UNIVERSIDAD NACIONAL DE LOJA

FACULTAD DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

TÍTULO

DISEÑO E IMPLEMENTACIÓN DE UN PILOTO DE RED WIRELESS MESH BASADA EN FIRMWARE LIBRE UTILIZANDO NODOS 802.11 EN EL A.E.I.R.N.N.R. DE LA UNIVERSIDAD NACIONAL DE LOJA.

TESIS DE GRADO PREVIO A OPTAR POR EL TÍTULO DE INGENIERA EN ELECTRÓNICA Y TELECOMUNICACIONES

AUTORA

María Fernanda Benítez León.

DIRECTOR

Ing. Daniel Armando Jaramillo Chamba, Mg. Sc.

LOJA-ECUADOR

2018

CERTIFICACIÓN

Señor Ingeniero
Daniel Armando Jaramillo Chamba, Mg. Sc.
DIRECTOR DE TESIS

CERTIFICA:

Haber dirigido, asesorado, revisado y corregido el presente trabajo de tesis de grado, en su proceso de investigación cuyo tema versa en **"DISEÑO E IMPLEMENTACIÓN DE UN PILOTO DE RED WIRELESS MESH BASADA EN FIRMWARE LIBRE UTILIZANDO NODOS 802.11 EN EL A.E.I.R.N.N.R. DE LA UNIVERSIDAD NACIONAL DE LOJA."**, previa a la obtención del título de **Ingeniera en Electrónica y Telecomunicaciones**, realizado por la señorita egresada **María Fernanda Benítez León**, la misma que cumple con la reglamentación y políticas de investigación, por lo que autorizo su presentación y posterior sustentación y defensa.

Loja, Enero del 2018

Ing. Daniel Armando Jaramillo Chamba, Mg. Sc.
DIRECTOR DE TESIS

AUTORÍA

Yo**, MARÍA FERNANDA BENÍTEZ LEON**, declaro ser autora del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi trabajo de tesis en el Repositorio Institucional- Biblioteca Virtual.

Firma:

Cédula: 1105826760

Fecha: 27/06/2018

CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LA AUTORA, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL, Y PUBLICACION ELECTRÓNICA DEL TEXTO COMPLETO.

Yo, **MARÍA FERNANDA BENÍTEZ LEÓN,** declaro ser autora de la tesis titulada: **"DISEÑO E IMPLEMENTACIÓN DE UN PILOTO DE RED WIRELESS MESH BASADA EN FIRMWARE LIBRE UTILIZANDO NODOS 802.11 EN EL A.E.I.R.N.N.R. DE LA UNIVERSIDAD NACIONAL DE LOJA"** como requisito para optar al grado de: **INGENIERA EN ELECTRÓNICA Y TELECOMUNICACIONES**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los veintisiete días del mes de junio del dos mil dieciocho.

Firma: …….

Autor: María Fernanda Benítez León

Cédula: 1105826760

Dirección: Loja (Ciudadela del Chofer La Banda: Belisario Moreno y Vilcabamba).

Correo Electrónico: mfer04bl@gmail.com

Teléfono: 072541397 **Celular:** 0981287313

DATOS COMPLEMENTARIOS

Director de Tesis: Ing. Daniel Armando Jaramillo Chamba, Mg. Sc.

Tribunal de Grado: Ing. Juan Gabriel Ochoa Aldeán, Mg. Sc.

Ing. Andy Fabricio Vega Leon, Mg. Sc.

Ing. Manuel Augusto Pesantez González, Mg. Sc.

DEDICATORIA

A Dios, que con su infinita misericordia ha permitido cristalizar mi más anhelado sueño, porque de Él, por Él y para Él son todas las cosas.

A mis padres, Amada y Mijaíl, por ser el más perfecto ejemplo de amor y apoyo incondicional, y por cada uno de los sacrificios que han hecho para que sea mejor persona.

A mis hermanas, Katty Alejandra y Paula Doménica por brindarme su apoyo constante y sincero que me ha motivado a seguir luchando.

A mis abuelos, Antonio y Martha por cada consejo lleno de sabiduría que cultivo en mí el deseo de superación; y a María Olivia por cuidarme desde el cielo.

María Fernanda

AGRADECIMIENTOS

Dejo constancia de mi agradecimiento a la Universidad Nacional de Loja recinto de grandeza y sabiduría, que a través de la carrera de Ingeniería en Electrónica y Telecomunicaciones de la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables, me ha permitido formarme académicamente.

Agradezco al Ing. Daniel Armando Jaramillo Chamba Mg. Sc. quien como director de la presente tesis me brindo su guía y asesoramiento constante en el trabajo que hoy pongo a vuestra consideración.

Hago extensivo este agradecimiento a mi familia y a cada una de las personas que de alguna u otra manera me orientaron en forma oportuna y desinteresada.

LA AUTORA

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN II
AUTORÍA III
CARTA DE AUTORIZACIÓN IV
DEDICATORIA V
AGRADECIMIENTOS VI
ÍNDICE DE CONTENIDOS VII
ÍNDICE DE FIGURAS X
ÍNDICE DE TABLAS XIII
1. TÍTULO 1
2. RESUMEN 2
ABSTRACT 3
3. INTRODUCCIÓN 4
4. REVISIÓN LITERARIA 6
4.1 CONCEPTOS GENERALES 6
4.1.1 Red. 6
4.1.2. Red Mesh. 6
4.1.3. Red Inalámbrica 6
4.2 IEEE 802.11 9
4.2.1 Variantes 10
4.2.2 Frecuencia y Potencia 13
4.2.3 Nivel Mínimo de señal recibida y perdidas en cables 15
4.3 WIRELESS MESH NETWORK 15
4.3.1 Clasificación de WMNs 16
4.3.2 Ejemplos de Redes Mesh 18
4.3.3 Equipos para redes MESH 19
4.4 PROTOCOLOS DE ENRUTAMIENTO 20
4.4.1 Protocolos de enrutamiento basados en topología 21
4.5 OLSR (Optimized Link State Routing) 23
4.5.1 MPR (MultiPoint Relays) 24
4.5.2 Mensajes y Formato de Paquete 25
4.5.3 Repositorios de información y cálculo de la tabla de enrutamiento 30

4.5.4 Métrica 32
4.6 LINKSYS WRT54G/GS/GL 33
4.6.1 Características comunes de la serie WRT54G 33
4.6.2 WRT54GS v 3.0 35
4.7 FIRMWARES OPENSOURCE 36
4.7.1 Firmware OEM 37
4.7.2 DD-WRT 37
4.7.3 Freifunk 37
4.7.4 OpenWRT 38
4.8 OpenWRT 39
4.9 HERRAMIENTAS DE ANÁSLISIS 41
4.9.1 Wireshark 41
4.9.2 Iperf 42
4.9.3 Ping 43
4.9.4 Elastix 43
4.9.5 Wampserver 43
4.9.6 Softphone 44
4.9.7 Wifi Analizer 44
5. MATERIALES Y MÉTODOS 45
5.1 Simulación de la Red Mesh 45
5.2 Diseño de la Red 47
5.2.1 Requerimientos de Hardware y Software 47
5.2.2 Parámetros a configurar 48
5.3 Implementación de la Red Mesh Inalámbrica 52
5.3.1 Etapa 1: Instalación 52
5.3.2 Etapa 2: Configuración 54
5.3.3 Etapa 3: Pruebas 56
5.4 Parámetros a evaluar 64
6. RESULTADOS 67
6.1 Simulación 67
6.2 Configuración de la Red 69
6.3 ANÁLISIS DEL PROTOCOLO. 72
6.3.1 Primer Escenario: Funcionamiento básico del protocolo. 72
6.3.2 Segundo Escenario: Tiempo de convergencia de un nuevo nodo. 75

