualización, las

cuales pueden ser: virtualización basada en host, en protocolo, en redes sobrepuestas o basada en redes activas y programables.

2.2.1. Virtualización basada en host

En este tipo de virtualización, dispositivos del nivel de red, tales como encaminadores son implementados en computadores mediante software de virtualización. Sin embargo, tienen la desventaja de ser más lentos que los dispositivos dedicados por cuanto, recursos como la memoria física son compartidos con otros procesos del computador.

2.2.2. Virtualización basada en protocolo

Dentro de esta categoría, se encuentran las VLAN y las redes privadas virtuales (*VPN*, del inglés *Virtual Private Network*).

Las VLAN, al estar constituidas por grupos de computadores que pueden formar redes, aisladas de manera lógica, dentro de una red física, creando cada uno de estos grupos de computadores, un solo dominio de broadcast, constituyen un caso de virtualización de red basada en protocolo.

En el caso las VPN, se crean redes dedicadas conectando sitios remotos entre ellos, utilizando túneles privados y seguros sobre redes de comunicación compartidas o públicas (Internet) [23].

2.2.3. Virtualización basada en redes sobre puestas

Las redes sobrepuestas se caracterizan por estar construidas sobre una o más redes físicas, aprovechando la infraestructura ya instalada para proveer servicios de telecomunicaciones. Uno de los casos, son las comunicaciones *Peer to Peer* (*P2P*) siendo un ejemplo de uso de estas redes, el servicio Skype [23] [24].

2.2.4. Virtualización basada en redes activas y programables

Constituyen un tipo de redes en las cuales se requiere la creación sobre la marcha de nuevos servicios, en respuesta a los requisitos de los

usuarios, esto apoyándose en interfaces programables y códigos activos [23].

2.3. Virtualización de Funciones de Red

La virtualización de funciones de red (*NFV*, del inglés *Network Functions Virtualization*), es una tecnología asociada a la virtualización de redes. En ella, el objetivo es implementar funciones de red mediante software, de tal manera que sea posible su inicialización, instanciación o reubicación, sin necesidad de adquirir nuevo hardware [25].

Las funciones que se virtualizan en NFV, son las que se encuentran desde el nivel 4 hasta el nivel 7 del modelo de interconexión de sistemas abiertos (*OSI,* del inglés *Open Systems Interconnection*) por ejemplo, balanceadores de carga, firewalls, controladores de sesión de borde... [1].

La organización que encabeza las investigaciones relacionadas con esta tecnología es el Instituto Europeo de Estándares de Telecomunicaciones (*ETSI,* del inglés *European Telecommunications Standards Institute*).

2.3.1. Infraestructura NFV

La infraestructura NFV, está compuesta por los recursos de hardware y software que se encuentran presentes en el ambiente donde operan las funciones de red virtual (*VNF,* del inglés *Virtual Network Functions)*. También se incluyen las redes que conectan los diferentes sitios en los que pudiera existir Infraestructura NFV [26].

2.3.2. Funciones de Red Virtual y Servicios

Las funciones de red (*NF*, del inglés *Network Functions*), pueden ser definidas como elementos de la red que tienen un funcionamiento definido para realizar una tarea específica, dotados de interfaces externas.

Las NF tales como puertas de enlace, servidores del protocolo de configuración de host dinámico (*DHCP,* del inglés *Dynamic Host Configuration Protocol*), servidores de nombre de dominio (*DNS,* del inglés *Domain Name System*), corta fuegos… podrían ser implementadas

en máquinas virtuales, por lo cual pasarían a denominarse VNF. Los servicios virtuales están compuestos por una o varias VNF [1].

2.3.3. Modelo de Negocio en la NFV

En [1] se ha planteado un modelo de negocio donde se proponen cuatro roles, los cuales se muestran en la Figura 2.2.

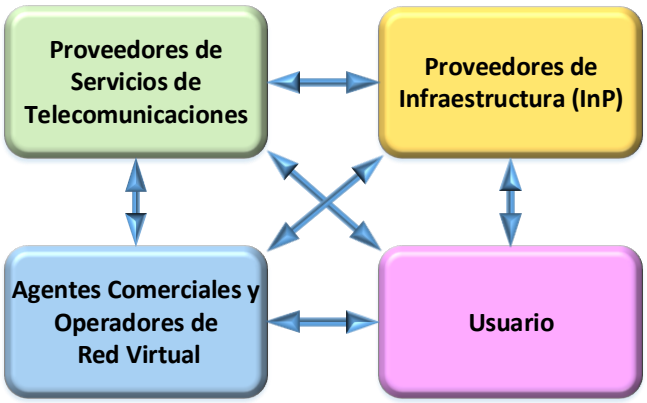


Figura 2.2: Modelo de Negocio de las NFV

- *Proveedor de Infraestructura* (*InP*, del inglés *Infrastructure Provider*): Son los propietarios de los recursos físicos tales como Data Centers y Redes Físicas.

- *Proveedor de Servicios de Telecomunicaciones* (*TSP*, del inglés *Telecommunications Service Provider*): arriendan recursos de uno o más InPs, con el fin de hacer funcionar a las VNF. También se encargan de establecer la secuencia de esas funciones para la entrega de servicios a los usuarios finales.

Existe también la figura de los Operadores de Red Virtuales Móviles (*MVNO*, del inglés *Mobile Virtual Network Operator*), quienes a su vez pueden sub arrendar recursos de los TSP.

- *Agente comercial* (*Broker*): se puede tener el caso, de que un solo servicio sea provisto por varios InP (ambiente multi dominio). En ese caso el Broker se encargaría de gestionar la entrega desde los diferentes InP al TSP. Un MVNO podría ser considerado un Broker que arrienda recursos de un TSP para proveer servicios a los usuarios [1].

- *Usuario Final*: consumen los servicios provistos por los TSP.

2.3.4. Consideraciones de Diseño

Antes de implementar VNF en una red, es necesario considerar ciertos aspectos a fin de garantizar un funcionamiento adecuado de los componentes de las VFN y por ende una entrega oportuna de los servicios de los TSP. Según [1] deben tenerse en cuenta los siguientes factores:

- *Arquitectura de Red y Desempeño*: existe una relación entre las arquitecturas en la virtualización de funciones de red y el desempeño que dichas Arquitecturas deben proveer, ya que este último debe ser similar al obtenido en las funciones implementadas en hardware.

Resulta evidente el hecho de que, si una función de red va a ser virtualizada, el desempeño que tenga dicha función, sea similar a la que se tendría en un hardware dedicado, de tal manera que se justifique la transición a NFV.

- *Seguridad y Resiliencia*: al igual que todo sistema tecnológico, es necesario que las VNF, cuenten con las seguridades necesarias a fin de evitar accesos no autorizados a la información que manejan, desde el exterior de la red o entre las funciones virtualizadas presentes en la red. Con esto se logrará añadir resiliencia a las funciones, ya que, si una de las funciones es atacada y deja de funcionar, las demás funciones podrán seguir trabajando normalmente.

- *Confiabilidad y Disponibilidad*: de igual manera, la entrega de los servicios provistos por las funciones dentro de la red, debe estar garantizada, dentro de los intervalos de tiempo acordes a los servicios que provee.

Por ejemplo, en la transmisión de voz, se requiere una alta disponibilidad y confiabilidad en que los paquetes de datos que se transmiten sean entregados en el tiempo y en la forma correcta, mientras que, en la transmisión de una página web, no es imprescindible este comportamiento.

- *Soporte para Heterogeneidad*: el término heterogeneidad se refiere a que sin importar la marca del hardware en que se implementen las funciones, estas puedan operar en el mismo entorno sin restricciones entre ellas, facilitando a los InP, la adquisición de nuevo hardware.

- *Soporte legado*: es de gran importancia que exista una compatibilidad entre las funciones anteriores implementadas en hardware y las nuevas funciones que se virtualizan. De esta manera se puede ahorrar tiempo a los InP en la transición hacia NFV, ya que no se tendrían que realizar mayores cambios en la infraestructura.

- *Escalabilidad de la Red y Automatización*: la posibilidad de expansión de la red debido al crecimiento de usuarios y por ende la posibilidad de un aumento en la demanda de los servicios, debe estar cubierta cuando se realiza el diseño para el despliegue de las VNF.

Debe también considerarse la opción de que estos cambios sean realizados de manera automática a fin de disminuir los tiempos de implementación de las nuevas VNF requeridas.

2.4. Redes Definidas por Software

Otro concepto asociado a la virtualización de redes inalámbricas es el de SDN.

2.4.1. Definición

De acuerdo a [8]: *un conjunto de técnicas que hacen posible directamente programar*, *orquestar*, *controlar y administrar recursos*, *lo cual facilita el diseño*, *entrega y operación de servicios de red de una manera dinámica y escalable*.

La Fundación de Redes Abiertas [7] (*ONF,* del inglés *Open Networking Foundation*), es la organización que lidera las investigaciones en torno a esta tecnología.

La ONF está conducida por una directiva compuesta por los directores de 7 compañías que operan y a las cuales pertenecen algunas de las

mayores redes en el mundo: Deutshe Telekom, Facebook, Google, Microsoft, Verizon, Yahoo y NTT [7].

Sin embargo, la ITU, ha estandarizado las SDN a través de la recomendación Y.3300: Marco de Referencia de las Redes Definidas por Software [8], la cual describe los fundamentos de las SDN.

Las SDN son también definidas como una arquitectura de red, en la cual, se divide a la red en un plano de control y un plano de datos, lo que le otorga a la red la capacidad de poderse expandir utilizando un control centralizado, sin tener que realizar configuraciones en cada uno de los equipos como sería en el caso de una red de datos tradicional.

2.4.2. Objetivos

Entre los objetivos que persiguen las SDN, se han identificado los siguientes:

- Facilitar la administración de los diferentes dispositivos de red, independientemente del vendedor, desde una ubicación central.

- Conseguir mediante la utilización de API, una mejora en la automatización y administración de la red.

- Lograr que modificaciones en la red relacionadas a capacidades y servicios puedan ser efectuadas de manera rápida.

- Permitir una programación de dispositivos de red mediante código abierto, permitiendo la interoperabilidad entre dispositivos de distintas marcas, sin restricciones de sistemas operativos y demás software propietario.

- Obtener de manera centralizada, información sobre el estado de la red.

2.4.3. Arquitectura SDN

La arquitectura de red básica de las SDN, según [7] [10] [27], se muestra en la Figura 2.3, y consta de los siguientes componentes:

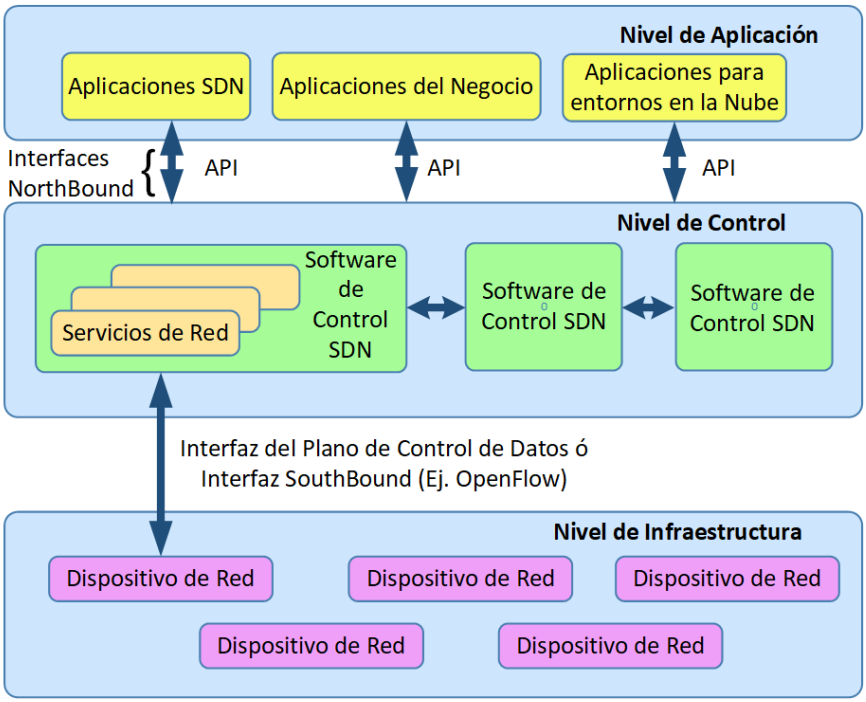


Figura 2.3: Arquitectura de SDN

a) *Nivel de Control:* el nivel de control/controlador presenta una visión abstracta de la infraestructura de red completa, otorgando al administrador la capacidad de aplicar políticas/protocolos personalizados a través del hardware de red. El controlador de sistema operativo de red (NOX), es el controlador más ampliamente utilizado.

En este nivel se encuentra una entidad llamada *controlador*. Esta entidad encapsula la lógica y es responsable de proveer una interfaz programable a la red, la cual es usada para implementar nueva funcionalidad y desempeñar varias tareas de administración.

SDN ha introducido la noción de abstracción del sistema operativo de red. Un sistema operativo de red ofrece una abstracción más general del estado de la red en los conmutadores, revelando una interfaz simplificada para controlar la red. Esta abstracción asume un modelo de control centralizado lógicamente, en el cual las aplicaciones ven a la red como un solo sistema. En otras palabras, el sistema operativo de red actúa como un nivel intermedio responsable de mantener una vista consistente del estado de la red, el cual es explotado mediante control lógico para proveer

varios servicios de red para descubrimiento topológico, encaminamiento, administración de movilidad, estadísticas...

b) *Nivel de Aplicación:* en la cima de la arquitectura de SDN está el nivel de aplicación, el cual incluye todas las aplicaciones que explotan los servicios provistos por el controlador para desempeñar tareas relacionadas con la red, como balanceo de carga, *virtualización de red*... Una de las características más importantes de SDN es la apertura que provee a terceros desarrolladores a través de la abstracción que define para el fácil desarrollo y despliegue de nuevas aplicaciones en varios entornos de red desde centros de datos hasta redes inalámbricas y celulares. La arquitectura SDN elimina la necesidad de cajas intermedias como cortafuegos, Sistemas de Detección de Intrusión (*IDS,* del inglés *Intrusion Detection System*) en la topología de la red, lo cual es posible ahora por su funcionalidad para ser implementada en la forma de aplicaciones de software que monitorean y modifican el estado de la red a través de los servicios del sistema operativo de red. Obviamente, la existencia de este nivel añade gran valor a las SDN, puesto que da lugar a un amplio rango de oportunidades para innovación, haciendo a las SDN una solución convincente tanto para investigadores como para la industria.

c) *Nivel de Infraestructura*: el plano de datos representa el hardware de reenvío en la arquitectura de red SDN.

En el nivel del fondo se puede observar el plano de datos, donde se encuentra la infraestructura de red (conmutadores, encaminadores, PA…). En el contexto de SDN estos dispositivos han sido despojados de toda lógica de control (por ejemplo, algoritmos de encaminamiento como BGP) simplemente implementando un conjunto de operaciones de reenvío manipulando paquetes y flujos, proporcionando una interfaz abierta abstracta para la comunicación con los niveles superiores. En la terminología SDN estos dispositivos se conocen comúnmente como conmutadores de red [27].

d) *Interfaces de Aplicación Norte:* las Interfaces de Programación de Aplicaciones (*API,* del inglés *Application Programming Interfaces*) Norte, representan las interfaces de software entre los módulos de software de la plataforma controladora y las aplicaciones SDN corriendo sobre la plataforma de red. Estas API exponen modelos de datos de abstracción de red universales y funcionalidad para ser usadas por aplicaciones de red. Las “API Norte” son basadas en código libre.

e) *Interfaces de Aplicación Sur:* en vista de que el controlador necesita comunicarse con la infraestructura de red, esto requiere ciertos protocolos para controlar y administrar la interfaz entre varias piezas de equipos de red. El más popular “protocolo Sur” es el protocolo OpenFlow.

f) Protocolos Este-Oeste: en el caso de arquitecturas basadas en múltiples controladores el Protocolo de Interfaz Este-Oeste, administra la interacción entre los varios controladores.

Además, para abstraer la red, la arquitectura SDN soporta un conjunto de API que hacen posible implementar servicios de red común incluyendo encaminamiento, multicast, seguridad, control de acceso, administración de ancho de banda, ingeniería de tráfico, QoS, optimización de procesamiento y almacenamiento, uso de energía, y todos los formatos de administración de políticas personalizados para cumplir con los objetivos del negocio.

2.4.4. Protocolo OpenFlow

El protocolo OpenFlow es una interfaz de comunicaciones estándar, que opera entre los niveles de control y reenvío en la arquitectura SDN [7]. OpenFlow permite el acceso directo y la manipulación del plano de reenvío de los dispositivos de red tales como conmutadores y encaminadores tanto físicos como virtuales.

La ausencia de una interfaz abierta para el plano de reenvío que ha liderado la caracterización de los dispositivos de red de hoy como monolíticos, cerrados y tipo mainframe.

Ningún otro protocolo standard hace lo que OpenFlow hace, y un protocolo como OpenFlow es necesario para mover el control de la red fuera de los conmutadores de red hacia un software de control centralizado lógicamente [7].

El protocolo OpenFlow está implementado en ambos lados de la interfaz entre los dispositivos de la infraestructura de red (Nivel de Infraestructura) y el software de control SDN (Nivel de Control). OpenFlow utiliza el concepto de flujos para identificar el tráfico de la red basado en reglas de emparejamiento predefinidas que pueden ser programadas estáticamente o dinámicamente por el software de control SDN.

Un elemento importante en la arquitectura SDN para lograr la separación del plano de control del plano de datos, es el *Conmutador OpenFlow*.

El *conmutador OpenFlow*, cuyo diagrama de bloques básico se muestra en la Figura 2.4, está constituido por una o más *tablas de flujo* y una *tabla de grupo*, las cuales realizan búsqueda de paquetes y reenvío, y un *canal OpenFlow* hacia un controlador externo. El conmutador se comunica con el controlador y el controlador administra el conmutador por medio del protocolo OpenFlow [28].

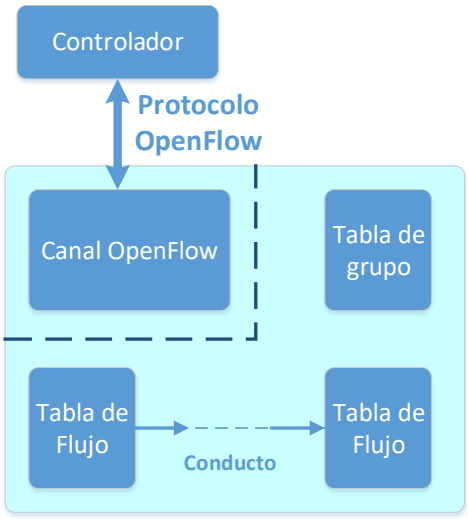


Figura 2.4: Componentes del Conmutador OpenFlow

El **canal OpenFlow**, es la interfaz que conecta cada conmutador OpenFlow a un controlador. A través de esta interfaz, el controlador

configura y administra el conmutador, recibe eventos desde el conmutador y envía paquetes fuera del conmutador [28].

El protocolo OpenFlow utiliza tres tipos de mensaje: *controlador a conmutador*, *asíncrono y simétrico*.

a) *Mensaje Controlador a Conmutador*: estos mensajes los inicia el controlador y los usa para administrar directamente o inspeccionar el estado del conmutador. Los mensajes, podrían o no requerir una respuesta del conmutador. Los mensajes Controlador a conmutador son los siguientes:

- *Características*: el controlador podría requerir la identidad y las capacidades básicas de un conmutador enviando un requerimiento de "Características"; el conmutador debe responder con una réplica que especifique la identidad y capacidades básicas del conmutador. Esto se realiza comúnmente al establecer el canal llamado OpenFlow.

- *Configuración*: mediante este mensaje, el controlador es capaz de establecer y consultar los parámetros de configuración en el conmutador. El conmutador sólo responde a una consulta desde el controlador.

- *Modificar Estado*: el controlador envía los mensajes de "Modificación de Estado" para administrar el estado en los conmutadores. Su propósito primario es añadir, borrar y modificar entradas de flujo/grupo en las tablas OpenFlow y establecer las propiedades de los puertos del conmutador.

- *Leer Estado*: el controlador usa los mensajes "Leer Estado" para recolectar información variada desde el conmutador, tales como configuración actual, estadísticas y capacidades.

- *Salida de paquete*: el controlador usa estos mensajes para enviar paquetes fuera de un puerto especificado en el conmutador, y para reenviar paquetes recibidos vía mensajes de "Ingreso de Paquetes".

Los mensajes de "Salida de Paquetes" deben contener un paquete lleno o un identificador de buffer (buffer ID) referenciando un paquete en el conmutador. El mensaje también debe contener una lista de acciones

para ser aplicadas en el orden en que ellas son especificadas; una lista de acciones vacía, descarta el paquete.

- *Barrera*: los mensajes de petición/respuesta de “Barrera” son utilizados por el controlador para asegurar que las dependencias de mensaje han sido reunidas o para recibir notificaciones de operaciones completadas.

- *Petición de Rol*: los mensajes de “Petición de Rol”, son usados por el controlador para establecer el rol de su canal OpenFlow o consultar ese rol. Esto es principalmente útil cuando el conmutador se conecta a múltiples controladores.

- *Configuración Asíncrona*: los mensajes de “Configuración Asíncrona”, son usados por el controlador para establecer un filtro adicional sobe los mensajes asíncronos que quiere recibir en su canal OpenFlow, o consultar ese filtro. Esto es mayormente usado cuando el conmutador se conecta a múltiples controladores y se realiza comúnmente al establecer el canal OpenFlow.

b) *Mensajes Asíncronos*: son iniciados por el conmutador y usados para notificar al controlador sobre eventos en la red red, cambios en el estado del conmutador y llegada de paquetes. Los mensajes asíncronos son enviados desde un conmutador sin que el controlador los solicite. Los tres tipos principales de mensajes asíncronos son:

- *Ingreso de paquete*: transfiere el control de un paquete al controlador. Para todos los paquetes reenviados al puerto reservado del controlador utilizando una entrada de flujo o la entrada en la tabla de flujo faltante, siempre es enviado un evento de “Ingreso de paquete” a los controladores. Otros procesamientos, tales como chequeo de *Time To Live* (*TTL*), también podrían generar eventos “Ingreso de paquete” para enviar paquetes al controlador.

- *Flujo Removido*: informa al controlador acera de la remoción de una entrada de flujo de una tabla de flujo. Estos mensajes son generados como resultado de una solicitud de eliminación de flujo del controlador o

el proceso de caducidad de flujo del conmutador cuando se excede uno de los tiempos muertos de flujo.

- *Estado del puerto*: informa al controlador de un cambio sobre un puerto. Se espera que el conmutador envíe mensajes “Estado del puerto” a los controladores como “Configuración” de puerto o cambios de “Estado de puerto”. Esos eventos incluyen cambios en eventos de configuración de puerto, por ejemplo, si este es dado de baja directamente por el usuario, y eventos de cambio en el estado del puerto, por ejemplo, si el enlace es dado de baja.

c)*Mensajes Simétricos*: son iniciados por el conmutador o por el controlador y son enviados sin ser solicitados. Los mensajes simétricos son los siguientes:

- *Hola*: los mensajes “Hola” son intercambiados entre el conmutador y el controlador al inicio de la conexión.

- *Eco*: la petición/réplica de mensajes “Eco” puede ser enviada tanto desde el conmutador como del controlador, y debe retornar una réplica de “Eco”. Ellos son usados principalmente para verificar la operatividad de una conexión controlador- conmutador, y podría también ser usada para medir su latencia o ancho de banda.

- *Error*: los mensajes de “Error” son usados por el conmutador o el controlador para notificar al otro lado de la conexión sobre problemas. Ellos son usados mayoritariamente por el conmutador para indicar una falla de un pedido iniciado por el controlador.

- *Experimentador*: los mensajes “Experimentador” proporcionan una forma estándar para conmutadores OpenFlow para ofrecer funcionalidades adicionales, dentro del espacio de tipo de mensajes OpenFlow. Este es un escenario promedio para futuras revisiones de OpenFlow.

2.5. Conceptos sobre Telefonía IP

Se revisan a continuación varios conceptos que son de mucha importancia a la hora de entender el funcionamiento de la Telefonía IP.

2.5.1. CODEC, Paquetizador y Buffer de Reproducción

Los tres componentes principales de la VoIP son: El Codificador/Decodificador (CODEC, del inglés COder/DECoder), el paquetizador y el búfer de reproducción [29], y se muestran en la Figura 2.5.

El emisor convierte muestras acondicionadas de señales de voz en señales digitales, las comprime y luego las codifica dentro de un formato predeterminado usando un CODEC de voz. Existen varios códecs desarrollados y estandarizados por la ITU-T tales como G.711, G.729, GSM...

En el siguiente proceso, se lleva a cabo la paquetización, el cual fragmenta la voz codificada en paquetes de igual tamaño. Además, en cada paquete, algunas cabeceras de protocolo de diferentes niveles son añadidas a la voz codificada. Las cabeceras de protocolo añadidas a los paquetes de voz pertenecen al Protocolo de Transporte en Tiempo Real (*RTP,* del inglés *Real-Time Protocol*), Protocolo Datagrama de Usuario (*UDP,* del inglés *User Datagram Protocol*), y al protocolo IP, además de un encabezado del nivel de enlace de datos.

Adicionalmente, RTP y el Protocolo de Control en Tiempo Real (*RTCP,* del inglés *Real-Time Control Protocol*) fueron diseñados en el nivel de aplicación para soportar aplicaciones en tiempo real. Aunque en la Internet, el protocolo usado comúnmente es el de control de transporte (*TCP,* del inglés *Transport Control Protocol*), en VoIP se prefiere utilizar el protocolo UDP, así como también en otras aplicaciones en tiempo real sensibles a los retrasos. El protocolo TCP es adecuado para paquetes de datos menos sensibles a los retrasos y no para paquetes de datos sensibles a los retrasos debido al esquema de acuse de recibo (*ACK,* del inglés *Acknowledgement*) que aplica TCP. Este esquema introduce retrasos ya que como receptor tiene que notificar al remitente por cada paquete recibido enviando un ACK. Por otro lado, UDP no aplica este esquema por lo que, es más adecuado para aplicaciones VoIP.

Los paquetes son enviados fuera sobre la red IP a su destino donde es llevado a cabo el proceso de reversa de decodificación y despaquetización de los paquetes recibidos. Durante el proceso de transmisión, podrían ocurrir variaciones de tiempo (jitter) en la entrega de paquetes. Aquí, un buffer de reproducción es utilizado en el receptor final para suavizar la reproducción mediante la mitigación del jitter introducido. Los paquetes son encolados en el buffer de reproducción por un tiempo de reproducción antes de ser reproducidos. Sin embargo, los paquetes que lleguen posterior al tiempo de reproducción son descartados.

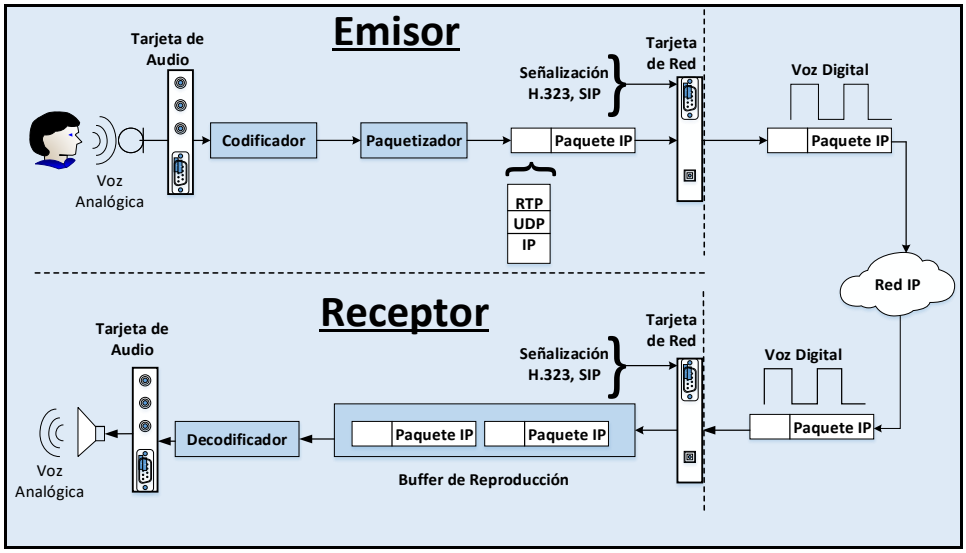


Figura 2.5: Componentes de la VoIP

2.5.2. Real Time Protocol

El protocolo en tiempo real (*RTP*, del inglés *Real-Time Protocol*) (RFC1889) entrega datos en tiempo real entre dos sistemas finales. Esto es hecho en tal forma que el sistema de recepción final es capaz de reconstruir la trama de datos enviada por el otro sistema final, incluso si los paquetes sufren un retraso o llegan desordenados. Si los paquetes se pierden en el camino, el protocolo es capaz de detectarlo, pero no soporta pedidos de retransmisiones [30].

La razón para no soportar retransmisión en el protocolo es que probablemente tomaría demasiado tiempo (al menos un RTT --del inglés

Round Trip Time--, el cual podría ser de varios cientos de mili segundos) para pedir que la fuente reenvíe los paquetes RTP perdidos y para que llegue esta copia. Una mejor solución, para el caso de audio al menos, es extrapolar el sonido desde muestra de audio previas para compensar las pérdidas. Otro algoritmo para resolver el problema debido a pérdida de paquetes es solo ignorar los datos perdidos y continuar como si nada hubiese sucedido. Simplemente trabajar ignorando los paquetes perdidos, ya que la duración del audio en un paquete es relativamente corta, en el rango de los 20 ms a cerca de 60 ms. La pérdida de sonido para ese corto período de tiemplo no tendrá una mayor influencia en la calidad. Es probable que ni siquiera se note en absoluto [30].

El tema de retransmisión es una razón mayor para no usar el protocolo TCP (RFC793). TCP, el cual es un protocolo orientado a conexión confiable, usa retransmisión como una forma para garantizar el reparto de los datos entregados al nivel TCP desde el nivel de aplicación [30].

En lugar de TCP, RTP normalmente usa UDP (RFC768) como el protocolo de transmisión por defecto. UDP no provee ninguna característica de confiabilidad. UDP en cambio usa IP, con entrega de menor esfuerzo para encapsular la data. Cualquier protocolo de nivel de aplicación que dependa sobre UDP para transmisión y todavía tenga el deseo de asegurar que cualquier dato enviado es también recibido, debe implementar su propio algoritmo de retransmisión [30].

El número de secuencia en la cabecera RTP, se muestra en la Figura 2.6, el cual se usa para detectar pérdida y desorden en los paquetes [29].

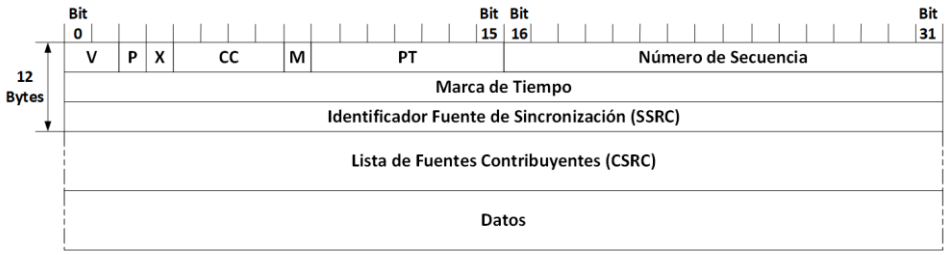


Figura 2.6: Cabecera del protocolo RTP

Los primeros doce octetos en una cabecera RTP (las primeras tres filas en la Figura 2.6) están incluidos en todos los paquetes RTP, mientras que la lista de identificadores de Fuentes Contribuyendo (*CSRC*, del inglés *Contributing Source*) están presentes cuando son insertadas por un *mezclador* [31]. A continuación, se describe la composición de un paquete RTP:

- *Versión* (*V*, *2 bits*): este campo identifica la versión de RTP. La versión actual, definida en RFC 1889, es la dos.

- *Relleno* (*P*, *1 bit*): si el bit de relleno está habilitado, los paquetes contienen uno o más octetos de relleno en el final de la carga útil. El último octeto de la carga útil contiene el número de octetos de relleno.

- *Extensión* (*X*, *1 bit*): si el bit extensión está habilitado, son permitidos el encabezado RTP fijo y posibles CSRCs por extensiones que usan el formato definido en RFC 1889.

- *Contador CSRC* (*CC*, *4 bits*): el contador CSRC contiene el número de identificadores de Fuentes Contribuyendo que siguen al encabezado fijo. Por lo general, este número toma el valor de cero.

- *Marcador* (*M*, *1bit*): la interpretación del bit marcador está definida por el perfil RTP. El bit marcador está destinado para marcar eventos significantes, tales como límite de trama, en el flujo de paquetes.

- *Tipo de Carga Útil* (*PT*, *7 bits*): este campo identifica el formato de la carga útil RTP, y determina su interpretación por la aplicación. Un perfil especifica el mapeo estático por defecto de los códigos de tipo de carga útil para formatos de carga útil.

- *Número de Secuencia* (*16 bits*): el número de secuencia empieza desde un valor aleatorio y es incrementado en uno por cada paquete RTP enviado. El número de secuencia es usado por el receptor para detectar pérdidas de paquetes y para reiniciar la secuencia de paquete.

- *Marca de Tiempo* (*32 bits*): la Marca de Tiempo refleja el instante de muestreo del primer octeto de la carga útil. La frecuencia del reloj está

definida por cada tipo de carga útil, y el reloj está inicializado con un valor aleatorio.

- *SSRC* (*32 bits*): el campo SSRC identifica la fuente de sincronización. Este identificador es escogido aleatoriamente, con la intención de que no existan dos fuentes de sincronización dentro de la misma sesión RTP con el mismo identificador SSRC.

- *Lista CSRC* (*0 a 15 ítems*, *32 bits cada uno*): la lista CSRC identifica las fuentes contribuyentes para la carga útil contenida en este paquete. El número de identificadores está dado por el campo CC. Solo 15 fuentes pueden ser identificadas. Los identificadores CSRC son insertados por mezcladores, utilizando identificadores SSRC de las fuentes contribuyentes.

Ahora resumimos el proceso de encapsulamiento, del audio para una sesión antes de que este sea enviado desde un host, el cual se describe en la Figura 2.7.

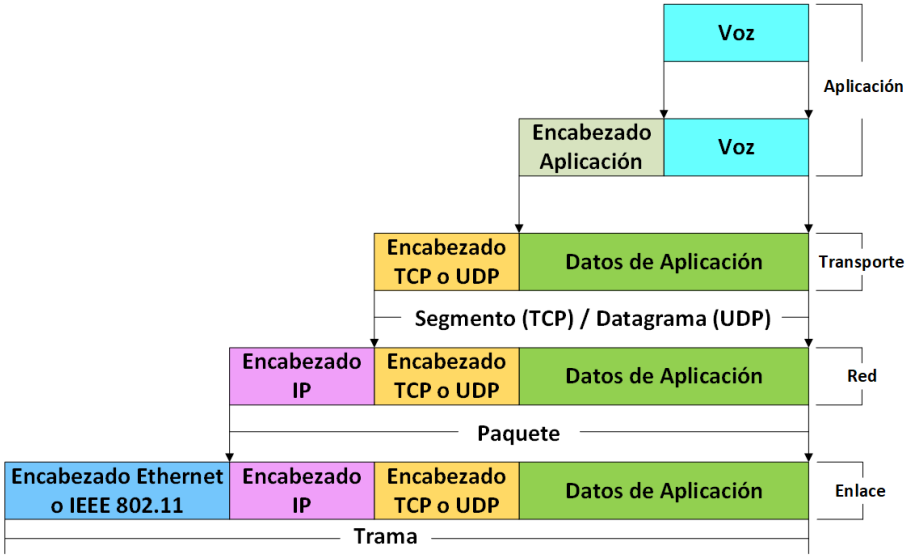


Figura 2.7: Encapsulamiento del audio

1. El sonido desde el micrófono es muestreado en determinados momentos. Cierta cantidad de muestras son empaquetadas juntas por la aplicación para convertirse en el dato encapsulado en un paquete RTP. Típicamente 20 ms de sonido es encapsulado dentro de un paquete RTP.

2. En el nivel de transportación, el paquete RTP es encapsulado dentro de un datagrama UDP.

3. Sobre el nivel de red el datagrama UDP es encapsulado dentro de un paquete IP. Para referencia los encabezados TCP y UDP son mostrados en la Figura 2.8 y Figura 2.9 respectivamente.

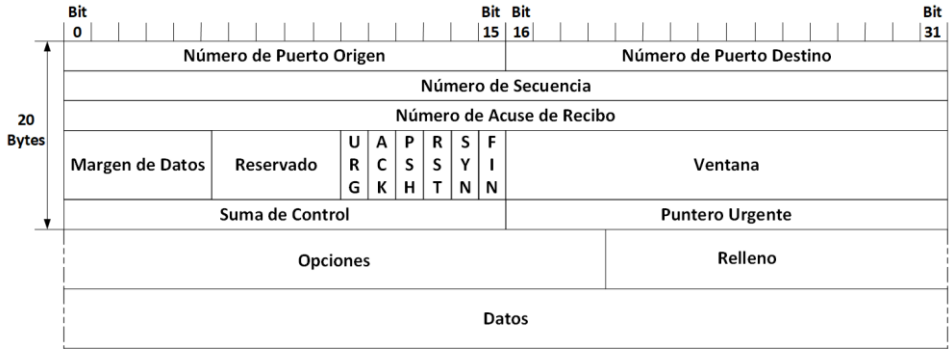


Figura 2.8: Encabezado TCP

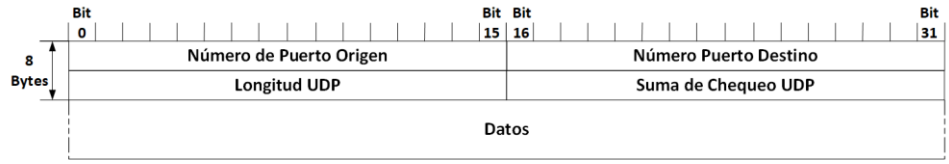


Figura 2.9: Encabezado UDP

4. El paquete IP, cuyo encabezado se muestra en la Figura 2.10, es encapsulado dentro de una trama IEEE 802.11, mostrada en la Figura 2.11 y luego la trama es enviada por la red, a través del medio inalámbrico.

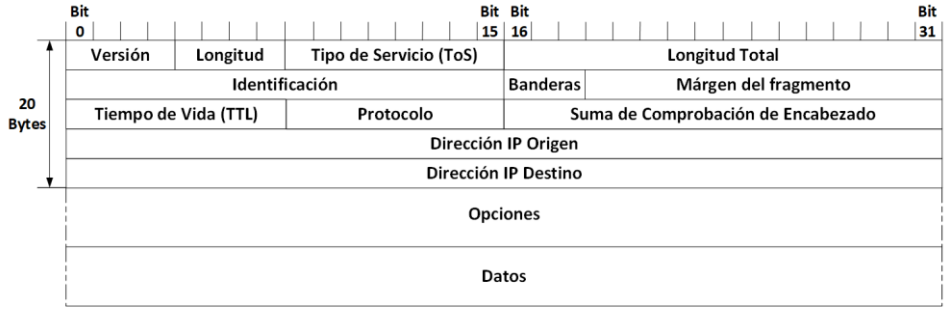


Figura 2.10: Encabezado IPv4

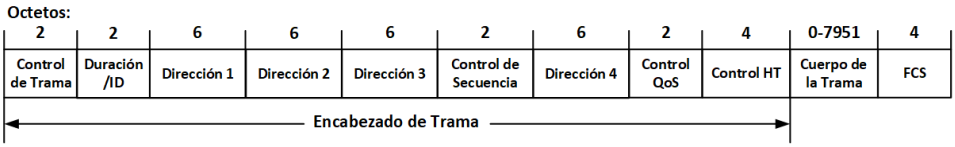


Figura 2.11: Formato de trama MAC del estándar IEEE 802.11n

RTP tiene su propio protocolo de control, el *Protocolo de Control en Tiempo Real* (*RTCP*) (RFC1889). El propósito principal para RTCP es dar retroalimentación sobre la calidad de la entrega de datos desde el receptor de los datos, a quienes los envían. La retroalimentación es dada en reportes enviados con RTCP y pueden por ejemplo contener el número de paquetes perdidos en la sesión e información acerca de retrasos en las redes intermedias. RTCP no es transmitido continuamente en la misma forma como RTP, en su lugar se envía periódicamente con un período típico de unos pocos segundos entre los reportes [30].

Cualquier transmisión multimedia RTP es siempre asignada a un número de puerto par, mientras los reportes RTCP asociados son enviados sobre el siguiente puerto mayor, por lo tanto, un número de puerto impar [30].

RTCP está basado en la transmisión periódica de paquetes de control para todos los participantes en la sesión. El protocolo subyacente debe proveer multiplexación de datos y control de paquetes, por ejemplo, usando números de puerto separado con UDP. RTP es usualmente asignado a un puerto UD par, y RTCP al siguiente puerto UDP impar [31].

RTCP desarrolla tres funciones de implementación obligatoria [31]:

1. La función primaria es proveer retroalimentación sobre la calidad de la distribución de datos. Esta función es desempeñada a través de reportes del remitente y del receptor.

2. RTCP lleva un identificador de nivel de transporte persistente para una fuente RTP, llamada Nombre Canónico (*CNAME,* del inglés *Canonical Name*). Ya que el identificador SSRC podría cambiar, rodos los receptores requieren el CNAME para mantener el control de cada participante.

3. Ya que las dos primeras funciones requieren que todos los participantes en una sesión envíen paquetes RTCP, la tasa de paquetes RTCP debe ser controlada para escalar hasta un gran número de participantes. Cada participante puede observar independientemente el número de otros participantes y así controlar la tasa de paquetes RCTP. La máxima tasa a la cual un participante puede enviar reportes RTCP es uno cada cinco segundos.

Existen varios tipos de paquete RTCP para llevar información de control:

- *SR (Emisor de Reporte)*: contiene estadísticas de transmisión y recepción para emisores activos.
- *RR (Reporte de Receptor)*: contiene estadísticas de recepción para los participantes que no son emisores activos.
- *SDES (Items de Descripción de Fuente)*: describe varios parámetros sobre la fuente, incluyendo el CNAME.
- *BYE*: este paquete es enviado por un participante cuando deja la sesión.
- *APP*: funciones específicas de aplicación [31].

2.5.3. Session Initiation Protocol

Se define como *Sesión* a un conjunto de tramas de datos llevando múltiples tipos de contenido entre emisor y receptor. Una sesión puede ser una llamada telefónica, una video conferencia, un usuario tomando control remoto de una computadora, o dos usuarios compartiendo datos, charlando o intercambiando mensajes instantáneos [19].

El *Protocolo de Inicio de Sesión* (*SIP*, del inglés *Session Initiation Protocol*), inicia, modifica y termina sesiones interactivas, basado en texto codificado sobre elementos desde el Protocolo de Transporte de Hiper Texto (*HTTP,* del inglés *HyperText Transport Protocol*), el cual es usado por navegadores web, y también por el Protocolo de Transporte de Correo Simple (*SMTP,* del inglés *Simple Mail Transport Protocol*), el cual es usado para correo sobre la Internet [33].

Como su nombre lo indica, la función primaria de SIP es la iniciación (establecimiento) de la sesión, pero también tiene otros usos y funciones importantes, tales como notificación de presencia y mensajes cortos. SIP es usado para comunicaciones par a par (que son aquellas en las cuales ambas partes en la llamada son consideradas iguales, no existe maestro ni esclavo). Sin embargo, SIP usa un modelo de transacción cliente-servidor similar a HTTP [33].

Direccionamiento en SIP

El estándar de direccionamiento en SIP es similar al utilizado en correos electrónicos, tomando una de las siguientes formas (el puerto es opcional y si no se lo especifíca, se usa el puerto 5060) [34]:

sip:*usuario@dominio*:*puerto*

sip:*usuario@host*:*puerto*

sip:*número telefónico@dominio*

Otro tipo de formato utilizado en direccionamiento es el Identificador de Recursos Uniforme (*URI,* del inglés *Uniform Resource Identifier*). El URI es comúnmente la dirección IP o el nombre de dominio totalmente calificado (*FQDN,* del inglés *Full Qualyfied Domain Name*) del host [34]. Dentro de SIP, el URI podría empezar con la palabra *sips*, indicando un URI SIP seguro, el cual usará el puerto 5061. El tipo de direccionamiento depende de la topología de la red y los servicios desplegados.

RFC 3261 recomienda el uso de nombres de dominio totalmente calificados (FQDN) para direccionamiento, por lo tanto, las implementaciones SIP son comúnmente integradas con los sistemas de nombre de dominio o DNS.

Cada URI contiene la dirección SIP del registro (*AOR,* del inglés *Addres Of Record*), el cual es a veces es visto como la dirección pública para un usuario. El AOR apunta al dominio para el usuario. Dentro de ese dominio, debe haber un servicio que pueda mapear el URI a un URI de la ubicación

actual del usuario. Otra forma de verlo es como la dirección pública del usuario. En otras palabras, es como alguien te contactaría.

Componentes (***Arquitectura***) ***en SIP***

Existen tres elementos principales en una red SIP: agentes usuarios, servidores y servicios de ubicación.

- *Agentes usuarios: los Agentes Usuarios* (*UA*, del inglés *User Agent*) *SIP*, son los dispositivos finales en una red SIP. Ellos originan los requerimientos para establecer las sesiones multimedia y enviar y recibir contenido multimedia. Un agente usuario puede ser un teléfono SIP o un software cliente SIP corriendo en un computador. De manera alternativa, un agente usuario puede ser una puerta de enlace a otra red, tal como una pasarela PSTN, la cual permite a un teléfono SIP recibir y hacer llamadas a la PSTN.

Un *Agente Usuario Cliente* (*UAC*, del inglés *User Agent Client*) es la parte del agente usuario que inicia el requerimiento, mientras el *Agente Usuario Servidor* (*UAS*, del inglés *User Agent Server*) es la parte del agente usuario que genera respuestas a los requerimientos recibidos. Cada agente usuario SIP contiene tanto un UAC y un UAS. Durante el curso de una sesión, ambas partes son típicamente usadas. Esta característica es diferente de la mayoría de arquitecturas cliente servidor, tales como la navegación web. Durante una sesión de navegación web, un computador siempre e el cliente HTTP (software de navegación web), y el servidor siempre en un servidor HTTP.

Se asume que los UA SIP son inteligentes, en el sentido de ser parte de un host de Internet totalmente calificado como se define en RFC 1121 y RFC 1122, y soporta muchos otros protocolos incluyendo DHCP, DNS, IMCP...

- *Servidores*: *los servidores* son dispositivo intermediarios que están localizados dentro de la red SIP y asisten a los UA en el establecimiento

de la sesión y otras funciones. Existen tres tipos de servidores SIP definidos en RFC 3261:

- *Proxy SIP*: recibe los requerimientos SIP de un UA o de otro Proxy y lo reenvía o lo asigna a otra ubicación.

- *Servidor Redirect*: recibe un requerimiento de agente usuario o proxy y retorna una respuesta de redirección (3XX), indicando donde el requerimiento debe ser re procesado.

- *Servidor Registrar*: recibe requerimientos de registro SIP y actualiza la información del UA en un servicio de ubicación u otra base de datos.

Los servidores SIP Proxy, Redirect y Registrar son elementos puramente para retransmitir señalización. Ellos no tienen capacidades multimedia y no inician requerimientos excepto en nombre de un UA.

- *Servicios de Ubicación*: un *servicio de ubicación* es un término general usado en la RFC 2543 para una base de datos. La base de datos podría contener información de los usuarios tales como URIs, direcciones IP, scripts, características, y otras preferencias. También podría contener información y encaminamiento sobre la red SIP, incluyendo la ubicación de proxys, puertas de enlace y otros servicios de ubicación. Los UA generalmente no interactúan directamente con un servicio de ubicación, pero atraviesan un servidor proxy, redirect o registrar. Los servidores SIP usan un protocolo que no es SIP para consultar, actualizar y recuperar registros del servicio de ubicación en el curso de encaminamiento de un mensaje SIP.

Mensajes SIP

El formato para los *mensajes SIP*, por lo general siguen el formato estándar para Mensajes de Internet (RFC5322) [35] y consisten de una *línea de inicio*, *cabeceras de mensaje*, y luego una línea vacía con un retorno de carro (CR) y avance de línea (LF). Las líneas de inicio son de dos tipos: para mensajes de requerimiento se denominan *líneas de*

requerimiento y para mensajes de respuesta se las denomina *líneas de estado* [34] o también llamadas de *respuesta* según [33].

- *Líneas de Requerimientos*: en un mensaje de requerimiento SIP, la línea de inicio es en realidad llamada una línea de requerimiento. La línea de requerimiento contiene el URI SIP y la versión del protocolo y termina con los caracteres CR y LF [34].

SIP cuenta con varios tipos de requerimientos, también llamados *métodos*, mostrándose a continuación los principales [33]:

- *INVITE*: establecimiento de la sesión.
- *ACK*: reconocimiento de respuesta final para INVITE.
- *BYE*: terminación de Sesión.
- *CANCEL*: cancelación de Sesión pendiente.
- *REGISTER*: registro del URI de un usuario.
- *OPCIONES*: consulta de opciones y capacidades.
- *INFO*: transporte de señalización de llamada intermedia.
- *PRACK*: reconocimiento de respuesta provisional.
- *UPDATE*: actualizar Información de la Sesión.
- *REFER*: transferir usuario a un URI.
- *SUBSCRIBE*: pedir notificación de un evento.
- *NOTIFY*: transporte de notificación de evento suscrito.
- *MESSAGE*: transporte de un cuerpo de mensaje instantáneo.
- *PUBLISH*: subir estado de presencia a un servidor.

En la Figura 2.12 se muestra el establecimiento de una sesión SIP utilizando el método INVITE. Mientras que en la Figura 2.13 se muestra una línea de requerimiento SIP (INVITE), en formato hexadecimal.

Figura 2.12: Establecimiento de Sesión SIP usando INVITE

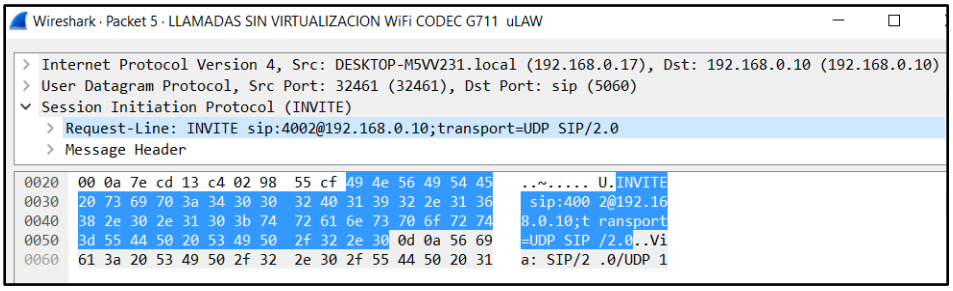


Figura 2.13: Línea de Requerimiento SIP en formato hexadecimal

- *Líneas de Respuestas*: las respuestas SIP, como la mostrada en la Figura 2.14, tienen una línea de estado para el inicio y un código de estado para la transacción [34].

En SIP, las respuestas son numéricas. Muchos códigos de repuesta han sido prestados de HTTP. Los códigos de respuesta SIP son divididos en seis clases, identificados por el primer dígito del código, como se detalla a continuación [33]:

- *1XX*: Provisional o Informacional – El requerimiento está siendo procesado, pero no se ha completado todavía.

- *2XX*: Satisfactorio – El requerimiento ha sido completado satisfactoriamente.

• *3XX*: Redirección – El requerimiento debería ser tratado en otra ubicación.

• *4XX*: Error de Cliente – El requerimiento no fue completado a causa de un error en el requerimiento, puede ser reprocesado en otra ubicación.

• *5XX*: Error de Servidor – El requerimiento no fue completado a causa de un error en el receptor, puede ser reprocesado en otra ubicación.

• *6XX*: Falla global – El requerimiento ha fallado y no debería ser reprocesado de nuevo.

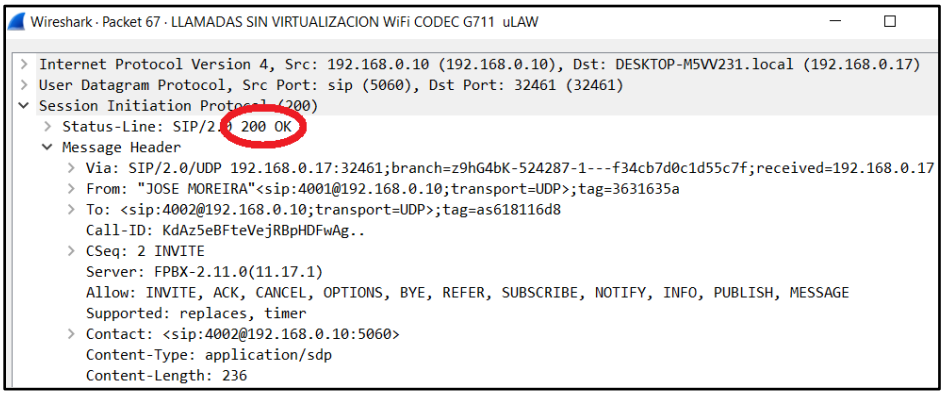


Figura 2.14: Respuesta SIP a una INVITACION

Los requerimientos SIP deben incluir como mínimo, los siguientes campos en el encabezado: Via, From, To, Call-ID, CSeq y Max-Forwards. Ellos contienen la información necesaria para encaminamiento, identificación y ordenamiento.

Campos del Encabezado

Como se indicó en la descripción de los requerimientos SIP, son mandatorios ciertos campos en la siguiente porción del paquete.

Las siguientes definiciones son generales, ya que los campos tienen condiciones especiales adjuntas o comportamiento dependiendo de la circunstancia:

- *Via*: este campo les dice a los nodos involucrados donde enviar el paquete SIP. Este campo tiene ciertas reglas, empezando con el requerimiento de iniciar con SIP/2.0 y los detalles de la pila de

comunicaciones. “Branch” es un número establecido por la llamada del UA, el cual debe comenzar con el número de siete caracteres “z9hG4bk” para decirle al Servidor que se trata de un valor globalmente único para cada requerimiento (excepto en el caso de un ACK, CANCEL, o falta de una respuesta 2XX) [35], asegurando de esta manera que antiguas implementaciones que usen RFC 2543 no emplearán esos valores. Esto provee una indicación de que RFC 3261 fue usado para la transmisión.

- *From*: contiene la identidad del iniciador del requerimiento en formato URI. Típicamente, se rellena con la información que ingresa el usuario o la información de configuración. Por ejemplo, los teléfonos conocen la dirección IP del servidor de llamadas, y esta es la forma en como ellos serán contactados. Además, los registros de usuario dentro del teléfono con un nombre de usuario o número telefónico.

Las etiquetas (tags) son usadas para especificar un diálogo. La única identificación es en realidad una combinación del campo “Call-ID” y las dos etiquetas en los campos “To” y “From”.

El “Call-ID” es un valor que agrupa todos los mensajes de un diálogo juntos. La RFC 3261 establece que todos los requerimientos y respuestas en un diálogo deben tener el mismo valor.

- *To*: este campo especifica el receptor del requerimiento, también en formato URI. Este no necesariamente tiene que ser el mismo que el nombre o URI del último receptor. También se lo une al nombre desplegado.

- *CSeq*: este campo provee un valor para ayudar a identificar transacciones y para ordenarlas. Este número es el mismo en todos los mensajes de un mismo tipo (Método) dentro de la llamada, pero cuando se envía un mensaje con un método distinto, el campo CSeq cambia su valor.

- *Max-Forwards*: este valor limita el número de saltos que un mensaje puede atravesar en su camino al destino. Si el campo alcanza el valor de

cero, el mensaje no será entregado, y se generará un código de error (estado). Este campo es usado en requerimientos y la recomendación es que el valor se establezca en 70.

- *Contact*: este campo debe estar presente en el requerimiento. Se supone que contiene un único URI que coincide con el formato usado anteriormente en el encabezado.

- *Allow*: esta es una lista de los métodos soportados por el agente usuario que genera el mensaje. Normalmente, el mensaje OPCIONES es usado para este propósito. Operacionalmente, los mensajes SIP podrían incluir los campos ALLOW para reducir el número total de mensaje requeridos.

- *Content-Length*: la cantidad de octetos en el cuerpo del mensaje [35].

Establecimiento de la llamada

Las sesiones SIP son establecidas usando un procedimiento de negociación de tres vías (igual que TCP), como el mostrado en la Figura 2.15.

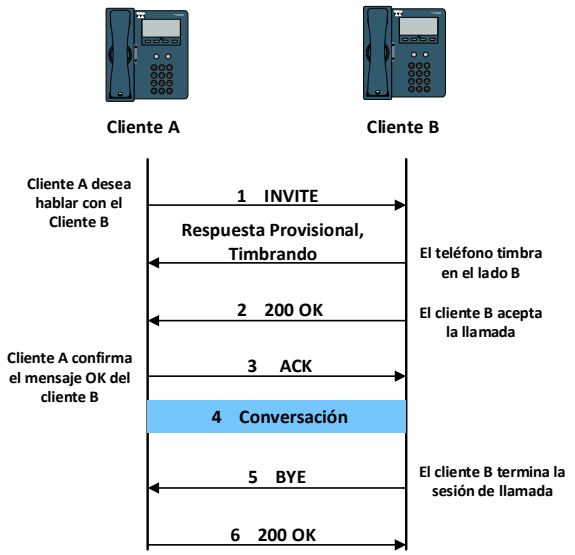


Figura 2.15: Establecimiento de una Sesión SIP

Cuando el cliente A quiere establecer una sesión telefónica con el cliente B, A envía un requerimiento INVITE a B. El mensaje contiene una carga útil con una descripción de la sesión que él quiere establecer con B. Si se

trata de una sesión telefónica de establecimiento, entonces la descripción de la sesión contiene información sobre el tipo de codificación de audio.

El cliente A puede entenderlo y también específica sobre cuales puertos quiere enviar los datos de audio del protocolo RTP. El protocolo para transmitir las descripciones es llamado Protocolo de Descripción de Sesión (SDP). No es mandatorio para SIP usar SDP, pero es el único que ha sido definido.

Cuando B, acepta la llamada, su agente usuario envía un mensaje con un código de respuesta de 200. Cualquier respuesta 2XX significa que el mensaje fue satisfactoriamente recibido, entendido y aceptado.

En la respuesta el cliente B agrega sus capacidades de códec y los números de puerto donde él quiere que el cliente A envíe sus datos. La parte final de la negociación de tres vías ocurre cuando A envía un mensaje ACK a B. Al enviar un ACK el llamante confirma que ha recibido la respuesta desde el cliente receptor. Después el procedimiento de establecimiento es completado y la conversación puede empezar.

La Figura 2.15 también muestra un mensaje opcional y una respuesta provisional. La respuesta provisional aquí es un mensaje informacional que provee retroalimentación al cliente que llama indicándole que el teléfono está timbrando en la parte receptora.

En el ejemplo anterior mostrado acerca del establecimiento de una llamada, se mostró que tipos de mensajes están involucrados al establecer una conexión SIP entre dos usuarios, pero no se dijo de qué manera el emisor descubrió donde estaba el receptor. Para descubrir donde enviar el mensaje INVITE en ese ejemplo, primero es necesario descubrir cual servidor SIP es responsable por un usuario en particular. Esto puede ser bastante complicado y está mejor explicado en (RFC 2543 – Sección 1.4.2), después de recibir información sobre donde está localizado el servidor SIP el emisor será capaz de enviar el INVITE a esa ubicación o ubicaciones (si fueron recibidas varias alternativas).

El servidor SIP leería el campo PARA en el mensaje, e iniciaría una búsqueda para el usuario particular, quien es señalado por el SIP URL en este campo. El usuario es ubicado por medio de una consulta para la ubicación del servidor, el cual podría ser un servidor de Protocolo Ligero/Simplificado de Acceso a Directorios (*LDAP,* del inglés *Lightweight Directory Access Protocol*) como se sugiere en (RFC 2543).

Cuando el servidor SIP recibe una ubicación o ubicaciones para ese usuario el cual reaccionará en una de dos formas. Reenviará el paquete a su destino, como se muestra en la Figura 2.16, o enviará un mensaje de respuesta al emisor como se muestra en la Figura 2.17, el cual contiene la(s) ubicación (es) del receptor permitiendo de esta manera que el emisor contacte al receptor él mismo. Se dice que el servidor está trabajando en modo PROXY o en modo REDIRECT.

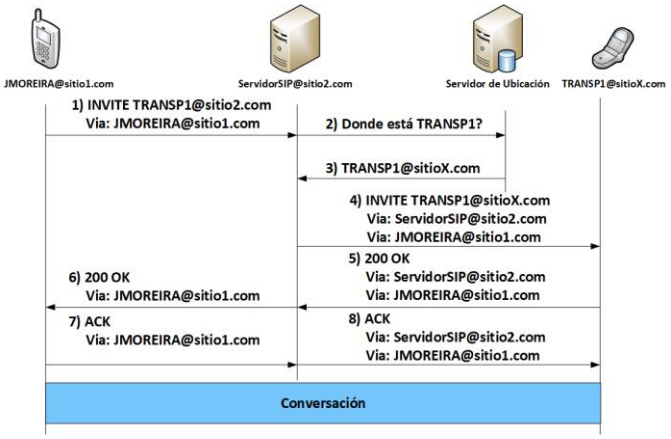


Figura 2.16: Servidor SIP en modo Proxy

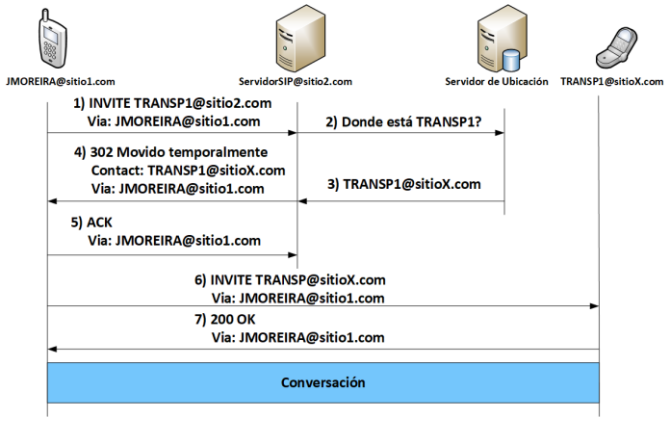


Figura 2.17: Servidor SIP en modo Redirect

Protocolo de Descripción de Sesión

El propósito del SDP es proveer suficiente información acerca de la sesión, para que [35]:

- El receptor pueda decidir si participa (basado en los requerimientos de ancho de banda, formato del contenido...).
- Si se une a una sesión, el receptor sabrá donde y como unirse.
- El receptor seguiría un puntero o URI para obtener más información o fuentes de contenido.

Cuando un UAC inicia un diálogo, típicamente incluye un bloque SDP en el cuerpo de mensaje INV. Una descripción de sesión SDP requiere:

- Nombre de la sesión y propósito.
- Tiempo(s) que la sesión está activa.
- El contenido que compone la sesión.
- Información necesaria para recibir contenido (direcciones, puertos, formatos...).

Podrían ser listados múltiples contenidos y versiones; el UAS entonces puede ver si soporta al menos uno de ellos y por lo tanto puede manipular la sesión requerida. El UAS podría responder con su propia lista de alternativas, por ejemplo, sus CODECs de audio y video [35].

El SDP tiene tres objetivos principales que necesitan ser logrados antes de que pueda empezar una sesión de telefonía IP. Primero, especificar el tipo de medio a usar: audio, video o ambos. Segundo, perfilar la comunicación. Tercero, informar a la otra parte sobre la dirección y puerto en que tú quieres que el contenido sea entregado. Para este trabajo la persona en el otro lado de la comunicación, también tendrá que enviar al emisor una descripción de la sesión con su información, o de lo contrario el emisor no será capaz de enviar ningún contenido multimedia a él.

La gramática para SDP es muy estructurada y estricta cuando se trata de decir lo que es una descripción de sesión. SDP no permite que los

encabezados vengan en cualquier otro orden que no sea el que se muestra en la Figura 2.18.

```
Descripción de Sesión
v= (versión del protocolo)
o= (propietario/creador e identificador de sesión).
s= (nombre de sesión)
i=* (información de sesión)
u=* (URI de descripción)
e=* (dirección email)
p=* (número telefónico)
c=* (información de conexión)
b=* (información del ancho de banda)

Una o más descripciones de tiempo
z=* (ajustes de la zona de tiempo)
k=* (clave de encriptación)
a=* (cero o más líneas de atributo de sesión)
Cero o más descripciones multimedia

Descripción de Tiempo
t= (tiempo durante el cual la sesión está activa)
r=* (cero o más tiempos repetidos)

Descripción Multimedia
m= (nombre multimedia y direcciones de transporte)
i=* (título multimedia)
c=* (información de)
b=* (información del ancho de banda)
k=* (clave de encriptación)
a=* (cero o más líneas de atributo multimedia)
* ítem opcional
```

Figura 2.18: Campos de un encabezado SDP

Una descripción de sesión típica se muestra en la Figura 2.19.

```
v=0
o=uabfrth 955720785594 955720785594 IN
IP4 134.138.242.7
s=Sesión Basica
c=IN IP4 134.138.242.7
t=955720785594 0
m=audio 2328 RTP/AVP 8 0 96 98 99 97
a=rtpmap:96 SC6/6000
a=rtpmap:98 SC6/3000
a=rtpmap:99 RT24/2400
a=rtpmap:97 VR15/1500
```

Figura 2.19: Ejemplo de una descripción

Una descripción de sesión SDP consiste de un número de líneas de texto de la forma <tipo>=<valor>, según [36]. Donde <tipo> debe ser exactamente un carácter y <valor> es texto estructurado cuyo formato depende de <tipo>. En general, <valor> es un número de campos delimitados por un solo espacio o una cadena de formato libre. Espacios en blanco no deben ser usados en ningún lado del signo “=”.

Una descripción de session SDP consiste de una sección de nivel de session seguida por cero o más secciones de nivel de contenido. La parte del nivel de sesión empieza con una línea “v=” y continua a la siguiente sección de nivel de contenido o termina toda la session de descripción. En general, los valores de nivel de session son los definidos por defecto para todos los contenidos a menos que se anule por un valor de nivel de contenido equivalente.

Algunas líneas en cada descripción son requeridas y algunas son opcionales, pero todas deben aparecer en exactamente el orden dado aquí (el orden fijado aquí mejora en gran medida la detección de error y permite un analizador simple). Los ítems opcionales son marcados con un “*”.

- *Versión de Protocolo (“v=”)*: el campo “v=” da la versión del Protocolo de Descripción de Sesión. La versión que se encuentra liberada es la 0. No existe un número de versión menor.

- *Origen (“o=”)*: el campo “o” da el iniciador de la sesión (su nombre de usuario y la dirección del host de usuario) más un identificador de sesión y número de versión.

- *Nombre de Sesión (“s=”)*: el campo "s=" es el nombre de sesión textual. Aquí debe estar uno y solo un campo "s=" por descripción de sesión. El campo "s=" NO DEBE estar vacío.

- *Sincronización (“t=”)*: las líneas “t=” especifican los tiempos de inicio y paro de una sesión. Se podrían usar múltiples líneas “t=” si una sesión está activa en múltiples instantes de tiempo espaciados irregularmente; cada línea “t=” adicional especifica un período de tiempo adicional para el cual la sesión estará activa. Si la sesión está activa en instantes de tiempo regulares, deberá ser usada además una línea “r=”, y seguidamente, una línea “t=” (en cuyo caso la línea “t=” especifica los tiempos de inicio y paro de la secuencia de repetición.

- *Descripciones de contenido (“m=”)*: Una descripción de sesión podría contener un número de descripciones de medio. Cada descripción de medio empieza con un campo “m=” y es terminado por el siguiente campo “m=” o por el final de la descripción de sesión. Un campo de contenido puede tener varios sub-campos, entre ellos el puerto al que se enviará el contenido.

2.6. Telefonía IP sobre WiFi

De manera similar lo que ocurre en las redes cableadas, la transmisión de voz sobre redes inalámbricas y en particular, sobre WiFi, debe sortear algunos obstáculos, tales como retrasos, jitter, pérdida de paquetes y la capacidad del canal, para que la información pueda ser percibida en el receptor de manera clara.

A continuación, se revisan los obstáculos mencionados y otros conceptos relacionados con Telefonía IP y con WiFi que se utilizan en la implementación del demostrador propuesto.

2.6.1. Generalidades y nivel de acceso al medio de WiFi con infraestructura

El estándar IEEE 802.11, fue desarrollado por el IEEE desde su Grupo de Trabajo para Estándares WLAN [37] y publicado inicialmente en el año 1.997, el cual ha sido mejorado en varias ocasiones mediante la emisión de *enmiendas* al estándar, lo que ha dado lugar a que existan varias *versiones* del mismo. Según [38], el propósito de este estándar es proporcionar conectividad inalámbrica para estaciones fijas, portátiles y móviles dentro de un área local.

Por otra parte, en el año de 1999, se juntaron un grupo de empresas interesadas en mejorar la QoE sin importar la marca de los dispositivos inalámbricos, mediante una nueva tecnología inalámbrica. Con este propósito, en el año 2000 crea la marca WiFi, con la que el grupo de empresas pasó a denominarse WiFi Alliance [39], la cual trabaja en la certificación del cumplimiento de los estándares IEEE 802.11 en nuevos dispositivos que son lanzados al mercado.

- Componentes básicos de la Arquitectura IEEE 802.11

A continuación, se revisan brevemente, varios componentes de la arquitectura IEEE 802.11,

- *Estación Inalámbrica*: Son los dispositivos direccionables, y por lo general son el origen y/o destino de un mensaje en una red IEEE 802.11. Una estación (*STA,* del inglés *Station*) inalámbrica puede ser portable o móvil.

Una *STA portable* es una en la que puede ser movida con facilidad desde una ubicación a otra, pero que sólo es usada mientras se encuentra en una ubicación fija. Las *STA móviles* pueden acceder a la WLAN mientras están en movimiento.

- *Punto de Acceso*: un PA es una entidad que posee la funcionalidad de una STA y habilita el acceso al DS, a través del WM para las STA asociadas.

- *Beacon*: es una trama en la que se incluyen información con las características de la red inalámbrica y se transmiten en determinados intervalos de tiempo para anunciar que la red está presente.

- *SSID*: el elemento SSID indica la identidad de un conjunto de servicios extendido (*ESS,* del inglés *Extended Service Set*). La longitud del campo SSID está entre 0 y 32 octetos.

- Modo de Operación en Infraestructura

El estándar IEEE 802.11 puede operar en un modo, en el cual se define una arquitectura básica llamada Conjunto de Servicios Básicos (*BSS*, del inglés *Basic Service Set*), mostrado en la Figura 2.20 . Esta arquitectura está compuesta por estaciones (*STA,* del inglés *Station*), las cuales se conectan a un sistema de distribución (*DS,* del inglés *Distribution System*) a través de un PA, el cual, es un dispositivo que posee la funcionalidad de una STA y, además, habilita el acceso al DS, a través del medio inalámbrico (*WM,* del inglés *Wireless Medium*) para las STAs asociadas. Este modo de operación descrito, se denomina Modo Infraestructura.

Un identificador de 6 bytes (48 bits) es utilizado para identificar los BSS. Este identificador se conoce como identificador de conjunto básicos de servicios (*BSSID,* del inglés *Basic Service Set Identifier*).

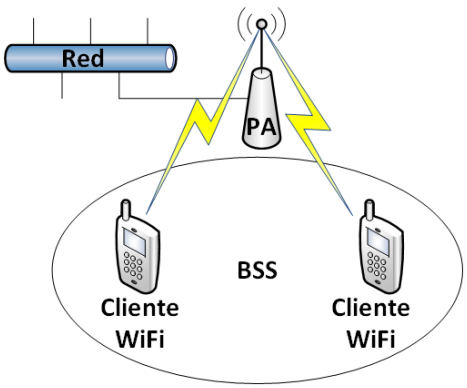


Figura 2.20: Conjunto de Servicios Básicos

- Nivel de Acceso al Medio

Tomando como referencia el modelo básico de referencia ISO / IEC de Interconexión de Sistemas Abiertos (OSI) (ISO/IEC 7498-1:1994) (*OSI,* del inglés *Open System Interconnection*), el nivel de enlace de datos dentro del estándar IEEE 802.11 consiste de dos sub niveles: el sub-nivel de control de enlace lógico (*LLC,* del inglés *Logical Link Control*) y el sub-nivel de control de acceso al medio (*MAC,* del inglés *Media Access Control*) [40].

- *Sub-Nivel de Control de Enlace Lógico (LLC)*: el estándar 802.11 utiliza el mismo nivel LLC del estándar 802.2 y el mismo direccionamiento de 48 bits al igual que otras redes LAN 802, permitiendo un puenteo muy simple desde una red inalámbrica a una red cableada IEEE.

- *Sub-Nivel de Control de Acceso al Medio (MAC)*: el sub-nivel MAC en el estándar 802.11, es muy similar en concepto al empleado en el estándar 802.3, el cual está diseñado para soportar múltiples usuarios sobre un medio compartido teniendo el emisor que escuchar el medio antes de accesarlo. En este nivel, se incluyen dos métodos de acceso al medio: la función de coordinación distribuida (*DCF,* del inglés *Distributed Coordination Function*) y la función de coordinación puntual (*PCF,* del

inglés *Point Coordination Function*), cuya ubicación dentro de la arquitectura del estándar, se muestra en la Figura 2.21.

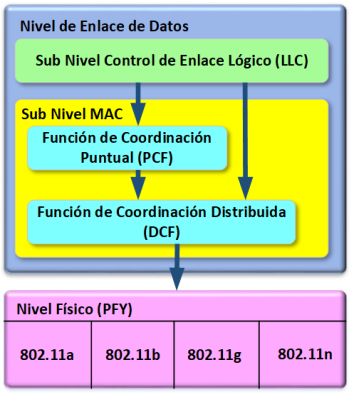


Figura 2.21: Ubicación de DCF y PCF en la arquitectura de IEEE 802.11

La razón para el uso de DCF, radica en que, mientras que en las redes LAN 802.3, el protocolo de Acceso Múltiple por Detección de Portadora con Detección de Colisión (*CSMA/CD,* del inglés *Carrier Sense Multiple Access with Collision Detection*) regula la forma en que las estaciones ethernet establecen el acceso al medio de cobre y como ellos detectan y manipulan las colisiones que ocurren cuando dos o más dispositivos tratan de comunicarse simultáneamente sobre la LAN.

En el caso de una WLAN 802.11, la detección de colisión no es posible debido a la existencia del problema “cerca del medio”: para detectar una colisión, una estación debe ser capaz de transmitir y escuchar al mismo tiempo, pero en sistemas de radio la transmisión ahoga la capacidad de la estación para “escuchar” una colisión.

Considerando esta diferencia, el estándar 802.11 emplea DCF, la cual, utiliza un protocolo ligeramente modificado conocido como Acceso Múltiple por Detección de Portadora con Evasión de Colisiones (*CSMA/CA,* del inglés *Carrier Sense Multiple Access with Collision Advoidance*). CSMA/CA trata de evitar colisiones usando reconocimiento explícito de paquetes (*ACK,* del inglés *Acknowledgment*), lo cual significa que un paquete ACK es enviado por la estación receptora para confirmar que el paquete de datos llegó intacto.

CSMA/CA trabaja de la siguiente manera: Una estación que desea transmitir sensa el medio, en este caso el aire y si no se detecta actividad, la estación espera un período de tiempo adicional seleccionado aleatoriamente, y luego transmite si el medio está todavía libre. Si el paquete es recibido intacto, la estación receptora emite una trama ACK que, una vez que es recibida satisfactoriamente por el emisor, completa el proceso. Si la trama ACK no es detectada por la estación emisora, ya sea porque el paquete de datos original no fue recibido intacto o el ACK no fue recibido intacto, se asume que ha ocurrido una colisión y el paquete de datos es transmitido de nuevo después de esperar otra cantidad de tiempo aleatorio.

De esta manera, CSMA/CA provee una forma de compartir acceso sobre el aire. Sin embargo, añade encabezado adicional al estándar 802.11 que el estándar 802.3 no tiene, por lo tanto, una LAN 802.11 siempre tendrá un desempeño más lento que una LAN Ethernet equivalente.

Haciendo un resumen del proceso que sigue la voz, para ser transmitida a través de dispositivos que cumplan el estándar IEEE 802.11, esta podría ser explicada en términos de su arquitectura la cual se muestra en la Figura 2.22.

El caso de la función PCF, se analizará en el Capítulo 3, por cuanto esta función es empleada principalmente en redes operando en modo infraestructura.

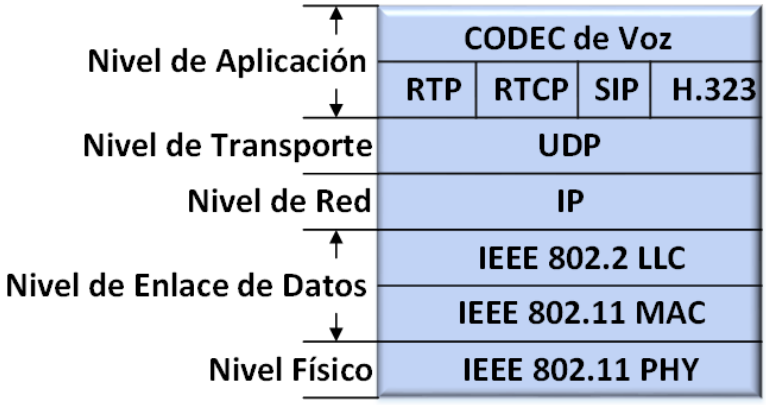


Figura 2.22: Arquitectura VoIP sobre WLAN IEEE 802.11

- Autenticación

Autenticación es el medio por el cual se verifica que una estación tenga autorización para comunicarse con una segunda estación en un área de cobertura dada. En el modo de infraestructura, la autenticación es establecida entre un AP y cada estación [41].

La Autenticación en el estándar IEEE 802.11 opera en el nivel de enlace entre las STAs IEEE 802.11. El estándar IEEE 802.11 no provee autenticación ni extremo a extremo (origen del mensaje a destino del mensaje) ni usuario a usuario [32].

El estándar IEEE 802.11 define cuatro métodos de autenticación 802.11: Autenticación de Sistema Abierto, Autenticación de clave compartida, Autenticación FT y Autenticación simultánea de iguales (*SAE,* del inglés *Simultaneous Authentication of Equals*). En este trabajo, se revisan los métodos de autenticación: sistema abierto y de clave compartida.

La autenticación de *sistema abierto* le permite a cualquier cliente autenticarse siempre y cuando se ajuste a cualquier política de filtrado de dirección MAC que pudiera haber sido establecida. Todos los paquetes de autenticación son transmitidos sin encriptación [41]. Este tipo de encriptación se muestra en la Figura 2.23.

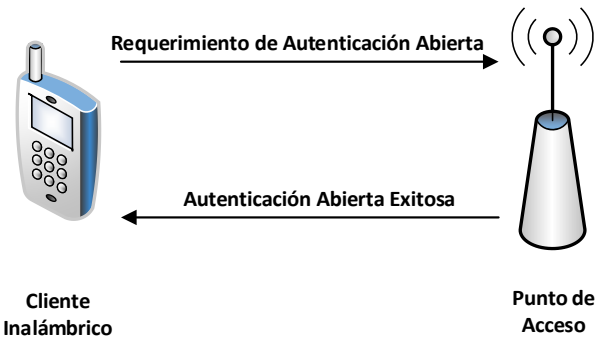


Figura 2.23: Autenticación abierta

Por otro lado, la autenticación mediante *clave compartida*, requiere que se habilite WEP, e idénticas claves WEP en el cliente y en el AP. El punto final iniciador requiere una autenticación mediante clave compartida, el cual retorna un texto de desafío no encriptado (128 bytes de texto

aleatoriamente generado). El iniciador encripta el texto y devuelve el dato [41]. Este tipo de autenticación se muestra en la Figura 2.24.

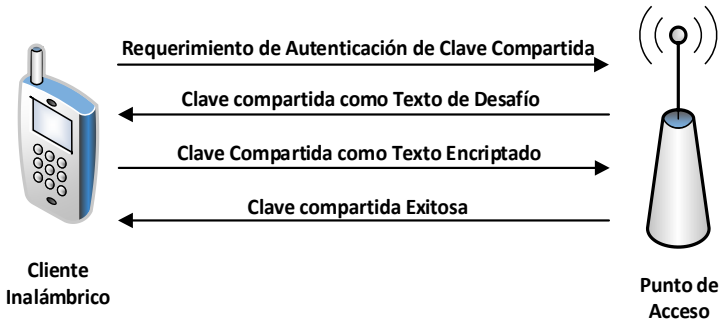


Figura 2.24: Autenticación de clave compartida

2.6.2. Características de WiFi y movimiento de terminales

La movilidad es la mayor motivación para desplegar redes 802.11 según [42], por lo que uno de los requerimientos de estándar IEEE 802.11 es el de manipular estaciones móviles en redes de área local (WLANs). El estándar IEEE 802.11 maneja la movilidad de la estación dentro del subnivel MAC, y aquí dicha movilidad es oculta para los niveles superiores en la red. Sin embargo, los eventos de desconexión y reconexión inducidos por la movilidad en una WLAN afectan significativamente el desempeño de los protocolos de niveles superiores tales como TCP. Por ejemplo, TCP interpreta las desconexiones debido a la movilidad, como congestión, y por lo tanto, disminuye multiplicativamente su tamaño de ventana de congestión. Después de la reconexión, el protocolo TCP toma un tiempo innecesariamente más largo para recuperar la ventana de congestión a un tamaño que coincida con el ancho de banda disponible [43].

- Servicios del estándar IEEE 802.11

Una forma de definir una tecnología de red es definir los servicios que ofrece y permitir a los vendedores de equipos implementar esos servicios en la forma que ellos crean convenientes. El estándar IEEE 802.11 provee nueve servicios. Solo tres de los servicios son usados para datos en movimiento; los seis restantes son operaciones de

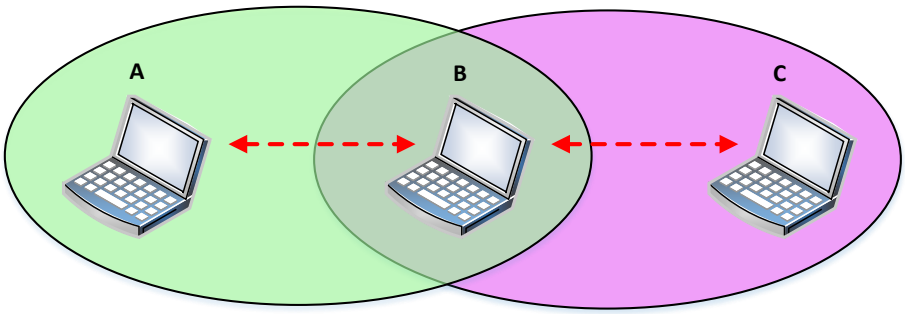
administración que permiten a la red realizar un seguimiento de los nodos móviles y entregar tramas en consecuencia. Los servicios están descritos en la Tabla 1 [42].

Servicio	Ubicación	Descripción
Distribución	Distribución	Los servicios usados en la entrega de tramas para determinar la dirección destino en redes de tipo Infraestructura.
Integración	Distribución	Entrega de trama a una LAN IEEE 802 fuera de la red inalámbrica.
Asociación	Distribución	Usada para establecer el AP que sirve como gateway a una estación móvil en particular.
Reasociación	Distribución	Usada para cambiar el AP el cual sirve como gateway a una estación móvil en particular.
Desasociación	Distribución	Remueve la estación inalámbrica de la red.
Autenticación	Estación	Establece la identidad antes de establecer la asociación.
Desautenticación	Estación	Usado para terminar la autenticación, y por extensión, la asociación.
Privacidad	Estación	Provee protección contra espionaje.
Entrega de MSDU	Estación	Entrega de datos al receptor.

Tabla 1: **Servicios del estándar IEEE 802.11**

- Problemas a enfrentrar en el Estándar IEEE 802.11

Nodo Oculto: el estándar IEEE 802.11 utiliza las tramas *request to send* (RTS) y *clear to send* (CTS) en varias circunstancias para minimizar aún más las colisiones. Las tramas RTS y CTS son especialmente útiles para resolver el problema con el "Nodo Oculto", que tienen las estaciones móviles en WLANs, y el cual se muestra en la Figura 2.25. Intercambiando el RTS y el CTS entre el emisor y el receptor se informa a las estaciones cercanas que está por empezar una transmisión.



A está hablando con **B**; **C** no conoce de esta comunicación y empieza a hablar con **B**; Por lo que se produce una colisión

Figura 2.25: Problema del Nodo Oculto

La información de duración en las tramas RTS/CTS se utiliza para establecer el vector de asignación de red (*NAV,* del inglés *Network Allocation Vector*) en todas las estaciones que están dentro del rango de recepción de las tramas RTS/CTS. De esta forma, el problema de un emisor oculto puede ser resuelto ya que cualquier estación que observe la trama CTS sabe que está cerca del receptor y, de por lo tanto, no puede transmitir durante el período de tiempo indicado en el NAV. Si las tramas de dato transmitidas son cortas, no se recomienda enviar tramas RTS/CTS, ya que añaden ineficiencia por el sobre encabezado. Por lo tanto, se define un umbral para usar RTS/CTS solamente sobre tramas mayores que una longitud específica [43].

Ya que los requerimientos RTS/CTS añaden encabezado adicional a la red mediante la reserva temporal del medio, este protocolo es usado típicamente solo sobre paquetes de gran tamaño, para los cuales la retransmisión sería costosa en términos de ancho de banda.

Transiciones: las estaciones pueden moverse mientras estén conectadas a la red y transmitir tramas mientras estén en movimiento. La movilidad puede causar uno de tres tipos de transición [42]:

- *No transición*: cuando las estaciones no se mueven fuera de las áreas de servicio de sus PA actuales, no es necesaria una transición. Este estado ocurre ya que la estación no se está moviendo o se está moviendo dentro del área de servicio básico de su PA actual.
- *Transición BSS*: las estaciones continuamente monitorean la potencia de la señal y la calidad de todos los PA asignados administrativamente para cubrir y extender el área de servicio. Dentro de un área de servicio extendido, el estándar 802.11 provee movilidad del nivel MAC. Las estaciones conectadas al sistema de distribución pueden enviar fuera tramas direccionadas a la dirección MAC de una estación móvil y dejar que los PA manejen el salto final para la estación móvil. Las estaciones del Sistema de Distribución no necesitan conocer la ubicación de una

estación móvil, siempre y cuando se encuentren dentro de la misma área de servicio extendida.

La Figura 2.26 ilustra una transición de BSS. Los tres PA en la figura están todos asignados al mismo ESS. En el instante *t=1*, la laptop con una tarjeta de red IEEE 802.11 es situada dentro del área de servicio básico del AP1 y está asociada con el AP1. Cuando la laptop se mueve fuera del área de servicio básico AP1 al área de servicio básico de AP2 en *t=2*, ocurre una transición de BSS. La estación móvil usa el servicio de re-asociación para asociarse con AP2, con lo cual luego empieza a enviar tramas a la estación móvil.

Las transiciones BSS requieren la cooperación de PA. En este escenario, AP2 necesita informar a AP1 que la estación móvil ahora está asociada con AP2. El estándar IEEE 802.11 no especifica los detalles de las comunicaciones entre los PA durante las transiciones BSS.