6.3.3 Tercer Escenario: Cálculo de una nueva ruta 76
6.4 ANÁLISIS DE LA RED.....76
6.4.1 Ping.....76
6.4.2 Iperf.....77
6.4.3 Prueba con Aplicaciones Reales.....78
6.5 COSTO DEL PROTOTIPO..... 82
7. DISCUSIÓN..... 84
8. CONCLUSIONES 88
9. RECOMENDACIONES 90
10. BIBLIOGRAFÍA 91
11. ANEXOS. 94
Anexo 1. Acrónimos 94
Anexo 2. Simulación NS3 99
Anexo 3. Instalación del Protocolo y de los Paquetes 102
Anexo 4. Configuración Mediante Interfaz Gráfica 103
Anexo 5. Archivos de Configuración. 107
Anexo 6. Visualización de la Red 111
Anexo 7. Archivo .bat.....113
Anexo 8. Configuración de las Extensiones y del Softphone.....114
Anexo 9. Tasa de Transferencias de Diferentes Aplicaciones 117

ÍNDICE DE FIGURAS

Figura 1. Modos de operación de 802.11 12
Figura 2. WMN y sus elementos 16
Figura 3. Clasificación de las Redes Inalámbricas Multisalto 18
Figura 4. Clasificación de Protocolos. 20
Figura 5. Selección de los MPRs en OLSR 25
Figura 6. Formato de paquete en OLSR 26
Figura 7. Detección de vecinos 28
Figura 8. Formato de mensaje HELLO 28
Figura 9. Formato de mensaje TC 29
Figura 10. Formato del mensaje MID 30
Figura 11. Formato de paquetes HNA 30
Figura 12. Flujo de Datos en OLSR 31
Figura 13. Router Linksys WRT54gs V3.0 36
Figura 14. Arquitectura interna del router Linksys WRT54GS v3.0 36
Figura 15. Página web de OpenWRT 40
Figura 16. Diagrama de trabajo 45
Figura 17. Topología de red que presenta el archivo OLSR-HNA de ns3. 46
Figura 18. Canales no solapados en la banda ISM 2,4 GHz 50
Figura 19. Sondeo de canales 50
Figura 20. Plano de la segunda panta del edifico de laboratorios. 51
Figura 21. Distribución de los equipos en el Laboratorio de Telecomunicaciones. 51
Figura 22. Etapas de la implementacion de la red 52
Figura 23. Instalación del firmware a través de Interfaz web 53
Figura 24. Reglas de firewall utilizadas para la creación de las topologías. 57
Figura 25. Pruebas realizadas para el análisis del protocolo y la red 57
Figura 26. Topología utilizada en el primer escenario 58
Figura 27. Topología de la red / 1er escenario 58
Figura 28. Grafica de la red/ 1er escenario 58
Figura 29. Topología utilizada para las pruebas del tercer escenario. 59
Figura 30. Resultado del comando Tracert/ Tercer escenario 60

Figura 31. Topología utilizada /3er escenario 60
Figura 32. Gráfica de la red/ 3er escenario 60
Figura 33. Escenario utilizado para el análisis de la red. 61
Figura 34. Resultado del comando Tracert/Topología para al análisis de la red 61
Figura 35. Topología de la red/ Análisis de la red 61
Figura 36. Gráfica obtenida/ Topología utilizada para el análisis de la red 61
Figura 37. Comando PING utilizado en las pruebas 62
Figura 38. Comandos para generación de tráfico en iperf. 62
Figura 39. Ubicación de los routers en cada piso del edificio de Laboratorios. 63
Figura 40. Disposición de los dispositivos para la realización de la prueba VoIP 63
Figura 41. Disposición de los dispositivos para la realización de la prueba HTTP 64
Figura 42. Archivo .pcap obtenido de la simulación 67
Figura 43. Detalles del frame 1 de la captura de paquetes 67
Figura 44. Detalles del frame 5 de la captura de paquetes 68
Figura 45. Detalles del frame 7 de la captura de paquetes 68
Figura 46. Detalles del frame 13 de la captura de paquetes 69
Figura 47. Tabla de vecinos, interfaz gráfica OpenWRT 69
Figura 48. Direcciones de las redes LAN y WLAN de cada nodo. 70
Figura 49. Topología de la red mesh, interfaz gráfica de OpenWRT 70
Figura 50. Redes anunciadas a través de los mensajes HNA 71
Figura 51. Visualización de la red, a través de la interfaz gráfica de OpenWRT 71
Figura 52. Visualizacion de la red mediante el plugin dot-draw 71
Figura 53. Captura de tráfico en la red mallada a través de Wireshark. 72
Figura 54. Frame 1 de la captura de tráfico de control de la red implementada 73
Figura 55. Frame 6 de la captura de paquetes de la red 74
Figura 56. Frame 19 de la captura de tráfico de la red 74
Figura 57. Frame 50 de la captura de paquetes en la red 75
Figura 58. Datos obtenidos al realizar ping. 77
Figura 59. Resultado obtenido de traceroute para análisis de retardo 77
Figura 60. Gráfica de resultados de inyeccion de tráfico TCP/UDP 78
Figura 61. Captura de paquetes, llamada a 1 salto 79
Figura 62. Captura de paquetes, llamada a 2 saltos 79

Figura 63. Retardo máximo obtenido en llamada VoIP 79
Figura 64. Jitter máximo obtenido en llamada VoIP.....79
Figura 65. Gráfica RETARDO (izquierda) y JITTER (derecha)- llamada 1 salto 80
Figura 66. Gráfica RETARDO (izquierda) y JITTER (derecha)-llamada 2 saltos 80
Figura 67. Throughput obtenido en la descarga del archivo. 1_SALTO 81
Figura 68. Throughput obtenido en la descarga del archivo. 2_SALTOS 81
Figura 69. Ingreso al nodo a través de la herramienta Putty 102
Figura 70. Cambio de contraseña. 103
Figura 71. Nombre del router y Timezone 103
Figura 72. Interfaz LAN configurada 104
Figura 73. Creación y configuración de la interfaz wlo 104
Figura 74. Configuración de la interfaz inalámbrica. 105
Figura 75. Elección de la interfaz en la que va a correr OLSR 105
Figura 76. Complementos de OLSR habilitados. 106
Figura 77. Configuración del parámetro HNA 106
Figura 78. Interfaz gráfica del servidor Elastix. 114
Figura 79. Extensión 101, creada a través de la interfaz web de Elastix.....115
Figura 80. Parámetros configurados en el Softphone Port Go en un PC 115
Figura 81. Habilitación del Softphone PortGo.....116

ÍNDICE DE TABLAS

Tabla 1. Bandas ISM Y U-NII utilizadas en IEEE 802.11 13
Tabla 2. Potencia máxima del transmisor para antenas con ganancia mayor a 6dbi..... 14
Tabla 3. Regla 3:1 para EIRP de enlaces PtP y PtMp 15
Tabla 4. Mensajes OLSR con su intervalo 27
Tabla 5. Configuración por defecto de las interfaces del Linksys WRT54gs v3.0 36
Tabla 6. Comparación de firmwares OpenSource para los enrutadores Linsksys 38
Tabla 7. Métodos para la instalación de firmwares 39
Tabla 8. Versiones de OpenWRT..... 40
Tabla 9. Versión instalada en cada router..... 48
Tabla 10. Direcciones asignadas a los nodos. 49
Tabla 11. Direcciones Ip del servidor y del AP 49
Tabla 12. ESSID y modo de operación de la red. 50
Tabla 13. Paquetes instalados en OpenWRT. 54
Tabla 14. Resultados primer y segundo escenario 76
Tabla 15. Resultados obtenidos de las pruebas realizadas con la herramienta IPERF 78
Tabla 16. Resultados obtenidos de las pruebas VoIP mediante Wireshark. 80
Tabla 17. Resultados obtenidos en la prueba de HTTP 82
Tabla 18. Presupuesto del prototipo..... 82
Tabla 19. Presupuesto del prototipo para ambientes outdoor 83
Tabla 20. Aplicaciones con sus diferentes tasas de transmisión. 117

1. TÍTULO

“DISEÑO E IMPLEMENTACIÓN DE UN PILOTO DE RED WIRELESS MESH BASADA EN FIRMWARE LIBRE UTILIZANDO NODOS 802.11 EN EL A.E.I.R.N.N.R. DE LA UNIVERSIDAD NACIONAL DE LOJA.”