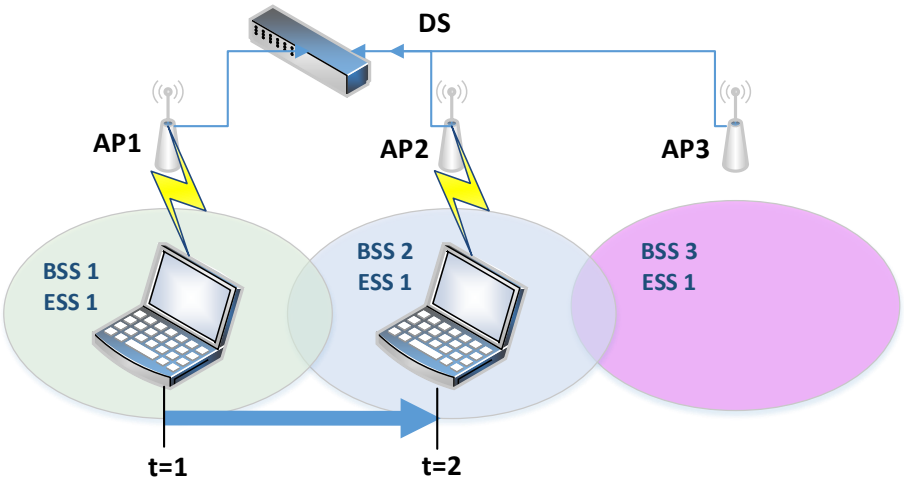


Figura 2.26: Transición BSS

- *Transición ESS*: una transición ESS se refiere al movimiento desde una ESS a un segundo y distinto ESS. El estándar IEEE 802.11 no soporta este tipo de transición, excepto para permitir a la estación que se asocie con un PA en el segundo ESS una vez que esta deje el primero. Está casi garantizado que las conexiones en el nivel alto sean interrumpidas. El mantener conexiones de nivel superior requiere

soporte de las suites de protocolos en cuestión. En el caso de TCP/IP, se requiere IP Móvil para soportar de forma transparente una transición ESS.

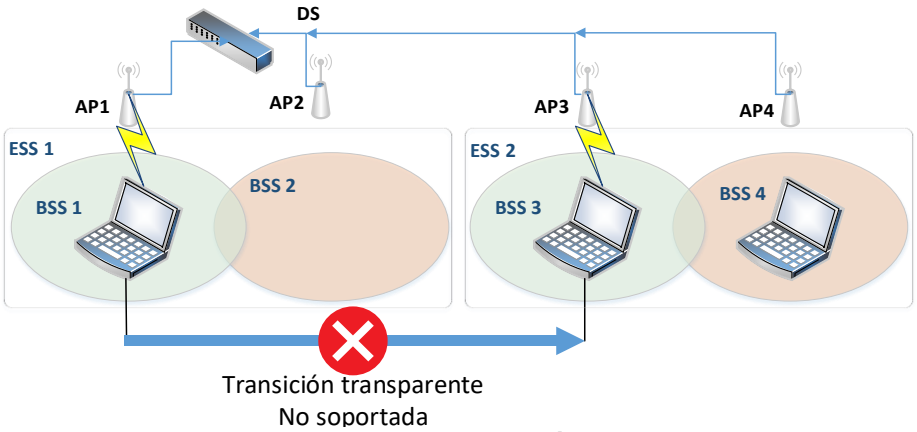


Figura 2.27: Transición ESS

En la Figura 2.27, se describe una transición ESS. Cuatro áreas de servicio básico son organizadas en dos áreas de servicio extendido. No son soportadas las transiciones de forma transparente desde la ESS de la izquierda hacia la de la derecha. Las transiciones de ESS sólo se admiten porque la estación móvil se asocia rápidamente con un punto de acceso en el segundo ESS. Es probable que cualquier conexión de red activa desaparezca cuando la estación móvil abandone el primer ESS.

2.6.3. Gestión de tráfico de telefonía IP en WiFi con herramientas software

En el ámbito de las herramientas de software disponibles para la Gestión de tráfico de Telefonía IP existe una gran variedad de programas para dicho fin, las cuales pueden ser del tipo Gratuito o de Pago por Licencia de Uso. De manera similar, existen herramientas para sistemas operativos del tipo Código Abierto (Open Source) como Linux y para Sistemas Operativos con licenciamiento como el caso del popular Windows. A continuación, se presentan cuatro de ellas: *Wireshark*, *VoIP Monitor*, *CommView*, *Pathtest*.

Wireshark: inicialmente llamado Ethereal, Wireshark es un analizador de paquetes de red, y como tal, trata de capturar paquetes de red e intenta desplegar esos paquetes de la manera más detallada posible [44].

Wireshark puede capturar tráfico desde diferentes tipos de medio de red, incluyendo redes LAN inalámbricas.

Entre las principales aplicaciones de Wireshark están: resolución de problemas de red, examinar problemas de seguridad, depurar implementaciones de protocolos, aprender la composición interna de algún protocolo...

En la Figura 2.28, se puede apreciar cómo se visualiza el contenido de un paquete que es capturado por el programa Wireshark.

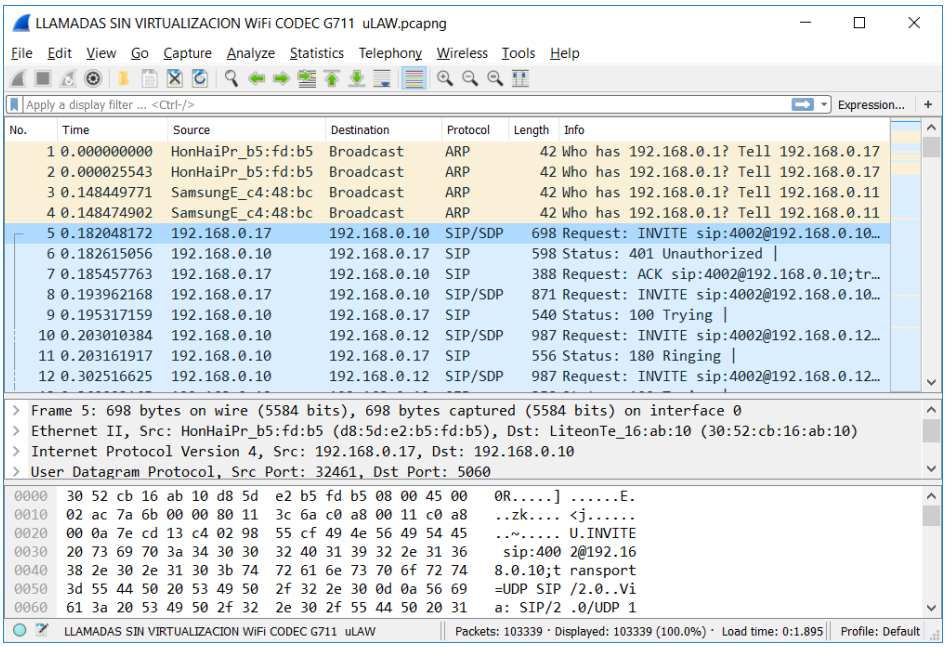


Figura 2.28: Visualización del contenido de un paquete capturado

Una característica importante de esta herramienta, es la del Análisis Gráfico de llamadas de VoIP, la cual permite visualizar de manera resumida el flujo que siguen los paquetes de una llamada de VoIP entre los dispositivos que conforman la red de telefonía IP. En la Figura 2.29, se puede apreciar una parte de una secuencia en una llamada de telefonía IP.

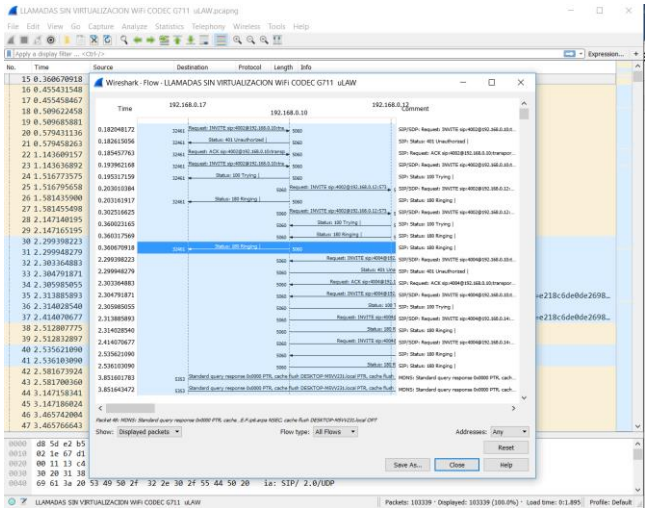


Figura 2.29: Análisis Gráfico de una llamada de VoIP

VoIP monitor: VoIPmonitor [45] es un husmeador (sniffer) de paquetes de red de Código Abierto (Open Source) con una interfaz comercial para protocolos de VoIP SIP, RTP, RTCP y SKINNY(SCCP) corriendo sobre Linux. VoIPmonitor está diseñado para analizar la QoS de llamadas VoIP basado en parámetros de red tales como variación de retrasos y pérdidas de paquetes de acuerdo a la recomendación ITU-T G.107 E-Model el cual predice la calidad sobre una escala MOS. Las llamadas con todas las estadísticas relevantes son guardadas en una base de datos MySQL u ODBC. Opcionalmente cada llamada puede ser guardada en un archivo de tipo *pcap* con los protocolos SIP/RTP/RTCP.

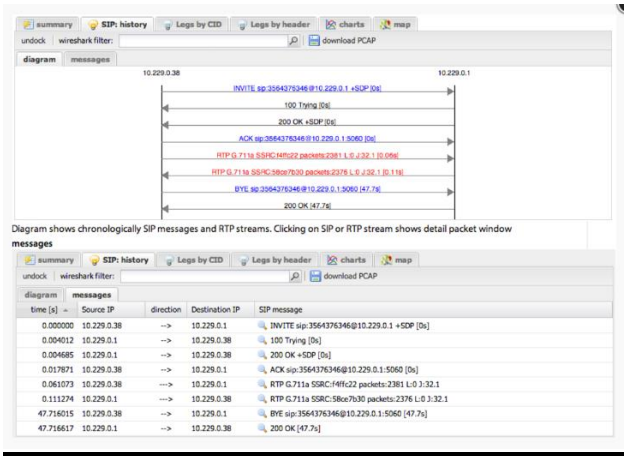


Figura 2.30: Visualización del intercambio de mensajes en una conexión de VoIP

En la Figura 2.30 se presenta un diagrama que muestra el intercambio de mensajes en una conexión de VoIP.

VoIPmonitor también puede decodificar conversaciones y reproducirlas sobre el WEB GUI comercial o guardarlas al disco como archivo WAV. Soporta los códecs G.711 alaw/ulaw y los plugins comerciales soportan G.722 G.729A G.723 iLBC Speex GSM Silk iSAC OPUS. VoIPmonitor también es capaz de convertir archivos del tipo T.38 FAX a PDF. En la Figura 2.31 se presentan los parámetros que se pueden medir durante una llamada de VoIP.

Figura 2.31: Parámetros registrados durante una llamada de VoIP

CommView: CommView [46] es un monitor de red y analizador, diseñado para administradores LAN, profesionales de seguridad, programadores de red, usuarios de hogar... Virtualmente, cualquiera que quiera una imagen completa del tráfico fluyendo a través de una PC o un segmento de LAN.

Esta aplicación captura cada paquete sobre el cable para desplegar importante información tal como una lista de paquetes y conexiones de red, estadísticas vitales, diagramas de distribución de protocolo... Se puede examinar, guardar, filtrar, importar y exportar paquetes capturados, la vista de protocolo decodifica hasta el nivel más bajo con un completo análisis de más de 100 protocolos soportados. Con esta información, puede ayudar a identificar problemas de red y solucionar problemas de

software y hardware. Incluye un analizador de VoIP para análisis exhaustivo, grabación, y reproducción de comunicaciones de voz H.323 y SIP.

Para tareas de monitoreo remoto, se puede usar opcionalmente el add-on especial: Agente Remoto CommView. Este permite a los usuarios de CommView capturar tráfico de red sobre cualquier computador donde el Agente Remoto esté corriendo, independientemente de la ubicación física del computador.

Esta tecnología expande el rango de monitoreo: no se está limitado al segmento LAN utilizado o al computador personal.

En la Figura 2.32 se puede apreciar un listado de las conexiones que están siendo monitoreadas por el programa, donde se detallan las direcciones IP locales y remotas, la cantidad de paquetes transmitidos, los puertos utilizados, nombre del equipo Host...

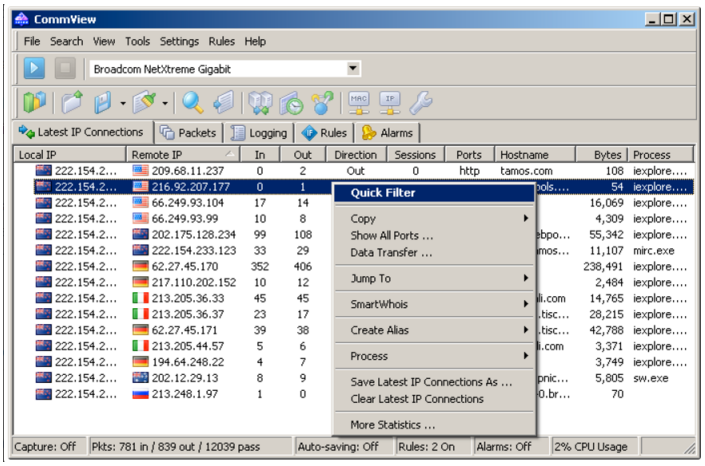


Figura 2.32: Vista de las conexiones monitoreadas por el programa

En la Figura 2.33, se puede visualizar el panel dedicado al monitoreo de conexiones basadas en VoIP. En esta parte del programa, las conexiones están catalogadas de acuerdo al protocolo que está siendo utilizado (SIP, H.323…), pudiéndose apreciar algunos datos importantes como direcciones IP origen y destino, hora de inicio, hora de finalización, duración, estado. Además, se pueden visualizar parámetros de QoS como el MOS y el factor R los cuales se revisan en la siguiente sección.

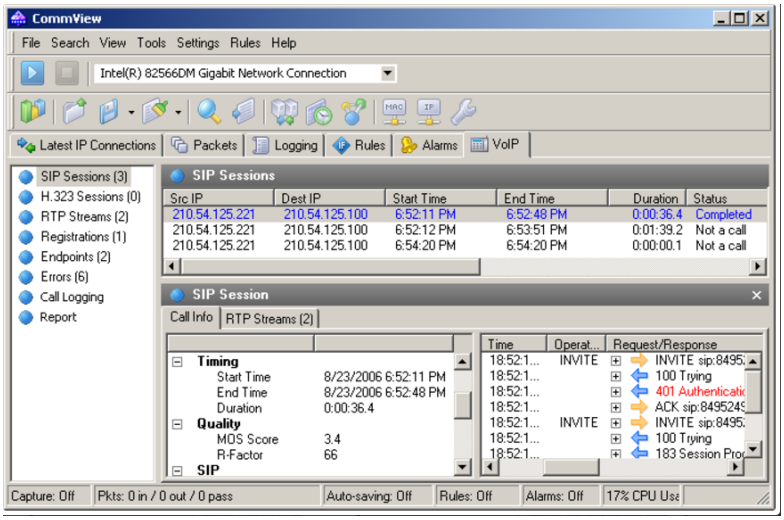


Figura 2.33: Panel de Monitoreo VoIP en CommView

Pathtest: el programa PathTest [47] es un software desarrollado por la empresa AppNeta, y es utilizado para probar la capacidad máxima de transmisión de una red y ofrece una alta precisión en los resultados además de ser una herramienta de prueba para el rendimiento del nivel 3 y nivel 4 y pruebas amplias y precisas.

Esta herramienta utiliza la técnica de inundación de paquetes en la red, por lo que debe ser utilizada con precaución ya que podría interferir en el desempeño de la red que está siendo analizada.

En la Figura 2.34 se puede observar el resultado que se obtiene al utilizar la herramienta PathTest en la medición de la capacidad de una conexión entre dos equipos (cliente y servidor) dentro de una red.

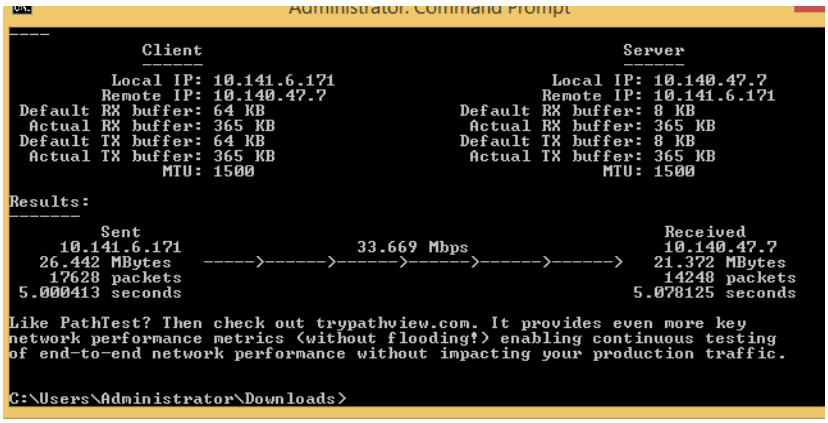


Figura 2.34: Resultados de Medición de Capacidad

2.6.4. Parámetros de Calidad de Servicio y Experiencia de usuario para Telefonía IP en WiFi

Los parámetros utilizados para determinar la QoS y la QoE en una comunicación inalámbrica son similares a los utilizados en una comunicación por cable.

Actualmente, el estándar IEEE 802.11, por ende, los dispositivos con la certificación WiFi incluyen el soporte de aplicaciones con requerimientos de QoS [32] mediante técnicas de priorización en la transmisión de paquetes.

A pesar de aquello, es necesario revisar los parámetros que permiten determinar la QoS y la QoE en las comunicaciones de telefonía IP en un ambiente WiFi, para lo cual se deben conocer algunos aspectos que pueden convertirse en un obstáculo para mantener una comunicación en una red de datos que transporta VoIP. Entre esos aspectos se tienen: retraso, jitter, pérdida de paquetes y el throughput.

Retraso: el *Retraso* o también conocido como *Latencia*, puede ser definido como el tiempo total que tarda un paquete de VoIP en ir desde el emisor hasta el receptor, en otras palabras, el tiempo total que le toma a la voz en viajar desde que esta se emite en el lado transmisor hasta que es percibida en el lado del receptor.

Este parámetro es determinante, durante la realización de comunicaciones basadas en VoIP, debido a que la transmisión de voz es muy sensible a los retrasos, aunque pueden ser tolerados hasta cierto límite estos retrasos, para el caso del ITU G.114 máximo 150ms y en el caso del ETSI máximo 100ms [48].

En [48] y [49], se menciona que existen tres componentes que ocasionan el retraso de los paquetes VoIP:

- El estado de la red, el cual puede provocar *retrasos de red* el cual es el retraso total de las redes inalámbricas y de backbone, adicionalmente los retrasos introducidos por encolamiento, transmisión y propagación. Los

retrasos de transmisión son los provocados por retrasos de encaminamiento y retransmisión. Mientras los retrasos por propagación son los introducidos por el medio físico de la red.

- La codificación y paquetización, usadas por el emisor introducen retrasos que se conocen como *retrasos en la fuente*.

- El proceso inverso, que se realiza en el receptor al realizado en el transmisor provoca *retrasos en el receptor* debido además del retraso introducido por el buffer del jitter.

Jitter: los retrasos obtenidos durante la transmisión y recepción de dos paquetes consecutivos, podrían no ser exactamente los mismos, lo que origina una variación en los retrasos de las dos transmisiones realizadas, variación a la cual se conoce como *Jitter*.

Existen varias técnicas por medio de algoritmos basados en memoria, que tratan de mitigar el problema del Jitter y de esta manera mejorar la calidad de la comunicación e VoIP.

El jitter tolerable en una comunicación de VoIP según [49] está entre 0ms y 50ms.

Pérdida de Paquetes: Una congestión en la red puede provocar que los paquetes lleguen tarde al buffer o por el contrario el buffer esté lleno a la llegada del paquete debido a un sobre flujo y el paquete se tenga que descartar. En ambos casos, se producen *pérdidas de paquetes*.

Al producirse la pérdida de paquetes, el emisor es informado, por lo cual retransmite el mensaje, aumentando la congestión, en consecuencia, aumentando el retraso ocasionando una degradación de la QoS de la comunicación.

En [48], se indica que los sistemas de VoIP pueden tolerar pérdidas de paquetes entre el 1% y el 3%. Más allá de ese valor, la calidad de la conversación se degrada por la pérdida de paquetes.

Throughput: El throughput es definido como la cantidad de datos que pueden ser enviados sobre un canal de comunicación en un período de tiempo dado (normalmente un segundo) incluyendo todo el sobre encabezado y elementos de retraso introducidos por los protocolos de la red WLAN. Según [50], el throughput que se puede obtener en una WLAN, es cerca del 50% de la tasa bruta de transmisión del estándar, pudiendo llegar hasta un 70% según se menciona en [48].

Por otro lado, el throughput alcanzado durante una llamada de VoIP, se ve influenciado por el CODEC elegido, para establecer la comunicación. Adicionalmente, los encabezados de los diferentes protocolos que intervienen en el encapsulamiento de la voz antes de ser transmitida influyen en el throughput utilizado en la comunicación. Un listado del tamaño de los encabezados de protocolos comúnmente utilizados en las comunicaciones de VoIP se muestra en la Tabla 2.

Protocolo u Origen del Encabezado	Nivel OSI	Tamaño Encabezado o Carga Util CODEC G.711 [Bytes]	Tamaño Encabezado o Carga Util CODEC G.729 [Bytes]	Tamaño Encabezado o Carga Util CODEC GSM [Bytes]
Carga Util Voz	7	160	20	33
RTP	5	12	12	12
UDP	4	8	8	8
IP	3	20	20	20
ETHERNET	2	18	18	18
WiFi (IEEE 802.11n)	2	36	36	36

Tabla 2: Resumen tamaño encabezados según nivel OSI

Por lo anterior, es necesario conocer cómo se calcula el throughput. Los parámetros que intervienen en el cálculo del throughput de acuerdo a [51] son: el intervalo de muestreo del CODEC (*CSI*, del inglés *CODEC Sample Interval)* [ms], el tamaño de muestras del CODEC (*CSS*, del inglés *CODEC Sample Size*) [Bytes], la tasa de bit del CODEC (*CBR*, del inglés *CODEC Bit Rate*) [Kbps] de la ecuación 2.1, el tamaño de la carga útil de voz (*VPS*, el inglés *Voice Payload Size*) [Bytes o ms], la cantidad de paquetes por segundo (*PPS*, del inglés *Packet Per Seconds*) de la

ecuación 2.2 y el tamaño del paquete total (*TPS*, del inglés *Total Packet Size)* [Bytes] de la ecuación 2.3.

$$\mathbf{CBR} = \frac{\mathbf{CSS}}{\mathbf{CSI}} \text{ [ms]} \tag{2.1}$$

$$\mathbf{PPS} = \frac{\mathbf{CBR}}{\mathbf{VPS}} \text{ [pps]} \tag{2.2}$$

$$\mathbf{TPS} = \mathbf{L2} + \mathbf{IP} + \mathbf{UDP} + \mathbf{L5} + \mathbf{VPS} \text{ [Bytes]} \tag{2.3}$$

Donde L2 es el número de bytes en la cabecera de la trama del nivel 2; IP el número de bytes en la cabecera del paquete del nivel 3; UDP el número de bytes de la cabecera del segmento de nivel 4; L5 el número de bytes de la cabecera del protocolo de nivel 5.

$$\mathbf{Throughput} = \mathbf{TPS} \times \mathbf{PPS} \text{ [Kbps]} \tag{2.4}$$

Para el caso de *n* cantidad de llamadas, al ser cada llamada una comunicación full-dúplex, se debe multiplicar por dos el valor del throughput por la cantidad n de llamadas que se realicen.

$$\mathbf{Throughput\_n} = \mathbf{2} \times \mathbf{n} \times \mathbf{Throughput} \text{ [Kbps]} \tag{2.5}$$

- Medición de la Calidad de Servicio de la VoIP en redes inalámbricas

Al igual que en las redes cableadas [52], en las redes inalámbricas, existen dos maneras de determinar la QoS y con ello la QoE: mediante Métodos Subjetivos y mediante Métodos Objetivos, de acuerdo a [48] y [53].

Métodos Subjetivos: esta forma de medir la QoS y QoE se basa en la percepción de los usuarios, esto es, mediante la opinión de los usuarios oyentes, quienes califican la comunicación en una escala que va de 1 (mala) a 5 (excelente), de cuyas mediciones se obtiene un promedio. A este procedimiento se le conoce como Puntuación de Opinión Media (MOS, del inglés Mean Opinion Score) [29] **[49] [52] [53]**, cuyo uso está estandarizado en la recomendación P.800 de la ITU-T.

Métodos Objetivos: estos métodos miden los niveles de QoS en función de parámetros de la red tales como el delay, el jitter y pérdida de paquetes.

Estos métodos se subdividen a su vez, en métodos *Intrusivos*, los cuales inyectan una señal de prueba durante un tiempo en el cual la red no debe transportar ningún tipo de tráfico; y métodos *no Intrusivos* los cuales basan sus resultados de QoS en la medición de los parámetros de la red tales como el delay, jitter y pérdida de paquetes.

PESQ: el uso del método de Medición Perceptiva de la Calidad del Habla (PESQ, del inglés Perceptual Speech Quality Measurement), está definido en la recomendación P.862 de la ITU-T y es utilizado para establecer de manera predictiva la calidad de la voz en comunicaciones telefónicas para lo cual se emplean los códecs de voz más comunes [52].

El funcionamiento de PESQ puede resumirse en la Figura 2.35 y consiste en comparar dos señales, una de ellas la señal original x(t), la cual sirve de referencia, y la otra es una señal degradada y(t) la cual es el resultado de transmitir la señal r(t) a través del sistema a evaluar.

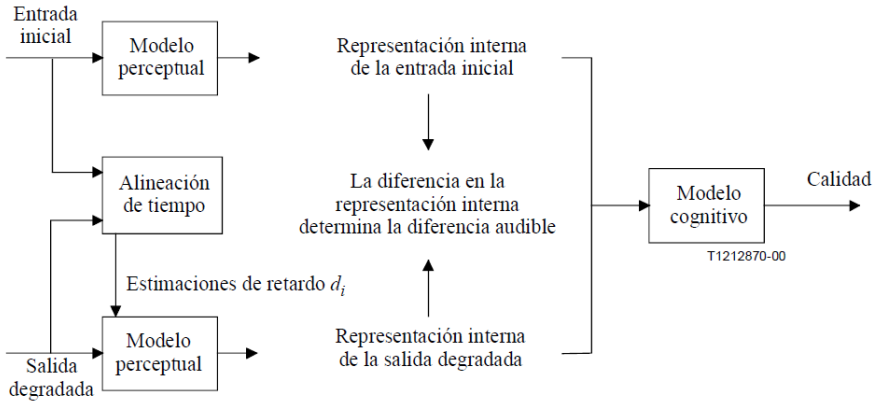


Figura 2.35: Procesamiento realizado sobre la señal de voz en PESQ

Ambas señales, siguen luego una serie de procesos, en los que primero se alinean en el tiempo, por cada intervalo recibido. Luego, mediante un proceso las señales son transformadas utilizando un modelo perceptual en el cual son transformadas en una representación interna que intenta reproducir la representación psico acústica de señales de audio en el sistema auditivo humano, teniendo en cuenta la frecuencia por percepción (Bark) y la sonoridad (Sone) [52].

Finalmente, se calcula una distancia entre la señal vocal inicial y la señal vocal degradada (nota PESQ). La nota PESQ se hace corresponder a una escala similar a la de MOS, un número único en una escala de 0.5 a 4.5, aunque en la mayoría de los casos la gama de las salidas estará entre 1.0 y 4.5, que es la gama normal de valores de MOS que suelen darse en un experimento sobre la calidad de voz [53].

E-model: el modelo E, ó comúnmente llamado E-model, es un modelo desarrollado por la ETSI y adoptado por la ITU-T [54] [55], el cual es utilizado para la planificación de la transmisión y es el modelo más ampliamente difundido [52].

Al emplear este método se proporciona un valor, el cual es una predicción de la calidad del audio, la que es representada por medio del factor *R*, el cual es obtenido de ecuación 2.6 [54].

$$\mathbf{R = R_0 - I_s - I_d - I_{e-eff} + A} \tag{2.6}$$

En la ecuación anterior, R_0 representa la relación señal a ruido básica; I_s es la suma de todos los deterioros los cuales podrán ocurrir más o menos simultáneamente con la transmisión de la señal de voz; I_d representa todos los deterioros debido a retrasos en las señales de voz; el factor de deterioro de equipo efectivo $I_{e\text{-}eff}$ representa los deterioros causados por códecs de bajas tasa de bit. El factor de ventaja A permite al usuario compensar los factores de deterioro cuando el usuario se beneficia de otros tipos de acceso [54].

Existe una relación entre el MOS y el factor R según [54], el cual puede expresarse mediante la ecuación 2.7.

$$\mathbf{MOS} = \begin{cases} \mathbf{1} & ; \quad \boldsymbol{R < 0} \\ \mathbf{1 + 0.035}\boldsymbol{R} + \mathbf{7}\boldsymbol{R}(\boldsymbol{R} - \mathbf{60})(\mathbf{100} - \boldsymbol{R})\boldsymbol{x}\mathbf{10}^{-\mathbf{6}} & ; \mathbf{0} \leq \boldsymbol{R} \leq \mathbf{100} \\ \mathbf{4.5} & ; \quad \boldsymbol{R > 100} \end{cases} \tag{2.7}$$

Dicha relación puede ser representada mediante la gráfica mostrada en la Figura 2.36.

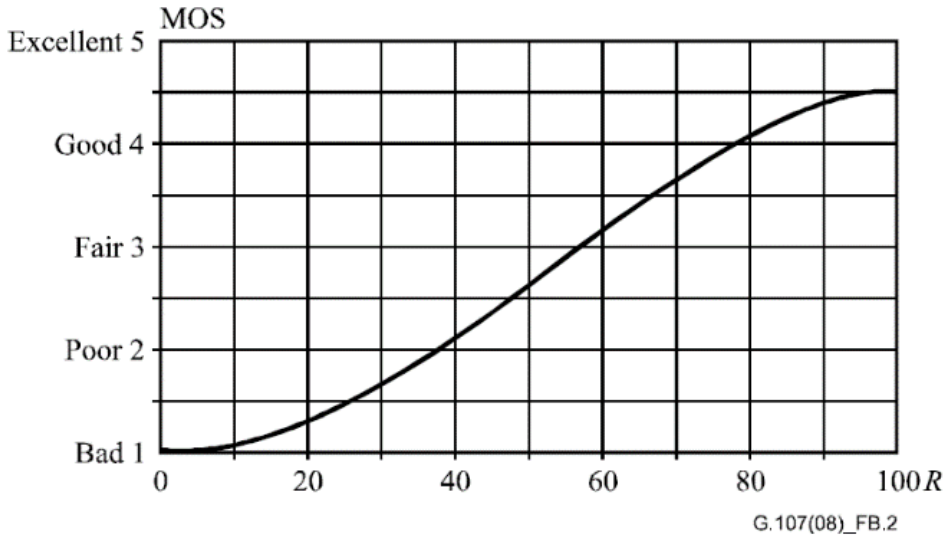


Figura 2.36: Gráfica de la relación entre el Factor R y el valor MOS

También los valores del factor R obtenidos, pueden relacionarse con la escala MOS mediante la tabla Tabla 3 [54].

Factor R (**Límite inferior**)	**MOS** (**Límite inferior**)	**Satisfacción de usuario**
90	4.34	Muy Satisfecho
80	4.03	Satisfecho
70	3.60	Algunos usuarios Insatisfechos
60	3.10	Muchos usuarios Insatisfechos
50	2.58	Casi todos los usuarios Insatisfechos

Tabla 3: **Relación entre el factor R y la satisfacción del usuario**

Por otra parte, existe una versión simplificada de la ecuación 2.6, puede encontrarse en [56], [57] y [58], donde previamente se calcula una *Latencia Efectiva* mediante la ecuación 2.8.

$$\textbf{Latencia Efectiva} = \textbf{Latencia Promedio} + (\mathbf{2} \times \textbf{Jitter}) + \mathbf{10} \qquad (2.8)$$

Posteriormente, se calcula un factor R inicial, cuya ecuación depende del valor de *Latencia efectiva* calculado, según la ecuación 2.9.

$$\boldsymbol{Ro} = \begin{cases} \mathbf{93.2} - \left(\frac{\boldsymbol{Latencia\ Efectiva}}{\mathbf{40}}\right) & ;\ \boldsymbol{Latencia\ Efectiva} < \mathbf{160} \\ \mathbf{93.2} - \left(\frac{\boldsymbol{Latencia\ Efectiva} - \mathbf{120}}{\mathbf{10}}\right) & ;\ \boldsymbol{Latencia\ Efectiva} \geq \mathbf{160} \end{cases} \qquad (2.9)$$

Finalmente, el valor del factor R puede ser calculado, descontando al valor obtenido en la ecuación 2.9, un factor de 2.5 veces el porcentaje de pérdida de paquetes, como se muestra en la ecuación 2.10.

$$\mathbf{R} = \mathbf{Ro} - (\mathbf{Paquetes\ Perdidos} \times \mathbf{2.5}) \quad (2.10)$$

A partir del factor R obtenido con la ecuación 2.10, el valor del MOS puede ser proyectado mediante la ecuación 2.11.

$$\mathbf{MOS} = \mathbf{1} + (\mathbf{0.035} \times \mathbf{R}) + (\mathbf{0.000007} \times \mathbf{R}) \times (\mathbf{R} - \mathbf{60}) \times (\mathbf{100} - \mathbf{R}) \quad (2.11)$$

Para el caso de los CODEC, existen valores característicos para varios parámetros. Entre esos parámetros se encuentra el throughput requerido para la comunicación y el MOS, como se muestran en la Tabla 4 [51] [59].

CODEC	Intervalo de Muestreo del CODEC [ms]	Tamaño de Muestras del CODEC [Bytes]	Tasa de Bit del CODEC [Kbps]	Tamaño de Carga Útil de Voz [Bytes o ms]	Tamaño Total del Paquete [Bytes]	Paquetes por segundo [pps]	Through-put * [Kbps]	MOS
G.711	10	80	64	20[ms] - 160[Bytes]	218	50	87,2	4.1
G.729	10	10	8	20[ms] - 20[Bytes]	80	50	32	3.92
GSM	20	33	13,2	20[ms] - 33[Bytes]	91	50	36,4	3.5

Tabla 4: Valores característicos de varios parámetros de diferentes CODEC

Por tratarse de comunicaciones en un sólo sentido, en caso de tener comunicaciones full duplex se deben multiplicar por dos los valores del throughput mostrados en la Tabla 4 para obtener el throughput total de una llamada.

CAPÍTULO 3

3. ESTADO DEL ARTE EN VIRTUALIZACIÓN DE WiFi CON INFRAESTRUCTURA

La virtualización de redes WiFi, es un concepto que promete varias ventajas en el manejo de dispositivos del mismo tipo, lo que podría resultar en una utilización más eficiente de los recursos de una red inalámbrica.

En este capítulo se profundiza en el tema de la Virtualización de redes inalámbricas.

3.1. Estratégias para la virtualización

En esta sección, se revisan varias técnicas y otros recursos para lograr la virtualización de una red inalámbrica. Además, se estudia el concepto de VPA, y cómo desarrollar una interfaz inalámbrica virtual para su utilización como PA.

3.1.1. Virtualización de redes inalámbricas basada en flujo

Dentro de esta perspectiva de virtualización, se define como *flujo* a la secuencia de datos que fluyen a través de un canal de comunicación; o como se lo define en [20], un “flujo de datos compartiendo una firma común”.

La virtualización de redes inalámbricas basada en flujo se ocupa de crear una ruta de datos personalizada y controlada de la red inalámbrica, que permita a los flujos de datos contar con servicios diferenciados, aislamiento, gestión y programación. Esta forma de virtualización puede ser implementada mediante filtros sobrepuestos o mediante un módulo de conmutación por software sobre el hardware existente [20].

Este tipo de virtualización toma como modelo la tecnología SDN revisada en el Capítulo 2 llamada OpenFlow y puede ser llamada “Virtualización de red Móvil” ya que tuvo sus orígenes en este tipo de redes [20].

Según [20] y [60], existen dos tipos de virtualización de redes inalámbricas basada en flujo: la virtualización sobrepuesta y la virtualización integrada cuyas arquitecturas se muestran en la Figura 3.1.

Esencialmente, la virtualización sobrepuesta basada en flujo, funciona a través de un hardware externo, diferente al hardware de acceso inalámbrico. En este hardware externo, se incluyen el hipervisor y el nivel de virtualización que tienen funciones similares a las de un filtro, de un conmutador y de un encaminador. Esto les permite poder controlar y configurar las rutas de datos de los nodos inalámbricos.

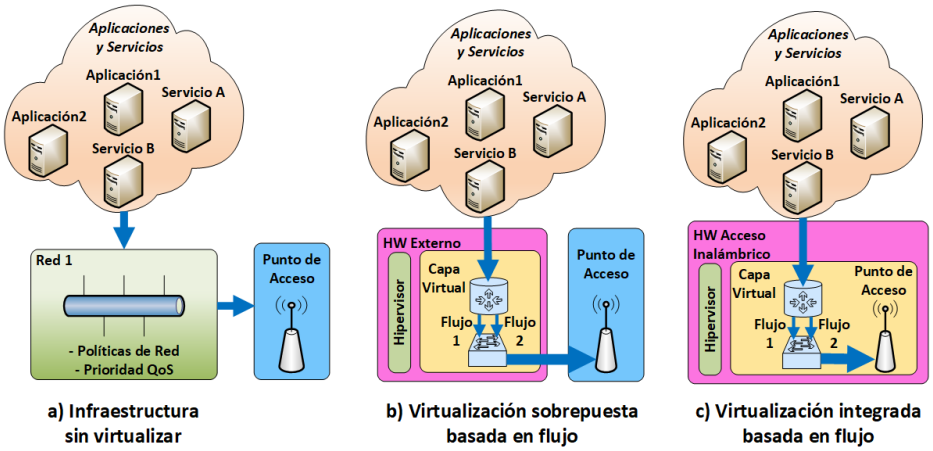


Figura 3.1: Representación en bloques de la Virtualización basada en Flujo Integrada y Sobrepuesta

Mientras que, en la virtualización integrada basada en flujo, la capa de virtualización se ubica dentro del hardware de acceso inalámbrico, lo que posibilita el acceso a las funciones de programación del nodo inalámbrico.

El comportamiento de este tipo de virtualización inalámbrica, posibilita la integración en una misma infraestructura de varias tecnologías de comunicación inalámbrica.

Según [60], la virtualización inalámbrica basada en flujo, es la más factible de implementar con beneficios inmediatos, siendo los principales beneficios la provisión de tráfico más eficiente y flexible y administración de recursos.

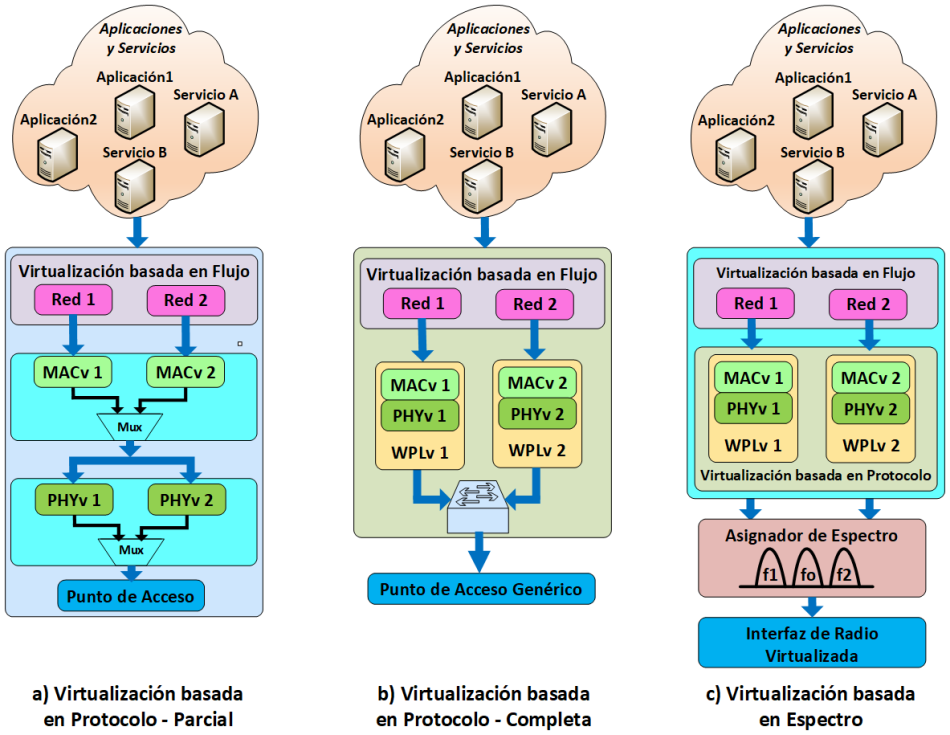
3.1.2. Virtualización de redes inalámbricas basada en protocolo

La tendencia actual de implementar radio definida por software (*SDR*, del inglés *Software Defined Radio*), ha dado lugar a que muchas funciones

del hardware de comunicaciones, sean implementadas en software, provocando que los protocolos inalámbricos tengan una independencia del hardware.

La virtualización de redes inalámbricas basadas en protocolo trata de aprovechar esta situación, mediante la separación en instancias de protocolos inalámbricos en el mismo hardware de radio [60]. En el caso de la virtualización basada en protocolo, los recursos que son virtualizados son los que se encuentran en el nivel MAC y en el nivel Físico.

Por otra parte, existen dos tipos de implementación en el caso de la virtualización inalámbrica basada en protocolo, dependiendo de la magnitud en que el protocolo es desacoplado del hardware: *Virtualización parcial* y *virtualización completa*. Las arquitecturas en ambos casos se muestran en la Figura 3.2.



a) Virtualización basada en Protocolo - Parcial

b) Virtualización basada en Protocolo - Completa

c) Virtualización basada en Espectro

Figura 3.2: Representación en bloques de la Virtualización basada en Protocolo: parcial y completa y Virtualización Basada en Espectro

En la virtualización inalámbrica basada en protocolo parcialmente, lo que se intenta es que la misma pila de protocolo sea compartida por varios *arrendatarios* cada uno con una configuración MAC o física diferente. Es común que en este tipo de implementaciones parámetros como la dirección MAC de los dispositivos de acceso inalámbrico puedan ser modificadas.

En el caso de la virtualización inalámbrica basada en protocolo completamente, se pueden tener varios protocolos sobre el mismo hardware de manera simultánea. Esta técnica se logra mediante la total separación de los protocolos inalámbricos del hardware de radio. Posteriormente, los recursos tanto del nivel MAC como Físico son entregados mediante un procedimiento de asignación temporizada, compartiendo la misma porción del espectro de frecuencias. De manera ideal se podría buscar que cada pila de protocolo implementada, utilice una porción de espectro, omitiendo el procedimiento de asignación temporizada, permitiendo que cada arrendatario utilice una porción de espectro diferente.

3.1.3. Virtualización de redes inalámbricas basada en Espectro e Interfaz RF

La virtualización inalámbrica basada en espectro e interfaz RF busca eliminar la restricción que tiene la virtualización inalámbrica basada en protocolo con respecto al espectro, esto es, permite la asignación de manera dinámica y la administración del espectro y de la interfaz de radio de los nodos [60] de comunicación.

En este caso los recursos que se virtualizan son el espectro de frecuencia y la interfaz RF, siendo la Radio Cognitiva una de las tecnologías en las cuales se emplea esta estrategia.

Para lograr las características de asignación dinámica del espectro, es necesario una función que perciba y mida el espectro para obtener la información que permita la administración de dicho recurso.

Cabe señalar, que las pilas de protocolos inalámbricos están desacopladas del nivel del espectro y de la interfaz RF, en este tipo de virtualización inalámbrica.

En la Figura 3.2 se representa el funcionamiento de la virtualización basada en espectro e interfaz RF.

Por lo expuesto, la posibilidad planteada en la virtualización de redes inalámbricas basadas en protocolo completamente, de que cada pila de protocolo utilice una porción del espectro de manera independiente, podría ser lograda mediante la virtualización de redes inalámbricas basada en espectro e interfaz RF.

3.2. Virtualización del Punto de Acceso

El surgimiento del concepto de redes inalámbricas virtuales, ha dado lugar a la realización de varios intentos para la virtualización de dispositivos en redes inalámbricas [15] [16] [61] , entre ellos el Punto de Acceso. Esta tarea, puede ser realizada de varias maneras, las cuales se presentan en esta sección junto con el concepto de VPA y su utilización.

3.2.1. Concepto de Punto de Acceso Virtual

Un VPA es una entidad lógica que reside dentro de un Punto de Acceso Físico [11] [13]. Un VPA opera transmitiendo en una misma trama Beacon, varios SSID, lo que da lugar a que las estaciones interpreten que existen varios PA independientes del punto de acceso físico. En la Figura 3.3, se puede apreciar el concepto de VPA.

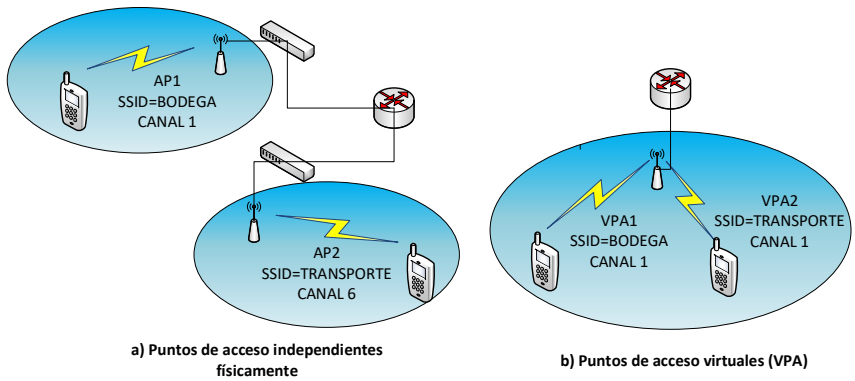


Figura 3.3: Puntos de acceso virtualizados

Existen varias técnicas que se pueden utilizar para crear un VPA, a partir de un punto de acceso físico. La técnica utilizada, depende de la topología de red que se vaya a implementar, pudiendo ser de Malla o en modo Infraestructura.

En el presente trabajo, se muestran las técnicas existentes para implementar VPA en modo infraestructura, las cuales pueden clasificarse en dos tipos: Virtualización de Punto de Acceso mediante MAC compartida y mediante Hipervisor.

3.2.2. Virtualización de Punto de Acceso mediante MAC Compartida

En esta técnica explicada en [12] y [13], el PA crea múltiples SSID y direcciones MAC para operar los VPA. A su vez, existen dos características del estándar IEEE 802.11 que pueden ser utilizadas para implementar un VPA mediante MAC compartida:

- Gestión de Ahorro de Energía (*PSM,* del inglés *Power Saving Management*).

- PCF.

En el caso de PSM, la virtualización del PA se lleva a cabo aprovechando los cambios en los modos de administración de energía de las estaciones inalámbricas que utilizan el PA, los cuales, según las especificaciones del nivel MAC del estándar IEEE 802.11 pueden ser: Modo activo y PSM.

El PA está pendiente del modo de cada estación. La estación informa al PA en qué modo se encuentra, el cual puede ser despierto (awake) o adormecido (dozen). La información del estado es transmitida al PA mediante una Petición de Ahorro de Energía transmitida en el bit de Administración de Energía de la trama de control, luego de lo cual la estación inalámbrica espera por una réplica de parte del PA para luego de esto entrar en modo PSM.

Durante el procedimiento de asociación, la estación informa al PA que está en período de escucha, el cuál es un período de tiempo en el cual

una estación en estado PSM podría optar por entrar en estado de "reposo".

En la técnica de virtualización basada en PSM, un punto de acceso físico utiliza la duración del período de reposo para desacoplar las configuraciones MAC de cada VPA, tales como canal, algoritmo back-off, políticas de calidad... Cabe señalar, que, en este tipo de implementación, no se puede soportar diferenciación entre varios servicios [12].

Por otro lado, la virtualización de PA mediante el uso de PCF, cuya operación se muestra en la Figura 3.4, otorga al PA, la característica de un Punto Coordinador (PC) y a las estaciones inalámbricas de esclavos.

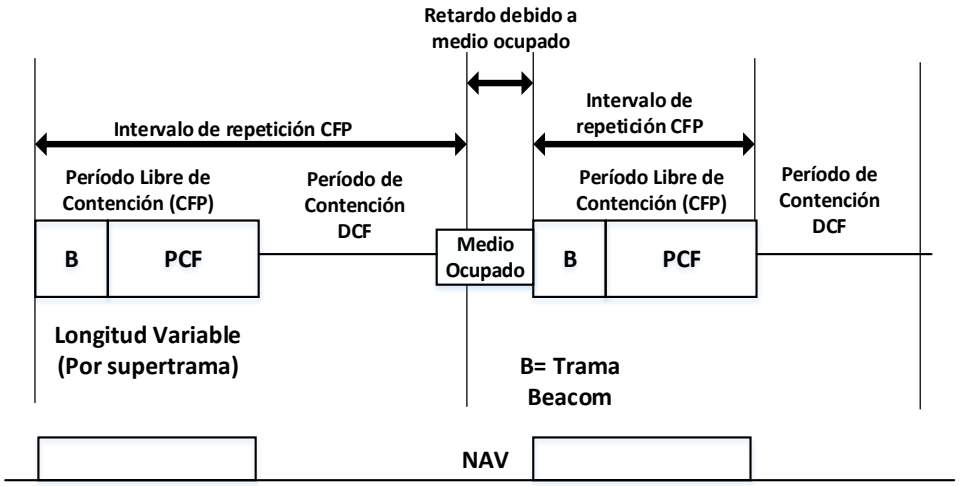


Figura 3.4: Función PCF

PCF utiliza una prioridad de acceso que el PC podría obtener usando un pequeño Espacio Inter Trama (EIT. Esta prioridad de acceso es usada por el PC para crear un período de Libre Contención (LC) en el cual se ejecuta el PCF. Un Período LC alterna con un Período de Contención (TC), en el cual está trabajando la Función de Coordinación Distribuida (*DCF*, del inglés *Distributed Coordination Function*). En el comienzo de cada período LC, el PC sensa el medio para saber si el medio está en estado idle, el Punto de Acceso espera un período PEIT para luego transmitir la trama Beacon conteniendo información usada por las estaciones inalámbricas para establecer sus temporizadores NAV. Luego, el PC obtiene el control

del medio y posterior a eso se prohíbe la utilización de DCF. Durante el período LC, el PC determina cual estación inalámbrica tiene el derecho de transmitir. Esto otorga un canal de acceso libre de contención a las estaciones individuales, agrupándolas por transmisiones, una a la vez.

Este control de acceso al medio centralizado, en el período libre de contención, puede ser explotado para soportar la virtualización de interfaces IEEE 802.11.

3.2.3. Virtualización de Punto de Acceso mediante Hipervisor

El hipervisor es un nivel responsable de administrar porciones de la infraestructura virtualizada [20]. Por otro lado, una máquina virtual es un computador ficticio basado en software que emula la arquitectura de un computador físico [12].

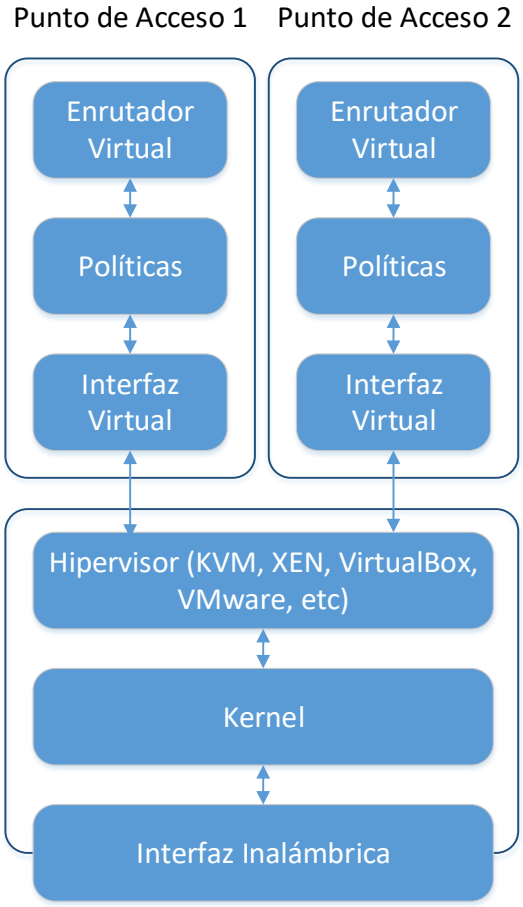


Figura 3.5: Virtualización de AP basada en Hipervisor

En el caso de la virtualización de PA mediante Hipervisor, el hipervisor inicia una máquina virtual sobe la máquina física para crear PA aislados utilizando una sola interfaz de red física. La virtualización basada en hipervisor divide un recurso físico en varios recursos lógicos y los asigna a cada VPA. En la Figura 3.5, se muestra la arquitectura de la Virtualización de Punto de Acceso basada en Hipervisor.

3.2.4. Utilización de Puntos de Acceso Virtuales

En [13] se resumen algunas de las posibles aplicaciones de la virtualización de PA. El *Conocimiento multicanal* puede ser logrado por un host que emplee una interfaz virtual para escanear periódicamente las características de todos los canales disponibles, mediante un escaneo en segundo plano, sin las pérdidas de asociación.

Se puede obtener una *Conectividad Simultánea* mediante un dispositivo inalámbrico que pueda ser conectado a varias redes simultáneamente; se podría conectar un host a un punto de acceso, al mismo tiempo que está en una red ad-hoc.

Podrían coexistir sobre el mismo banco de pruebas, varios experimentos de comunicación, mediante la multiplexación de facilidades inalámbricas dando lugar a la *Multiplexación de Experimentos*.

Se podría lograr una *Extensión o Retransmisión de la Cobertura de la Red* al emplear la virtualización de PA, para dotar de una segunda interfaz virtual en modo de PA a las estaciones móviles que son parte de una celda inalámbrica. Esto permitiría que los nodos que originalmente estaban fuera del rango del PA, se puedan conectarse a la red.

Mediante una *Red Inalámbrica en modo Híbrido*, una estación que esté conectada a una red ad hoc podría convertirse en una retransmisora de Internet, cuando esté en el rango un PA con conectividad a Internet.

Se podrían utilizar las interfaces virtuales, sobre canales ortogonales, para reducir la interferencia y de este modo *Incrementar la Capacidad* de una

red inalámbrica. Además, la capacidad de convertirse en un retransmisor bajo demanda, también puede aumentar la capacidad.

3.3. Virtualización de la Infraestructura

Antes de finalizar este capítulo, se presentan algunos temas adicionales, de interés para el despliegue de infraestructura inalámbrica virtualizada.

3.3.1. Virtualización de Interfaces Inalámbricas IEEE 802.11

Tal como se revisó en la Sección 2.6.1, el modo de operación básico de las estaciones inalámbricas IEEE 802.11, es el BSS.

En este modo de operación, las estaciones se conectan e intercambian información con los otros miembros del área de servicio básico a través de su interfaz de red inalámbrica. Por consiguiente, la topología de red que se puede implementar entre estaciones inalámbricas de manera inmediata, es la que se conoce como *De igual a igual* o en inglés *Peer to peer*. En consecuencia, es posible implementar redes del tipo *malla* o en inglés *Mesh*.

Sin embargo, algunas técnicas existentes en el campo de la virtualización de host, permiten convertir una interfaz inalámbrica de red común, en un PA inalámbrico, aprovechando las características en común que comparten, tanto las interfaces de red Ethernet que se utilizan en equipos host y las interfaces WiFi que se utilizan en dispositivos móviles, al ser en ambos casos dispositivos finales de red, lo que hace posible la aplicación de las técnicas mencionadas.

Las técnicas a las que nos referimos pueden ser implementadas mediante un enfoque hacia el hardware o el software, pudiéndose combinar también ambos enfoques según [63].

En el caso del enfoque basado en software, se necesita la intervención de máquinas virtuales e hipervisores para su implementación. Sin embargo, este requisito, hace complicada su implementación bajo el enfoque de software, por cuanto se requiere de grandes cantidades de recurso

computacional, lo que dificulta su despliegue en dispositivos móviles inteligentes.

Por otro lado, bajo el enfoque basado en hardware, se conocen al menos dos maneras de implementar la virtualización de interfaces inalámbricas IEEE 802.11: Virtualización de entrada/salida de una sola raíz (*SR-IOV*, del inglés Single *Root Input/Output Virtualization*) y la técnica *Ballesta* o en inglés *Crossbow* [20] la cual es una técnica propuesta por el grupo de Redes del Kernel Open Solaris de la empresa Oracle. En el presente estudio, se analiza la primera de ellas.

La técnica SR-IOV, es aplicada sobre dispositivos del tipo Interconexión de Componente Periférico expreso (*PCIe*, del inglés *Periphereal Component Interconnect express*), y se apoya en el uso de la Unidad de Administración de Memoria de Entrada/Salida (*IOMMU*, del inglés *Input/Output Memory Management Unit*) del dispositivo.

La IOMMU es la parte donde se realizan las operaciones de memoria de entrada/salida y traducción de direcciones, y no en el hipervisor, por cuanto el uso de este elemento durante la transferencia de datos es una de las mayores fuentes de degradación en virtualización de entrada/salida [20].

Otra característica que presenta SR-IOV, es la división del dispositivo PCIe en Funciones Virtuales (FV), las cuales son separadas de las Funciones Físicas (FF), siendo estas últimas, las funciones por defecto del dispositivo.

A cada FF, se le pueden asignar varias FVs, operando sobre distintas máquinas virtuales (*VM*, del inglés *Virtual Machine*), con muy pocas funcionalidades. Cada Función Física tiene su propio conjunto de recursos críticos para su desempeño, aunque, por otra parte, debe compartir los recursos generales de todo el dispositivo, tales como el procesamiento del nivel Físico y clasificación de paquetes. En la Figura 3.6, se puede apreciar la arquitectura de la técnica SR-IOV.

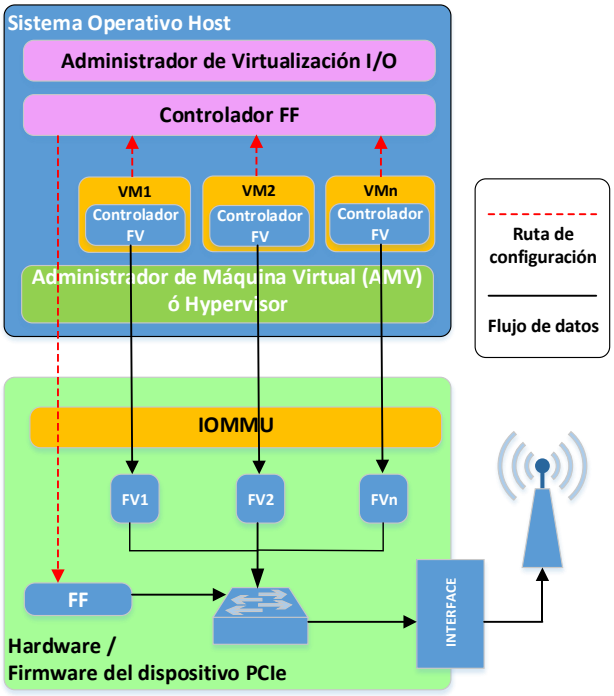


Figura 3.6: Arquitectura de la técnica SR-IOV

3.3.2. Visión completa de WiFi con infraestructura Virtualizada

Como ya se ha explicado, para lograr la virtualización de redes WiFi, es posible aplicar algunas técnicas utilizadas tanto en el ámbito de la virtualización de computadores, como así también, de técnicas del dominio de las redes WiFi.

Además, en el caso de las técnicas relacionadas a virtualización de computadores, es posible aplicar este tipo de técnicas en un ambiente inalámbrico, gracias a la similitud que existe entre las interfaces de redes cableadas e inalámbricas, por cuanto ambas son consideradas dispositivos de terminación de la red.

En el caso de las redes WiFi es factible su operación en modo infraestructura, y por ende su virtualización, gracias al uso de técnicas como PCF [13], la cual es predominante su uso en el modo de Infraestructura.

Sin embargo, de lo revisado hasta aquí, las redes WiFi virtualizadas también están sometidas a los mismos condicionamientos a los que son

sometidas las demás tecnologías inalámbricas existentes. Esto es, parámetros como el retraso (latencia), jitter, paquetes descartados y el throughput, son cruciales para mantener niveles aceptables de QoS y QoE, a la hora de desplegar redes WiFi virtualizadas en modo infraestructura para la transmisión de VoIP.

Por otra parte, el despliegue masivo de este paradigma denominado Virtualización de Redes Inalámbricas, depende en gran medida del aporte que brinden a ella las empresas fabricantes de hardware y para ello sería necesario demostrar la conveniencia de su implementación. Las argumentaciones para aquello se pueden revisar en la siguiente sección relacionada con los casos de uso de la Virtualización Redes Inalámbricas. Finalmente, en la Figura 3.7 se muestra de una manera general, una posible arquitectura para la implementación de una red WiFi virtualizada para transportar comunicaciones de VoIP, en base a los conceptos estudiados hasta aquí.

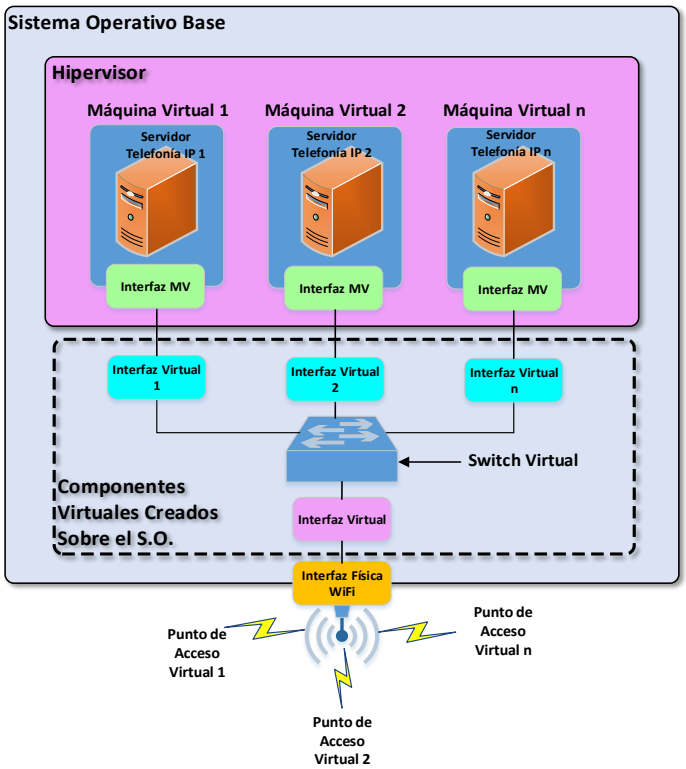


Figura 3.7: Arquitectura propuesta para el despliegue de una red WiFi para transportar VoIP

3.3.3. Casos de Uso

Plataformas para la realización de experimentos, como la plataforma GENI, sistemas de telefonía móvil de próxima generación como la Telefonía Celular 5G, tecnologías emergentes como el Internet de las Cosas (*IoT*, del inglés *Internet of Things*), son entre otros, algunos de los casos más relevantes que se pueden mencionar y en los cuales se está estudiando la incorporación de la virtualización de redes inalámbricas para el despliegue de las tecnologías mencionadas.

Plataformas Experimentales: ***El caso GENI***

El Ambiente Global para Innovaciones de Red (*GENI*, del inglés *Global Environment for Network Innovations*), es un laboratorio de pruebas a gran escala, desplegado en varios estados de los Estados Unidos de América, impulsado por la Fundación de Ciencia Nacional Americana. Está conformado por varias entidades universitarias, centros de investigación y empresas.

El objetivo de esta iniciativa, es el de facilitar la creación de escenarios experimentales para probar nuevas arquitecturas de red, en las que se incluyen dispositivos de acceso inalámbrico. Este objetivo es posible, mediante la creación de particiones (slides), los cuales pueden estar conformados a su vez, por distintos recursos, los cuales pueden estar ubicados en distintas localidades dentro del ambiente. Estas particiones son tratadas como “experimentos”. La creación de las particiones es posible por cuanto GENI tiene funcionalidades de programación profunda (por medio de SDN y virtualización), un sistema de instrumentación avanzada y características de seguridad avanzadas.

La plataforma GENI, mostrada en la Figura 3.8, posee en su núcleo, una red de Internet 2. Este ambiente está compuesto, además, por una serie de recursos computacionales y de telecomunicaciones situados en distintas localidades del país, los cuales pueden ser utilizados por los miembros que tienen acceso a GENI.

Uno de los sistemas sobre los cuales podría experimentarse en este ambiente, sería el de las comunicaciones de VoIP, aprovechando la posibilidad que otorga esta plataforma a sus participantes, de compartir recursos de redes inalámbricas entre ellos dispositivos IEEE 802.11 en un ambiente que se apoya en la virtualización de redes.

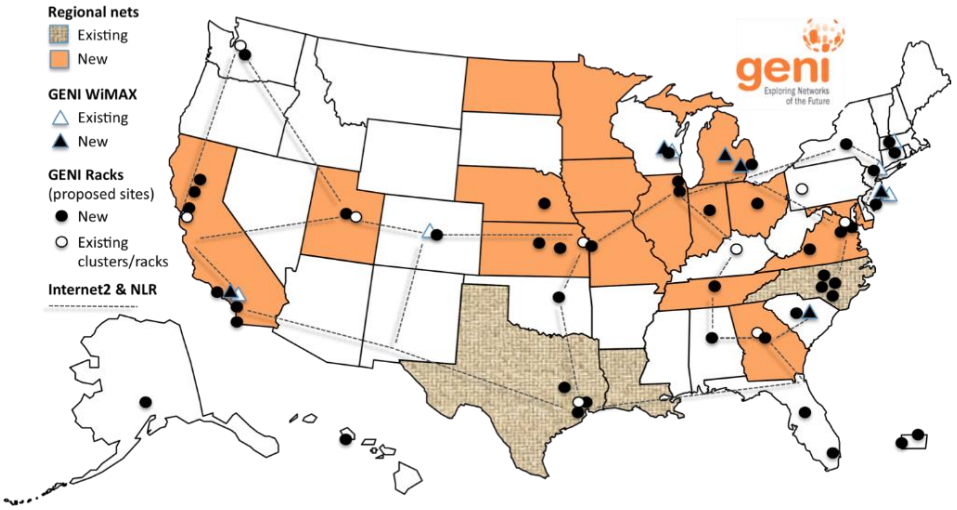


Figura 3.8: Plataforma GENI desplegada en territorio de los EE.UU.

Sistemas de Telefonía Móvil de Próxima Generación: ***Caso Telefonía Celular 5G***

Otro de los casos en los que la virtualización de redes inalámbricas podría encontrar una aplicación, es en el despliegue y operación de la Telefonía Celular 5G.

En estudios recientes como en [65] y [66] se proyecta la posibilidad de utilizar la virtualización de redes inalámbricas como tecnología habilitante para implementar redes inalámbricas 5G, para lo cual, se prevé aprovechar las características ya revisadas en capítulos anteriores del presente trabajo, tales como, la reutilización de recursos de infraestructura, en vista del aumento en la demanda de capacidad de los canales de transmisión de datos, prevista para la tecnología 5G.

Una de las metas que se han propuesto alcanzar junto con la implementación de las redes 5G es la densificación de las redes celulares

mediante la adición de pequeñas celdas de radio comunicación, las cuales podrían ser implementadas aplicando técnicas de virtualización similares a las estudiadas en el capítulo anterior, tales como SDN, VNF, y WNV.

En el caso de [65], realiza una explicación de cómo se podría representar a un alto nivel, la virtualización de una red celular utilizando componentes del modelo de negocio de las SDN, esto es, mediante MVNOs e InPs.

Mientras que en [66], los autores hacen énfasis en la utilización de SDN y los protocolos derivados de esta tecnología, tales como OpenFlow, para facilitar el despliegue y operación de las redes inalámbricas 5G.

Evidentemente, la VoIP será parte de la información que fluirá a través de las futuras redes 5G, de allí que se considera a la tecnología 5G como un caso de uso de la virtualización de la infraestructura.

Tecnologías emergentes: ***Uso de la virtualización en el Internet de las Cosas*** (***IoT***)

La irrupción del IoT está motivando investigaciones para determinar los requisitos que se deben cumplir para implementar de mejor manera esta prometedora tecnología.

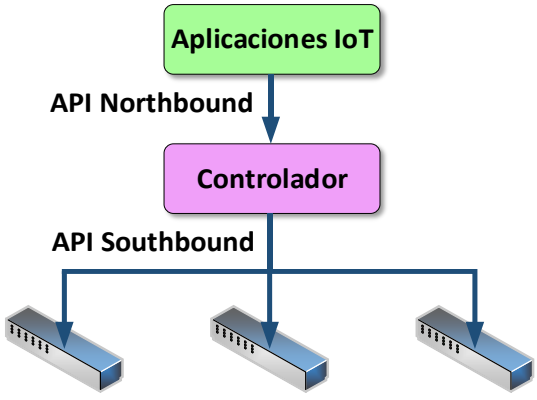


Figura 3.9: Arquitectura con SDN propuesta para IoT

En [67] vuelven a converger los conceptos de SDN y WNV, y los autores estudian una posible arquitectura, la cual se resume en la Figura 3.9 donde se utilizan elementos derivados de las SDN y la WNV. Esto con el fin de preparar la infraestructura existente para la llegada del IoT. Se

mencionan las bondades de las SDN que ya han sido revisadas en el presente trabajo entre ellas la posibilidad de separar la red en un plano de control y otro plano de datos. Se revisa la posibilidad de que los nodos miembros de la red de sensores, puedan servir a más de una red, dando lugar a sensores virtuales múltiples.

En resumen, la virtualización de redes inalámbricas podría ser considerada como una tecnología, que se puede prestar en la colaboración del despliegue de nuevas tecnologías.

CAPÍTULO 4

4. DISEÑO DEL DEMOSTRADOR, PRUEBAS A REALIZAR Y RECURSOS UTILIZADOS

En este capítulo se realiza una explicación de cómo se despliega el demostrador propuesto, empezando por definir las condiciones iniciales presentadas para su despliegue. Se presenta también el escenario que se implementa basado en técnicas de virtualización inalámbrica. Además, se indican las variables que se miden en función de las cuales se intenta establecer los niveles de QoS y QoE que se pueden presentar en un escenario como el propuesto.

También se detallan, los diferentes recursos de hardware y software que se utilizan para montar el escenario que nos permite demostrar la operación de la Virtualización de Redes inalámbricas.

4.1. Consideraciones Iniciales

En esta sección se muestran las condiciones que se presentan previo a la implementación del demostrador planteado.

4.1.1. Descripción del escenario propuesto

El escenario desplegado está compuesto por un computador portátil en el cual se encuentra instalado el sistema operativo Linux Ubuntu 16.0. Dentro de ese computador, y sobre un software de administración de máquinas virtuales, se crean dos servidores de Telefonía IP, que simulan por separado a los servidores de dos secciones de una empresa o que incluso pueden utilizarse para representar a los servidores de dos empresas distintas.

La ventaja de utilizar un computador portátil para desplegar el demostrador planteado, es que este equipo posee una tarjeta de red inalámbrica, la cual se utiliza para aplicar la virtualización de redes inalámbricas.

Se realiza la virtualización de los componentes de la red inalámbrica, haciendo uso de paquetes de software disponibles para el sistema operativo empleado.

Se emplean como dispositivos clientes, varios teléfonos celulares y al menos un computador portátil, que se conectan indistintamente a cualquiera de las redes inalámbricas mediante sus respectivas tarjetas de red WiFi.

La arquitectura que se implementa con el demostrador planteado, está basada en la arquitectura mostrada en la Figura 3.7. En este caso, está limitado a dos redes inalámbricas virtuales. La arquitectura implementada, utilizando virtualización, se muestra en la Figura 4.1.

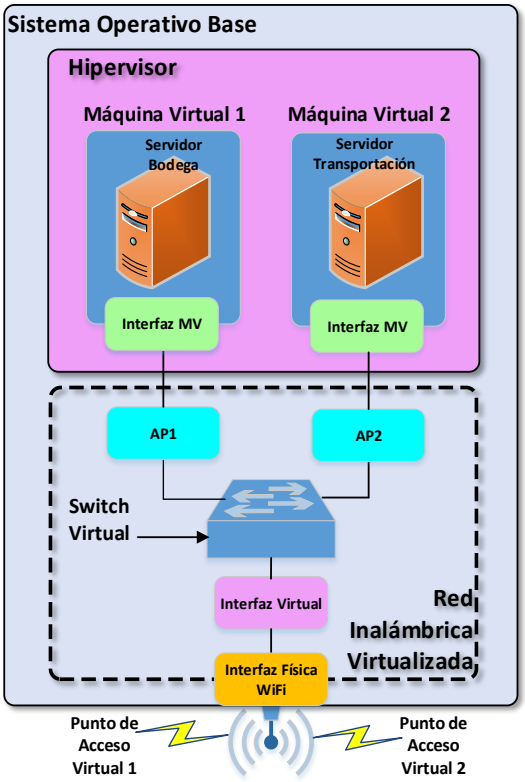


Figura 4.1: Arquitectura del Demostrador a implementar

4.1.2. Selección de variables a estudiar y pruebas a realizar

Las variables escogidas para su medición son: el retraso (latencia) de los paquetes, el jitter, la pérdida de paquetes y el throughput, en función de

las cuales se calculan los valores de QoE para cada prueba realizada. Las variables mencionadas han sido escogidas, en vista de su influencia en la QoS y QoE de las llamadas.

Según los conceptos revisados en la sección 2.6.4, las variables que se pretenden medir, tienen valores máximos tolerables pre establecidos, dentro de los cuales se pueden llevar a cabo las comunicaciones mediante VoIP, sin que se afecte la QoS, por lo que se espera que las mediciones que se obtengan durante las pruebas, estén por debajo de estos límites, de tal manera que se compruebe la viabilidad de emplear WNV en la realización de llamadas de telefonía IP sobre WiFi.

Los niveles de QoE son obtenidos mediante el parámetro MOS el que a su vez es calculado utilizando las ecuaciones 2.2, 2.3, 2.4 y 2.5, las cuales relacionan los parámetros de red como el Retraso Promedio, Jitter y Pérdida de Paquetes con el MOS. Como se examinó en la sección 2.6.4, el MOS presenta valores en una escala que va del 1 al 5, siendo el 1 el nivel más pobre de QoE y 5 el nivel más alto.

4.1.3. Condiciones y limitaciones del demostrador propuesto

Los equipos terminales inalámbricos en las pruebas que se realizan, no tienen mayor movilidad, prácticamente estarán fijos. No existen transiciones entre BSS, por lo que no se mide el impacto que las transiciones entre BSS, tienen sobre las variables medidas.

En el caso del Throughput, al utilizar el estándar *IEEE 802.11n*, se tiene disponible una tasa de Transmisión de 100Mbps, y por ende un Througput de 60Mbps aproximadamente en el canal.

No se ha desarrollado en este trabajo un software que automatice el despliegue del demostrador, por lo que la ejecución del demostrador, se la realiza siguiendo una secuencia de pasos de manera manual, los cuales se detallan en el Capítulo 5.

Los componentes de software del demostrador propuesto están basados en Software Libre, en vista de las facilidades en cuanto a la adquisición de los mismos, al no tener que pagar por su licenciamiento.

En el caso del software Zoiper que actúa como cliente de telefonía IP, se utiliza la versión gratuita la cual tiene una cantidad limitada de CODEC de audio que pueden ser utilizados en las llamadas telefónicas. Por esta razón, las pruebas se realizan utilizando los CODEC G.711 y GSM.

Para poder desplegar los PA sobre la tarjeta de red WiFi, la tarjeta de red que se vaya a virtualizar debe poseer la característica de Multiple SSID. Esta característica permite configurar varios PA en una tarjeta de red inalámbrica, cada uno con un SSID distinto. En el presente caso, la tarjeta de red WiFi del computador portátil sólo permite crear dos VPA, lo que limita la comprobación de resultados para mayores cantidades de redes.

4.2. Recursos utilizados en la implementación

A continuación, se detallan los recursos materiales y metodológicos utilizados para implementar el demostrador planteado.

4.2.1. Metodología a utilizar en la recolección de datos

Para verificar la factibilidad de utilizar virtualización de redes inalámbricas sobre redes WiFi que transporten telefonía IP, se comparan los resultados obtenidos al realizar las pruebas, en dos escenarios distintos: un escenario sin virtualización y otro con virtualización. Las mediciones se realizan capturando los paquetes de datos que fluyen a través de las

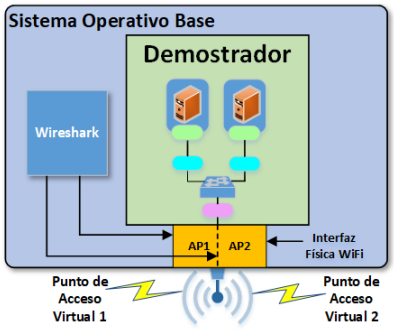


Figura 4.3: Medición de variables con WNV

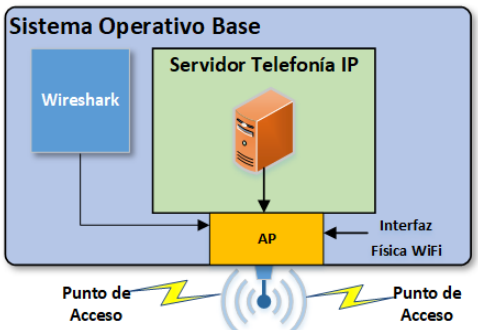


Figura 4.2: Medición de variables sin WNV

interfaces que actúan como PA en ambos casos, mediante el programa Wireshark, como se muestra en la Figura 4.3 y Figura 4.2.

Las llamadas durante las pruebas que se realizan, se mantienen activas por aproximadamente 5 minutos, con el fin de confirmar la estabilidad y continuidad de las conexiones.

En el caso del MOS, este se calcula aplicando las ecuaciones 2.2, 2.3, 2.4 y 2.5 de la sección 2.6.4. sobre los valores de Latencia, Jitter y Pérdida de Paquetes que se obtienen para cada paquete RTP en cada ejecución del demostrador sobre los diferentes escenarios (sin y con virtualización), valores que el Wireshark permite guardar en un archivo CSV.

El archivo CSV se lo procesa en una hoja de cálculo y de allí se obtienen, tanto los parámetros requeridos en las ecuaciones utilizadas como los gráficos de Correlación que se muestran en el capítulo 5. Esta forma de medir el MOS se realiza para agilizar su obtención, siendo una forma más objetiva de medición.

Los valores de MOS calculados de esta manera, se los compara con los valores que se muestran en los registros de llamadas realizadas en el softphone Zoiper utilizado para hacer las llamadas. La obtención del MOS mediante software se advierte en [68].

4.2.2. Selección de equipos utilizados en la implementación

El principal componente del demostrador planteado es la WNIC, en el caso del demostrador planteado, la del computador portátil que aloja la plataforma virtualizada o *Demostrador*.

Por cuanto la WNIC utilizada, permite la creación de múltiples VPA sobre si misma, se utiliza el sistema operativo Linux, ya que el proceso llamado *Hostapd*, que se emplea para la creación de los VPA, se instala y opera únicamente sobre este sistema operativo.

Además, al estar basado en software libre, Linux permite fácilmente la modificación y ejecución de diversos procesos internos tales como servicios de enrutamiento, conmutación virtual, virtualización, dhcp, …

Figura 4.5: Dispositivos Clientes

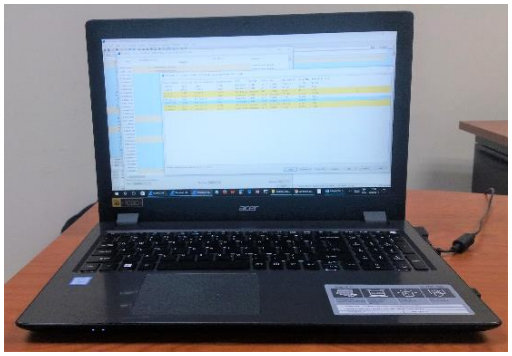


Figura 4.4: Equipo que aloja el Demostrador

4.2.3. Hardware y software empleados en la implementación

Hardware

En el caso del computador portátil, el cual se muestra en la Figura 4.4, se utiliza un equipo de las siguientes características:

Marca: Acer

Modelo: Aspire V3-575T

Sistema Operativo: Linux Ubuntu 16.0

Procesador: Intel Core I7-6500U CPU @ 2.50 GHz

Memoria RAM: 8GB

Disco Duro: 1 Tb

Tarjeta de red Inalámbrica: Qualcomm Atheros QCA61x4A

Los equipos que operan como estaciones inalámbricas son teléfonos celulares con sistema operativo Android. También se utiliza, al menos un computador portátil, con sistema operativo Windows, como cliente para realizar mediciones. Estos dispositivos se muestran en la Figura 4.5.

Software

Se utiliza como hipervisor para la creación y administración de máquinas virtuales, el programa VirtualBox (versión gratuita) de la empresa Solaris. Para el despliegue de los servidores de telefonía IP se emplea la

plataforma Elastix versión 2.5.0, la cual brinda prestaciones suficientes para las pruebas de VoIP que se pretenden realizar.

Existen dos paquetes que son instalados en el sistema operativo Linux llamados *iw* y *hostapd* para poder implementar la virtualización.

En el caso del paquete *iw*, esté nos permitie crear las interfaces virtuales sobre la tarjeta de red inalámbrica del computador portátil.

Por otro lado, *hostapd* nos da la posibilidad de configurar las interfaces virtuales creadas, para que operen en modo de Punto de Acceso.

Mientras que, en las estaciones inalámbricas se utiliza el programa Zoiper como cliente de telefonía IP.

4.2.4. Herramientas utilizadas para medición de parámetros de red

Las herramientas de medición que se utilizan para recolectar los valores de los parámetros han sido ya presentadas en la sección 2.6.3. Se emplea principalmente el Programa Wireshark para analizar el establecimiento de sesiones SIP y el intercambio de mensajes que se realizan.

Adicionalmente, el softphone Zoiper empleado para realizar las llamadas desde los dispositivos clientes, cuenta con un indicador del valor MOS de las conexiones, con el cual se puede comparar los resultados que se obtengan de las mediciones con el capturador de paquetes Wireshark.

CAPÍTULO 5

5. IMPLEMENTACIÓN Y RESULTADOS DE LAS PRUEBAS EXPERIMENTALES CON Y SIN WiFi VIRTUALIZADO PARA TELEFONÍA IP

En este capítulo, se explica el proceso de implementación del demostrador propuesto en esta investigación. Se describen los componentes necesarios y la forma en que son configurados, para luego dar paso a la revisión de las pruebas realizadas y los resultados obtenidos sobre el mismo.

5.1. Implementación de la red Inalámbrica Virtualizada

Para poder realizar las modificaciones y configuraciones necesarias para la creación de las interfaces inalámbricas virtuales, es necesario accesar mediante la aplicación *Terminal* del sistema operativo Linux Ubuntu, con privilegios de administración mediante el usuario *root*.

5.1.1. Virtualización de Interfaz Inalámbrica WiFi: creación de VPA

Antes de proceder con la creación de las interfaces virtuales, es necesario detener el servicio de administración de red del computador. Para esto se utiliza la orden *service nombre-servicio stop.*

root@srvmultissidvoip:/home/jmoreira# service network-manager stop

Seguidamente, se verifica la dirección MAC y el nombre interno que tienen inicialmente la interfaz WiFi, como se muestra en la Figura 5.1.

root@srvmultissidvoip:/home/jmoreira# iw dev

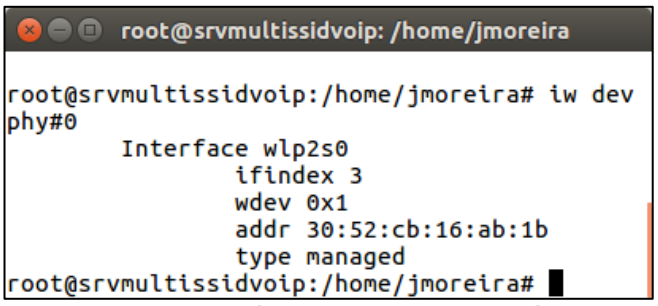


Figura 5.1: Dirección MAC de interfaz física WiFi

En este caso, el nombre interno de la interfaz WiFi es *wlp2s0* y la dirección MAC actual de la interfaz WiFi es la 30:52:cb:16:ab:1**b**.

A continuación, es necesario modificar la dirección MAC de la tarjeta de red WiFi o Interfaz Física, de tal modo que el primer octeto comenzando por el lado derecho de la MAC, tenga el valor de cero. Esto para que se pueden crear a partir de este valor, las demás interfaces virtuales inalámbricas con la orden *hostapd*. Antes de modificar la dirección MAC de la interfaz física, podría ser necesario desactivar la interfaz con la orden *ifconfig nombre de interfaz down*. Luego, se realiza la modificación de la dirección MAC, como se observa en la Figura 5.2.

root@srvmultissidvoip:/home/jmoreira# ifconfig wlp2s0 down

root@srvmultissidvoip:/home/jmoreira# ifconfig wlp2s0 hw ether 30:52:cb:16:ab:10

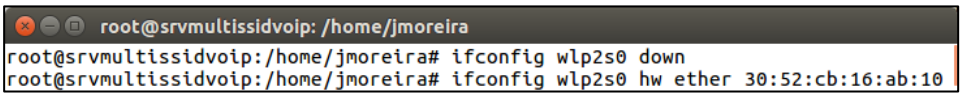


Figura 5.2: Modificación de Dirección MAC de interfaz WiFi

Luego de esto, se crean las interfaces virtuales, las cuales en este estudio se llaman AP1 y AP2, usando el paquete *iw*, y al mismo tiempo, se modifican las direcciones MAC de las interfaces virtuales creadas, como se demuestra en la Figura 5.3, las cuales deberán seguir la secuencia de la dirección MAC de la interfaz física.

*root@srvmultissidvoip:/home/jmoreira# iw dev wlp2s0 interface add **AP1** type managed*

*root@srvmultissidvoip:/home/jmoreira# ifconfig AP1 hw ether 30:52:cb:16:ab:1**1***

*root@srvmultissidvoip:/home/jmoreira# iw dev wlp2s0 interface add **AP2** type managed*

*root@srvmultissidvoip:/home/jmoreira# ifconfig AP2 hw ether 30:52:cb:16:ab:1**2***

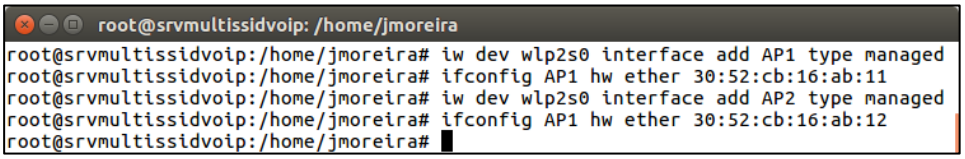


Figura 5.3: Creación de interfaces virtuales

Seguidamente, se verifican como quedan las direcciones MAC tanto de la Interfaz Física como de las interfaces virtuales creadas, con la orden *iw dev*, como se muestra en la Figura 5.4.

root@srvmultissidvoip:/home/jmoreira# iw dev

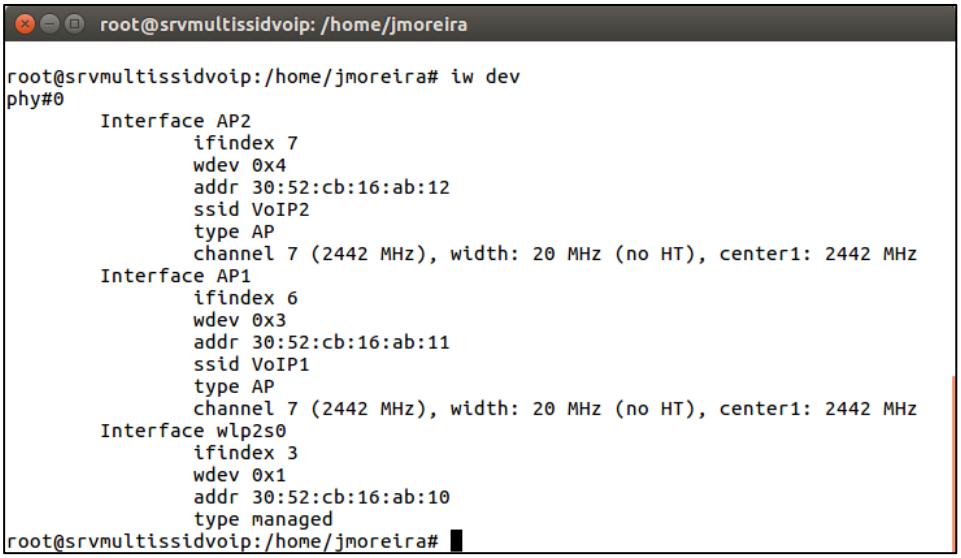


Figura 5.4: **Verificación de direcciones MAC de interfaces instaladas en equipo**

Luego, se preparan los archivos de configuración del paquete *hostapd* donde se configuran los diferentes parámetros con los cuales operan los dos VPA: AP1 y AP2. Esto es, los SSIDs, el canal, la versión del estándar IEEE 802.11 que utilizan, entre otros. Esto se hace utilizando la orden *gedit.*

root@srvmultissidvoip:/home/jmoreira# gedit /etc/hostapd/ hostapd.conf

El contenido de los archivos de configuración para cada Punto de Acceso, se muestra en la Figura 5.5 y Figura 5.6.