2. RESUMEN

El presente documento ofrece conocimientos teóricos y técnicos sobre la Redes Inalámbricas Malladas, el protocolo OLSR, el firmware OpenWrt y los routers Linksys WRT54GS versión 3.0. El desarrollo del proyecto se realizó en varias fases, primero se procedió a elegir la versión (release) adecuada del firmware de acuerdo a la versión y características técnicas de los equipos. Elegido el release compatible con los dispositivos, este se instaló en cada uno de ellos, al igual que el demonio del protocolo OLSR y los complementos necesarios para el correcto funcionamiento del mismo.

Para la configuración de la red se tomó en cuenta algunos factores previos, como el canal, el nombre de la red, el parámetro HNA entre otros, realizadas las configuraciones de las interfaces LAN, WLAN, el firewall, y los parámetros del protocolo, se obtuvo una gráfica de la red por medio de la instalación de diferentes plugins. El prototipo se implementó en el Laboratorio de Telecomunicaciones, y consta de 6 nodos, para el análisis del protocolo se utilizó la herramienta Wireshark, con la que se realizó el análisis de la captura de tráfico permitiendo conocer los campos y la información que maneja el protocolo en los mensajes de control, también se tomó el tiempo de convergencia de un nuevo router a la red, y el tiempo que tarda la red en calcular una nueva ruta cuando un nodo cae mediante el comando ping. Para analizar la viabilidad de la red se realizaron pruebas que permitieron analizar el impacto del número de saltos en métricas como el delay y tasa de transmisión para lo cual se utilizó herramientas de diagnóstico como ping, iperf y Wireshark.

Como parte complementaria se realizó una simulación para conocer el comportamiento del protocolo en la red, el software elegido fue NS3 ya que contiene ejemplos sobre el protocolo OLSR, el archivo modificado fue “olsr-hna”, en el cual se cambió el número de routers y su posición; y se procedió a analizar la captura de tráfico de la simulación a través de Wireshark.

ABSTRACT

This document offers theoretical and techniques knowledges about Wireless Mesh Networks, the OLSR protocol, the OpenWrt firmware and WRT54GS Linksys routers version 3.0. The development of the project was made in several phases, first it was proceeded to choose the appropriate release of the firmware according to the version and techniques characteristics of the devices. Chosen the compatible release with the devices, it was installed in each one of them, as well as the demon of OLSR protocol with the necessary complements in order to its proper functioning.

For the configuration of the network some previous factors were taken in account, like the channel, name of the network, HNA parameter among others, it was made the configurations LAN interfaces, WLAN, the firewall, and parameters of the protocol, it was got a graphic of the network through the installation of differents plugins. The prototype was implemented in Telecommunications Laboratory, and consists of 6 nodes, for the analyses of the protocol the tool Wireshark was used, with which was made the analyses of the traffic capture, allowing us to know the fields and information what the protocol handles in the control messages, furthermore it was taken the converged time of a new router to the network, and the time what network takes for calculate a new path when the node falls by the ping command. To analyze the viability of the network tests were performed which allowed us to analyze the impact of the number of hops in metrics such as the delay and transmission rate for which diagnostic tools such as ping, iperf and Wireshark were used.

As a complementary part one simulation was made to know the behavior of the protocol in the network, the software chosen was NS3 because its contains examples about the OLSR protocol, the modified file was “olsr-hna”, in which was changed the number of routers and its spot; and it was proceeded to analyze the traffic capture of the simulation trough wireshark.

3. INTRODUCCIÓN

En la actualidad las nuevas tecnologías de la información en particular las redes han evolucionado la forma tradicional de comunicarse, facilitando el acceso a la información y su implementación debe responder a las necesidades de los usuarios. Las comunicaciones inalámbricas han tomado fuerza debido a su gran diversidad de configuraciones y métodos de acceso, los firmwares libres para diferentes dispositivos se han popularizado en los últimos años pues presentan un gran abanico de funcionalidades que permiten controlar varias características del nodo contribuyendo a la interoperabilidad entre dispositivos de diferentes tecnologías. Las Redes Inalámbricas Malladas fusionan la tecnología inalámbrica con la topología de red en malla, dotándole de características como tolerancia a fallos, escalabilidad, movilidad entre otras. Esta tecnología se encuentra en constante evolución convirtiéndose en una propuesta innovadora para diversas actividades. Por ser redes descentralizadas son menos costosas, además pueden implementarse en entornos urbanos y rurales lo que las transforma en una tecnología adecuada para un sin número de aplicaciones. En la actualidad es posible encontrar gran cantidad de información y diversos proyectos desplegados en varias partes del mundo. Es por eso que se plantea la construcción de un prototipo de Red Inalámbrica Mallada (del inglés, Wireless Mesh Network) con OpenWRT y protocolo OLSR en un ambiente de Laboratorio, que permitirá a través de varias pruebas analizar el funcionamiento de la red y del protocolo, además esta investigación servirá para que los estudiantes tengan una guía de la configuración de los equipos existentes en el Laboratorio y para la investigación de diferentes protocolos para redes Mesh. Para la realización de esta investigación se plantearon los siguientes objetivos:

- Configurar los routers Linksys existentes en el Laboratorio de Telecomunicaciones del A.E.I.R.N.N.R de la Universidad Nacional de Loja con el firmware OpenWRT y protocolo OLSR.

- Implementar el prototipo de Red Inalámbrica Mallada en un ambiente de Laboratorio.

- Analizar el comportamiento del protocolo OLSR sobre una red mallada configurada con firmware libre.
- Verificar el correcto funcionamiento y desempeño del piloto, mediante parámetros como la tasa de transmisión y retardo con ayuda de herramientas de análisis de red.

4. REVISIÓN LITERARIA

4.1 CONCEPTOS GENERALES

4.1.1 Red.

Una red es un conjunto de nodos conectados entre sí de forma alámbrica o inalámbrica con la finalidad de enviar y/o recibir datos. En la actualidad las redes forman parte de nuestro entorno, se usan para servicios financieros, procesos de fabricación, mensajería electrónica, teleconferencia, televisión por cable, Voz sobre IP (VoIP, del inglés Voice-over-IP), etc. El término topología en una red define la configuración, sea física o lógica. La topología física de red, indica cómo están los nodos conectados entre sí mientras que una topología lógica indica de qué manera se realizan la comunicación dentro de la red. Son varias las topologías que una red de datos puede tener, entre ellas: anillo, árbol, estrella, malla entre otras.

4.1.2. Red Mesh.

Una red Mesh o mallada es aquella que tiene una topología física en malla, en donde cada nodo tiene un enlace punto a punto y dedicado con cualquier otro nodo, los nodos pueden estar conectados de manera total o parcial entre sí, de esta manera existen varios caminos para enviar y/o recibir datos. Algunas de las ventajas que presentan este tipo de red son robustez, escalabilidad, adaptabilidad y tolerancia a fallos. La principal desventaja que presenta, es la gran cantidad de cable y puertos entrada/salida de los nodos necesarios para ser implementada, lo que hace que sea una red costosa; aunque combinada con tecnología inalámbrica son una opción muy atractiva. (Forouzan, 2002)

4.1.3. Red Inalámbrica

Una Red Inalámbrica (WN, del inglés Wireless Network) es una red que no requiere un medio guiado para la transmisión, sino que realizan su comunicación a través de ondas electromagnéticas. Existen un sinnúmero de redes que utilizan tecnología inalámbrica para establecer comunicación entre sus nodos, como las redes celulares, redes LAN inalámbricas, entre otras. Entre las soluciones para redes LAN inalámbricas más

popular y que tiene más aceptación en el mercado, está el estándar IEEE 802.11. (Salazar, 2016)

De forma general se pueden identificar los siguientes componentes de una red inalámbrica:

- *Hosts inalámbricos:* dispositivos que actúan como sistema final, y que permite la interacción de la red y el usuario a través de la ejecución de las aplicaciones.
- *Enlaces inalámbricos:* la comunicación entre el host inalámbricos y la estación base se realiza a través de un enlace de comunicación inalámbrico, que tiene distintas velocidades y distancias dependiendo del estándar utilizado.
- *Estación base / STA:* es la encargada de enviar y/o recibir datos hacia/desde un host inalámbrico. En las redes LAN inalámbricas 802.11 los puntos de acceso AP (del inglés, Acces Point) actúan como estación base, mientras que en las redes celulares son las torres. (Kurose & Roos, 2010)

Las redes inalámbricas presentan tres tipos de configuraciones lógicas:

- ***Punto a Punto, PtP (del inglés, Point to Point)***

Es una relación uno a uno, es decir una conexión dedicada entre dos dispositivos inalámbricos, se suelen utilizar para redes de gran escala, incluso que abarquen ciudades. Este tipo de conexiones utilizan antenas direccionales para establecer la comunicación en la dirección deseada.