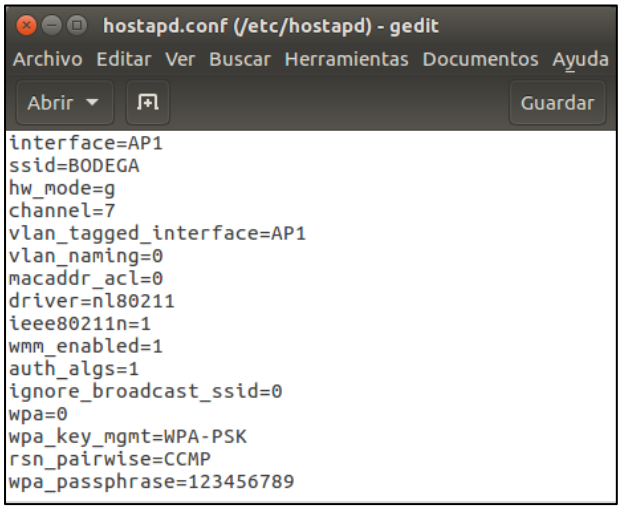


Figura 5.5: **Archivo de configuración VPA 1**

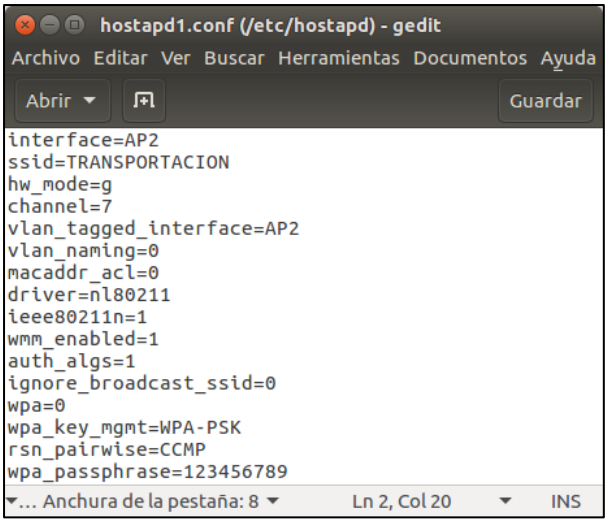


Figura 5.6: **Archivo de configuración VPA 2**

El significado de cada instrucción en el archivo *hostapd*, puede encontrarse en el *Apéndice I*.

Luego, se inicializan los VPA utilizando la orden *hostapd*, ejecutándose cada una de las líneas a continuación, en ventanas del *Terminal* distintas.

root@srvmultissidvoip:/home/jmoreira# hostapd /etc/hostapd/hostapd.conf

root@srvmultissidvoip:/home/jmoreira# hostapd /etc/hostapd/hostapd1.conf

Cuyo resultado se muestra en la Figura 5.7 y Figura 5.8.

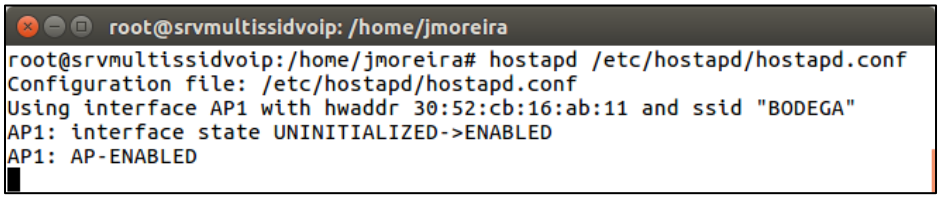


Figura 5.7: **Activación del VPA 1**

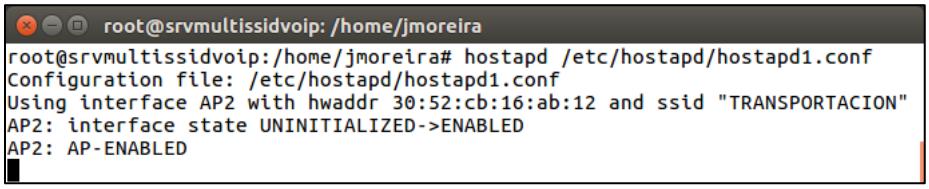


Figura 5.8: **Activación del VPA 2**

Posteriormente, se configuran las direcciones IP de cada VPA y se activa dichas interfaces utilizando la orden *ifconfig*.

root@srvmultissidvoip:/home/jmoreira# ifconfig AP1 192.168.0.1 up

root@srvmultissidvoip:/home/jmoreira# ifconfig AP2 10.0.0.1 up

Se procede a reiniciar el servicio de red del sistema.

root@srvmultissidvoip:/home/jmoreira#/etc/init.d/ networking restart

Finalmente, como se muestra en la Figura 5.9, se reinicia el servicio de provisión de direcciones de red dinámicas.

root@srvmultissidvoip:/home/jmoreira# /etc/init.d/isc-dhcp-server restart

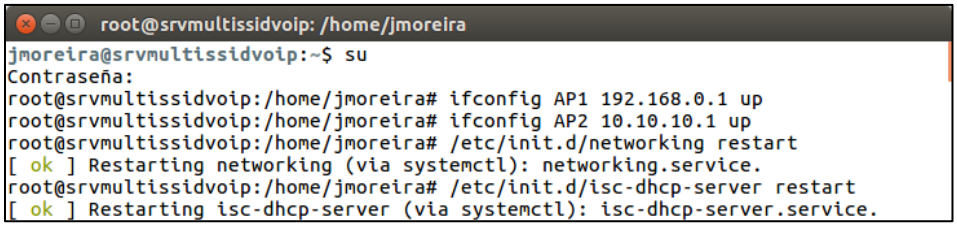


Figura 5.9: **Direccionamiento de Interfaces Virtuales creadas y reinicio de servicios de red**

5.1.2. Instalación y configuración de Servidores de Telefonía IP virtuales

La instalación de los servidores de Telefonía IP, se la realizó sobre el administrador de máquinas virtuales *VirtualBox*, el cual se muestra en la Figura 5.10.

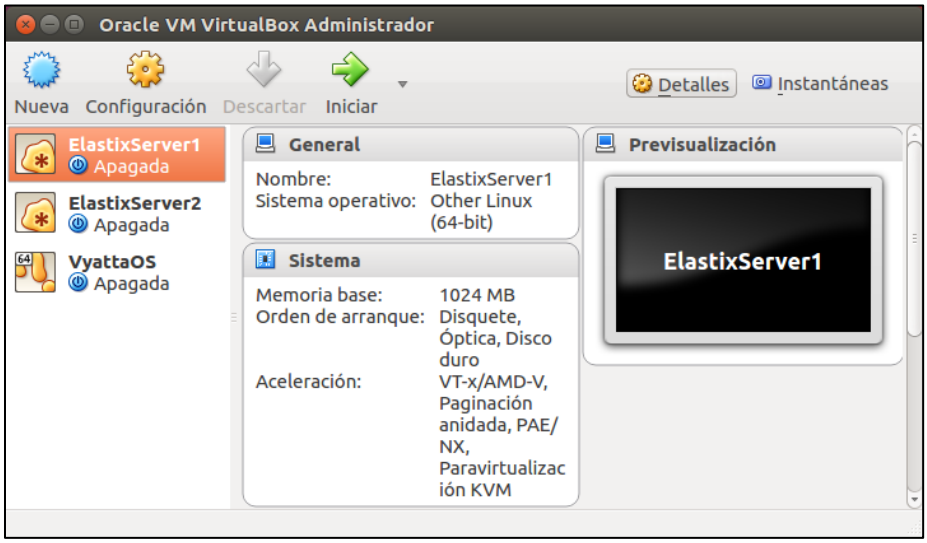


Figura 5.10: **Pantalla principal de Virtual Box**

Se crearon dos máquinas virtuales llamadas *ElastixServer1* y *ElastixServer2*, utilizando para ello como se muestra en la Figura 5.11, el archivo imágen *Elastix-2.5.0-STABLE-x86_64-bin-08may2015.iso*, el cual contiene el instalador de la plataforma de telefonía Elastix.

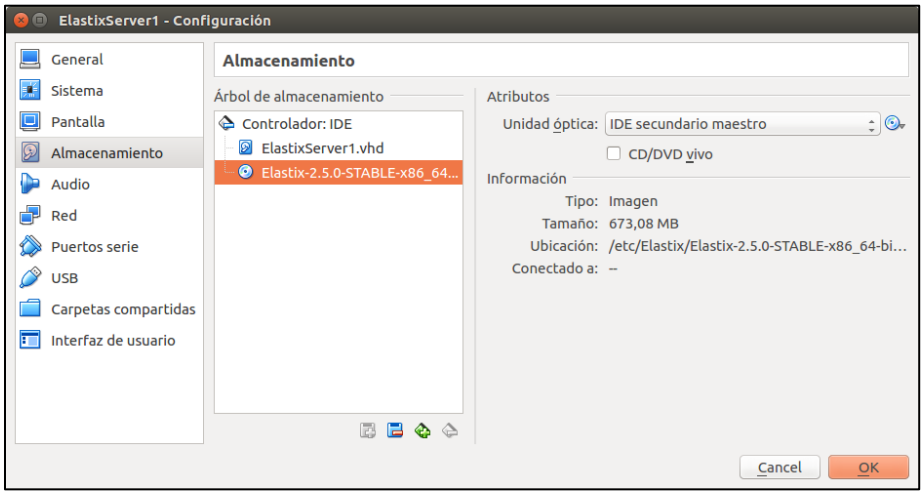


Figura 5.11: **Selección de archivo de imágen de Plataforma Elastix**

El proceso de puesta en marcha de los servidores de telefonía IP que se describe a continuación, es prácticamente el mismo en ambos servidores, variando únicamente los nombres de los servidores y el direccionamiento IP en ambos casos.

Los dispositivos de red de las máquinas virtuales creadas, deben configurarse en modo *Bridge* con las interfaces que conectan los servidores de telefonía IP con el exterior.

Inicialmente para probar la conectividad entre el servidor y los dispositivos terminales, se podrían utilizar las interfaces virtuales creadas AP1 y AP2. Adicionalmente, se escoge la opción *Permitir todo* en el *Modo Promiscuo*, como se muestra en la Figura 5.12.

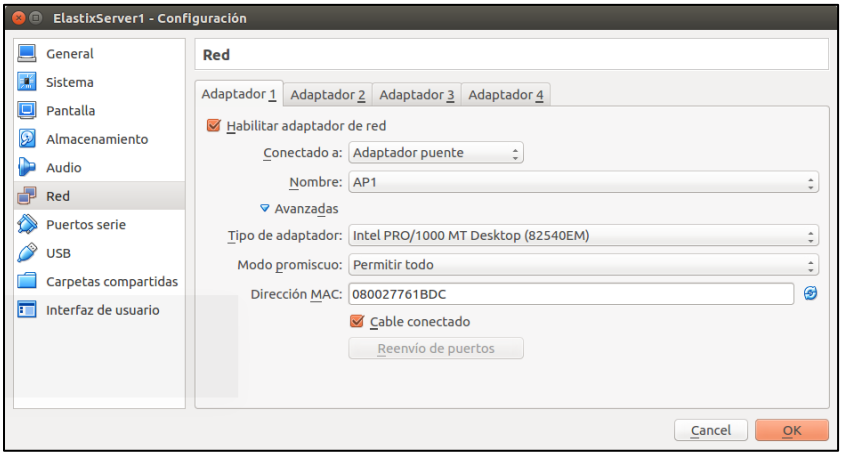


Figura 5.12: **Configuración de interfaz de red para máquina virtual**

Luego de esto, el proceso de instalación de Elastix bajo VirtualBox, es similar al de cualquier sistema operativo y se inicia con la pantalla mostrada en la Figura 5.13.

Figura 5.13: **Asignación de nombre a máquina virtual**

Se asigna a ambos servidores, una memoria virtual de 1024Mb para poder funcionar con un rendimiento óptimo, como se muestra en la Figura 5.14.

Figura 5.14: **Asignación de memoria a máquina virtual**

Luego, en la selección del tipo de archivo de Disco Duro, se selecciona el tipo genérico *VHD* de tal manera que sea posible la utilización del archivo de disco duro en cualquier administrador de máquinas virtuales, como se observa en la Figura 5.15.

Figura 5.15: **Elección de Disco Duro Virtual**

Seguidamente se escoge el modo de asignación de espacio en disco como se muestra en la Figura 5.16. Para el caso de este estudio, se escoge el modo *Dinámico* a fin de optimizar el uso del espacio que se toma del sistema operativo anfitrión.

Figura 5.16: **Asignación de unidad de disco duro físico**

Posteriormente, se asigna la cantidad de espacio en disco que se requiere para el almacenamiento de la plataforma Elastix, como se observa en la Figura 5.17. Para el caso del presente estudio, se utilizan 20Gb para trabajar sin inconvenientes. Para finalizar se crea la máquina virtual con los parámetros definidos, para alojar la plataforma Elastix.

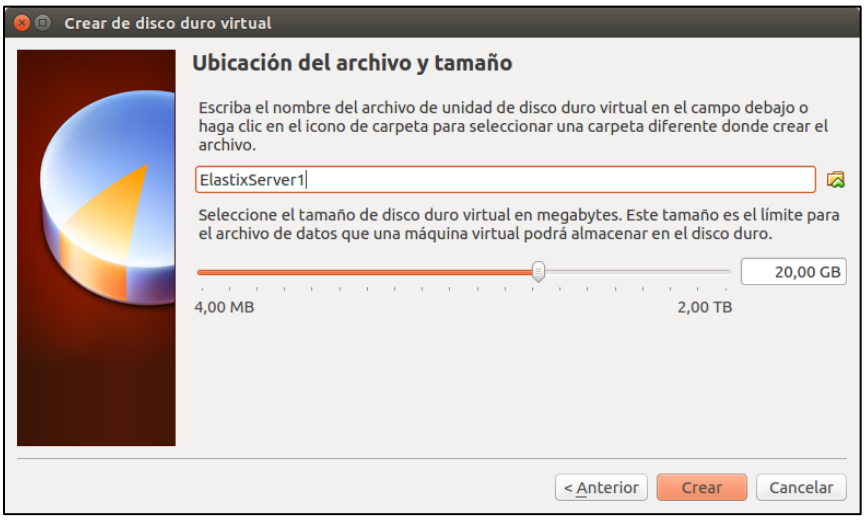


Figura 5.17: **Asignación de espacio en disco**

Al empezar el proceso de creación de la máquina virtual, se solicita confirmar la ubicación del Disco de Instalación de la Plataforma Elastix, como se muestra en la Figura 5.18.

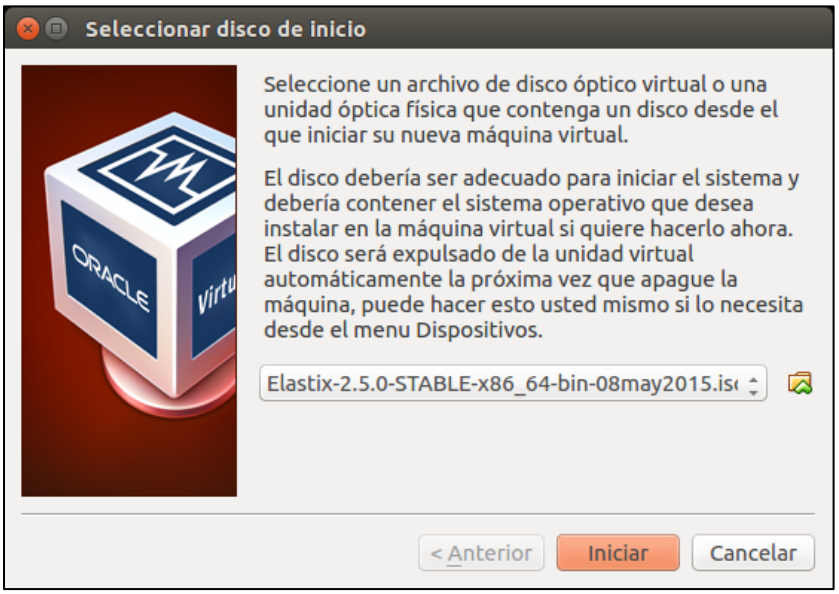


Figura 5.18: **Selección de ubicación de Disco de Instalación de Servidor Elastix**

Con esto empieza el proceso de instalación de la plataforma Elastix en la máquina virtual creada, como se muestra en la Figura 5.19.

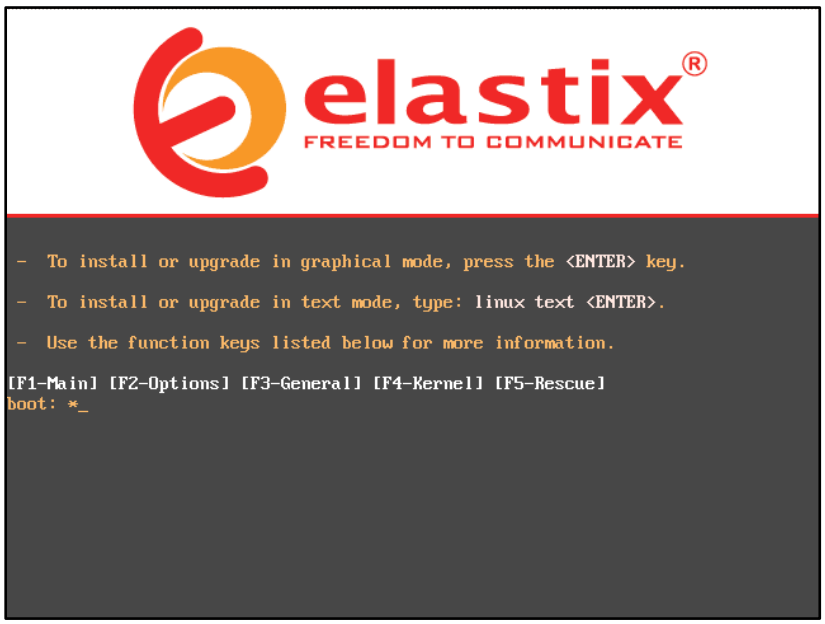


Figura 5.19: **Arranque del instalador del Servidor Elastix**

Inicialmente, se configuran los parámetros básicos como idioma de la interfaz de instalación e idioma del teclado. Posterior a esto, se procede a formatear el disco virtual creado, usando para ello la opción Utilizar espacio disponible *en dispositivos seleccionados y crear diseño predeterminado*, y se confirma la cantidad de espacio en disco necesario. Para este demostrador, los 20GB, como se observa en la Figura 5.20.

Figura 5.20: **Preparación del disco duro**

Luego se pide confirmar si se desea borrar la información y proceder con el formateo del disco. Se da la opción de modificar el esquema de particiones del disco. En este punto no es necesario modificar dichas particiones, por lo que se procede al formateo con las particiones sugeridas por defecto en el instalador, como se observa en la Figura 5.21.

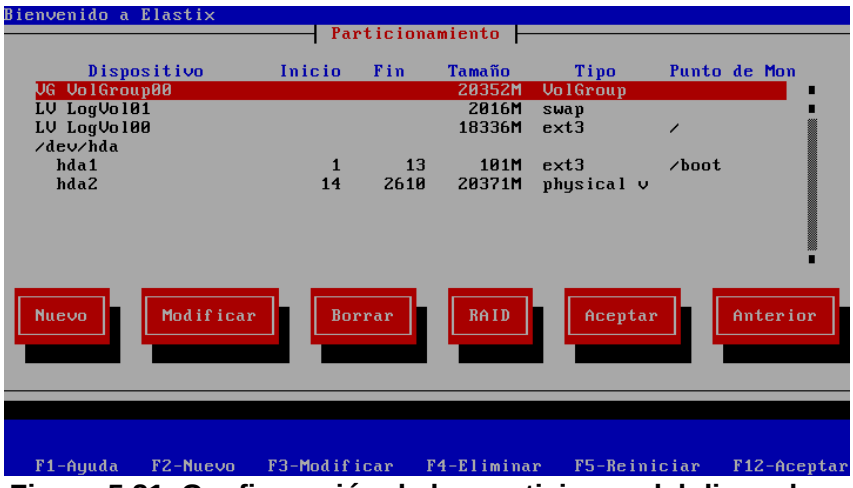


Figura 5.21: **Configuración de las particiones del disco duro**

Luego de esto, se pide configurar los parámetros de red del servidor de telefonía Elastix, como se muestra en la Figura 5.22. Se puede aprovechar este punto, para introducir las direcciones IP del servidor, máscara de sub-red, direcciones IP del gateway, DNS y el nombre que va a tener el servidor en la red.

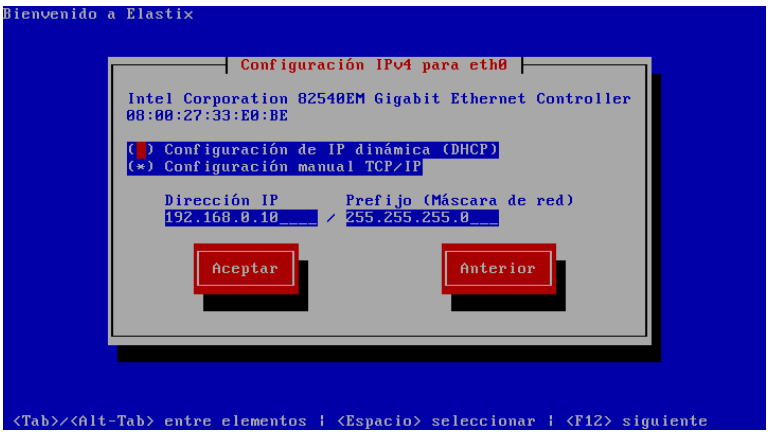


Figura 5.22: **Configuración de parámetros de red del servidor**

Por el momento, no se configuran servidores DNS ya que, para las pruebas a realizar, no son necesarios, según se observa en la Figura 5.23.

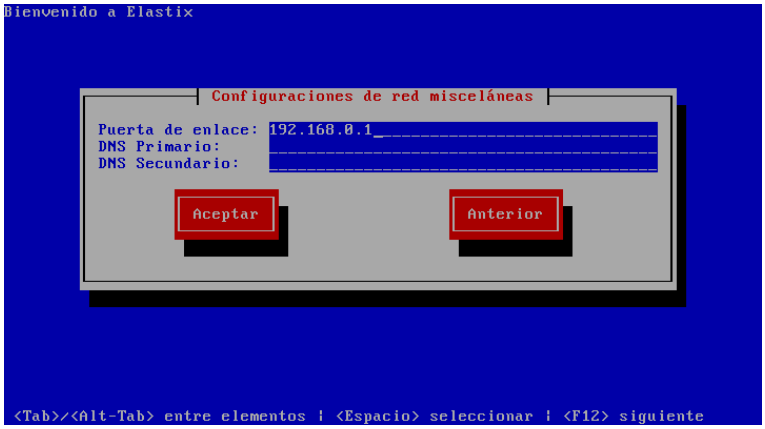


Figura 5.23: **Configuración de servidores DNS**

A continuación, se configura el nombre de cada servidor, como se muestra en la Figura 5.24. Se ha optado por llamarlos multissid1 y multissid2, en alusión a la característica MultiSSID que tiene la tarjeta WiFi utilizada en este demostrador.

Figura 5.24: **Configuración nombre del servidor**

Luego se selecciona la zona horaria y posteriormente se configura la contraseña de root, según se muestra en la Figura 5.25.

Figura 5.25: **Configuración de contraseña de root**

Una vez ingresada la contraseña y dado un click en Aceptar, arranca el proceso de instalación de los paquetes. Al finalizar este proceso, y al inicializarse los servicios del servidor, se pedirá la clave de root de la base de datos MySQL que utiliza el servidor, como se aprecia en la Figura 5.26. Además, pedirá se reconfirme la contraseña.

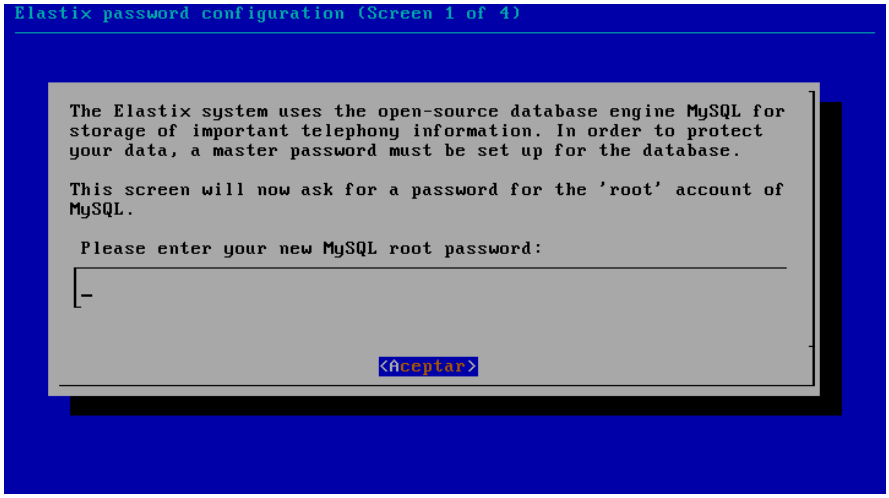


Figura 5.26: **Configuración de contraseña MySQL**

Luego, el asistente de instalación solicita la contraseña de la interfaz web del servidor Elastix, que será también usada como contraseña del servidor Free PBX sobre el cual está basado Elastix, como se aprecia en la Figura 5.27. Además, pedirá se reconfirme la contraseña.

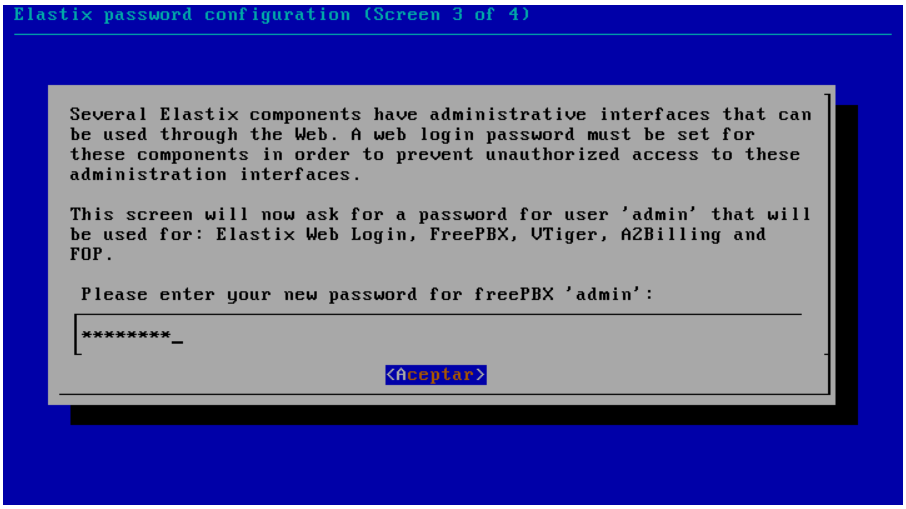


Figura 5.27: **Configuración contraseña Free PBX e Interfaz Web**

Luego de esto, finalmente queda instalado el servidor de telefonía IP, para accesar desde la interfaz de línea de comandos. Posterior a esto, podría ser necesario configurar desde la línea de comandos, nuevamente la dirección IP del servidor y reiniciar el servidor, para que los cambios surtan efecto y el servidor sea accesible en la red. La configuración de la

dirección IP por la interfaz de línea de comandos se realiza con la orden *ifconfig*.

[root@multissid1 ~]# ifconfig eth0 192.168.0.10 up

Si se desea confirmar los parámetros de red de servidor, desde la línea de comandos se puede utilizar la orden *setup*, el cual muestra un menú para realizar dicha verificación, como se observa en la Figura 5.28.

Figura 5.28: **Setup para verificación de parámetros de red del servidor Elastix**

Con esto, el servidor se encuentra listo para ser accedido ya sea desde la interfaz de línea de comandos o desde la interfaz web.

5.1.3. Segmentación de la Red mediante Conmutador Virtual y VLANs

Con el fin de introducir aislamiento entre los flujos de datos que se transmiten por la interfaz física Wifi como medida de seguridad, es conveniente implementar algún tipo de mecanismo que segmente las rutas que siguen los paquetes por la red.

Para lograr este propósito, se puede recurrir a la utilización de conmutadores virtuales junto con VLANs como parte del proceso de virtualización de la red inalámbrica.

El proceso de implementación, empieza de manera similar al realizado en la sección 5.1.1, donde se crean los VPA, AP1 y AP2.

Sin embargo, en el esquema propuesto ahora, se incorporan dos nuevos dispositivos *VSW1* y *VSW2*, llamados Conmutadores Virtuales, creados a partir del paquete Open Virtual Switch. Se incluye también, la utilización de dos interfaces virtuales de tipo TAP, las cuales simulan un cable físico, y que se las utiliza para conectar los conmutadores virtuales a las máquinas virtuales donde están alojados los servidores de telefonía IP.

La creación de los conmutadores virtuales *VSW1* y *VSW2* se lo realiza desde la consola de *Terminal*, usando la orden *add-br*.

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-br VSW1

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-br VSW2

Luego, se deben asignar las interfaces virtuales AP1 y AP2 a los conmutadores VSW1 y VSW2 respectivamente, con la orden *add-port*.

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-port VSW1 AP1 tag=10

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-port VSW2 AP2 tag=11

En las ordenes anteriores, mediante la orden *tag*, se definen las VLANs a las cuales pertenecerán los puertos que están siendo añadidos a los conmutadores.

Luego de esto, se crean dentro del sistema operativo anfitrión, las dos interfaces tipo TAP mencionadas anteriormente, a las cuales se las llama *vnet10* y *vnet11*.

root@srvmultissidvoip:/home/jmoreira# ip tuntap add mode tap vnet10

root@srvmultissidvoip:/home/jmoreira# ip tuntap add mode tap vnet11

Luego de esto, las interfaces que se encuentran creadas en este punto, pueden ser visualizadas como se muestra en la Figura 5.29.
root@srvmultissidvoip: /home/jmoreira# ifconfig | egrep dirección
Para poder configurarlas correctamente, se deben levantar las interfaces TAP.
root@srvmultissidvoip:/home/jmoreira# ifconfig vnet11 up
root@srvmultissidvoip:/home/jmoreira# ifconfig vnet10 up

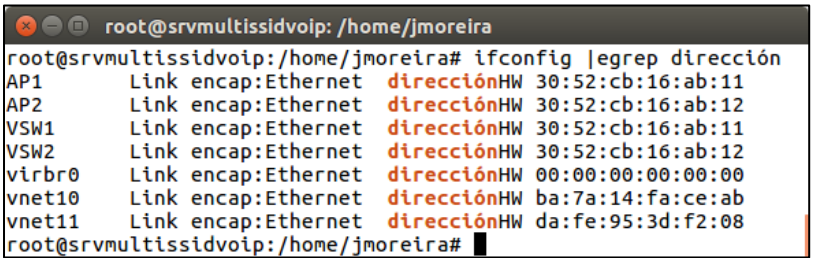


Figura 5.29: **Verificación de las interfaces creadas en el sistema operativo anfitrión**

Y luego, se añaden las interfaces TAP a los conmutadores VSW1 y VSW2.

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-port VSW1 vnet10 tag=10

root@srvmultissidvoip:/home/jmoreira# ovs-vsctl add-port VSW2 vnet11 tag=11

Seguidamente, es conveniente establecer las interfaces tipo TAP creadas, como *internas*.

root@srvmultissidvoip:/home/jmoreira# set interface vnet10 type=internal

root@srvmultissidvoip:/home/jmoreira# set interface vnet11 type=internal

En este punto, los conmutadores virtuales, internamente se encuentran configurados con las interfaces que se muestra en la Figura 5.30.

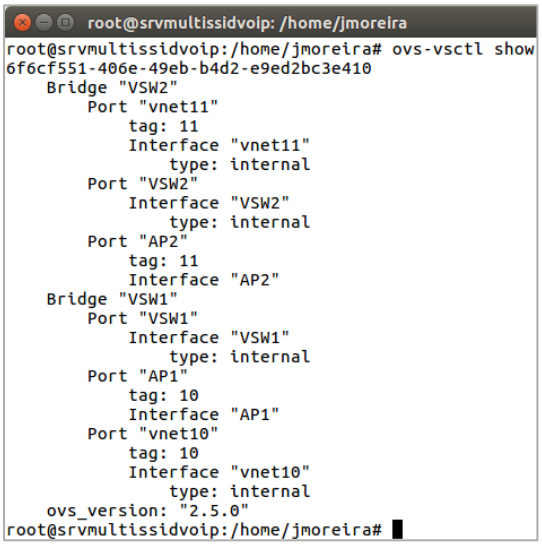


Figura 5.30: **Visualización de configuración de conmutadores virtuales**

A continuación, se configuran las direcciones IP tanto de los conmutadores virtuales como de las interfaces TAP, con las siguientes ordenes:

root@srvmultissidvoip:/home/jmoreira# ifconfig vnet10 192.168.0.9 netmask 255.255.255.0

root@srvmultissidvoip:/home/jmoreira# ifconfig VSW1 192.168.0.5 netmask 255.255.255.0

root@srvmultissidvoip:/home/jmoreira# ifconfig vnet11 10.0.0.9 netmask 255.0.0.0

root@srvmultissidvoip:/home/jmoreira# ifconfig VSW2 10.0.0.5 netmask 255.0.0.0

Finalmente, para asegurarse que los paquetes sean reenviados a través de la interfaz que les corresponde, para lo cual se deben añadir a la tabla de ruteo del servidor anfitrión, como se aprecia en la Figura 5.31, las rutas que se muestra a continuación:

root@srvmultissidvoip:/home/jmoreira# route add -net 192.168.0.0 netmask 255.255.255.0 AP1

root@srvmultissidvoip:/home/jmoreira# route add -net 10.0.0.0 netmask 255.0.0.0 AP2

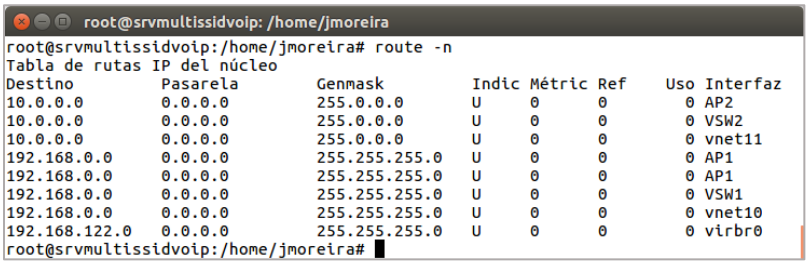


Figura 5.31: **Visualización de las rutas por defecto para las redes creadas**

Es importante señalar que, en este esquema, no se puede utilizar el método de autenticación WPA2, por cuanto los drivers de las tarjetas wireless junto con el paquete *hostapd*, entran en conflicto imposibilitando la operación del esquema.

Un esquema de la arquitectura final del demostrador que se despliega con el presente trabajo, se muestra en la Figura 5.32.

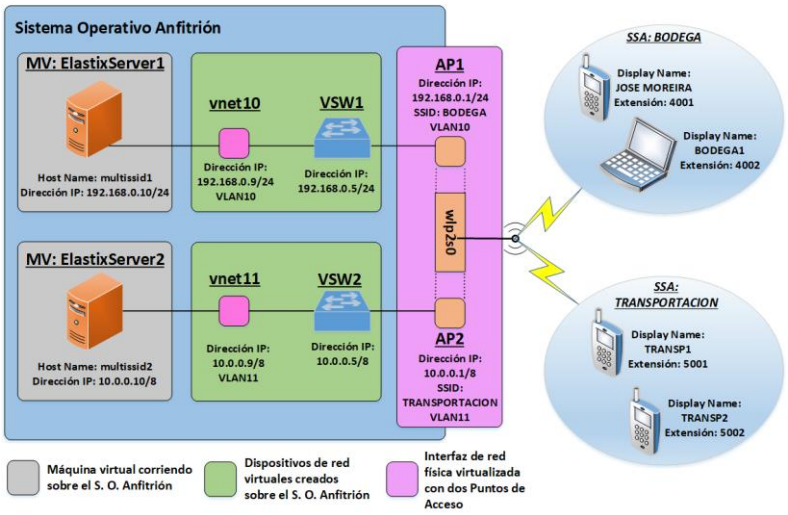


Figura 5.32: **Esquema propuesto para implementación del demostrador propuesto**

5.1.4. Instalación y configuración de Servidor y Clientes de Telefonía IP

En esta parte, se revisa la configuración del servidor de telefonía IP, con las extensiones, claves y demás parámetros para la comunicación entre las terminales. Es necesario conectar un equipo, ya sea un computador portátil o un teléfono móvil, que cuente con navegador web, a la misma red de cada servidor de telefonía IP, como se muestra en la Figura 5.33.

Figura 5.33: Conexión a misma red de servidor de telefonía IP

Mediante el navegador se debe dirigir a la dirección IP de cada servidor.

- ElastixServer1: multissid1: 192.168.0.10

- ElastixServer2: multissid2: 10.0.0.10

Donde aparecerán en primera instancia, las pantallas de inicio de sesión, como se muestra en la Figura 5.35 y Figura 5.34.

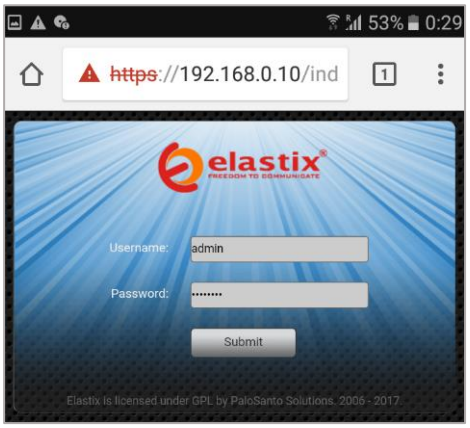


Figura 5.35: Inicio de sesión servidor multissid1

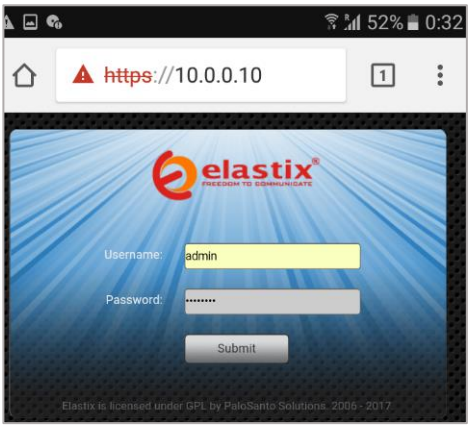


Figura 5.34: Inicio de sesión servidor multissid2

Luego, en las pantallas que se muestran en la Figura 5.36 y Figura 5.37, se escoge la opción "*PBX*".

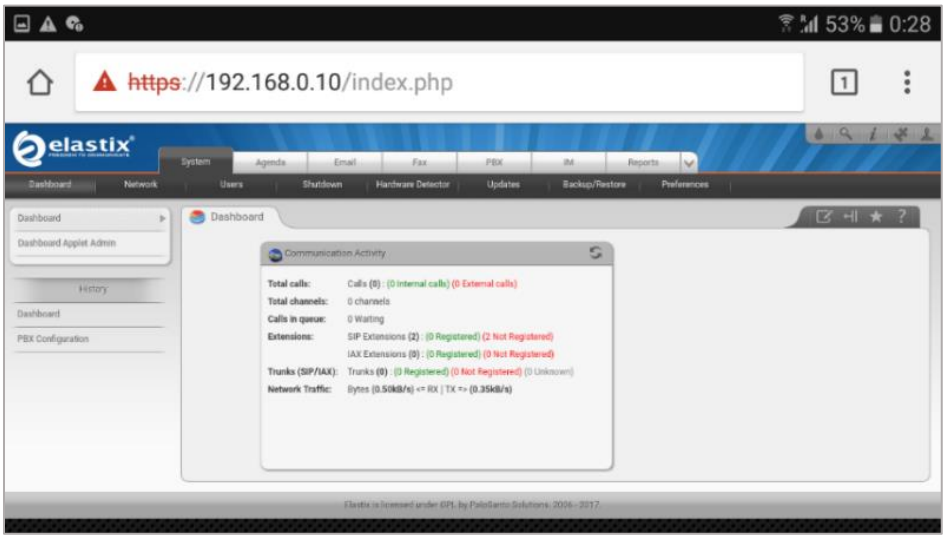


Figura 5.36: Pantalla principal servidor multisssid1

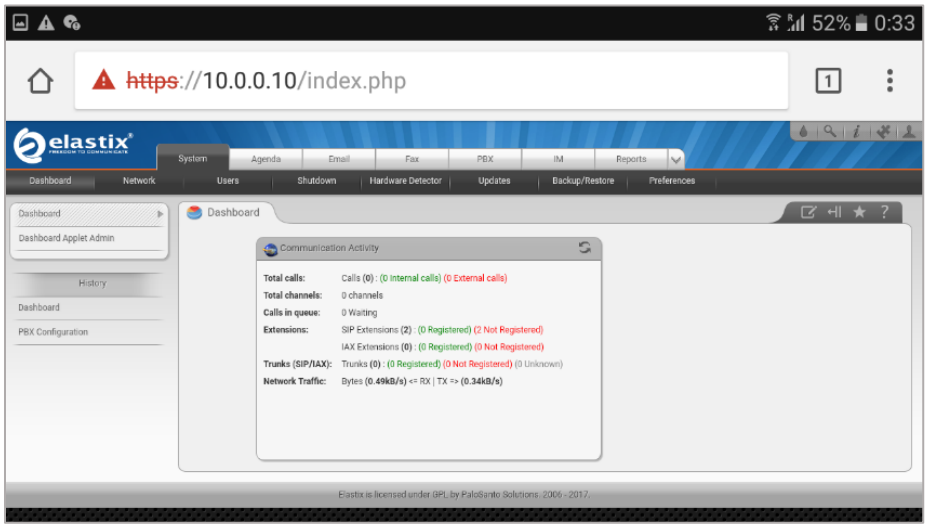


Figura 5.37: Pantalla principal servidor multisssid2

En ambos servidores, se muestra el módulo para añadir extensiones telefónicas, como se muestra en la Figura 5.38. En la pantalla que aparece, se da click en el botón "*Submit*", y luego se presenta la pantalla de la Figura 5.39.

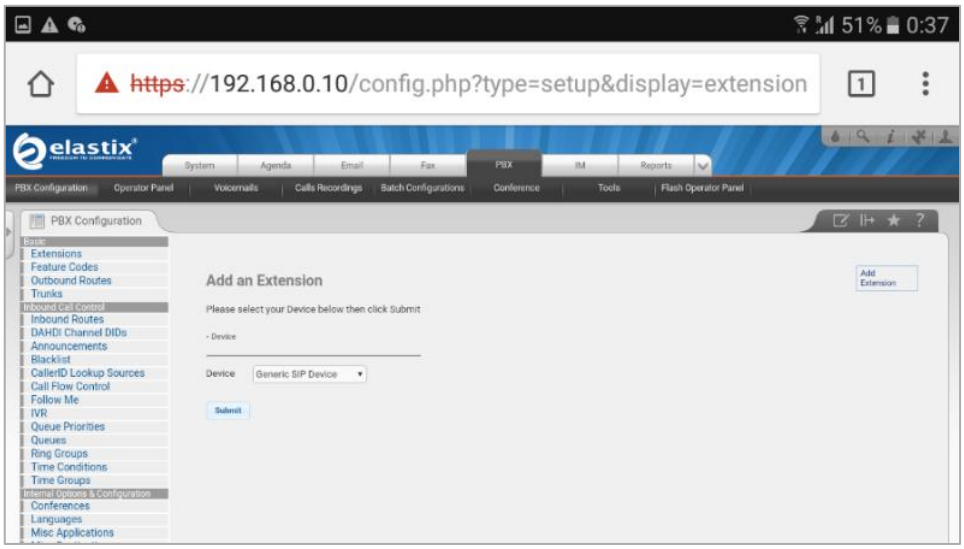


Figura 5.38: Módulo para añadir extensiones

En la pantalla de la Figura 5.39, se ingresan los datos de la extensión SIP que se crea para ser utilizada por cada equipo cliente.