- ***Punto a Multipunto, PtMP (del inglés Point to Multipoint)***

Es una relación uno a muchos, la conexión se realiza entre varios dispositivos a un dispositivo central. Comúnmente el dispositivo central hace uso de antenas omni o semidireccionales y los otros dispositivos antenas semi o direccionales.

- ***Multipunto a Multipunto, MPtMP (del inglés Multipoint to Multipunto)***

Es una relación de muchos a muchos, que se asemeja a una red mallada inalámbrica, en este tipo de conexión todos los dispositivos se comunican entre sí. (Carpenter, 2008)

Las redes inalámbricas se pueden clasificar por el radio de cobertura o alcance, entre las cuales tenemos:

- ***Redes Inalámbricas de Área Personal***

Las WPAN (Redes Inalámbricas de Área Personal, del inglés Wireless Personal Area Network), son redes que permiten la comunicación inalámbrica en distancias pequeñas dentro de un POS (Espacio Operativo Personal, del inglés Personal Operating Space), con una cobertura máxima de 10 m aunque en la actualidad se están trabajando en estándares que permitan cubrir distancias mayores. Entre las tecnologías utilizadas en las WPANs se encuentran Bluetooth, RFID (Identificación Por Radio Frecuencia, del inglés Radio Frequency IDentification), Zigbee.

Las WPANs son soluciones económicas, con bajo consumo de energía y que puede ser implementada en una variedad de dispositivos de uso común, es por esto que este tipo de tecnología ha tomado fuerza en aplicaciones de domótica. El grupo encargado de desarrollar los estándares para las WPAN es el IEEE 802.15. Este estándar trabaja en las bandas ISM de 2,4GHz, de 16 canales con un ancho de banda de 5 MHz. (Grupo NAP del COIT, 2004) (Salazar, 2016)

- ***Redes Inalámbricas de Área Local***

Una WLAN (Red Inalámbrica de Área Local, del inglés Wireless Local Area Network), cubre áreas equivalentes al de un campus o un edificio con un alcance aproximado de 100 m. Esta tecnología se basa en el estándar IEEE 802.11 o también conocido como WiFi, y también en HiperLAN2[1] (del inglés High Performance Radio LAN 2.0). Estas tecnologías se diferencian por la banda de frecuencia que utilizan (Wifi generalmente 2.4GHz y 5GHz, HiperLAN2 5,15 y 5,3 GHz), la velocidad de transmisión (Wifi 300Mbps, HiperLAN2 54Mbps) y el área de cobertura. (Salazar, 2016)

- ***Redes Inalámbricas de Área Metropolitana***

Una WMAN (Red Inalámbrica de Área Metropolitana, del inglés Wireless Metropolitan Area Network), también conocidas como Bucle Local Inalámbrico (WLL, Wireless Local Loop) son redes que permiten la comunicación inalámbrica en áreas urbanas o ciudades. Basan su funcionamiento en el estándar IEEE 802.16 conocido también como WiMAX (Interoperabilidad Mundial para Acceso de Microondas, del inglés Worldwide

[1] Estándar inalámbrico europeo aprobado por la ETSI, esta tecnología opera en banda de frecuncia de 5 GHz y utiliza OFDM

Interoperability for Microwave Access). Pueden alcanzar distancias de hasta 10 Km. En las WMANs se pueden diferenciar dos partes, la estación base (BS) y los abonados o suscriptores de la estación base (SS), además utilizan una arquitectura punto-multipunto (PMP). (Kurose & Roos, 2010) (Salazar, 2016)

- ***Redes Inalámbricas de Área Amplia***

Las WWAN (Red Inalámbrica de Área Amplia, del inglés Wireless Wide Area Networks Area Network) pueden alcanzar distancias de más de los 50 kilómetros permitiendo la comunicación en una región (país o grupo de países), permiten la interconexión de varios sistemas de comunicaciones ayudando a que ésta sea cada vez más globalizada y suelen utilizar frecuencias con licencia. Este tipo de redes se pueden mantener en grandes áreas, tales como ciudades o países, a través de los múltiples sistemas de satélites o ubicaciones con antena atendidos por un proveedor de servicios de Internet. Existen principalmente dos tecnologías disponibles: la telefonía móvil y los satélites (Salazar, 2016)

4.2 IEEE 802.11

El comité 802.11 fue formado en 1990 por el IEEE (Instituto de Ingenieros Eléctricos y Electrónicos, del inglés Institute of Electrical and Electronics Engineers) con el propósito de desarrollar un método de acceso al medio, MAC (Control de Acceso al Medio, del inglés Medium Access Control) y una especificación del medio físico (PHY) para redes LAN inalámbricas, con el paso del tiempo se han publicado una variedad de normas que no son más que una versión mejorada de su antecesora. La primera versión del estándar IEEE 802.11 conocida como IEEE 802.11 legacy, se publicó en 1997 haciendo uso de frecuencias en bandas ISM de 2,4 GHz, con velocidades de 1 o 2 Mbps, y con el método de DSSS (Espectro ensanchado de secuencia directa, del inglés Direct Sequence Spread Spectrum) contemplando también la opción de Espectro expandido por salto de frecuencia (FHSS, Frecuency Hopping Spread Spectrum). 802.11 opera en el rango de frecuencias de 2,4 GHz a 2,485 GHz, dentro de esta banda de 85 MHz, 802.11 define 11 canales parcialmente solapados y solo el conjunto de canales 1, 6 y 11 es el único conjunto de tres canales no solapados.

4.2.1 Variantes

- **IEEE 802.11a:** su principal diferencia con el 802.11 legacy es que opera en la banda de UNII de 5GHz, permite una velocidad de transmisión de datos teórica de hasta 54 Mbps y velocidades reales de 20 Mbps mediante el uso de Multiplexación por División De Frecuencia Ortogonal, OFDM[2] (del inglés, Orthogonal Frecuency Division Multiplexing). Además las distancias que cubre se ven reducidas, en exteriores 300m y en interiores 90m a su máxima velocidad de transmisión ya que las frecuencias alta tienen un rango de cobertura más corto
- **IEEE 802.11b:** opera en la banda ISM 2,4GHz y es una extensión de 802.11 legacy; con velocidad de transmisión de datos de 5.5 y 11 Mbps gracias a la modulación DSSS y codificación CCK (Modulación por Código Complementario, del inglés Complementary Code Keying). Este estándar es el más popular y el que ha liderado en redes inalámbricas. Cuenta también con soporte de hasta 3 canales sin solapamiento y el radio de cobertura dependerá de la velocidad, número de usuarios, antenas, y otros parámetros; en espacios abiertos a su máxima velocidad permitida puede llegar a 120m y en interiores a 30m.
- **IEEE 802.11c:** permite la comunicación entre redes distintas, es muy utilizado debido al bajo costo que representan otras tecnologías.
- **IEEE 802.11d:** permite el uso internacional de redes 802.11 locales
- **IEEE 802.11e:** permite introducir nuevos mecanismo a nivel MAC ofreciendo QoS (del inglés, Quality of Service). A nivel de capa MAC se introduce una nueva función de coordinación llamado HCF (del inglés, Hybrid Coordination Function) con dos tipos de acceso: Función Mejorada de Distribución de Acceso al Canal, EDCA (del inglés, Enhanced Distributed Channel Access); y la Función HCF Mejorada de Distribución de Acceso al Canal, HCCA (del inglés, HCF Controlled Channel Access).
- **IEEE 802.11f:** permite la interoperabilidad entre dispositivos (roaming), funciona bajo el estándar 802.11g.
- **IEEE 802.11g:** aprobado en 2003, este estándar trabaja con la banda ISM de 802.11b y velocidades de trasmisión de 802.11a. Trabaja en la banda ISM 2,4 GHz y puede

[2] Técnica en donde la banda de frecuencia se divide en sub-canales y cada uno de ellos se modula individualmente

utilizar DSSS y OFDM como métodos de modulación, lo que permite que sea posible llegar a 54 Mbps teóricamente mientras que su velocidad real es de 25 Mbps, además es compatible con Wi-Fi, lo que permite la coexistencia entre ambos estándares.

- **IEEE 802.11h:** este estándar permite asignación dinámica de canales y control automático de potencia para minimizar los efectos de posibles interferencias, es una evolución de estándar 802.11 a
- **IEEE 802.11i:** permite implementar protocolos de seguridad y autentificación 802.11
- **IEEE 802.11n:** este estándar permite trabajar en las bandas de 2.4 y 5GHz con velocidades de hasta 600Mbps, haciendo uso de la tecnología conocida como MIMO (Múltiples Entrada Múltiples Salida, del inglés Multiple Input Multiple Output). (Grupo NAP del COIT, 2004) (Carpenter, 2008)

Las redes 802.11 a/b o g pueden operar en dos modos

- **Infraestructura**/ Conjuntos de Servicios Básicos, BSS (del inglés, Basic Server Set), en este modo la red inalámbrica se crea mediante un AP que normalmente tiene una conexión a una infraestructura cableada. Cada nodo móvil se comunica directamente con el AP, toda la comunicación se realiza a través del punto de acceso. Si dos o más BSS se conectan entre si utilizando una infraestructura esto se denomina Conjunto de Servicios Extendidos, ESS (del inglés, Extended Service Set)
- **Ad-Hoc**/ Conjunto de Servicios Básicos Independientes, IBSS (del inglés, Independent Basic Server Set), en este modo la red se establece por un conjunto de estaciones sin ninguna clase de infraestructura, este tipo de red se crean con dispositivos que poseen una interfaz inalámbrica, los dispositivos se comunican directamente sin necesidad de un AP. (SoCalFreeNet.org Wireless User Group, 2004). En la Figura 1 se presenta los dos modos de operación de 802.11

Un SSID (Identificador de Conjunto de Servicio, del inglés Service Set Identifier) es un identificador único, permite la asociación entre los dispositivos de red dependiendo del modo de operación. En las redes BSS o ESS, tiene entre 2 y 32 caracteres alfanuméricos y también se lo llama Identificador del Conjunto del Servicios Extendidos, ESSID (del inglés, Service Set Identifier Extended). En cambio el

Identificador del Conjunto de Servicios Básicos, BSSID (del inglés, Basic Service Set Identifier) tiene una longitud de 48 bits y es utilizado en las redes IBSS, suele ser la dirección MAC.

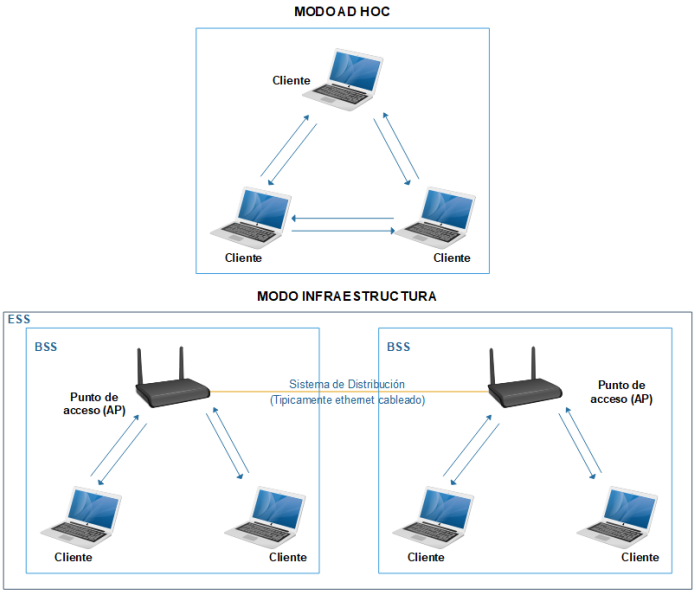

Figura 1. Modos de operación de 802.11

Fuente: Recuperado de Wireless Hacking: Projects por Wi-Fi Enthusiasts por SoCalFreeNet.org Wireless User Group

En la práctica los dispositivos 802.11a/b/g presentan 4 opciones o modos en los cuales puede operar:

1) ***Modo master/modo AP/modo infraestructura:*** en este modo se crea un servicio como el que proporciona un punto de acceso tradicional, en la cual se crea una red con un nombre y un canal (llamado SSID: Service Set Identifer). Las interfaces inalámbricas en modo máster solo se pueden comunicar inalámbricamente con interfaces asociadas a ellas en el modo administrado.
2) ***Modo administrado /modo cliente.*** Las interfaces inalámbricas en modo administrado se asocian a una red creada por el máster utilizando automáticamente el canal escogido por este. Las interfaces en modo administrado no se comunican entre sí directamente sino solamente a través de un máster con el cual están asociadas.
3) ***Modo ad hoc:*** en este modo el dispositivo puede crear una red multipunto a multipunto donde no hay dispositivos operando en modo administrativo o master, es decir la comunicación se realiza directamente entre vecinos. Los nodos deben compartir el mismo nombre y canal.

4) ***Modo monitor:*** este modo se utiliza para la escucha pasiva del tráfico de radio en un canal determinado, en este modo no se realiza transmisión datos y es muy útil cuando se analizan problemas en un enlace inalámbrico o para la observación del uso del espectro en al área correspondiente. (WNDW, 2013)

4.2.2 Frecuencia y Potencia

En Ecuador la Agencia de Regulación y Control de las Telecomunicaciones (ARCOTEL) es la entidad encargada de la administración, regulación y control de las telecomunicaciones y del espectro radioeléctrico y su gestión, así como de los aspectos técnicos de la gestión de medios de comunicación social que usen frecuencias del espectro radioeléctrico o que instalen y operen redes. Ecuador toma en cuenta e incluye muchas características técnicas del Reglamento de Radiocomunicaciones de la UIT (Unión Internacional de Telecomunicaciones).

- **Frecuencia**

La UIT provee de dos bandas sin licencia para las comunicaciones por radio, la banda ISM (Ciencia Médica Industrial, del inglés Industrial Scientific and Medical) y bandas U-NII (Infraestructura Nacional de Información Sin Licencia, del inglés Unlicensed National Information Infrastructure). El rango de frecuencias de las bandas ISM y U-NII es amplio pero el utilizado para la familia 802.11 se limita a la banda ISM de 2,4GHz y el rango de frecuencia de 5GHz de la banda U-NII, como se observa en la Tabla 1

Tabla 1. Bandas ISM Y U-NII utilizadas en IEEE 802.11

Banda de Frecuencia	Ambientes de operación	Ancho de Banda	Banda No icenciada
2400–2500 MHz	Indoor/outdoor	100 MHz	ISM
5.15–5.25 GHz	Solo operaciones indoor	100 MHz	U-NII
5.25–5.35 GHz	Indoor/outdor	100 MHz	
5.470–5.725 GHz	Indoor/outdor	255 MHz	
5.725–5.825 GHz	Operaciones outdoor	100 MHz	

Fuente: Recuperado de CWNA Certified Wireless Network Administrator por T. Carpenter

- **Potencia.**

Como 802.11a/b/g trabaja en bandas que no requieren autorización hay que tener en cuenta algunas de las limitaciones de potencia que presenta cada país, en el caso de Estados Unidos la FCC (del inglés, Federal Communications Commission) en la

Norma FCC 15.247 considera 1000 mW como potencia máxima, en cambio en los países europeos la ETSI (del inglés, European Telecommunications Standards Institute) permiten como potencia máxima 100mW para 802.11 b. Ecuador para redes de corto alcance en el artículo 23 del Reglamento de las Radiocomunicaciones de la ARCOTEL nos dice: *"**Art. 23.- Sistemas que no Requieren Autorización.-** Los usuarios del espectro radioeléctrico que operen equipos de radiocomunicaciones con potencias menores a 100 mW sin antenas directivas y que no correspondan a sistemas de última milla y los que operen al interior de locales, edificios y en general áreas privadas con potencias menores a 300 mW sin antenas exteriores, en cualquier tecnología, no requieren autorización del CONATEL."*

Cuando se utiliza antenas directivas con ganancias mayores a 6dBi, se tendrá que aplicar ciertas combinaciones entre la ganancia de la antena y potencia de salida de dispositivo radiador, estas combinaciones se encuentran descritas en la resolución del CONATEL 560 denominada "Sistemas de Modulación Digital de Banda Ancha", en la Tabla 2 se resume de manera breve la potencia máxima según el tipo de configuración del sistema.