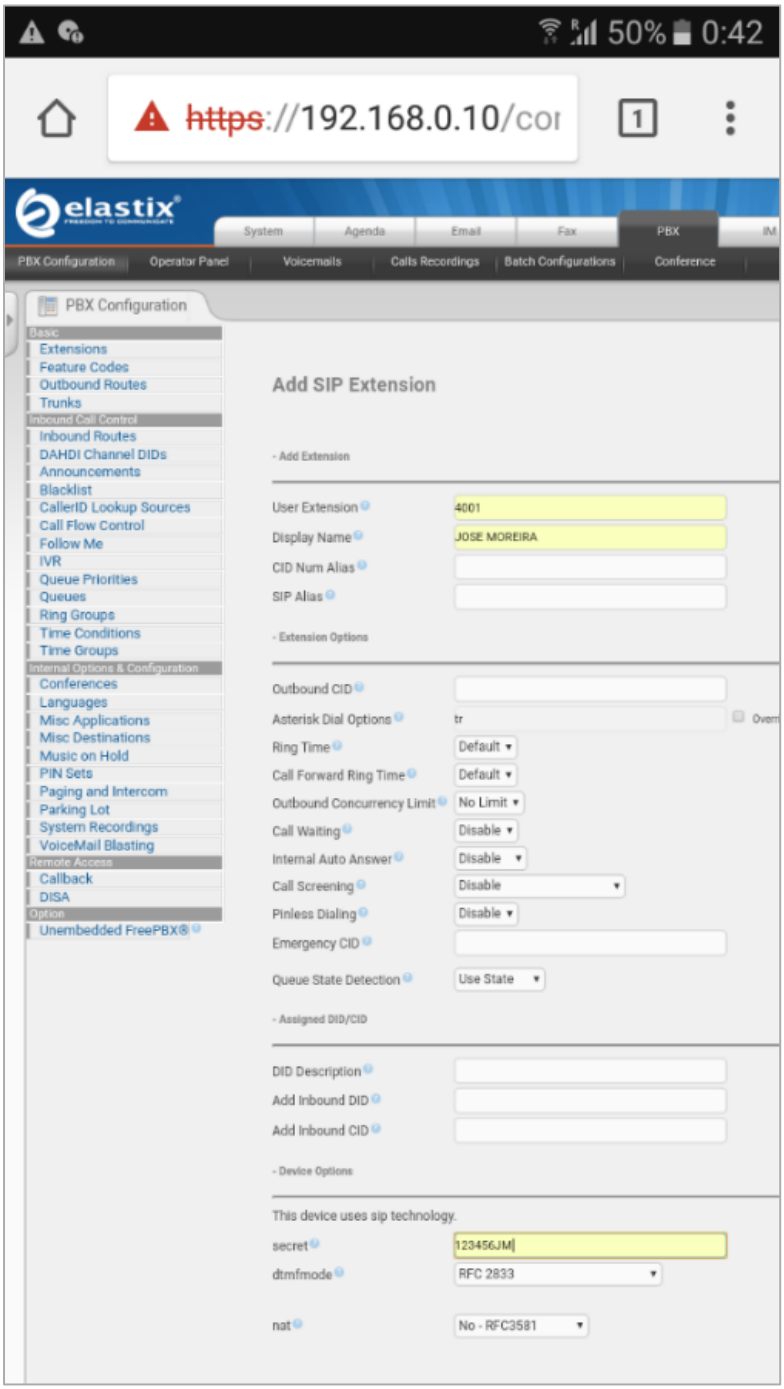


Figura 5.39: Parámetros de la extensión

Como se muestra en la figura anterior, los campos que se necesitan ingresar son: Extensión de usuario (User Extension), Display Name (Nombre a Mostrar) y secret (Contraseña).

Luego de ingresar los datos requeridos, se da click en el botón submit que se encuentra en la parte inferior de la pantalla, como se aprecia en la Figura 5.40.

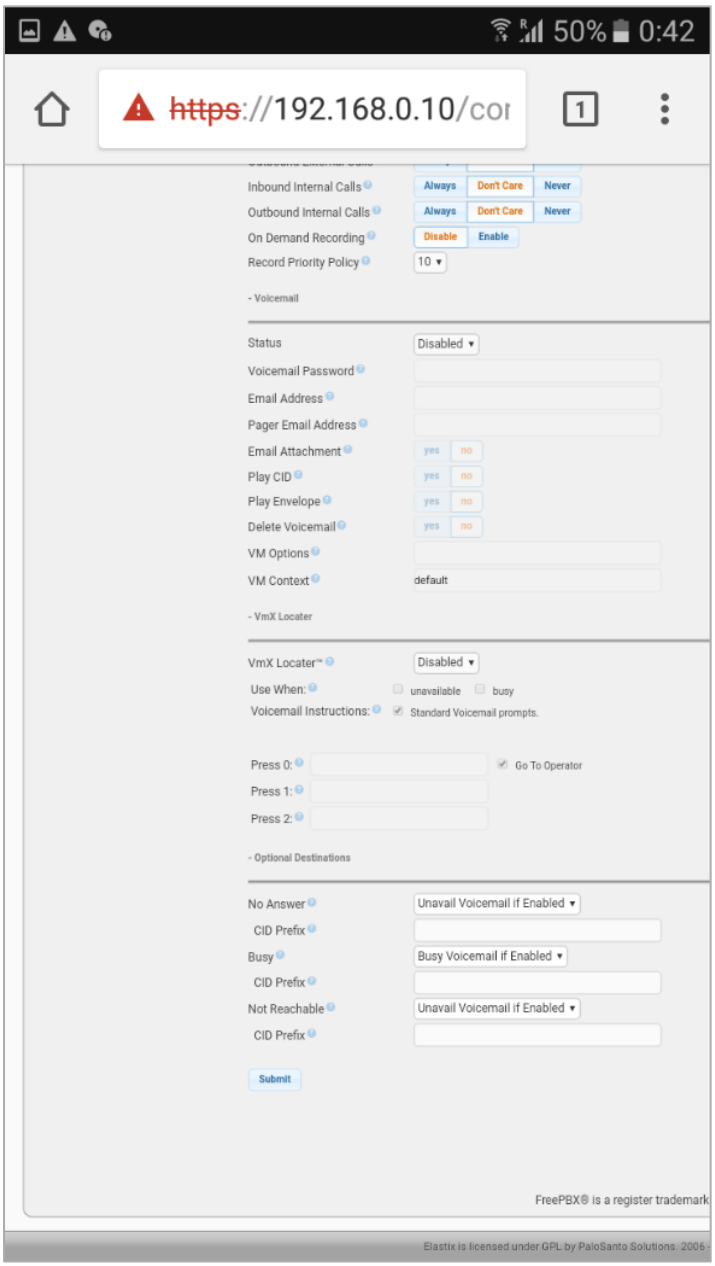


Figura 5.40: Envío de configuración

La acción anterior, lleva a la pantalla que se muestra en la Figura 5.41 a continuación, en la cual se debe dar click en la zona sombreada de color rojo "*Apply Config*".

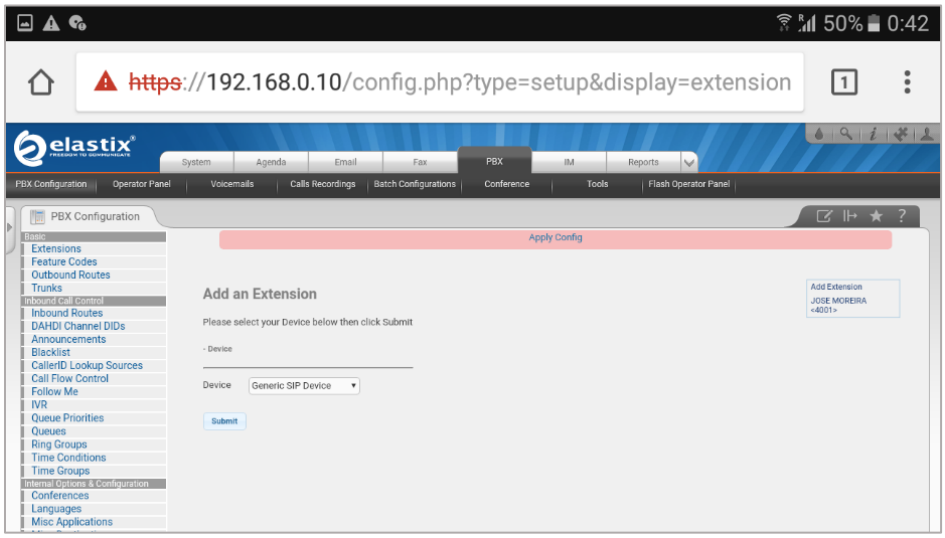


Figura 5.41: Aplicación de configuración

Y con esto, quedaría concluida la configuración de la primera extensión SIP. Se debería realizar el mismo procedimiento para las demás líneas del escenario planteado, para que los servidores, queden de la manera en que se muestra en la Figura 5.42 y Figura 5.43.

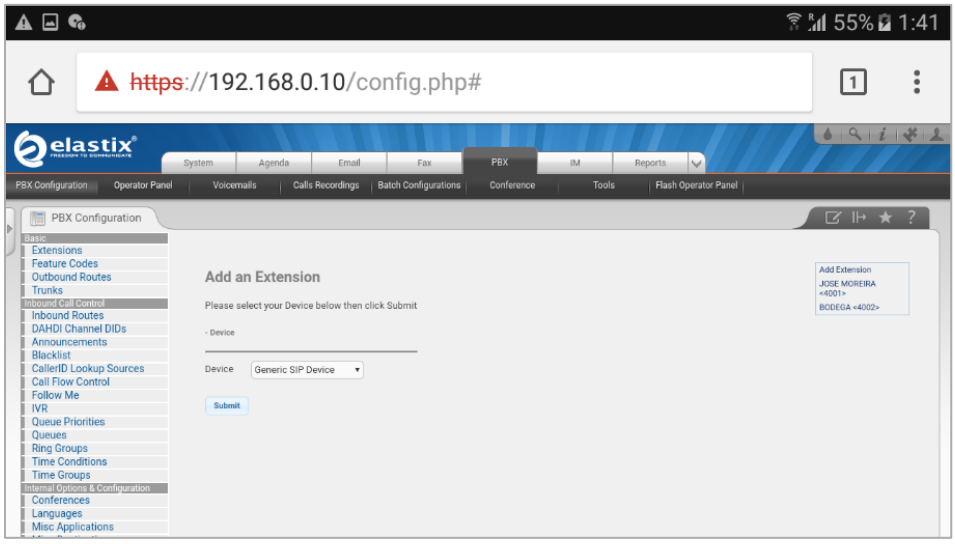


Figura 5.42: Extensiones configuradas en el servidor multissid1

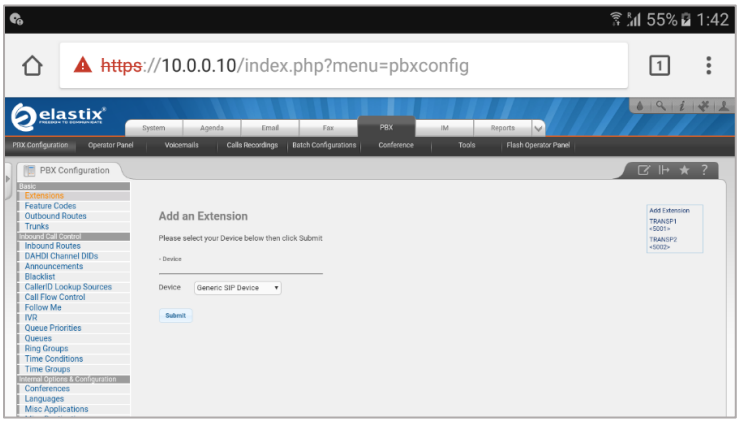


Figura 5.43: Extensiones configuradas en el servidor multissid2

Luego, se realiza la configuración de los clientes de telefonía IP. En el caso de este demostrador, se ha instalado desde la página www.zoiper.com/downloads, el softphone llamado "*Zoiper*", el cual presta las funcionalidades mínimas necesarias para el despliegue del escenario planteado.

Una vez instalado el softphone ya sea en un computador portátil o en un teléfono inteligente, se debe abrir la pestaña ajustes, como se muestra en la Figura 5.44.

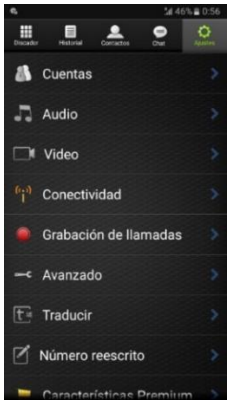


Figura 5.44: Sección Ajustes del Softphone

Se selecciona la opción *Cuentas-> Agregar Cuentas*, con lo que se logra mostrar la pantalla para agregar cuentas, como se aprecia en la Figura 5.45.

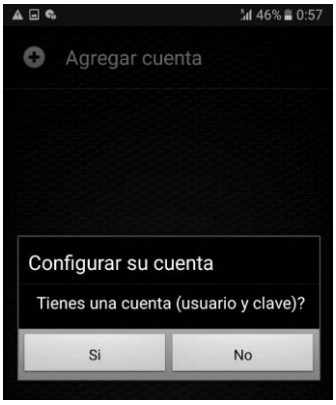


Figura 5.45: Opción Agregar Cuentas

Se da click en la opción "*si"*. Posterior a eso, se solicita escoger el tipo de cuenta que se necesita configurar, para lo cual se debe escoger la opción "*SIP*", como se muestra en la Figura 5.46.

Figura 5.46: Selección del tipo de cuenta

En este punto, se configura en cada campo correspondiente, como se muestra en la figura a continuación, los parámetros de la extensión que se crearon en el equipo terminal, parámetros que fueron previamente definidos en ambos servidores de telefonía IP.

Figura 5.47: Configuración de parámetros de la extensión en el softphone

Finalmente, se deben realizar los pasos anteriores de configuración de extensiones SIP, en los demás dispositivos del demostrador planteado.

5.2. Resultados de parámetros de Calidad de Servicio

Una vez que ha sido implementado el demostrador, en esta sección se revisan y analizan las mediciones obtenidas luego de realizar varias pruebas. Los datos, fueron recogidos por medio del programa *Wireshark*, y han sido utilizados para estudiar el comportamiento del tráfico de paquetes de los diferentes protocolos involucrados, en presencia de interfaces virtuales.

Se realizaron pruebas en dos escenarios: el primero sin utilizar virtualización de la interfaz WiFi, esquema que se muestra en la Figura 5.48 y otro, en el que se empleó el esquema propuesto en la figura (sección 5.1.3).

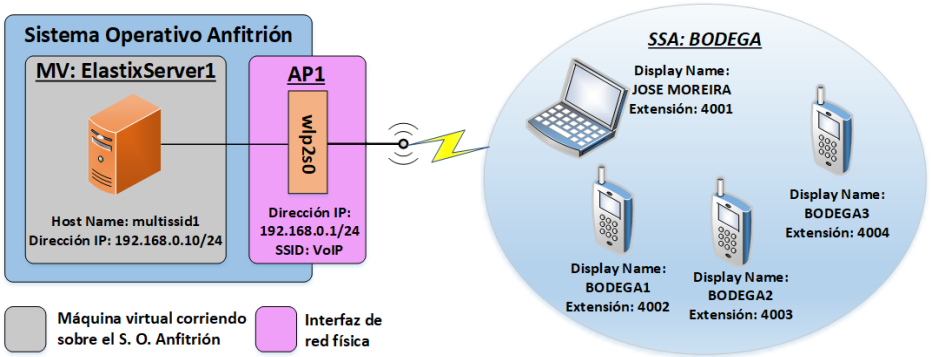


Figura 5.48: **Esquema propuesto para pruebas sin virtualización**

La duración de las llamadas realizadas, en ambos escenarios, fue de 5 minutos con el fin de comprobar la continuidad en la conexión durante las llamadas.

Además, se probaron dos códecs de audio: el G. 711 uLaw y el GSM, para comparar el desempeño del demostrador en presencia de ambos códecs.

5.2.1. Throughput y latencia

En primer lugar, se analizan los resultados para el caso en el que no se virtualizan los PA teniendo en cuenta varios CODEC diferentes.

Códec G.711 uLaw

El throughput alcanzado por cada dispositivo cliente durante las llamadas sin el uso de los VPA, se muestra en la Figura 5.49. Se observa, que el

valor aproximado del Throughput alcanzado durante las llamadas por cada extensión es de 110Kbps, el cual coincide con el valor mostrado por Wireshark en la venta de estadísticas de protocolos para el nivel *Frame* que se muestra en la Figura 5.50, estadísticas que se obtienen de todos los paquetes transmitidos y que contienen, en este caso, la dirección IP origen 192.168.0.12 que corresponde a la extensión 4002 (Bodega1).

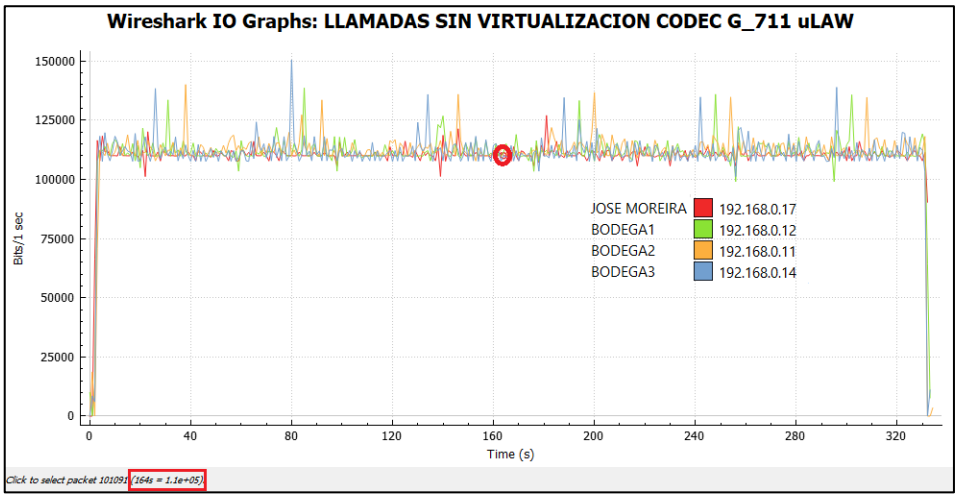
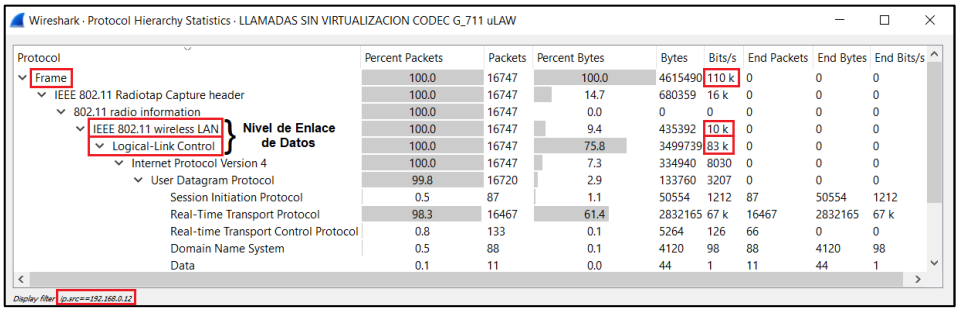


Figura 5.49: **Throughput en escenario sin VPA – Códec G.711 uLaw**



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	16747	100.0	4615490	110 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	16747	14.7	680359	16 k	0	0	0
802.11 radio information	100.0	16747	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	16747	9.4	435392	10 k	0	0	0
Logical-Link Control	100.0	16747	75.8	3499739	83 k	0	0	0
Internet Protocol Version 4	100.0	16747	7.3	334940	8030	0	0	0
User Datagram Protocol	99.8	16720	2.9	133760	3207	0	0	0
Session Initiation Protocol	0.5	87	1.1	50554	1212	87	50554	1212
Real-Time Transport Protocol	98.3	16467	61.4	2832165	67 k	16467	2832165	67 k
Real-time Transport Control Protocol	0.8	133	0.1	5264	126	66	0	0
Domain Name System	0.5	88	0.1	4120	98	88	4120	98
Data	0.1	11	0.0	44	1	11	44	1

Figura 5.50: Estadísticas de protocolos utilizados en la comunicación sin VPA – CODEC G.711 uLaw

Por otra parte, si se utilizan las ecuaciones 2.3, 2.4 junto con los valores de encabezados de la Tabla 2, que aparecen en la sección 2.6.4 para el caso del estándar IEEE 802.11n, y además, un valor de 50pps para el CODEC G.711 de la Tabla 4 de la misma sección, se obtiene el valor del throughput teórico incluyendo el nivel de enlace de datos.

$$\text{TPS} = 36 + 20 + 8 + 12 + 160 \text{ [Bytes]} \quad (5.1)$$

$$\text{TPS} = 236 \text{ [Bytes]} = 1.888 \text{ [bits]} \quad (5.2)$$

$$\text{Throughput}_{Teórico} = 1.888 \times 50 \text{ [bps]} \quad (5.3)$$

$$\mathbf{Throughput_{Teórico} = 94.4 \text{ [Kbps]}} \quad (5.4)$$

Por otra parte, el throughput real producido por los paquetes capturados en el nivel de enlace de datos, se obtendría de la suma del throughput resultante en los niveles *IEEE 802.11 wireless LAN* y *Logical-Link Control*.

$$\text{Throughput}_{Real} = \text{Throughput}_{IEEE\ 80211n} + Througput_{llc} \text{ [Kbps]} \quad (5.5)$$

$$\text{Throughput}_{Real} = 10 + 83 \text{ [Kbps]} \quad (5.6)$$

$$\mathbf{Throughput_{Real} = 93 \text{ [Kbps]}} \quad (5.7)$$

Códec GSM

En este caso, el throughput alcanzado en cada una de las conexiones desde las diferentes extensiones, se muestra en la Figura 5.51, el cual alcanza un valor de 60Kbps aproximadamente para cada una de las conexiones realizadas desde las distintas extensiones. Mientras que las estadísticas de los protocolos utilizados en las comunicaciones se muestran en la Figura 5.52.

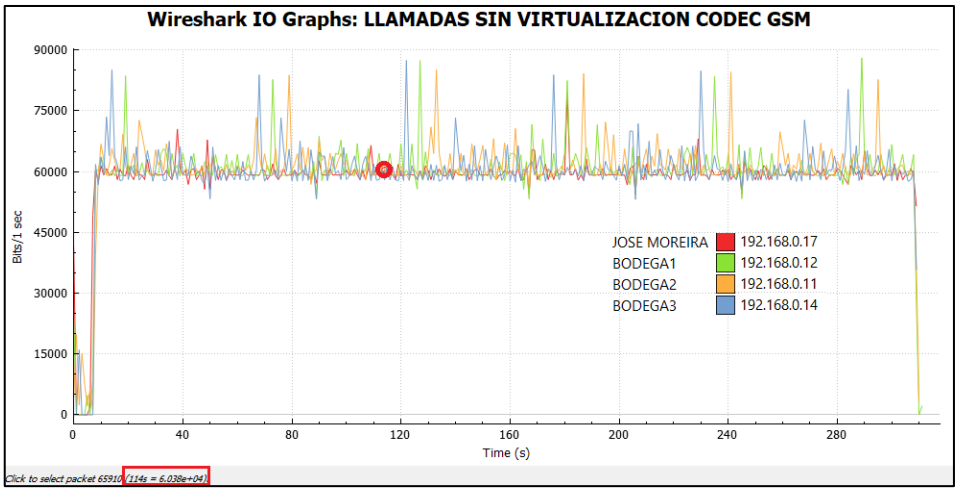
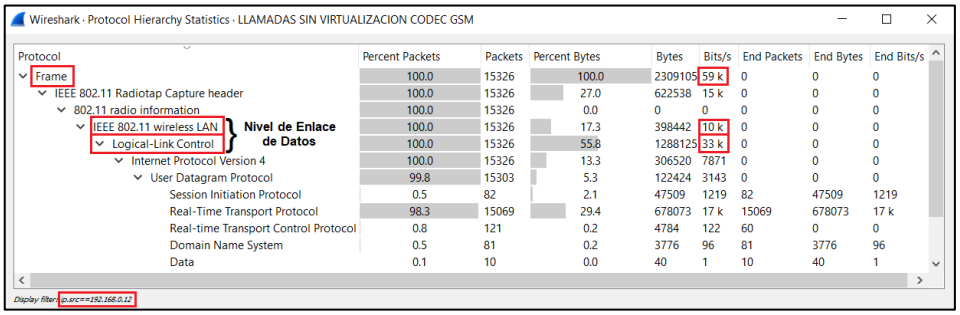


Figura 5.51: **Throughput en escenario sin VPA – Códec GSM**



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	15326	100.0	2309105	59 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	15326	27.0	622538	15 k	0	0	0
802.11 radio information	100.0	15326	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	15326	17.3	398442	10 k	0	0	0
Logical-Link Control	100.0	15326	55.8	1288125	33 k	0	0	0
Internet Protocol Version 4	100.0	15326	13.3	306520	7871	0	0	0
User Datagram Protocol	99.8	15303	5.3	122424	3143	0	0	0
Session Initiation Protocol	0.5	82	2.1	47509	1219	82	47509	1219
Real-Time Transport Protocol	98.3	15069	29.4	678073	17 k	15069	678073	17 k
Real-time Transport Control Protocol	0.8	121	0.2	4784	122	60	0	0
Domain Name System	0.5	81	0.2	3776	96	81	3776	96
Data	0.1	10	0.0	40	1	10	40	1

Figura 5.52: **Estadísticas de protocolos utilizados en la comunicación sin VPA – CODEC GSM**

A diferencia del escenario anterior (CODEC G.711uLaw sin VPA), para calcular el throughput teórico, se utiliza ahora para el CODEC GSM, una carga útil de voz de 33 Bytes, a partir de la Tabla 2. Y de manera similar, se calcula el throughput teórico.

$$\text{TPS} = 36 + 20 + 8 + 12 + 33 \text{ [Bytes]} \quad (5.8)$$

$$\text{TPS} = 109 \text{ [Bytes]} = 872 \text{ [bits]} \quad (5.9)$$

$$\text{Throughput}_{Teórico} = 872 \times 50 \text{ [bps]} \quad (5.10)$$

$$\mathbf{Throughput_{Teórico} = 43.6 \text{ [Kbps]}} \quad (5.11)$$

Mientras que el throughput real se obtiene, al igual que en el caso anterior, de la suma del throughput resultante en los niveles *IEEE 802.11 wireless LAN* y *Logical-Link Control* de las estadísticas de los protocolos utilizados, en este caso para las llamadas con dirección IP origen 192.168.0.12.

$$\text{Throughput}_{Real} = 10 + 33 \text{ [Kbps]} \quad (5.12)$$

$$\mathbf{Throughput_{Real} = 43 \text{ [Kbps]}} \quad (5.13)$$

A continuación, se analiza el caso en el que se emplean AP virtuales, para cada CODEC utilizado.

Códec G.711 uLaw

En este escenario, el throughput que se obtiene, con las direcciones IP de origen mostradas en la Figura 5.53, en cada conexión desde cada extensión creada, alcanza los 110Kbps aproximadamente, un valor que es igual al obtenido en el escenario donde no se utilizan VPA.

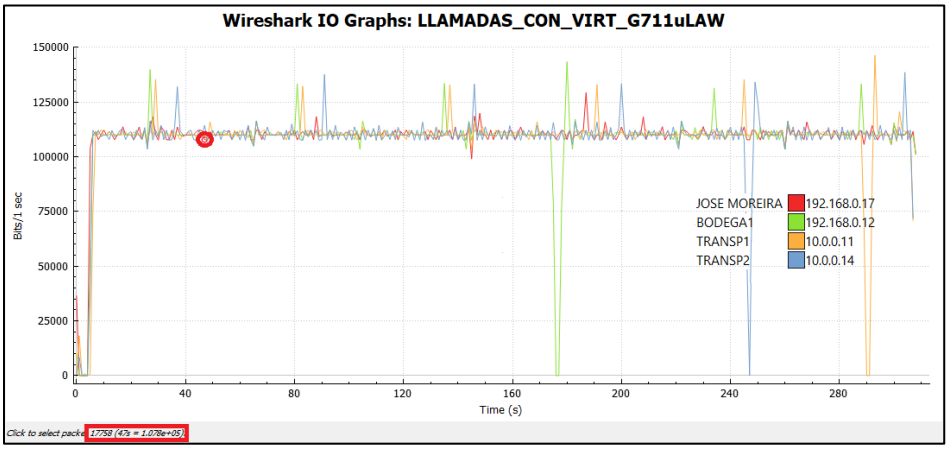
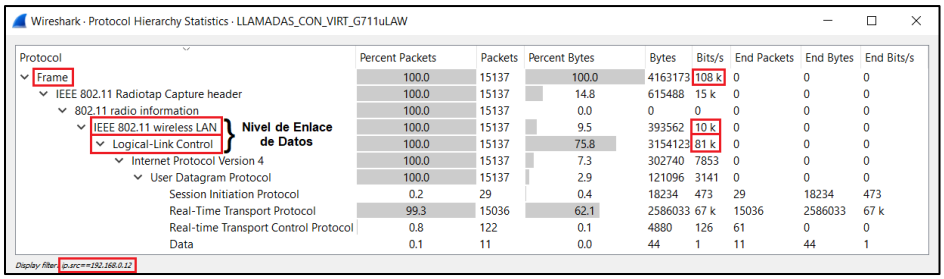


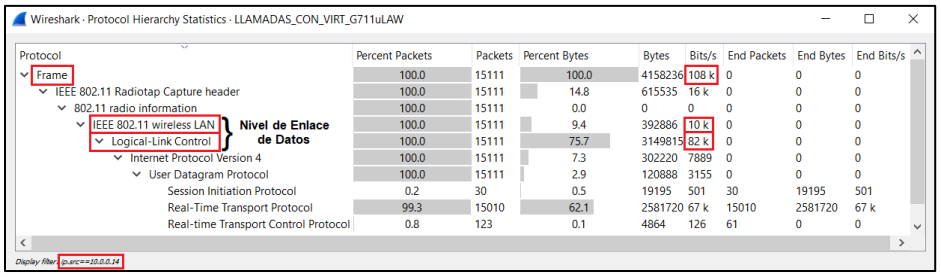
Figura 5.53: **Throughput en escenario con VPA – Códec G.711 uLaw**

En el caso del throughput real en el nivel de enlace de datos, como se muestra en la Figura 5.54 y Figura 5.55, este varía de manera insignificante en las dos redes si se compara con el obtenido en el escenario sin VPA.



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	15137	100.0	4163173	108 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	15137	14.8	615488	15 k	0	0	0
802.11 radio information	100.0	15137	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	15137	9.5	393562	10 k	0	0	0
Logical-Link Control	100.0	15137	75.8	3154123	81 k	0	0	0
Internet Protocol Version 4	100.0	15137	7.3	302740	7853	0	0	0
User Datagram Protocol	100.0	15137	2.9	121096	3141	0	0	0
Session Initiation Protocol	0.2	29	0.4	18234	473	29	18234	473
Real-Time Transport Protocol	99.3	15036	62.1	2586033	67 k	15036	2586033	67 k
Real-time Transport Control Protocol	0.8	122	0.1	4880	126	61	0	0
Data	0.1	11	0.0	44	1	11	44	1

Figura 5.54: **Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC G.711 uLaw – SSID BODEGA**



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	15111	100.0	4158236	108 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	15111	14.8	615535	16 k	0	0	0
802.11 radio information	100.0	15111	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	15111	9.4	392886	10 k	0	0	0
Logical-Link Control	100.0	15111	75.7	3149815	82 k	0	0	0
Internet Protocol Version 4	100.0	15111	7.3	302220	7889	0	0	0
User Datagram Protocol	100.0	15111	2.9	120888	3155	0	0	0
Session Initiation Protocol	0.2	30	0.5	19195	501	30	19195	501
Real-Time Transport Protocol	99.3	15010	62.1	2581720	67 k	15010	2581720	67 k
Real-time Transport Control Protocol	0.8	123	0.1	4864	126	61	0	0

Figura 5.55: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC G.711 uLaw – SSID TRANSPORTACION

$$\text{Throughput}_{Real-Red192} = 10 + 81 \text{ [Kbps]} \quad (5.14)$$

$$\mathbf{Throughput}_{\boldsymbol{Real-Red192}} = \mathbf{91\ [Kbps]} \quad (5.15)$$

$$\text{Throughput}_{Real-Red10} = 10 + 82 \text{ [Kbps]} \quad (5.16)$$

$$\mathbf{Throughput}_{\boldsymbol{Real-Red10}} \mathbf{= 92 \text{ [Kbps]}} \quad (5.17)$$

Códec GSM

En este caso, el throughput total que se obtiene, al transmitir utilizando el CODEC GSM, es de aproximadamente 60Kbps, valor que es el mismo en las dos redes configuradas: 192.168.0.0/24 y 10.0.0.0/8, tal como se muestra en la Figura 5.56 y que se pueden confirmar en las estadísticas de protocolos utilizados en las llamadas, como se muestra en el nivel *Frame* de la Figura 5.57 y Figura 5.58.

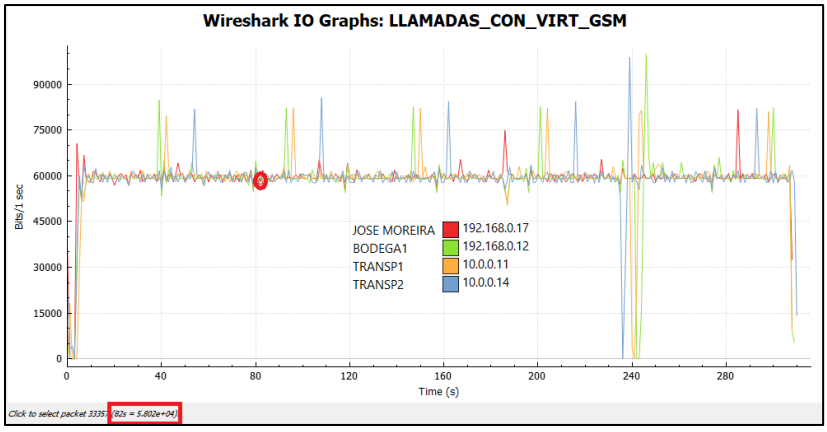
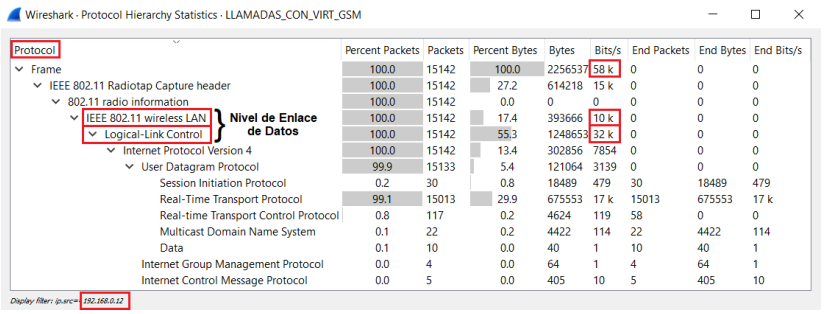


Figura 5.56: **Throughput en escenario con virtualización – Códec GSM**

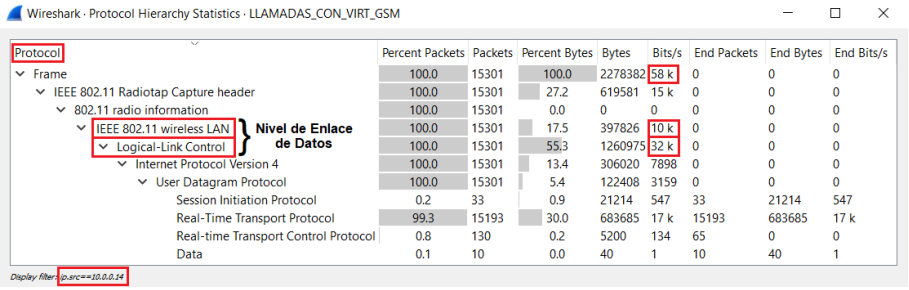


Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	15142	100.0	2256537	58 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	15142	27.2	614218	15 k	0	0	0
802.11 radio information	100.0	15142	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	15142	17.4	393666	10 k	0	0	0
Logical-Link Control	100.0	15142	55.3	1248653	32 k	0	0	0
Internet Protocol Version 4	100.0	15142	13.4	302856	7854	0	0	0
User Datagram Protocol	99.9	15133	5.4	121064	3139	0	0	0
Session Initiation Protocol	0.2	30	0.8	18489	479	30	18489	479
Real-Time Transport Protocol	99.1	15013	29.9	675553	17 k	15013	675553	17 k
Real-time Transport Control Protocol	0.8	117	0.2	4624	119	58	0	0
Multicast Domain Name System	0.1	22	0.2	4422	114	22	4422	114
Data	0.1	10	0.0	40	1	10	40	1
Internet Group Management Protocol	0.0	4	0.0	64	1	4	64	1
Internet Control Message Protocol	0.0	5	0.0	405	10	5	405	10

Figura 5.57: **Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC GSM – SSID BODEGA**

$$\text{Throughput}_{Real-Red192} = 10 + 32 \text{ [Kbps]} \quad (5.18)$$

$$\mathbf{Throughput}_{\boldsymbol{Real-Red192}} \mathbf{= 42 \text{ [Kbps]}} \quad (5.19)$$



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	15301	100.0	2278382	58 k	0	0	0
IEEE 802.11 Radiotap Capture header	100.0	15301	27.2	619581	15 k	0	0	0
802.11 radio information	100.0	15301	0.0	0	0	0	0	0
IEEE 802.11 wireless LAN	100.0	15301	17.5	397826	10 k	0	0	0
Logical-Link Control	100.0	15301	55.3	1260975	32 k	0	0	0
Internet Protocol Version 4	100.0	15301	13.4	306020	7898	0	0	0
User Datagram Protocol	100.0	15301	5.4	122408	3159	0	0	0
Session Initiation Protocol	0.2	33	0.9	21214	547	33	21214	547
Real-Time Transport Protocol	99.3	15193	30.0	683685	17 k	15193	683685	17 k
Real-time Transport Control Protocol	0.8	130	0.2	5200	134	65	0	0
Data	0.1	10	0.0	40	1	10	40	1

Figura 5.58: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC GSM – SSID TRANSPORTACION

$$\text{Throughput}_{Real-Red10} = 10 + 32 \text{ [Kbps]} \quad (5.20)$$

$$\mathbf{Throughput}_{\boldsymbol{Real-Red10}} \mathbf{= 42 \text{ [Kbps]}} \quad (5.21)$$

Por otro lado, el throughput en el nivel de enlace de datos, es aproximadamente igual al obtenido en los escenarios anteriores, en ambas redes.

Sobre la latencia, se puede adelantar, en función de los gráficos mostrados, que esta no ha tenido valores significativamente altos, ya que no se aprecian mayores interrupciones en la transmisión de los paquetes. Esta apreciación, se verificará en la sección 5.3, cuando se midan los parámetros de QoE, donde se requiere conocer los valores de *latencia.*

5.2.2. Paquetes perdidos e influencia de buffers

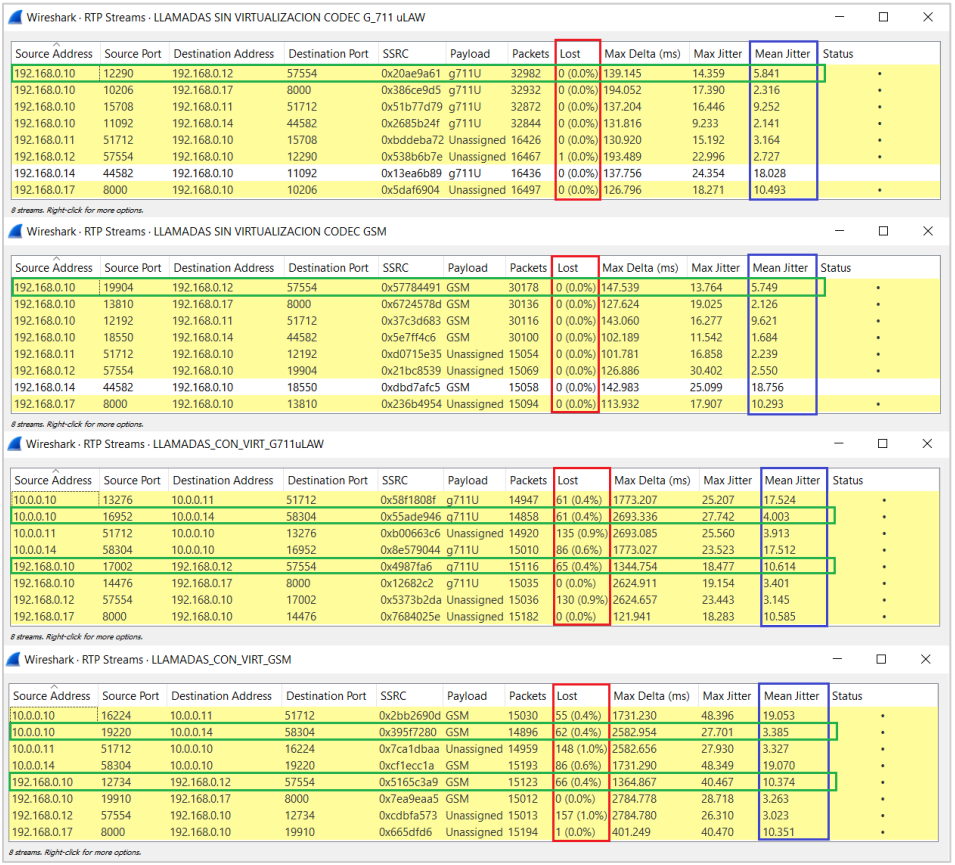
En la Figura 5.59, se encuentran remarcadas en color verde, las comunicaciones entre servidor y cliente basadas en protocolo RTP que han servido para este análisis.

En la misma figura, remarcado en color rojo, se puede apreciar un resumen de los paquetes perdidos, cuando son transmitidos entre los servidores de telefonía IP y los equipos terminales en ambos sentidos, en la columna *Lost* la cual se encuentra remarcada en rojo.

Se puede observar, la diferencia en cuanto a la cantidad de paquetes perdidos entre los escenarios sin VPA y con VPA. Mientras que, en el primero, la pérdida de paquetes es prácticamente 0%, en el escenario con VPA, existen pérdidas de paquetes las cuales, sin embargo, están dentro de los niveles aceptados (menor al 3%) para tener una comunicación

audible. Estas pérdidas de paquetes, de algún modo son previsibles, en vista de que ahora la tarjeta de red inalámbrica al haber sido virtualizada debe transmitir las tramas de las dos redes virtuales creadas, de manera alternada mediante conmutación, como se revisó en la sección 3.3, lo que introduce retrasos en la transmisión de los paquetes.

Esta pérdida de paquetes, se debe probablemente la existencia de congestión del enlace al momento de las transmisiones desde ambas interfaces virtuales, o de otra manera, a un sobre flujo momentáneo en el receptor, que hace que el buffer de supresión de jitter tenga de descartar algunos paquetes.