Tabla 2. Potencia máxima del transmisor para antenas con ganancia mayor a 6dbi

Tipo de Configuración del Sistema	Bandas de Operación	Potencia máxima del Transmisor (mW)
Punto-punto	2400-2483.5 Mhz	1000
Punto-multipunto		
Móviles		
Punto-punto	5.15–5.25 GHz	50
Punto-multipunto		
Móviles		
Punto-punto	5.25–5.35 GHz	250
Punto-multipunto		
Móviles		
Punto-punto	5.470–5.725 GHz	250
Punto-multipunto		
Móviles		
Punto-punto	5.725–5.825 GHz	1000
Punto-multipunto		
Móviles		

Fuente: Recuperado de Anexo 1 de la resolución del CONATEL 560 "Sistemas de Modulación Digital de Banda Ancha"

En las bandas de 5150 - 5250 MHz, 5250 -5350 MHz y 5470 - 5725 MHz, la potencia de transmisión pico deberá reducirse en la cantidad de dB que superen la ganancia de la

antena direccional que exceda los 6 dBi, en cambio para la banda ISM en enlaces PtP y PtMp , se deberá aplicar la regla 3:1, es decir que a cada 3dBi que se aumente de ganancia de antena se tendrá que reducir 1dBm de la potencia máxima de salida del transmisor, como se observa en la Tabla 3.

Tabla 3. Regla 3:1 para EIRP de enlaces PtP y PtMp

Potencia del radiador intencional (dBm)	Ganancia de la antena (dBi)	EIRP	
		dBm	Watts
30	6	36	4
29	9	38	6.4
28	12	40	10
27	15	42	16
26	18	44	25
25	21	46	39.8
24	24	48	63
23	27	50	100
22	30	52	158

Fuente: Recuperado de CWNA Certified Wireless Network Administrator por T. Carpenter

4.2.3 Nivel Mínimo de señal recibida y perdidas en cables

El nivel mínimo de señal recibida, RSL (del inglés, Received Signal Level) o la sensibilidad del receptor es el nivel más bajo de señal que un dispositivo puede distinguir y se expresa en dBm negativos. Para operación en la banda de 2,4GHz la IEEE propone que los equipos deben tener una antena con una sensibilidad igual o menor a -80 dBm. El RSL mínimo depende de la tasa de transmisión, la tasa más baja (1 Mbps) tiene la sensibilidad más alta. Parte de la energía de la señal se pierde en los cables, los conectores y otros dispositivos desde los radios a las antenas, la pérdida depende del tipo de cable usado y de su longitud, la pérdida de señal para cables coaxiales cortos incluyendo los conectores es bastante baja, en el rango de los 2-3 dB.

4.3 WIRELESS MESH NETWORK

Una Red Inalámbrica Mallada, WMN (del inglés Wireless Mesh Network) se puede definir como aquella en donde cada nodo tiene un enlace punto a punto con cualquier otro nodo, los mismos pueden estar conectados de manera total o parcial entre sí, de tal manera que existan varios caminos para la transmisión de datos. Según Gilbert Held

(Held, 2005) una "*WMN representa una serie de transmisiones peer-to-peer y cada nodo realiza funciones de router y a su vez de repetidor*".

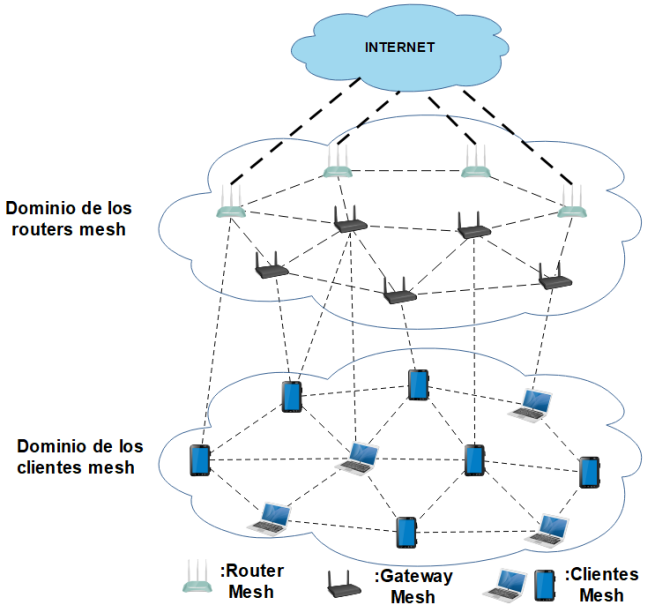

Figura 2. WMN y sus elementos

Fuente: Recuperado de Wireless Mesh Networks: Architectures and Porotocols por Hossain E

Las redes WMN usan retransmisión inalámbrica multisalto sobre una topología de malla. En la práctica para que los nodos puedan comunicarse directamente entre sí, deben operar en modo Ad-hoc, además de coincidir en el canal y nombre de red denominado ESSID (WNDW, 2013). En una arquitectura mesh se puede diferenciar dos partes, los routers mesh y los clientes mesh, los clientes mesh pueden ser cualquier tipo de dispositivos con una Tarjeta de Interfaz de Red Inalámbrica (NIC, del inglés Network Interface Cards) (WNDW, 2013), en la Figura 2 se puede observar bien diferenciadas estas dos partes

4.3.1 Clasificación de WMNs

Por su arquitectura las WMN se clasifican en 3 tipos:

- ***WMN plana:*** esta red está formada por clientes mesh, los mismos que actúan como hosts y enrutadores, esta arquitectura es la más cercana a una red inalámbrica ad hoc. La principal ventaja de esta arquitectura es su simplicidad, y sus desventajas incluyen la falta de escalabilidad de la red y las altas limitaciones de recursos.

- ***WMN jerárquica:*** en este tipo de arquitectura la red tiene niveles jerárquicos, los clientes mesh forman el nivel más bajo y se pueden comunicar con la red troncal que estará conformada por los enrutadores mesh. En la mayoría de los casos, los nodos WMN son nodos dedicados que forman una red troncal de WMN.
- ***WMN híbrida:*** es un caso especial donde se utiliza otras redes inalámbricas para la comunicación. Por ejemplo, redes celulares, redes WiMAX o redes satelitales. (Zhang, Luo, & Honglin, 2007)

Muchas de las veces se confunden las redes inalámbricas ad-hoc con las redes inalámbricas mesh, pero hay que tener claro que están redes son diferentes, aunque ambas utilizan retransmisión inalámbrica multisalto, existen diferencias entre ellas.

- ***Movilidad de los nodos,*** en las WMNs los nodos son relativamente estáticos, en cambio las redes inalámbricas ad-hoc están formadas en su mayoría de nodos móviles, la movilidad de los nodos de las WMNs es muy baja en comparación con las redes inalámbricas ad-hoc.
- ***Topología,*** las WMN presentan un topología estática, en las redes inalámbricas ad-hoc la topología de red cambia dinámicamente.
- ***Energía,*** debido a la topología estática que presentan las WMN, estas redes presentan un mejor almacenamiento de energía, en la redes ad-hoc una de las mayorías limitaciones es la energía
- ***Escenario de aplicación,*** a diferencia de las redes ad hoc inalámbricas, las WMN se utilizan tanto para aplicaciones militares como para civiles, como por ejemplo el suministro de servicios de Internet de bajo costo para centros comerciales, calles y ciudades.

También existen otras redes que utilizan retransmisión inalámbrica multisalto, entre las que tenemos: las redes de sensores inalámbricos, formadas por pequeños nodos denominados sensores que pueden recopilar parámetros físicos y transmitir a un nodo central de monitoreo y las redes inalámbricas híbridas tales como redes celulares y en bucles locales inalámbricos, WLL. (Zhang, Luo, & Honglin, 2007) En la Figura 3 se observa la clasificación de las redes inalámbricas multisalto.

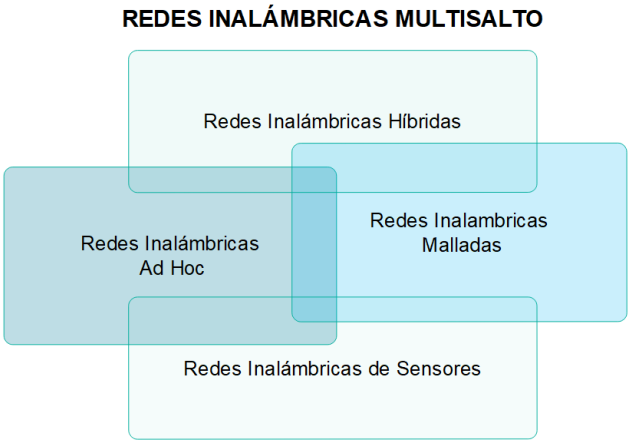

Figura 3. Clasificación de las Redes Inalámbricas Multisalto
Fuente: Recuperado de Wireless Mesh Networking, por Yan Zhang, Jijun Luo, Honglin Hu

4.3.2 Ejemplos de Redes Mesh

El uso masivo de las redes malladas inalámbricas comenzó con el nacimiento de las Redes Libres Comunitarias, que tienen como objetivo el acceso libre y sin restricciones a la información y comunicación, además de ser una herramienta que ayude a la disminución de la brecha digital. Generalmente este tipo de redes se despliegan en base a un firmware de libre distribución y con equipos de bajo coste. Algunas de las redes comunitarias más importantes son:

- **Freifunk:** es una de las redes comunitarias más antiguas y sin ánimo de lucro que da soporte a redes inalámbricas libres principalmente en regiones de Alemania. Gran parte de sus infraestructuras se basan en una arquitectura mallada. Freifunk es parte de un movimiento internacional para redes inalámbricas abiertas, que actualmente cuenta con un amplio apoyo en muchos países, incluidos los países pobres. Actualmente, los nodos trabajan con OLSR y B.A.T.M.A.N. Se puede considerar a Freifunk como la precursora de las redes comunitarias libres. (Freifunk, 2003)
- **Guifi.net** es una red de telecomunicaciones desarrollada en España, abierta, libre y neutral y mayoritariamente inalámbrica. Los usuarios de Guifi.net son particulares, empresas y administraciones que construyen una red de telecomunicaciones comunitaria. Se trata de la red libre comunitaria más grande del mundo, con más 19000 nodos operativos en la actualidad. Sin embargo, la implementación mallada solo está extendida en ciertas partes de la red. Dichas implementaciones trabajan con OLSR y B.A.T.M.A.N. (Guifi.net, 2004)

- **Bogota-mesh10** es la red comunitaria libre del proyecto Bogotá libre. Surgió con el objetivo de servir como plataforma para ayudar a disminuir la brecha digital en Bogotá y ayudar a la creación de una red independiente donde los proyectos personales, educativos, culturales y sociales tengan acogida y difusión. Es una red que en su mayor parte esta implementada con arquitectura ad-hoc. El número de nodos es de varias decenas y el protocolo usado es B.A.T.M.A.N. Bogotá-mesh está actualmente trabajando con un gran número de proyectos de redes comunitarias en Latinoamérica que comparten los mismos objetivos, como Valledupar-mesh, Medellín-libre, Largo-mesh, Red Fusa Libre, Caribe Mesh, Quintana Libre, etc. (NetworkBogota, 2010)
- **Villaje Telco11** es una iniciativa para construir una red comunitaria libre inalámbrica totalmente mallada de bajo coste que de servicio telefónico en áreas rurales o aisladas. Se basa en un conjunto de software y hardware específico con el que se puede configurar de forma fácil y rápida una red comunitaria mallada en cualquier parte del mundo con unos requisitos mínimos de energía. Los equipos usados son los Mesh Potato, basados en una antena con capacidad de encaminamiento y conexiones de VoIP, donde se instala una versión de OpenWrt que trabaja con el protocolo Batman-adv y algunos módulos de telefonía IP como Asterix. Villaje Telco surgió debido a la necesidad de millones de personas de tener un acceso a las comunicaciones, principalmente llamadas telefónicas locales y acceso a Internet de forma barata en lugares poco accesibles o aislados. (VillageTelco, 2011)

4.3.3 Equipos para redes MESH

En la actualidad cualquier dispositivo inalámbrico que permita la modificación de su firmware puede convertirse en un nodo mesh, a continuación se presenta algunos de los equipos utilizados en varios proyectos de redes mesh comunitarias:

- **4G AccessCube:** Este equipo fue utilizado de manera exitosa en el año 2005 en muchos experimentos MESH, aunque ya no se fabrica en la actualidad aún existen AccessCube funcionando en redes comunitarias; es compatible con la versión Kamikaze de OpenWRT. Entre sus característica encontramos: dimensiones de 7x5x7cm, 2 interfaces inalámbricas (802.11a/b/g), un puerto Ethernet de 100Mbps, conectores RP-SMA, procesador MIPS de 400 MHz, memoria Flash de 32MB,

memoria RAM de 64MB, el precio de este equipo está entre 200 y 500 dólares. (OpenWRT, 2004) (Büttrich, 2007)

- **MeshBox:** desarrollado por la Compañía Locustworld, entre las características que presenta están: procesador de 500 MHz, 128 MB de RAM, tarjetas de radio, 32 MP compact flash drive, y no tiene partes móviles; el precio se encuentra alrededor de 400 dólares MeshBox es un elemento de hardware que se utiliza para proporcionar redes inalámbricas de banda ancha de gran escala. Fabricados por LocustWorld, los dispositivos están diseñados para cooperar con otros MeshBox, el área de cobertura de una malla se mide en millas cuadradas o kilómetros cuadrados. (Büttrich, 2007)
- **Linksys WRT54G, GS, GL:** Aunque no fue diseñado para redes mesh ni para operar en ambientes outdoor es ampliamente utilizado por su bajo costo y fácil manejo. El precio de estos equipos varía entre 60-80 dólares y sus características varían dependiendo del modelo. (Asadoorian & Pesce, 2017)

4.4 PROTOCOLOS DE ENRUTAMIENTO

Un protocolo de enrutamiento es un algoritmo que tiene como objetivo calcular la mejor ruta entre un nodo origen y un nodo destino, los protocolos de enrutamiento para las redes mesh pueden clasificarse en: protocolos de enrutamiento basados en topología y basados en posición como se observa en la Figura 4.