Wireshark · RTP Streams · LLAMADAS SIN VIRTUALIZACION CODEC G_711 uLAW

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter	Status
192.168.0.10	12290	192.168.0.12	57554	0x20ae9a61	g711U	32982	0 (0.0%)	139.145	14.359	5.841	•
192.168.0.10	10206	192.168.0.17	8000	0x386ce9d5	g711U	32932	0 (0.0%)	194.052	17.390	2.316	•
192.168.0.10	15708	192.168.0.11	51712	0x51b77d79	g711U	32872	0 (0.0%)	137.204	16.446	9.252	•
192.168.0.10	11092	192.168.0.14	44582	0x2685b24f	g711U	32844	0 (0.0%)	131.816	9.233	2.141	•
192.168.0.11	51712	192.168.0.10	15708	0xbddeba72	Unassigned	16426	0 (0.0%)	130.920	15.192	3.164	•
192.168.0.12	57554	192.168.0.10	12290	0x538b6b7e	Unassigned	16467	1 (0.0%)	193.489	22.996	2.727	•
192.168.0.14	44582	192.168.0.10	11092	0x13ea6b89	g711U	16436	0 (0.0%)	137.756	24.354	18.028	
192.168.0.17	8000	192.168.0.10	10206	0x5daf6904	Unassigned	16497	0 (0.0%)	126.796	18.271	10.493	•

8 streams. Right-click for more options.

Wireshark · RTP Streams · LLAMADAS SIN VIRTUALIZACION CODEC GSM

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter	Status
192.168.0.10	19904	192.168.0.12	57554	0x57784491	GSM	30178	0 (0.0%)	147.539	13.764	5.749	•
192.168.0.10	13810	192.168.0.17	8000	0x6724578d	GSM	30136	0 (0.0%)	127.624	19.025	2.126	•
192.168.0.10	12192	192.168.0.11	51712	0x37c3d683	GSM	30116	0 (0.0%)	143.060	16.277	9.621	•
192.168.0.10	18550	192.168.0.14	44582	0x5e7ff4c6	GSM	30100	0 (0.0%)	102.189	11.542	1.684	•
192.168.0.11	51712	192.168.0.10	12192	0xd0715e35	Unassigned	15054	0 (0.0%)	101.781	16.858	2.239	•
192.168.0.12	57554	192.168.0.10	19904	0x21bc8539	Unassigned	15069	0 (0.0%)	126.886	30.402	2.550	•
192.168.0.14	44582	192.168.0.10	18550	0xdbd7afc5	GSM	15058	0 (0.0%)	142.983	25.099	18.756	
192.168.0.17	8000	192.168.0.10	13810	0x236b4954	Unassigned	15094	0 (0.0%)	113.932	17.907	10.293	•

8 streams. Right-click for more options.

Wireshark · RTP Streams · LLAMADAS_CON_VIRT_G711uLAW

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter	Status
10.0.0.10	13276	10.0.0.11	51712	0x58f1808f	g711U	14947	61 (0.4%)	1773.207	25.207	17.524	•
10.0.0.10	16952	10.0.0.14	58304	0x55ade946	g711U	14858	61 (0.4%)	2693.336	27.742	4.003	•
10.0.0.11	51712	10.0.0.10	13276	0xb00663c6	Unassigned	14920	135 (0.9%)	2693.085	25.560	3.913	•
10.0.0.14	58304	10.0.0.10	16952	0x8e579044	g711U	15010	86 (0.6%)	1773.027	23.523	17.512	•
192.168.0.10	17002	192.168.0.12	57554	0x4987fa6	g711U	15116	65 (0.4%)	1344.754	18.477	10.614	•
192.168.0.10	14476	192.168.0.17	8000	0x12682c2	g711U	15035	0 (0.0%)	2624.911	19.154	3.401	•
192.168.0.12	57554	192.168.0.10	17002	0x5373b2da	Unassigned	15036	130 (0.9%)	2624.657	23.443	3.145	•
192.168.0.17	8000	192.168.0.10	14476	0x7684025e	Unassigned	15182	0 (0.0%)	121.941	18.283	10.585	•

8 streams. Right-click for more options.

Wireshark · RTP Streams · LLAMADAS_CON_VIRT_GSM

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter	Status
10.0.0.10	16224	10.0.0.11	51712	0x2bb2690d	GSM	15030	55 (0.4%)	1731.230	48.396	19.053	•
10.0.0.10	19220	10.0.0.14	58304	0x395f7280	GSM	14896	62 (0.4%)	2582.954	27.701	3.385	•
10.0.0.11	51712	10.0.0.10	16224	0x7ca1dbaa	Unassigned	14959	148 (1.0%)	2582.656	27.930	3.327	•
10.0.0.14	58304	10.0.0.10	19220	0xcf1ecc1a	GSM	15193	86 (0.6%)	1731.290	48.349	19.070	•
192.168.0.10	12734	192.168.0.12	57554	0x5165c3a9	GSM	15123	66 (0.4%)	1364.867	40.467	10.374	•
192.168.0.10	19910	192.168.0.17	8000	0x7ea9eaa5	GSM	15012	0 (0.0%)	2784.778	28.718	3.263	•
192.168.0.12	57554	192.168.0.10	12734	0xcdbfa573	Unassigned	15013	157 (1.0%)	2784.780	26.310	3.023	•
192.168.0.17	8000	192.168.0.10	19910	0x665dfd6	Unassigned	15194	1 (0.0%)	401.249	40.470	10.351	•

8 streams. Right-click for more options.

Figura 5.59: Resumen de parámetros de QoS

5.2.3. Comparación entre WiFi con y sin virtualización

En la Tabla 5, se presenta un resumen de los valores de throughput y pérdida de paquetes obtenidos en las pruebas realizadas.

A simple vista se puede observar, que no ha existido un impacto significativo en el throughput al utilizar VPA, ya que este parámetro se aproxima a los valores teóricos calculados en la sección anterior, para los dos CODEC utilizados en las pruebas.

Escenario	WiFi Sin Virtualización		WiFi Con Virtualización			
Extensión	4002		4002	5002	4002	5002
Dirección IP	192.168.0.12		192.168.0.12	10.0.0.14	192.168.0.12	10.0.0.14
Interfaz	wlp2s0		AP1	AP2	AP1	AP2
CODEC	G.711uLaw	GSM	G.711uLaw		GSM	
Throughput [Kbps]	110	60	110	110	60	60
Paquetes Perdidos [%]	0	0	0,43	0,41	0,43	0,41
Jitter Promedio [ms]	5,84	5,75	10,60	4,00	10,36	3,38

Tabla 5: Throughput y paquetes perdidos en ambos escenarios

Así mismo, los valores de pérdida de paquete sufren variación en el escenario con VPA comparado con el escenario sin VPA.

De manera general, se observa que los parámetros de QoS se ven afectados por el uso de virtualización inalámbrica, sin embargo, estos niveles se encuentran dentro de los parámetros tolerables.

5.3. Resultados de Calidad de experiencia de usuario

Los niveles de QoE para los diferentes escenarios, fueron calculados de dos maneras: la primera, mediante el parámetro MOS, el que a su vez, fue obtenido a partir del factor R utilizando la ecuación 2.11 que relaciona ambos parámetros. La otra forma fue, revisando los registros que se graban en el softphone Zoiper el cual mide el valor MOS producido en cada llamada, valor que es almacenado luego en los dispositivos terminales. El analizador de paquetes de red utilizado, Wireshark, permite generar un archivo en formato CSV de todos los segmentos de tipo RTP durante una conexión.

El mencionado archivo, fue procesado con el programa Microsoft Excel, de donde se obtuvieron, los diferentes factores con los cuales se determinaron los valores de R y MOS.

5.3.1. Medición del MOS

El valor MOS de las llamadas, ha sido obtenido empleando la ecuación 2.11, para lo cual se requiere inicialmente conocer los parámetros de QoS, detallados a continuación, el cálculo del factor R con el cual se proyecta el valor MOS.

- *Cantidad Total de Paquetes*: se lo obtuvo de la contabilización de los paquetes RTP contenidos en el archivo CSV.

- *Retraso (Latencia o Delta) promedio*: se lo obtuvo sumando los retrasos (delta) individuales de todos los paquetes RTP, dividiendo esta sumatoria, entre la cantidad total de paquetes RTP cargados en el archivo CSV.

- *Jitter Promedio*: se obtuvo este valor sumando el valor del Jitter individual de cada paquete RTP, dividiendo esta sumatoria entre la cantidad total de paquetes RTP cargados en el archivo CSV. El jitter promedio, también se muestra en la ventana de *Análisis de Tramas RTP* de wireshark.

-*Paquetes Perdidos*: Se lo obtuvo de la ventana de análisis de tramas RTP.

Escenario	**Sin Virt. PA**		**Con Virt. PA**			
Interfaz	wlp2s0	wlp2s0	AP1	AP2	AP1	AP2
Extensión	4002	4002	4002	5002	4002	5002
Dirección IP	192.168.0.12	192.168.0.12	192.168.0.12	10.0.0.14	192.168.0.12	10.0.0.14
CODEC	G.711 uLaw	GSM	G.711 uLaw	G.711 uLaw	GSM	GSM
Cantidad de Paquetes	32.982	30.178	15.116	14.858	15.123	14.896
Latencia Promedio [ms]	10,00	10,00	20,09	20,26	20,09	20,28
Jitter Promedio [ms]	5,84	7,75	10,60	4,00	10,36	3,38
Paquetes Perdidos [%]	0,00	0,00	0,43	0,41	0,43	0,41

Tabla 6: Parámetros para el cálculo de factor R y valor MOS

En la Tabla 6 se presentan los diferentes valores obtenidos de los parámetros para el cálculo del factor R y el valor MOS en todos los escenarios implementados.

A continuación, se presentan los resultados obtenidos para los escenarios implementados sin VPA, para los diferentes CODEC usados.

CODEC G.711 uLaw

A partir de los datos de la Tabla 6, para el caso del CODEC G.711 uLaw sin VPA, se obtienen los valores R y MOS con las ecuaciones 2.8, 2.9, 2.10 y 2.11. El análisis de tramas RTP se lo realizó sobre el dispositivo configurado con la dirección IP 192.168.0.12 configurado con la extensión 4002, análisis que se muestra en la Figura 5.60 y el valor registrado en el softphone Zoiper del cliente, se muestra en Figura 5.61.

$$\textbf{Latencia Efectiva} = \mathbf{10,00} + (\mathbf{2} \times \mathbf{5,84}) + \mathbf{10} \tag{5.22}$$

$$\textbf{Latencia Efectiva} = \mathbf{31,68[ms]} \tag{5.23}$$

$$\mathbf{Ro} = \mathbf{93.2} - \left(\frac{\mathbf{31,68}}{\mathbf{40}}\right) = \mathbf{92,41} \tag{5.24}$$

$$\mathbf{R} = \mathbf{92,41} - (\mathbf{0,00} \times \mathbf{2,5}) \tag{5.25}$$

$$\mathbf{R} = \mathbf{92,41} \tag{5.26}$$

$$\text{MOS} = 1 + (0{,}035 \times 92{,}41) + (0.000007) \times 92{,}41 \times (92{,}41 - 60) \times \times (100 - 92{,}41) \tag{5.27}$$

$$\mathbf{MOS} = \mathbf{4,39} \tag{5.28}$$

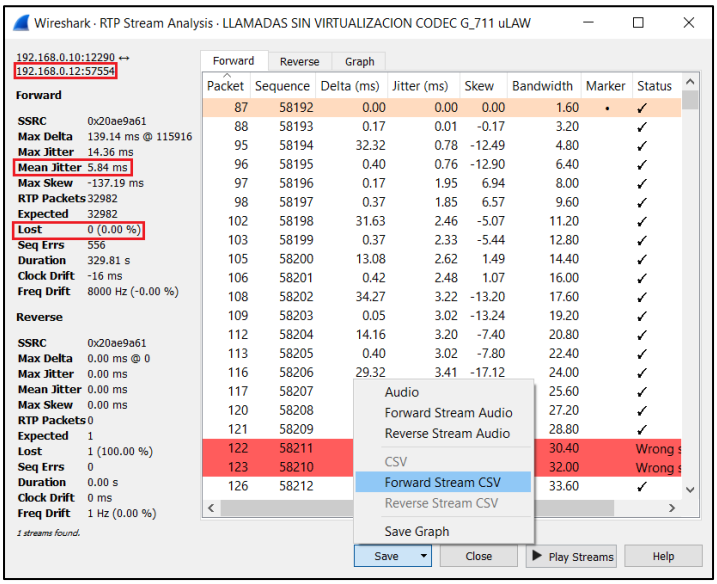


Figura 5.60: **Análisis de Tramas RTP en escenario sin VPA – CODEC G.711 uLaw**

Figura 5.61: **MOS usando CODEC G.711 uLAW sin VPA ext. 4002**

CODEC GSM

Igual que en el escenario anterior, se toman ahora de la Tabla 6, los parámetros correspondientes para el caso del CODEC GSM sin VPA, para obtener los valores R y MOS de este escenario. El análisis de tramas RTP se lo realizó sobre el dispositivo configurado con la dirección IP 192.168.0.12 configurado con la extensión 4002, análisis que se muestra en la Figura 5.62 mientras que el valor medido en el dispositivo cliente por el softphone se muestra en la Figura 5.63.

$$\textbf{Latencia Efectiva} = \mathbf{10,00} + (\mathbf{2} \times \mathbf{5,75}) + \mathbf{10} \quad (5.29)$$

$$\textbf{Latencia Efectiva} = \mathbf{31,49[ms]} \quad (5.30)$$

$$\text{Ro} = 93.2 - \left(\frac{31,49}{40}\right) = 92,41 \quad (5.31)$$

$$\mathbf{R} = \mathbf{92,41} - (\mathbf{0} \times \mathbf{2,5}) \quad (5.32)$$

$$\mathbf{R} = \mathbf{92,41} \quad (5.33)$$

$$\text{MOS} = 1 + (0,035 \times 92,41) + (0.000007) \times 92,41 \times (92,41 - 60) \times \times (100 - 92,4) \quad (5.34)$$

$$\mathbf{MOS} = \mathbf{4,39} \quad (5.35)$$

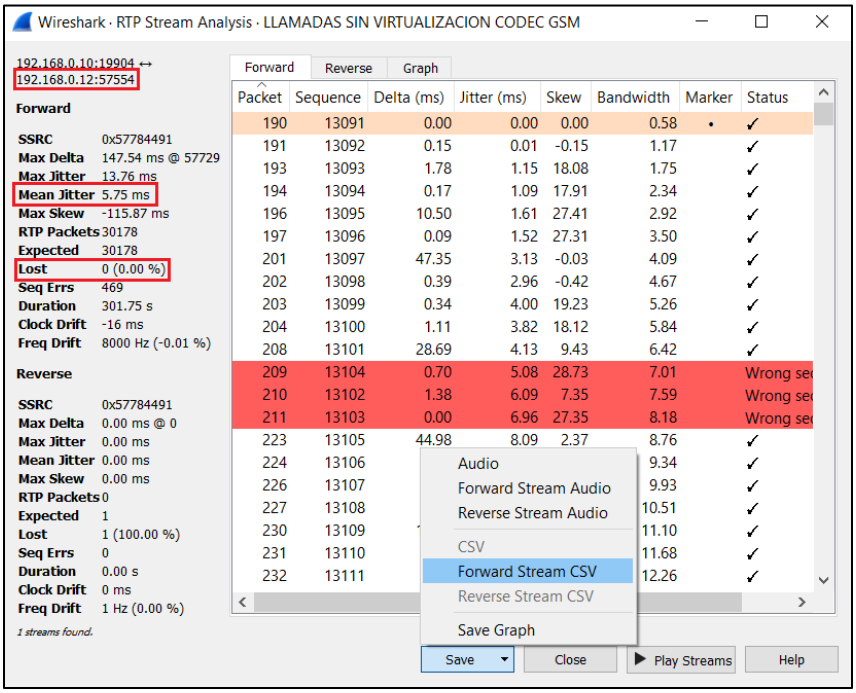


Figura 5.62: Análisis de Tramas RTP en escenario sin VPA – CODEC GSM

Figura 5.63: MOS usando códec GSM sin VPA ext. 4002

A continuación, se presenta el resultado de las pruebas en escenarios que utilizan VPA para los diferentes CODEC utilizados.

CODEC G.711uLaw

Se toman de la Tabla 6 los parámetros que corresponden al escenario con VPA usando CODEC G.711uLaw para la extensión 4002 en la red 192.168.0.0/24 con el SSID BODEGA, para el cálculo de los valores R y

MOS. El análisis de tramas RTP se muestra en la Figura 5.64 y el valor medido en el dispositivo cliente se muestra en la Figura 5.65.

$$\textbf{Latencia Efectiva} = \mathbf{20,09} + (\mathbf{2} \times \mathbf{10.60}) + \mathbf{10} \quad (5.36)$$

$$\textbf{Latencia Efectiva} = \mathbf{51,30[ms]} \quad (5.37)$$

$$\text{Ro} = 93.2 - \left(\frac{51,30}{40}\right) = 91,92 \quad (5.38)$$

$$\mathbf{R} = \mathbf{91,92} - (\mathbf{0,0043} \times \mathbf{2,5}) \quad (5.39)$$

$$\mathbf{R} = \mathbf{91,91} \quad (5.40)$$

$$\text{MOS} = 1 + (0{,}035 \times 91{,}91) + (0.000007) \times 91{,}91 \times (91{,}91 - 60) \times$$

$$\times (100 - 91{,}91) \quad (5.41)$$

$$\mathbf{MOS} = \mathbf{4,38} \quad (5.42)$$

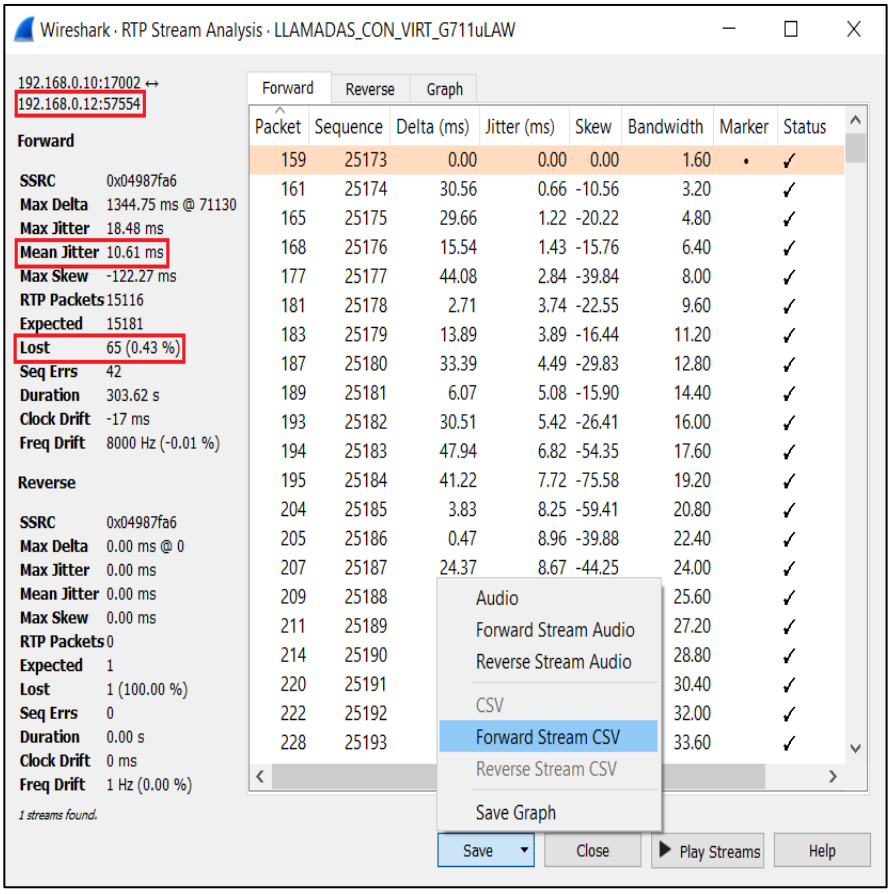


Figura 5.64: Análisis de Tramas RTP en escenario con VPA – CODEC G.711 uLaw

Figura 5.65: **MOS usando códec G.711 uLAW con VPA ext. 4002**

Ahora se toman de la Tabla 6 los parámetros que corresponden al escenario con VPA usando el mismo CODEC para la extensión 5002 en la red 10.0.0.0/8 con el SSID TRANSPORTACION, para el cálculo de los valores R y MOS. El análisis de tramas RTP se muestra en la Figura 5.66 y el valor medido en el dispositivo cliente se muestra en la Figura 5.67.

$$\mathbf{Latencia\ Efectiva} = \mathbf{20,26} + (\mathbf{2} \times \mathbf{4}) + \mathbf{10} \tag{5.43}$$

$$\mathbf{Latencia\ Efectiva} = \mathbf{38,26[ms]} \tag{5.44}$$

$$\mathrm{Ro} = 93.2 - \left(\frac{38,26}{40}\right) = 92,24 \tag{5.45}$$

$$\mathbf{R} = \mathbf{92,24} - (\mathbf{0,0041} \times \mathbf{2,5}) \tag{5.46}$$

$$\mathbf{R} = \mathbf{92,23} \tag{5.47}$$

$$\mathrm{MOS} = 1 + (0{,}035 \times 92{,}23) + (0.000007) \times 92{,}23 \times (92{,}23 - 60) \times \times (100 - 92{,}23) \tag{5.48}$$

$$\mathbf{MOS} = \mathbf{4,39} \tag{5.49}$$

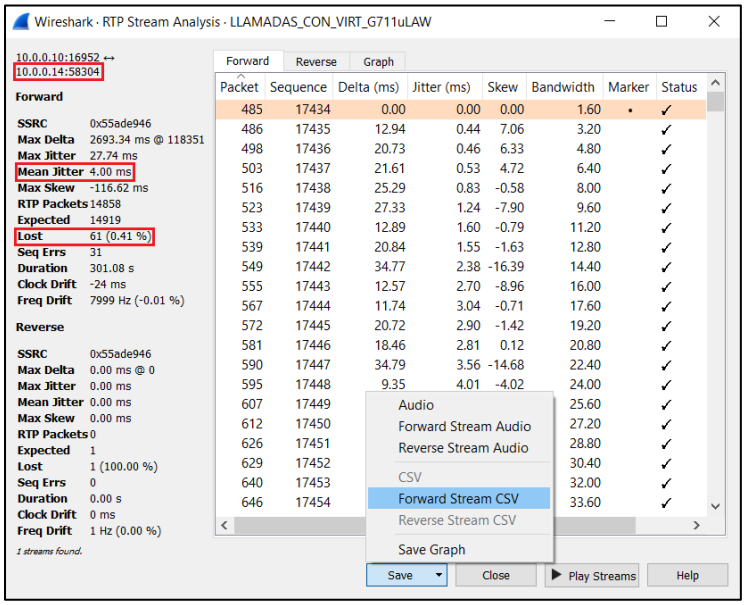


Figura 5.66: Análisis de Tramas RTP en escenario con VPA – CODEC G.711 uLaw - SSID TRANSPORTACION

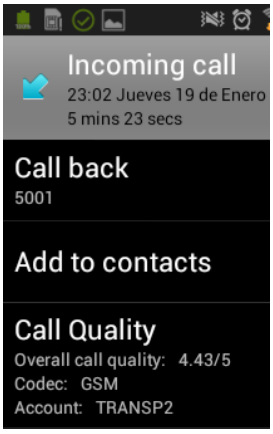


Figura 5.67: MOS usando códec G.711 uLaw con VPA ext. 5002

CODEC GSM

Se toman de la Tabla 6 los parámetros que corresponden al escenario con VPA usando CODEC GSM para la extensión 4002 en la red 192.168.0.0/24 con el SSID BODEGA, para el cálculo de los valores R y MOS. El análisis de tramas RTP se muestra en la Figura 5.68 y el valor medido en el dispositivo cliente se muestra en la Figura 5.69.

$$\textbf{Latencia Efectiva} = \mathbf{20,09} + (\mathbf{2} \times \mathbf{10.36}) + \mathbf{10} \quad (5.50)$$

$$\textbf{Latencia Efectiva} = \mathbf{50,81[ms]} \quad (5.51)$$

$$\text{Ro} = 93.2 - \left(\frac{50,81}{40}\right) = 91,93 \quad (5.52)$$

$$\mathbf{R} = \mathbf{91,93} - (\mathbf{0,0043} \times \mathbf{2,5}) \quad (5.53)$$

$$\mathbf{R} = \mathbf{91,92} \quad (5.54)$$

$$\text{MOS} = 1 + (0{,}035 \times 91{,}92) + (0.000007) \times 91{,}92 \times (91{,}92 - 60) \times \times (100 - 91{,}92) \quad (5.55)$$

$$\mathbf{MOS} = \mathbf{4,38} \quad (5.56)$$

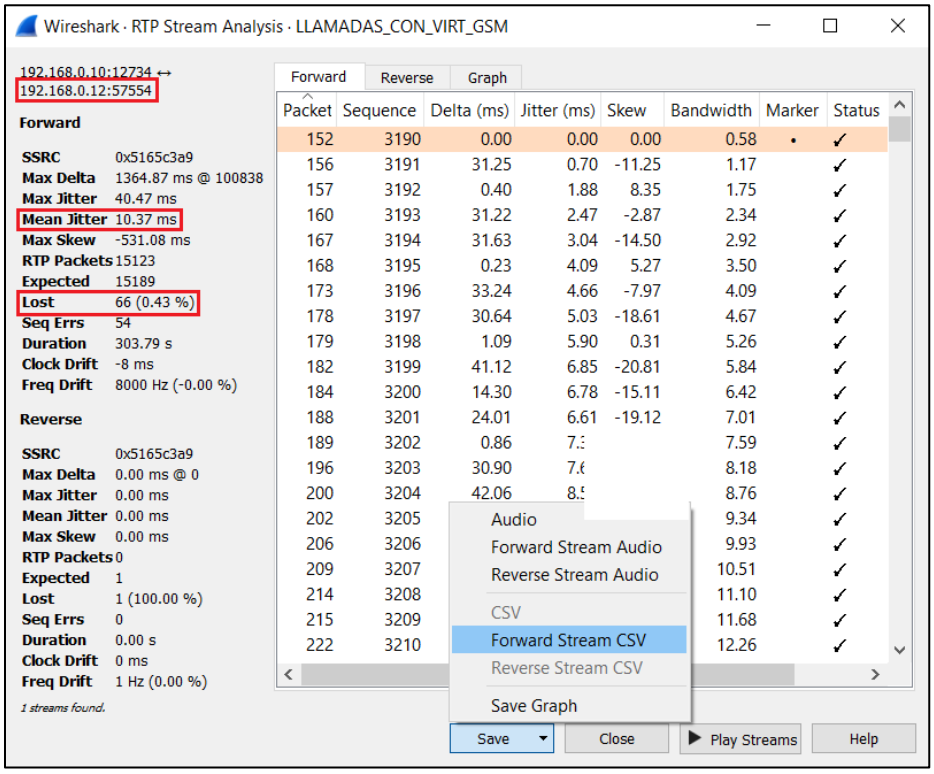


Figura 5.68: Análisis de Tramas RTP en escenario con VPA – CODEC GSM – SSID BODEGA

Figura 5.69: **MOS usando códec GSM con VPA ext. 4002**

Finalmente se toman de la Tabla 6 los parámetros correspondientes al escenario con VPA usando el mismo CODEC para la extensión 5002 en la red 10.0.0.0/8 con el SSID TRANSPORTACION, para el cálculo de los valores R y MOS. El análisis de tramas RTP se muestra en la Figura 5.70 y el valor medido en el dispositivo cliente se muestra en la Figura 5.71.

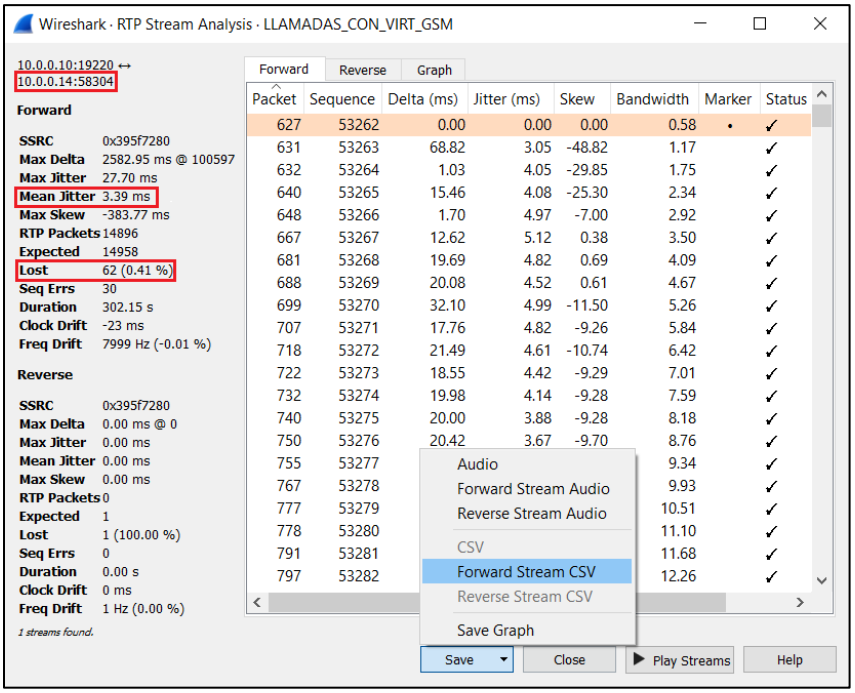


Figura 5.70: Análisis de Tramas RTP en escenario con VPA – CODEC GSM – SSID TRANSPORTACION

$$\textbf{Latencia Efectiva} = \mathbf{20,28} + (\mathbf{2} \times \mathbf{3,38}) + \mathbf{10} \quad (5.57)$$

$$\textbf{Latencia Efectiva} = \mathbf{37,05[ms]} \quad (5.58)$$

$$\text{Ro} = 93.2 - \left(\frac{37,05}{40}\right) = 92,27 \quad (5.59)$$

$$\mathbf{R} = \mathbf{92,27} - (\mathbf{0,0041} \times \mathbf{2,5}) \quad (5.60)$$

$$\mathbf{R} = \mathbf{92,26} \quad (5.61)$$

$$\text{MOS} = 1 + (0,035 \times 92,26) + (0.000007) \times 92,26 \times (92,26 - 60) \times \times (100 - 92,26) \quad (5.62)$$

$$\mathbf{MOS} = \mathbf{4,39} \quad (5.63)$$

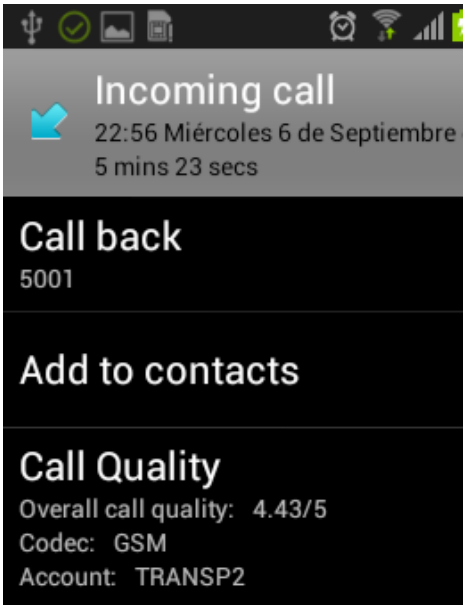


Figura 5.71: **MOS usando códec GSM con VPA ext. 5002**

5.3.2. Comparación entre WiFi con y sin virtualización

Observando los valores de la Tabla 7, se observa que en el caso del throughput, los valores tanto, usando VPA como sin usar VPA, se mantienen en ambos escenarios en sus niveles teóricos aproximadamente, para los dos CODEC utilizados.

Parámetro	Escenario					
	WiFi Sin Virtualización		WiFi Con Virtualización			
Extensión	4002		4002	5002	4002	5002
Dirección IP	192.168.0.12		192.168.0.12	10.0.0.14	192.168.0.12	10.0.0.14
Interfaz	wlp2s0		AP1	AP2	AP1	AP2
Códec	G.711uLaw	GSM	G.711uLaw		GSM	
Throughput [Kbps]	110	60	110	110	60	60
Latencia Promedio [ms]	10,00	10,00	20,09	20,26	20,09	20,28
Jitter Promedio [ms]	5,84	5,75	10,60	4,00	10,36	3,38
Latencia Efectiva [ms]	31,68	31,49	51,3	38,26	50,81	37,05
Paquetes Perdidos [%]	0	0	0,43	0,41	0,43	0,41
Factor R	**92,41**	**92,41**	**91,91**	**92,23**	**91,92**	**92,26**
MOS	**4,39**	**4,39**	**4,38**	**4,39**	**4,38**	**4,39**

Tabla 7: Comparación de parámetros de QoS y QoE entre WiFi sin VPA y con VPA

Con respecto a la latencia, se observa en el caso de WiFi utilizando VPA, que la latencia se ha incrementado al doble del valor obtenido en el escenario sin VPA. Este incremento es predecible, por cuanto en este escenario, los VPA “compiten” por transmitir sus paquetes a través de un mismo canal. Por lo que el incremento de la latencia es el resultado del tiempo que le toma a uno de los VPA, esperar a que el otro VPA deje de transmitir.

Para el caso del jitter, los valores se duplican en el escenario con VPA utilizando la interfaz virtual AP1 por la que se transmiten las llamadas de la red 192.168.0.0/24 (SSID: BODEGA), comparados con los valores obtenidos a través de la interfaz AP2. Adicionalmente, la distancia a la que se encontraron los terminales en el momento de la captura de los paquetes de VoIP, ya que los terminales de la red 10.0.0.0/8 (SSID: Transportación), estuvieron a una distancia desde el demostrador,

ligeramente menor (10cm aprox.) que los terminales usados para probar la otra red.

En el caso de los parámetros de QoE, el MOS mantuvo su valor para el escenario sin VPA para los dos CODEC usados. En el escenario con VPA, los valores de MOS obtenidos durante las llamadas a través de la interfaz inalámbrica virtual AP2, son similares a los del escenario sin VPA para ambos CODEC. Sin embargo, en el caso de la interfaz virtual inalámbrica AP1, se aprecia una ligera disminución del MOS, producto de la latencia y el jitter introducidos.

5.3.3. Correlación entre parámetros de Calidad de Servicio y Calidad de Experiencia de usuario

Los datos sombreados en azul en las Tabla 7, fueron utilizados para establecer la correlación entre los parámetros de QoS y QoE que se muestran a continuación junto con los gráficos obtenidos.

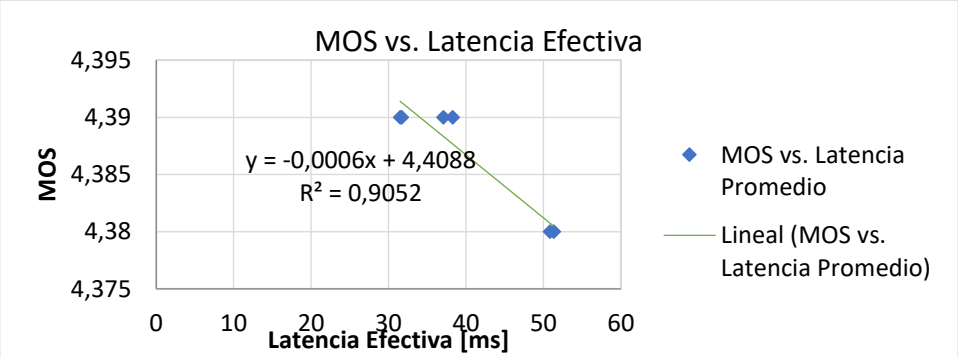


Gráfico 5.1: Correlación MOS vs. Latencia Efectiva

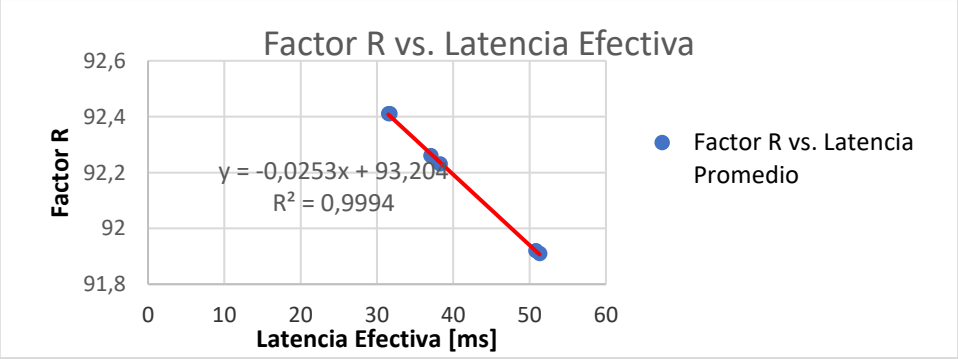


Gráfico 5.2: Correlación Factor R vs. Latencia Efectiva

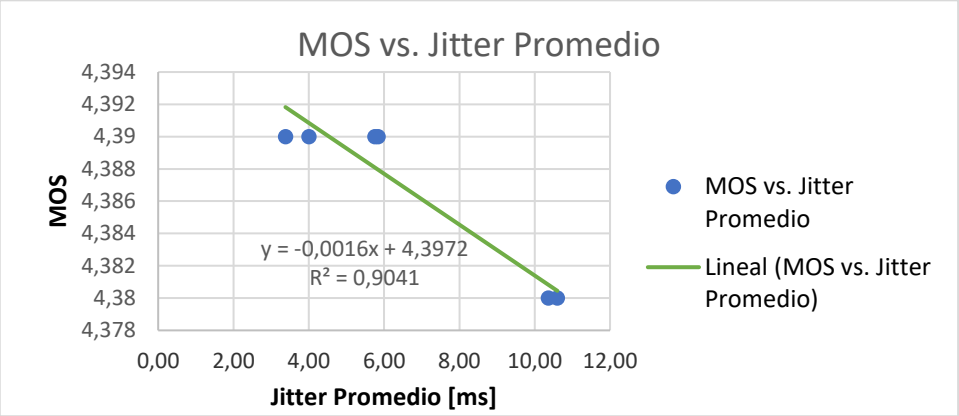


Gráfico 5.3: Correlación MOS vs. Jitter Promedio

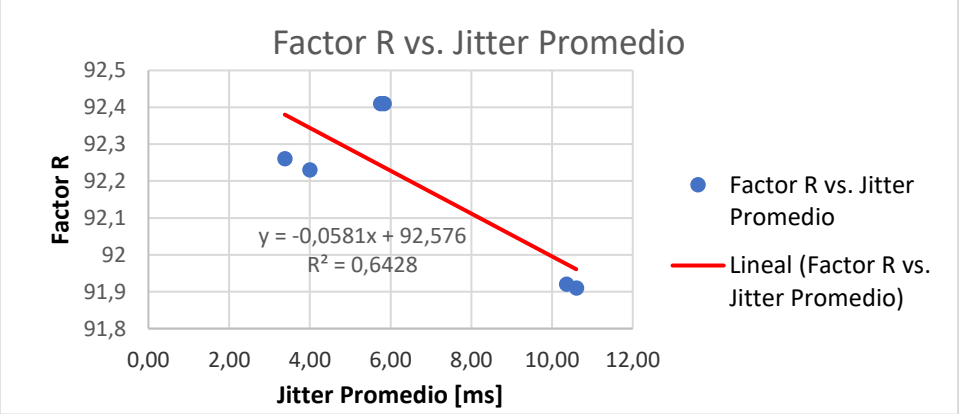


Gráfico 5.4: Correlación Factor R vs. Jitter Promedio

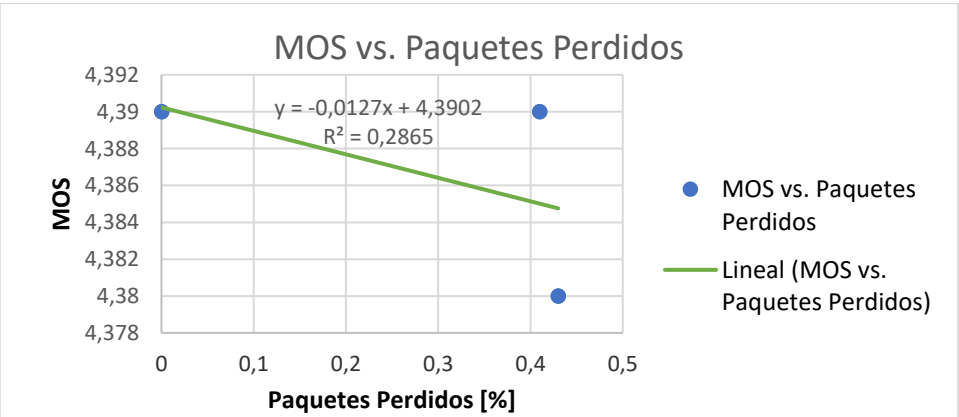


Gráfico 5.5: Correlación MOS vs. Paquetes Perdidos

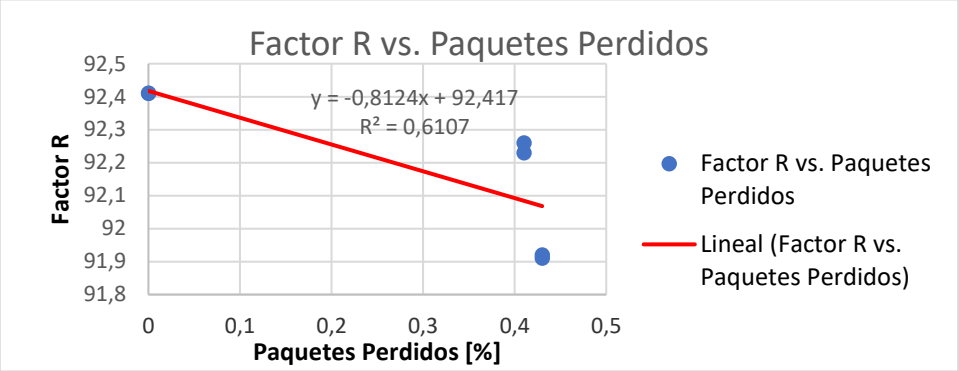


Gráfico 5.6: Correlación Factor R vs. Paquetes Perdidos

Como se puede observar en los gráficos anteriores, existe una relación inversamente proporcional entre los parámetros de QoS y los parámetros de QoE, donde se aprecia que el aumento de los valores de Latencia, Jitter y pérdida de paquetes, tienden a disminuir la QoS.

En cuanto a los coeficientes de determinación, se observa que existe una correlación fuerte del MOS y del factor R con la latencia efectiva. Una correlación fuerte existe también entre el MOS y el jitter promedio. Sin embargo, en el caso del factor R, la correlación de este con el jitter promedio es moderada.

El factor R también tiene una correlación moderada con la pérdida de paquetes, mientras que con el MOS la correlación con la pérdida de paquetes es baja.

5.4. Estudio Económico y de Viabilidad del Demostrador Propuesto

En esta sección se verifica si es viable desde el punto de vista financiero, el implementar la tecnología de virtualización inalámbrica, utilizando para ello un equipo físico real, con prestaciones similares a la del Demostrador planteado en este trabajo.

Los análisis realizados, se basan en la comparación de los valores actuales neto (VAN) resultantes de la implementación de dos redes inalámbricas: una sin VPA utilizando 2 PA físicos y otra utilizando dos VPA en un solo PA físico. Los análisis serán realizados sobre dos períodos de tiempo: 1 y 2 años.

5.4.1. Cálculo de los costos de implementación

Para la evaluación de los costos de implementación, los cuales se muestran en la Tabla 8, se han tomado como referencia los costos asociados con la utilización de Puntos de Acceso marca Aruba-HP modelo Instant 225. Se trata de un equipo con la funcionalidad de Multi SSID, permitiendo la implementación de varias redes virtuales con uno solo de estos equipos.

Costos	Valor
Costo por PA	$1.256,90
Costo de Mantenimiento Anual por PA	$140,00
Costo Total Anual Consumo Energía por PA (131,4Kw/h)	$25,00

Tabla 8: Costos asociados a un AP

5.4.2. Cálculo del Valor Actual Neto del Demostrador

El cálculo del Valor Actual Neto (VAN), se lo realiza de acuerdo con la ecuación 5.1.

$$\mathbf{VAN} = -\mathbf{I_0} + \sum_{\mathbf{t=1}}^{\mathbf{n}} \frac{\mathbf{F_t}}{\mathbf{(1+k)^t}} = -\mathbf{I_0} + \frac{\mathbf{F_1}}{\mathbf{(1+k)}} + \frac{\mathbf{F_2}}{\mathbf{(1+k)^2}} + \cdots \frac{\mathbf{F_n}}{\mathbf{(1+k)^n}} \qquad (5.64)$$

Donde I_0 es la inversión inicial; F_t son los flujos de dinero, *k* es la tasa de interés para la inversión siendo en en este caso un valor de 0.093 y *t* el tiempo sobre el cual se evaluará la viabilidad de la implementación del proyecto.

Costos e Ingresos por Inversión	Inversión Inicial Escenario sin VPA	Flujo Año 1	Inversión Inicial Escenario con VPA	Flujo Año 1
Costo Total por PA	-2513,80	0,00	-1.256,90	0,00
Costo Total de Mantenimiento Anual por PA	0,00	-280,00	0,00	-140,00
Costo Total Anual por consumo de energía	0,00	-50,00	0,00	-25,00
Ingresos por ventas, asociadas al uso de VoIP sobre WiFi	0,00	1.540,00	0,00	1.540,00
Total Anual (F)	-2.513,80	1.210,00	-1.256,90	1.375,00

Tabla 9: Inversiones iniciales y flujos de dinero anual - período 1 año

En ambos casos (t=1 año y t=2años), se ha añadido un Ingreso resultante de ventas que han podido ser realizadas por la utilización de un sistema de comunicación basado en VoIP sobre WiFi. Las inversiones iniciales y los flujos de dinero anuales para el período de 1 año se muestran en la Tabla 9.

Utilizando los valores mostrados en la Tabla 9 sobre la ecuación 5.1, se procede a calcular el VAN en ambos escenarios.

VAN Escenario sin VPA (Horizonte de 1 año):

$$\mathrm{VAN} = -\mathrm{I}_0 + \frac{\mathrm{F}_1}{(1+\mathrm{k})} = -2.513{,}80 + \frac{1.210{,}00}{(1+0.093)} \tag{5.65}$$

$$\mathbf{VAN} = \mathbf{-1.406{,}76} \tag{5.66}$$

VAN Escenario con VPA (Horizonte de 1 año):

$$\mathrm{VAN} = -\mathrm{I}_0 + \frac{\mathrm{F}_1}{(1+\mathrm{k})} = -1.256{,}90 + \frac{1.540{,}00}{(1+0.093)} \tag{5.67}$$

$$\mathbf{VAN} = \mathbf{1{,}11} \tag{5.68}$$

Para el caso del período de evaluación de 2 años, los valores de Inversión inicial y los flujos anuales de dinero se muestran en la Tabla 10.

Costos e Ingresos por Inversión	Inversión Inicial Escenario sin VPA	Flujo Año 1	Flujo Año 2	Inversión Inicial Escenario con VPA	Flujo Año 1	Flujo Año 2
Costo Total por PA	-2513,80	0,00	0,00	-1.256,90	0,00	0,00
Costo Total de Mantenimiento Anual por PA	0,00	-280,00	-280,00	0,00	-140,00	-140,00
Costo Total Anual por consumo de energía	0,00	-50,00	-50,00	0,00	-25,00	-25,00
Ingresos por ventas, asociadas al uso de VoIP sobre WiFi	0,00	883,00	883,00	0,00	883,00	883,00
Total Anual (F)	-2.513,80	553,00	553,00	-1.256,90	718,00	718,00

Tabla 10: Inversiones iniciales y flujos de dinero anual - período 2 años

Del mismo modo que en el caso con período de 1 año, se utiliza la ecuación 5.1 para evaluar ambos escenarios, ahora sobre un período de 2 años.

VAN Escenario sin VPA (Horizonte de 2 años):

$$\mathrm{VAN} = -\mathrm{I}_0 + \frac{\mathrm{F}_1}{(1+\mathrm{k})} + \frac{\mathrm{F}_2}{(1+\mathrm{k})^2} \tag{5.69}$$

$$\mathrm{VAN} = -2.513{,}80 + \frac{533{,}00}{(1+0.093)} + \frac{533{,}00}{(1+0.093)^2} \tag{5.70}$$

$$\mathbf{VAN = -1.544{,}96} \tag{5.71}$$

VAN Escenario con VPA (Horizonte de 2 años):

$$\mathrm{VAN} = -\mathrm{I}_0 + \frac{\mathrm{F}_1}{(1+\mathrm{k})} + \frac{\mathrm{F}_2}{(1+\mathrm{k})^2} \tag{5.72}$$

$$\mathrm{VAN} = -1.256{,}90 + \frac{718{,}00}{(1+0.093)} + \frac{718{,}00}{(1+0.093)^2} \tag{5.73}$$

$$\mathbf{VAN = 1{,}02} \tag{5.74}$$

5.4.3. Análisis de resultados de Estudio Económico

Con los resultados obtenidos para el demostrador planteado, en ambos casos (t=1año y t=2años), el VAN siempre es positivo en el escenario que utiliza VPA.

En vista de que las reglas de decisión definidas para el indicador VAN establecen que, de ser el VAN mayor a cero, la inversión debe aceptarse, por lo que se demuestra, que la implementación del demostrador es viable financieramente.

CONCLUSIONES Y RECOMENDACIONES

Como conclusiones se puede señalar, que la virtualización de redes inalámbricas presenta un potencial desarrollo en el futuro próximo en vista de los avances que están teniendo otras tecnologías a las cuales servirá para su despliegue tales como IoT y la telefonía móvil 5G.

La virtualización de redes inalámbricas presenta un campo de aplicaciones variado, tanto en el campo académico como en el comercial, que va desde escenarios de pruebas (como el implementado en el presente trabajo), optimización de infraestructura tecnológica empresarial, hasta el acompañamiento en el despliegue de nuevas tecnologías.

Para el caso del demostrador planteado, se comprobó que existe un aumento en los valores de Jitter, Latencia y Pérdida de paquetes cuando se utiliza VAP en una red WiFi que transporta VoIP, si se compara con una transmisión de comunicaciones de VoIP en una red que no utiliza VAP.

Mediante las pruebas realizadas, se verificó que el aumento del Jitter, Latencia y Pérdida de paquetes disminuyen los parámetros de QoE. Mientras que, en relación al throughput, este se mantiene en sus niveles teóricos al utilizar VPA.

En el aspecto financiero, queda demostrado que es conveniente invertir en el despliegue de la virtualización de una red inalámbrica, en vista del impacto positivo que esta tiene sobre la re - utilización de recursos.

La medición de los parámetros de QoS, arrojaron valores dentro de los rangos tolerables, lo que permite una QoE satisfactoria para el usuario. Razón por la cual se comprueba la viabilidad de utilizar la virtualización de redes inalámbricas de tipo WiFi, para transmitir llamadas de VoIP.

Como recomendación, se podría verificar el desarrollo mediante software de este demostrador tanto en computadores como en dispositivos móviles inteligentes, como teléfonos y tablets, apoyado en el uso de sistemas operativos de código libre vigentes en la actualidad (Ej. Android) y a su vez analizar el comportamiento de la VoIP en este escenario.

Además, realizar simulaciones de esta tecnología, para estudiar el desempeño de la VoIP bajo diferentes condiciones de: equipos, capacidades de enlace, ubicaciones...

También, se podrían emprender investigaciones en el campo del marco regulatorio, las ventajas y desventajas de su implementación a gran escala.

Adicionalmente, profundizar en la investigación del Balanceo de Carga como caso de uso de la Virtualización de Redes Inalámbricas.

Otra posible aplicación del presente trabajo sería la de implementar un banco de pruebas que apoye la enseñanza de diferentes tecnologías de telecomunicaciones, basado en la virtualización de redes inalámbricas.

Por otro lado, se podrían estudiar las repercusiones de implementar esta tecnología en Convergencia de Redes.

Se recomienda además, profundizar en el estudio del comportamiento de las comunicaciones de VoIP en una red inalámbrica virtualizada, utilizando esquemas de seguridad más sofisticados.

Finalmente, se recomienda utilizar el demostrador planteado, como una opción para explicar mediante una analogía, la operación de un Proveedor de Infraestructura y de un Operador Móvil Virtual.

BIBLIOGRAFÍA

[1] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck y R. Boutaba, «Network Function Virtualization: State-of-the-Art and Research Challenge,» *IEEE Communications Surveys & Tutorials,* vol. 18, nº 1, pp. 236-262, 2016.

[2] M. K. Chowdhury y R. Boutaba, «A survey of networks virtualization,» *Journal Computer Networks,* vol. 54, nº 5, pp. 862-876, 2009.

[3] S. d. E. ITU-T, «Recommendation ITU-T Y.3011: Framework of network virtualization for future networks,» Ginebra, 2012.

[4] T. T. W. Yee y K. Wipusitwarkun, «Network Virtualization and Aplication Areas,» *International Journal of Advances in Science and Technology,* pp. 8-14, 2014.

[5] M. El Barachi, N. Kara, S. Rabah y M. Forgues, «An open virtual multi-services networking architecture for the future internet,» *Journal of Internet Services and Applications,* vol. 6, nº 3, 2015.

[6] G. Aljabari y E. Eren, «Virtualization of Wireless LAN Infrastructures,» *Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems,* pp. 837-841, 2011.

[7] Open Networking Foundation, «Software-Defined Networking: The New Norm for Networks,» 2012.

[8] S. d. E. ITU-T, «Recommendation ITU-T Y.3300: Framework of software-defined networking,» Ginebra, 2014.

[9] M. Yang, Y. Li, D. Jin, L. Zeng, X. Wu y A. Vasilakos, «Software-Defined and Virtualized Future Mobile and Wireless Networks: A Survey,» *Journal Mobile Networks and Applications,* 2014.

[10] M. Jammal, T. Singh, A. Shami, R. Asal y Y. Li, «Software defined networking: State of the art and research challenges,» *Computer Networks,* vol. 72, pp. 74-98, 2014.

[11] T. Kittappa, "Virtual Access Points: Performance Impacts in an 802.11 environment and Alternative Solutions to overcome the problems," 2006. [Online]. Available: https://community.arubanetworks.com/aruba/attachments/aruba/115/1358/1/AppNote.MultipleBSSIDs.pdf.

[12] J. Kim y J.-B. Ko, «On the AP Virtualization of IEEE 802.11 WLANs,» *3rd International Confrerence on Green and Human Information Technology (ICGHIT),* pp. 237-242, 2014.

[13] H. Coskun, I. Schieferdecker y Y. Al-Hazmi, «Virtual WLAN: Going beyond Virtual Access Points,» vol. 17, 2009.

[14] Y. Al-Hazmi y H. De Meer, «Virtualization of 802.11 Interfaces for wireless Mesh Networks,» *Eighth International Conference on Wireless On-Demand Network Systems and Services (WONS),* 2011.

[15] L. Xia, S. Kumar, X. Yang y P. Gopalakri, «Virtual WiFi: Bring Virtualization from Wired to Wireless,» *VEE '11 Proceedings of the 7th ACM SIGPLAN/SIGOPS international conference on Virtual execution environments,* pp. 181-192, 2011.

[16] R. Chandra, P. Bahl y P. Bahl, «MultiNet: Connecting to multiple IEEE 802.11 Networks Using a Single Wireless Card,» *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies,* vol. 2, pp. 882-893, 2004.

[17] G. Aljabari y E. Eren, «Virtual WLAN: Extension of Wireless Networking into Virtualized Environments,» *International Journal of Computing,* vol. 10, nº 4, pp. 322-329, 2011.

[18] K. Nakauchi y Y. Shoji, «WiFi Network Virtualization to Control the Connectivity of a Target Service,» *IEEE Transactions on Network and Service Management,* vol. 12, nº 2, 2015.

[19] O. Hersent, IP Telephony Deploying VoIP Protocols and IMS Infrastructure, Segunda ed., Chichester: John Wiley & Sons Ltd., 2011.

[20] H. Wen, P. K. Tiwary y T. Le-Ngoc, Wireless Virtualization, New York: Springer, 2013, p. 112.

[21] C. Liang y R. Yu, «Wireless Network Virtualization: A Survey, Some Research Issues and Challenges,» *IEEE Communications Surveys & Tutorials,* vol. 17, nº 1, pp. 358-380, 2015.

[22] L. F. Espino Barrios, «Virtualización de Redes como elemento clave para Cloud Computing,» 2009. [En línea]. Available: http://docshare01.docshare.tips/files/23439/234397328.pdf.

[23] M. K. Chowdhury y R. Boutaba, «Network Virtualization: State of the Art and Research Challenges,» *Journal IEEE Communications Magazine,* vol. 47, nº 7, pp. 20-26, Julio 2009.

[24] M. Pilla Barcellos, D. Stefani Marcon y R. Ruas Oliveira, «Virtualized Networks for Cloud Computing: State of the Art and Research Challenges,» de *ACM Symposium on Applied Computing*, Coimbra, 2013.

[25] M. Chiosi, D. Clarke, C. Cui, J. Benitez, U. MIchel, K. Ogaki, M. Fukui, D. Delisle, I. Guardini, D. López, F. Ruhl y P. Sen, «Network Functions Virtualisation: An Introduction, Benefits, Enablers, Challenges & Call for Action,» de *SDN and OpenFlow World Congress*, Darmstadt, 2012.

[26] European Telecommunications Standards Institute (ETSI), «Network Functions Virtualisation (NFV); Architectural Framework,» Sophia Antipolis Cedex, 2013.

[27] X. Foukas, M. K. Marina y K. Kontovasilis, «Software Defined Networking Concepts,» de *Software Defined Mobile Networks (SDMN)*, Chichester, UK., John Wiley & Sons, Ltd, 2015, pp. 21-44.

[28] Open Networking Foundation, «OpenFlow Switch Specification,» Open Networking Foundation, Palo Alto, 2013.

[29] H. Kazemitabar, S. Ahmed, K. Nisar, A. Said y H. Hasbullah, «A comprehensive review on VoIP over Wireless LAN networks,» *Computer Science Letters,* vol. 2, nº 2, pp. 1-16, 2010.

[30] F. Thernelius, «SIP, NAT, and Firewalls,» Ericsson, Estocolmo, 2000.

[31] J. Lakkakorpi, «Voice in Packets: RTP, RTCP, Header Compression, Playout Algorithms, Terminal Requirements and Implementations,» *IP Telephony Protocols, Architecures and Issues,* pp. 31-38, 2001.

[32] Institute of Electrical and Electronic Engineers (IEEE), «Estándar IEEE 802.11-2012,» 2012.

[33] H. Sinnreich y A. B. Johnston, Internet Communications Using SIP, Segunda ed., Indianapolis: John Wiley & Sons, Inc, 2006.

[34] B. Hartpence, Packet Guide to Voice over IP, Sebastopol: O´Reily, 2013.

[35] W. A. Flanagan, VoIP and unified communications: Internet telephony and the future voice network, Hoboken: John Wiley & Sons, Inc, 2012.

[36] I. E. T. Force, «IETF Tools,» [En línea]. Available: https://tools.ietf.org/html/rfc4566. [Último acceso: 29 Abril 2017].

[37] Institute of Electrical and Electronic Engineers, «IEEE 802.11TM WIRELESS LOCAL AREA NETWORKS - Working Group for WLAN Standards,» [En línea]. Available: http://www.ieee802.org/11/. [Último acceso: 11 Mayo 2017].

[38] Institute of Electrical and Electronic Engineers (IEEE), «IEEE Standards Association-802_11 1997,» [En línea]. Available: http://standards.ieee.org/develop/project/802.11.html. [Último acceso: 12 Mayo 2017].

[39] Wi-Fi Alliance, «Wi-Fi Alliance,» [En línea]. Available: http://www.wi-fi.org/who-we-are/history. [Último acceso: 12 Mayo 2017].

[40] 3COM, «Computer Science - University of Colorado,» [En línea]. Available: http://www.cs.colorado.edu/~rhan/CSCI_7143_Fall_2007/Papers/IEEE_802_11b.pdf. [Último acceso: 15 Mayo 2017].

[41] M. Haenggi, «802.11 Data link Layer,» [En línea]. Available: http://www3.nd.edu/~mhaenggi/NET/wireless/802.11b/Data%20Link%20Layer.htm. [Último acceso: 18 Mayo 2017].

[42] M. Gast, 802.11 Wireless Networks: The Definitive Guide, O´Reilly, 2002.

[43] L. Peterson y B. Davie, «Computer Networks: A systems approach,» [En línea]. Available: https://booksite.elsevier.com/9780123850591/Lab_Manual/Lab_09.pdf. [Último acceso: 25 mayo 2017].

[44] Wireshark Foundation, «Wireshark,» [En línea]. Available: https://www.wireshark.org/. [Último acceso: 5 Mayo 2017].

[45] VoIPMonitor, «VoIPMonitor,» [En línea]. Available: http://www.voipmonitor.org/. [Último acceso: 5 Mayo 2017].

[46] TamoSoft, «CommView,» [En línea]. Available: http://www.tamos.com/products/commview/. [Último acceso: 5 Mayo 2017].

[47] AppNeta, «PathTest - Free Network Capacity Test!,» [En línea]. Available: http://info.appneta.com/Path-Test.html. [Último acceso: 5 Mayo 2017].

[48] H. Kazemitabar, S. Ahmed, K. Nisar, A. B. Said y H. B. Hasbullah, «A Survey on Voice over IP over Wireless LANs,» *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering,* vol. 4, nº 11, pp. 1617-1623, 2010.

[49] K. AlAlawi y H. Al-Aqrabi, «Quality of Service Evaluation of VoIP over Wireless Networks,» *2015 IEEE 8th GCC Conference Exhibition,* pp. 1-6, 2015.

[50] M. Finneran, Voice over WLANs: The Complete Guide, Burlington: Elsevier, 2008.

[51] O. Salcedo, D. López y C. Hernández, «SCIELO,» Facultad de Tecnologia Universidad Distrital Francisco José de Caldas, 13 Noviembre 2011. [En línea]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0123-921X2012000400013. [Último acceso: 31 Agosto 20117].

[52] J. Joskowicz y R. Sotelo, «Medida de la calidad de voz en redes IP,» *Memorias de Trabajos de Difusión Científica y Técnica,* nº 5, 2007.

[53] P. Zach, M. Pokorný y J. Balej, «Voice quality estimation in wireless networks,» *Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis,* vol. 63, nº 6, pp. 2179-2185, 2015.

[54] S. d. E. ITU-T, «Recomendation G.107: The E-model, a computational model for use in transmission planning,» 2015.

[55] H. Zhang, Z. Gu y Z. Tian, «QoS Evaluation Based on Extend E-Model in VoIP,» de *13th International Conference on Advanced Communication Technology (ICACT2011)*, Seoul, 2011.

[56] Pingman Tools, «How is MOS calculated in PingPlotter Pro?,» [En línea]. Available: https://www.pingman.com/kb/article/how-is-mos-calculated-in-pingplotter-pro-50.html.

[57] VoiceHost, «Call Quality - MOS or Mean Opinion Scores,» VoiceHost, [En línea]. Available: https://www.voicehost.co.uk/help/call-quality-mos-or-mean-opinion-scores. [Último acceso: 5 Septiembre 2017].

[58] D. Malone y J. Dunne, «Monitoring VoIP call quality using improved simplified E-model,» de *Conference: Networking and Communications (ICNC), 2013 International Conference on Computing*, 2013.

[59] Cisco Systems, «Voice Over IP - Per Call Bandwidth Consumption,» Cisco Systems, 13 Abril 2016. [En línea]. Available: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html. [Último acceso: 30 Agosto 2017].

[60] H. Wen, «Virtualization and Software-Defined Infrastructure Framework for Wireless Access Networks,» Montreal, 2014.

[61] K.-K. Yap, Y. Yiakoumis y M. Kobayashi, «Separating Autehntication, Access and Accounting: A Case Study with OpenWiFi,» 2011.

[62] L. Cai, Y. Xiao, X. Shen y J. W. Mark, «VoIP over WLAN: Voice capacity, admission control, QoS, and MAC,» *International Journal of Communication System,* vol. 19, nº 4, p. 491–508, 2006.

[63] Y. Li, «Virtualization in Wireless Network,» 2014. [En línea]. Available: http://www.cse.wustl.edu/~jain/.

[64] GENI, «GENI Wiki,» [En línea]. Available: http://groups.geni.net/geni.

[65] E. Hossain y M. Hassan, «5G Cellular: Key Enabling Technologies and Research Challenges,» *IEEE Instrumentation and Measurement Magazine,* vol. 18, nº 3, pp. 11-21, 2015.

[66] L. B. Le, V. Lau, E. Jorswieck, N.-D. Dao, A. Haghighat, D. I. Kim y T. Le-Ngoc, «Enabling 5G mobile wireless technologies,» *EURASIP Journal onWireless Communications and Networking,* 2015.

[67] N. Bizanis y F. Kuipers, «SDN and Virtualization Solutions for the Internet of Things: A Survey,» *IEEE Access,* vol. IV, pp. 5591-5606, 2016.

[68] N. Vogel, T. Fisher y B. Schulties, «Lifewire,» 02 Febrero 2017. [En línea]. Available: https://www.lifewire.com/measure-voice-quality-3426718. [Último acceso: 24 Agosto 2017].

[69] J. Malinen, «Hostapd,» [En línea]. Available: https://w1.fi/cgit/hostap/plain/hostapd/hostapd.conf. [Último acceso: 20 08 2017].

[70] Y. Zaki, L. Zhao y C. Goerg, «LTE Wireless Virtualization and Spectrum Management,» *Wireless and mobile networking conference (wmnc),* pp. 1-6, 2010.

[71] A. Ben Letaifa, A. Haji, M. Jebalia y S. Tabbane, «State of the Art and Research Challenges of new services architecture technologies: Virtualization, SOA and Cloud Computing,» *International Journal International Journal of Grid and Distributed Computing,* vol. 3, nº 4, pp. 69-88, 2010.

[72] X. Wang, P. Krishnamurthy y D. Tipper, «Wireless Network Virtualization,» *Journal of Communication,* vol. 8, nº 5, pp. 337-344, 2013.

[73] K. Katsalis, K. Choumas, T. Korakis, M. Anastasopoulos, A. Tzanakaki y J. Ferre, «Wireless Network Virtualization: The CONTENT Project Approach,» *IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks,* pp. 90-94, 1 Diciembre 2014.

[74] L. R. Bays, R. Ruas Oliveira, M. Pilla Barcellos, L. P. Gaspary y E. M. Madeira, «Virtual network security: threats, countermeasures, and challenges,» *Journal of Internet Services and Applications,* vol. 6, nº 1, 2015.

[75] M. Richart, J. Baliosian y J. Serrat, «Resource Slicing in Virtual Wireless Networks: A Survey,» *IEEE Transactions on Network and Service Management,* vol. 13, nº 3, 2016.

[76] I. Tanzeena Haque y N. Abu-Ghazaleh, «Wireless Software Defined Networking: A Survey and Taxonomy,» *IEEE Communications Surveys & Tutorials,* vol. 18, nº 4, pp. 2713-2737, 2016.

[77] V. Sivaraman, T. Moors, H. Habibi Gharakheili, D. Ong, J. Matthews y C. Russell, «Virtualizing the Access Network via Open APIs,» de *CoNEXT´13*, Santa Barbara, 2013.

[78] J. Carapinha y J. Jiménez, «Network Virtualization-a View from de Bottom,» *Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures,* pp. 73-80, 2009.

[79] Institute of Electrical and Electronic Engineers (IEEE), «IEEE Standards Association-P802.11ba,» [En línea]. Available: http://standards.ieee.org/develop/project/802.11ba.html. [Último acceso: 11 05 2017].

[80] Institute of Electrical and Electronic Engineers (IEEE), «IEEE 802 LAN/MAN Standards Committee-802_11 Timelines,» [En línea]. Available:

http://www.ieee802.org/11/Reports/802.11_Timelines.htm. [Último acceso: 11 Mayo 2017].

[81] Institute of Electrical and Electronic Engineers (IEEE), «IEEE Standards Association - 802_11ba,» [En línea]. Available: https://development.standards.ieee.org/get-file/P802.11ba.pdf?t=91647800003. [Último acceso: 11 05 2017].

[82] J. Leary y P. Roshan, 802.11 Wireless LAN Fundamentals, Cisco Press, 2003, p. 312.

[83] J. Geier, Designing and deploying 802.11 wireless networks, Cisco Press, 2010, p. 499.

[84] S. Banerji y R. Singha Chowdhury, «On IEEE 802.11: Wireless LAN Technology,» *International Journal of Mobile Network Communications & Telematics (IJMNCT),* vol. 3, nº 4, pp. 45-64, 2013.

[85] S. d. E. ITU-T, «Recomendación ITU-T P.862: Calidad de Transmisión Telefónica, Instalaciones Telefónicas y Redes Locales,» 2001.

[86] P. Roshan y J. Leary, 802.11 Wireless LAN Fundamentals, Cisco Press, 2003, pp. 174-196.

[87] O. Salcedo, D. López y C. Hernández, «Estudio comparativo de la utilización de ancho de banda con los protocolos SIP e IAX,» *Tecnura,* vol. 16, nº 34, pp. 171-187, 2012.

ANEXOS

A. ÍNDICE DE FIGURAS

Figura 1.1: Escenario propuesto para demostrar el despliegue y operación 3
Figura 1.2: Red Inalámbrica Virtualizada 4
Figura 1.3: Demostrador de aplicación de WiFi con topología de infraestructura 6
Figura 2.1: Arquitectura de la Virtualización de Redes 9
Figura 2.2: Modelo de Negocio de las NFV 15
Figura 2.3: Arquitectura de SDN 19
Figura 2.4: Componentes del Conmutador OpenFlow 22
Figura 2.5: Componentes de la VoIP 27
Figura 2.6: Cabecera del protocolo RTP 28
Figura 2.7: Encapsulamiento del audio 30
Figura 2.8: Encabezado TCP 31
Figura 2.9: Encabezado UDP 31
Figura 2.10: Encabezado IPv4 31
Figura 2.11: Formato de trama MAC del estándar IEEE 802.11n 32
Figura 2.12: Establecimiento de Sesión SIP usando INVITE 38
Figura 2.13: Línea de Requerimiento SIP en formato hexadecimal 38
Figura 2.14: Respuesta SIP a una INVITACION 39
Figura 2.15: Establecimiento de una Sesión SIP 41
Figura 2.16: Servidor SIP en modo Proxy 43
Figura 2.17: Servidor SIP en modo Redirect 43
Figura 2.18: Campos de un encabezado SDP 45
Figura 2.19: Ejemplo de una descripción de sesión 45
Figura 2.20: Conjunto de Servicios Básicos 49
Figura 2.21: Ubicación de DCF y PCF en la arquitectura de IEEE 802.11 50
Figura 2.22: Arquitectura VoIP sobre WLAN IEEE 802.11 51
Figura 2.23: Autenticación abierta 52
Figura 2.24: Autenticación de clave compartida 53
Figura 2.25: Problema del Nodo Oculto 54
Figura 2.26: Transición BSS 56
Figura 2.27: Transición ESS 57

Figura 2.28: Visualización del contenido de un paquete capturado 58
Figura 2.29: Análisis Gráfico de una llamada de VoIP 59
Figura 2.30: Visualización del intercambio de mensajes en una conexión de VoIP. 59
Figura 2.31: Parámetros registrados durante una llamada de VoIP 60
Figura 2.32: Vista de las conexiones monitoreadas por el programa 61
Figura 2.33: Panel de Monitoreo VoIP en CommView 62
Figura 2.34: Resultados de Medición de Capacidad 62
Figura 2.35: Procesamiento realizado sobre la señal de voz en PESQ 67
Figura 2.36: Gráfica de la relación entre el Factor R y el valor MOS 69
Figura 3.1: Representación en bloques de la Virtualización basada en Flujo Integrada y Sobrepuesta 72
Figura 3.2: Representación en bloques de la Virtualización basada en Protocolo: parcial y completa y Virtualización Basada en Espectro 73
Figura 3.3: Puntos de acceso virtualizados 75
Figura 3.4: Función PCF 77
Figura 3.5: Virtualización de AP basada en Hipervisor 78
Figura 3.6: Arquitectura de la técnica SR-IOV 82
Figura 3.7: Arquitectura propuesta para el despliegue de una red WiFi para transportar VoIP 83
Figura 3.8: Plataforma GENI desplegada en territorio de los EE.UU 85
Figura 3.9: Arquitectura con SDN propuesta para IoT 86
Figura 4.1: Arquitectura del Demostrador a implementar 89
Figura 4.2: Medición de variables sin WNV 91
Figura 4.3: Medición de variables con WNV 91
Figura 4.4: Dispositivos Clientes 93
Figura 4.5: Equipo que aloja el Demostrador 93
Figura 5.1: Dirección MAC de interfaz física WiFi 95
Figura 5.2: Modificación de Dirección MAC de interfaz WiFi 96
Figura 5.3: Creación de interfaces virtuales 97
Figura 5.4: Verificación de direcciones MAC de interfaces instaladas en equipo 97
Figura 5.5: Archivo de configuración VPA 1 98
Figura 5.6: Archivo de configuración VPA 2 98

Figura 5.7: Activación del VPA 1 99
Figura 5.8: Activación del VPA 2 99
Figura 5.9: Direccionamiento de Interfaces Virtuales creadas y reinicio de servicios de red 99
Figura 5.10: Pantalla principal de Virtual Box 100
Figura 5.11: Selección de archivo de imágen de Plataforma Elastix 100
Figura 5.12: Configuración de interfaz de red para máquina virtual 101
Figura 5.13: Asignación de nombre a máquina virtual 101
Figura 5.14: Asignación de memoria a máquina virtual 102
Figura 5.15: Elección de Disco Duro Virtual 102
Figura 5.16: Asignación de unidad de disco duro física 103
Figura 5.17: Asignación de espacio en disco 103
Figura 5.18: Selección de ubicación de Disco de Instalación de Servidor Elastix . 104
Figura 5.19: Arranque del instalador del Servidor Elastix 104
Figura 5.20: Preparación del disco duro 105
Figura 5.21: Configuración de las particiones del disco duro 105
Figura 5.22: Configuración de parámetros de red del servidor 106
Figura 5.23: Configuración de servidores DNS 106
Figura 5.24: Configuración nombre del servidor 107
Figura 5.25: Configuración de contraseña de root 107
Figura 5.26: Configuración de contraseña MySQL 108
Figura 5.27: Configuración contraseña Free PBX e Interfaz Web 108
Figura 5.28: Setup para verificación de parámetros de red del servidor Elastix 109
Figura 5.29: Verificación de las interfaces creadas en el sistema operativo anfitrión 111
Figura 5.30: Visualización de configuración de conmutadores virtuales 111
Figura 5.31: Visualización de las rutas por defecto para las redes creadas 112
Figura 5.32: Esquema propuesto para implementación del demostrador propuesto 113
Figura 5.33: Conexión a misma red de servidor de telefonía IP 113
Figura 5.35: Inicio de sesión servidor multissid2 114
Figura 5.34: Inicio de sesión servidor multissid1 114

Figura 5.36: Pantalla principal servidor multisssid1 114
Figura 5.37: Pantalla principal servidor multisssid2 115
Figura 5.38: Módulo para añadir extensiones 115
Figura 5.39: Parámetros de la extensión 116
Figura 5.40: Envío de configuración 117
Figura 5.41: Aplicación de configuración 118
Figura 5.42: Extensiones configuradas en el servidor multissid1 118
Figura 5.43: Extensiones configuradas en el servidor multissid2 119
Figura 5.44: Sección Ajustes del Softphone 119
Figura 5.45: Opción Agregar Cuentas 120
Figura 5.46: Selección del tipo de cuenta 120
Figura 5.47: Configuración de parámetros de la extensión en el softphone 120
Figura 5.48: Esquema propuesto para pruebas sin virtualización 121
Figura 5.49: Throughput en escenario sin VPA – Códec G.711 uLaw 122
Figura 5.50: Estadísticas de protocolos utilizados en la comunicación sin VPA – CODEC G.711 uLaw 122
Figura 5.51: Throughput en escenario sin VPA – Códec GSM 123
Figura 5.52: Estadísticas de protocolos utilizados en la comunicación sin VPA – CODEC GSM 124
Figura 5.53: Throughput en escenario con VPA – Códec G.711 uLaw 125
Figura 5.54: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC G.711 uLaw – SSID BODEGA 125
Figura 5.55: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC G.711 uLaw – SSID TRANSPORTACION 125
Figura 5.56: Throughput en escenario con virtualización – Códec GSM 126
Figura 5.57: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC GSM – SSID BODEGA 126
Figura 5.58: Estadísticas de protocolos utilizados en la comunicación con VPA – CODEC GSM – SSID TRANSPORTACION 127
Figura 5.59: Resumen de parámetros de QoS 128
Figura 5.60: Análisis de Tramas RTP en escenario sin VPA – CODEC G.711 uLaw 131

Figura 5.61: MOS usando CODEC G.711 uLAW sin VPA ext. 4002 132
Figura 5.62: Análisis de Tramas RTP en escenario sin VPA – CODEC GSM 133
Figura 5.63: MOS usando códec GSM sin VPA ext. 4002 133
Figura 5.64: Análisis de Tramas RTP en escenario con VPA – CODEC G.711 uLaw 134
Figura 5.65: MOS usando códec G.711 uLAW con VPA ext. 4002 135
Figura 5.66: Análisis de Tramas RTP en escenario con VPA – CODEC G.711 uLaw - SSID TRANSPORTACION 136
Figura 5.67: MOS usando códec G.711 uLaw con VPA ext. 5002 136
Figura 5.68: Análisis de Tramas RTP en escenario con VPA – CODEC GSM – SSID BODEGA 137
Figura 5.69: MOS usando códec GSM con VPA ext. 4002 138
Figura 5.70: Análisis de Tramas RTP en escenario con VPA – CODEC GSM – SSID TRANSPORTACION 138
Figura 5.71: MOS usando códec GSM con VPA ext. 5002 139

B. ÍNDICE DE TABLAS

Tabla 1: Servicios del estándar IEEE 802.11 54
Tabla 2: Resumen tamaño encabezados según nivel OSI 65
Tabla 4: Relación entre el factor R y la satisfacción del usuario 69
Tabla 5: Valores característicos de varios parámetros de diferentes CODEC 70
Tabla 6: Throughput y paquetes perdidos en ambos escenarios 129
Tabla 7: Parámetros para el cálculo de factor R y valor MOS 130
Tabla 8: Comparación de parámetros de QoS y QoE entre WiFi sin VPA y con VPA 140
Tabla 9: Costos asociados a un AP 144
Tabla 10: Inversiones iniciales y flujos de dinero anual - período 1 año 144
Tabla 11: Inversiones iniciales y flujos de dinero anual - período 2 años 145

C. ABREVIATURAS

ACK Acknowledgement

AOR Addres Of Record

API Application Programming Interfaces

BSS Basic Service Set

BSSID Basic Service Set Identifier

CNAME Canonical Name

CODEC COder/DECoder

CSI CODEC Sample Interval

CSMA/CA Carrier Sense Multiple Access with Collision Advoidance

CSMA/CD Carrier Sense Multiple Access with Collision Detection

CSRC Contributing Source

CSS CODEC Sample Size

DCF Distributed Coordination Function

DCF Distributed Coordination Function

DHCP Dynamic Host Configuration Protocol

DNS Domain Name System

DS Distribution System

ESS Extended Service Set

ETSI European Telecommunications Standards Institute

FQDN Full Qualyfied Domain Name

GENI Global Environment for Network Innovations

HTTP HyperText Transport Protocol

IDS Intrusion Detection System

IEEE Institute of Electrical and Electronics Engineers

InP Infrastructure Provider

IOMMU Input/Output Memory Management Unit

IP Internet Protocol

ITU-T International Union of Telecommunications – Standardization

LDAP Lightweight Directory Access Protocol

LLC Logical Link Control

MAC Media Access Control

MOS	Mean Opinion Score
MVNO	Mobile Virtual Network Operator
NAV	Network Allocation Vector
NF	Network Functions
NFV	Network Function Virtualization
NFV	Network Functions Virtualization
ONF	Open Networking Foundation
OSI	Open Systems Interconnection
PCF	Point Coordination Function
PCIe	Periphereal Component Interconnect express
PESQ	Perceptual Speech Quality Measurement
PPS	Packet Per Seconds
PSM	Power Saving Management
QoE	Quality of Experience
QoS	Quality of Service
RTCP	Real-Time Control Protocol
RTP	Real-Time Protocol
RTT	Round Trip Time
SDR	Software Defined Radio
SIP	Session Initiation Protocol
SMTP	Simple Mail Transport Protocol
SR-IOV	Single Root Input/Output Virtualization
STA	Station
STA	Station
TCP	Transport Control Protocol
TPS	Total Packet Size
TSP	Telecommunications Service Provider
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
URI	Uniform Resource Identifier

VLAN	Virtual Local Area Network
VM	Virtual Machine
VNF	Virtual Network Functions
VPN	Virtual Private Network
WM	Wireless Medium
WNV	Wireless Network Virtualization

D. DETALLE DE VARIABLES ARCHIVO DE CONFIGURACION HOSTAPD

A continuación, se describe el contenido de un archivo de configuración típico para el paquete *Hostapd*, el cual se puede encontrar en [69].

Nombre del dispositivo de red sobre el cual se creará el Virtual PA
interface=AP1

SSID que será utilizado en las tramas de administración IEEE 802.11
ssid=BODEGA

 # Modo de Operación (a= IEEE 802.11a (5GHz), b=IEEE 802.11b (2.4GHz), g=IEEE 802.11g #(2.4GHz), ad=IEEE 802.11ad (60GHz). Las opciones a/g son usadas también con IEEE 802.11n #(HT) para especificar la banda). Para IEEE 802.11ac (VHT), este parámetro necesita ser establecido #como hw_mode=a.
hw_mode=g

 # Número de canal, de acuerdo al estándar IEEE 802.11
channel=7

 # Interfaz donde los paquetes etiquetados 802.1q deben aparecer cuando se utiliza un servidor #RADIUS para determinar cuál estación VLAN está activa.
vlan_tagged_interface=AP1

 # Cuando hostapd crea una interfaz VLAN sobre la interfaz configurada en el parámetro #vlan_tagged_interface, se necesita saber cómo nombrarla.
vlan_naming=0

 # Autenticación basada en la dirección MAC de la estación. Se requiere un controlador que use #hostapd para cuidar el procesamiento de la trama de administración. 0 = aceptar a menos que esté #en la lista denegados; 1 = denegar a menos que esté en la lista aceptados; 2= usar servidor externo RADIUS.
macaddr_acl=0

#Tipo de interfaz del controlador (hostap/wired/none/nl80211/bsd). Se utiliza nl80211 con todos los #controladores Linux mac80211.
diver=nl80211

 # Si se habilita IEEE 802.11n (HT); 0=deshabilitado, 1=habilitado. Se necesita habilitar WMM para #habilitar la funcionalidad HT completamente. Se utiliza #hw_mode=g (2.4GHz) y hw_mode=a (5GHz) #para especificar la banda.
ieee80211n=1

 # Este parámetro es enviado a los clientes WMM cuando se asocian. Será usado por los clientes #WMM para las tramas transmitidas a los PA.
wmm_enabled=1

 # Define el algoritmo de autenticación a utilizar en la conexión. Se pueden usar los dos algoritmos #existentes: Sistema Abierto, y de Clave compartida. El campo de bit para seleccionar el algoritmo #permitido pueden tomar los valores: bit = 0 –> Autenticación de Sistema Abierto; bit = 1 -> #Autenticación de Clave Compartida (Requiere WEP).
auth_algs=1

 # Envía SSID vacíos en los beacons e ignora las tramas de petición de prueba que no especifican #completamente un SSID, por ejemplo, las estaciones que requieren conocer el # SSID.
ignore_broadcast_ssid=0

 # Este es un campo de bit que puede ser usado para habilitar WPA y/o WPA2. Si bit = 0 ->WPA; bit = #1 -> WPA2.
wpa=0

 # Establece el conjunto aceptado de algoritmos de administración de claves (WPA-PSK, WPA-EAP o #ambos). Las entradas se separan con espacios.
wpa_key_mgmt=WPA-PSK

 # Conjunto de algoritmos de encriptación aceptados: CCMP, TKIP, o ambos.
rsn_pairwise=CCMP

 # Claves compartidas previamente para WPA-PSK.
wpa_passphrase=123456789

E. CONFIGURACION SERVIDOR DHCP

E.1. CONFIGURACION ARCHIVO DE INTERFACES DEL SERVIDOR DHCP

Establecer las interfaces por las que se entregarán las direcciones IP, dentro del archivo de interfaces del servidor DHCP:

root@srvmultissidvoip:/home/jmoreira# gedit /etc/default/isc-dhcp-server

INTERFACES="AP1 AP2"

E.2. CONFIGURACION DEL SERVICIO ISC-DHCP-SERVER

Abrir y modificar el archivo de configuración del servicio ISC-DHCP-SERVER:

```
root@srvmultissidvoip:/home/jmoreira# gedit /etc/dhcp/dhcpd.conf
ddns-update-style none;
authoritative;
log-facility local7;
subnet 192.168.0.0 netmask 255.255.255.0 {
  range 192.168.0.11 192.168.0.100;
  option subnet-mask 255.255.255.0;
  option routers 192.168.0.1;
  option broadcast-address 192.168.0.255;
  default-lease-time 600;
  max-lease-time 7200;
}
subnet 10.0.0.0 netmask 255.0.0.0 {
  range 10.0.0.11 10.0.0.100;
  option subnet-mask 255.0.0.0;
  option routers 10.10.10.1;
  option broadcast-address 10.255.255.255;
  default-lease-time 600;
  max-lease-time 7200;
}
```

Reiniciar el servicio de red desde la ventana de *Terminal*:

root@srvmultissidvoip:/home/jmoreira# /etc/init.d/networking restart

Reiniciar el servicio *DHCP* desde la ventana de *Terminal*:

root@srvmultissidvoip:/home/jmoreira# /etc/init.d/isc-dhcp-server restart

F. COTIZACION PUNTO DE ACCESO

• GUAYAQUIL: Cdla. La Garzota, Av. Hermano Miguel, Sl.6 y Agustín Freire
• Contactos:(04) 5125636 - 5125492 - 5125534 - info@gensystems.net
(04) 2626148 - 2626328 - 2626480 (ext. 413)

• QUITO: Alpallana E7 - 123 y Whymper, Ed.Caminos del parque Of.:202
• Contactos: (02) 3530130 - (02) 3238387 - infouio@gensystems.net
RUC: 0992238402001

GENSYSTEMS
Tecnología y Software de Negocios

www.gensystems.net

Contribuyente Especial

Nombre: ING. JOSE MOREIRA
Empresa:
Dirección:
Telef: 0990894188

Cotización Nº: C20170913-0050
Cot. Solicitada: **12/09/2017**
Cot. Enviada: **13/09/2017**
Hora: **14:20**

Atendiendo su gentil solicitud tenemos el agrado de ofrecer la siguiente cotización de .....

Parte No.	Descripcion	Cant.	P.Unit	P.Total
	Aruba Instant IAP-225 (RW) 802.11n/ac Dual 3x3:3 Radio Integrated Antenna AP	1	$ 1,256.90	$ 1,256.90
	Aruba 1Y FC NBD Exch IAP 225 SVC [for JW240A]	1	$ 55.10	$ 55.10
	PD-3510G-AC 15.4W 802.3af PoE 10/100/1000Base-T Ethernet Midspan Injector	1	$ 64.70	$ 64.70
	AP-220-MNT-W1W Flat Surface Wall/Ceiling White AP Basic Flat Surface Mount Kit	1	$ 17.60	$ 17.60
	Instalación y Mantenimiento		$ 140.00	$ 140.00
	SON: MIL QUINIENTOS TREINTA Y CUATRO 30/100 DOLARES AMERICANOS + IVA		**TOTAL**	**$1,534.30**
Notas			12 % IVA	$184.12

a) *Instalación incluye: Instalación del equipo INDOOR.
b) *Mantenimiento incluye: 1 limpieza del equipo y actualización del firmware en el periodo de un año (a partir de la instalación).

Condiciones de Negociacion:
Forma de pago: CONTADO
Tiempo de entrega: 3 a 6 días laborables si hay stock, 30 a 45 días si se agota el stock

Validez de la oferta: 30 días, mientras no hayan cambios de aranceles

GENSYSTEMS S.A.
Atentamente,
Henry Javier
Ejecutivo de Cuentas Corporativas