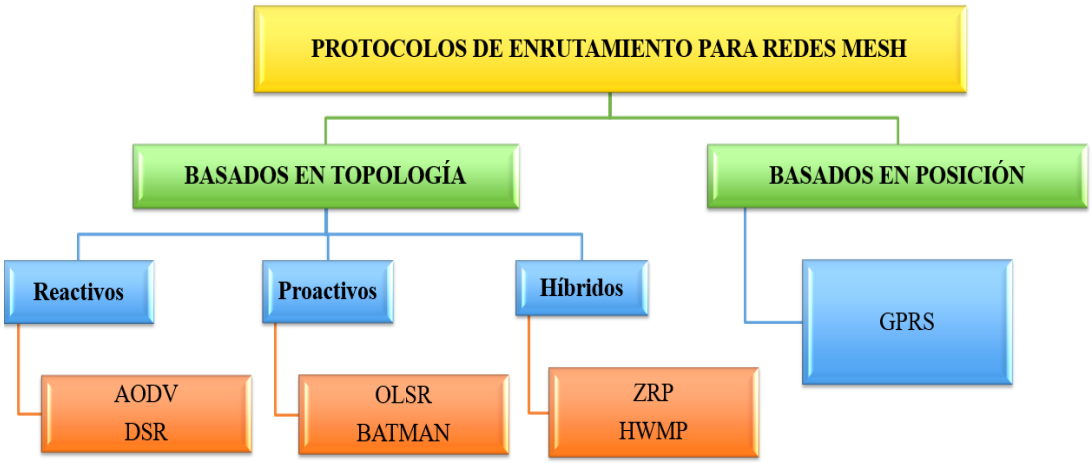

Figura 4. Clasificación de Protocolos.

Fuente: Recuperado de Wireless Mesh Networking por Yan Zhang, Jijun Luo, Honglin Hu

Los protocolos basados en posición calculan sus rutas mediante información geográfica de los nodos y los basados en topología eligen la mejor ruta gracias a la información topológica (número de saltos, calidad del enlace entre nodos, etc.).

4.4.1 Protocolos de enrutamiento basados en topología

Los protocolos de enrutamiento basados en topología se clasifican en protocolos reactivos, proactivos e híbridos, como se indica en la Figura 4

4.4.1.1 Reactivos

Calculan su ruta solo cuando existe un cambio topológico, por ejemplo cuando un nodo deja de funcionar, utilizan la técnica de flooding o inundación de paquetes de control para el cálculo de la ruta, una de sus ventajas es que reduce el tráfico de control ya que este se envía solo cuando existe un cambio y como desventaja es que la retardo aumenta hasta que la ruta sea calculada nuevamente y puede llegarse a saturar la red. También se lo conoce como protocolos “bajo demanda”. (Zhang, Luo, & Honglin, 2007)

AODV (del inglés, Ad-hoc On-demand Distance Vector) es un protocolo de enrutamiento utilizado en redes ad-hoc, desarrollado por Nokia Research Center, la Universidad de California, la Universidad de Cincinnati, Charles Perkins, Elizabeth Belding-Royer y Samir Das. Está descrito en la RFC 3561, en la cual se define tres tipos de mensajes de control: 1) RREQ que es un mensaje de solicitud de ruta, 2) RREP, un mensaje de respuesta de ruta que se devuelve a un RREQ y 3) RERR notifica sobre la pérdida de enlace. (Perkins, y otros, 2003)

DSR (del inglés, Dynamic Source Routing): El protocolo de Enrutamiento de Origen Dinámico está descrito en la RFC 4728, fue diseñado para redes inalámbricas multisalto donde los nodos son móviles. El protocolo se compone de dos mecanismos principales de "Descubrimiento de Rutas" y "Mantenimiento de rutas", que funcionan en conjunto para permitir que los nodos descubran y mantengan rutas. Es un protocolo que se basa en el enrutamiento de origen, en donde cada nodo tiene una memoria caché local para almacenar la información de enrutamiento (Johnson, Maltz, & Hu, 2007)

4.4.1.2 Proactivos

Utilizan el envío periódico de paquetes de control para descubrir nuevos rutas en la red, de esta manera la información de enrutamiento estará actualizada y se alojará en una tabla. Su ventaja en comparación con los reactivos es que cada nodo conoce las rutas a los otros nodos todo el tiempo reduciendo el retardo pero aumentando la carga de control (Zhang, Luo, & Honglin, 2007)

OLSR (del inglés, Optimized Link State Routing) Es un protocolo de enrutamiento que hereda la estabilidad del algoritmo de estado de enlace. El protocolo reduce el tráfico de los paquetes de control, mediante el uso de los MPRs (del inglés, Multipoint Relays), que se selecciona de entre los vecinos de un salto cuando se establece un enlace bidireccional, de manera que se establezca rutas a todos los nodos que se encuentren a una distancia de dos saltos. Solo los MPRs son los encargados de retransmitir los mensajes de control reduciendo significativamente el número de retransmisiones en un proceso de flooding o broadcast. (Clausen & Jacquet, 2003)

BATMAN (del inglés, Better Approach to Mobile Ad hoc Networking) BATMAN es un protocolo de enrutamiento que fue creado para suplir algunas de las deficiencias del protocolo OLSR. BATMAN solo conoce el siguiente salto a diferencia de OLSR que mantiene las rutas para todos los vecinos de la red. Los mensajes de enrutamiento que se envían en el protocolo BATMAN se denominan Originator Messages (OGMs, del inglés Originator Message), y son paquetes UDP (del inglés, User Datagram Protocol) de 52 portes de longitud. El campo más importante de estos mensajes es el campo Originator Address, que contiene la dirección de host del nodo emisor. Cuando otro nodo recibe este mensaje, comprueba si la dirección del originador y la dirección fuente de la cabecera IP coinciden, y en caso afirmativo los dos nodos son vecinos directos. Todos los OGMs dentro de la red BATMAN son difundidos y re-difundidos por los nodos de la red hasta que el valor del campo TTL (del inglés, Time To Live) sea cero, hasta que se hayan perdido los paquetes, o hasta que el nodo recibe un OGM que él mismo ha enviado previamente. Todos los nodos conocerán de la existencia de los demás y qué nodos constituyen el primer salto entre ellos y el resto de los nodos de la red, es decir, el primer salto de la ruta. Los nodos y los primeros saltos en sus trayectos se almacenan en una lista llamada Originator List. Existen dos versiones completamente

diferentes del protocolo de enrutamiento BATMAN, y el descrito anteriormente es el BATMAN Daemon o batmand, que es un protocolo de capa de red. Sin embargo, existe una versión más reciente del protocolo denominada BATMAN Advanced o batman-adv. (Neumann, Aichele, Lindner, & Wunderlich, 2008) (open-mesh, 2006)

4.4.1.3 Híbridos

Combinan las ventajas de los protocolos reactivos para los nodos o rutas utilizadas con frecuencia y proactivos para los nodos o rutas menos utilizadas. (Zhang, Luo, & Honglin, 2007)

Existen muchas más clasificaciones de protocolos de enrutamiento como vector distancia versus estado enlace, según la capa del modelo OSI, etc. Aunque todos los protocolos pueden implementarse en redes mesh no todos funcionaran bien, la elección del protocolo depende del escenario de aplicación, de la movilidad de los nodos, requerimientos de hardware, numero de nodos, cambios topológicos, entre otros. (Held, 2005) (Zhang, Luo, & Honglin, 2007). Existen algunos trabajos en donde realizan estudios comparativos de algunos protocolos, por ejemplo Modassir Ishfaq (Ishfaq, 2014) compara los protocolos AODV, DSR (protocolos reactivos) y OLSR (protocolo proactivo) dando como resultado que el protocolo OLSR tiene mayor rendimiento, menor retardo y sobrecarga en la red que los reactivos, hay que tomar en cuenta que en este estudio los nodos son estáticos y no móviles, característica que comparte con nuestra red, además a diferencia de los demás protocolos, este ha sido implementado en varios proyectos de redes inalámbricas libres de gran tamaño y ha sido precursor de otros protocolos, es por esta razones que se eligió OLSR como protocolo de enrutamiento en nuestra red.

4.5 OLSR (Optimized Link State Routing)

El protocolo OLSR (del inglés, Optimized Link State Routing) fue estandarizado en el año 2003 por el IETF (del inglés, Internet Engineering Task Force), en la norma RFC 3626 (Clausen & Jacquet, 2003). Fue diseñado por investigadores del Instituto Nacional de Investigación en Informática y Automatización, INRIA (del francés, Institut National

de Recherche en Informatique et en Automatique). OLSR funciona de manera distribuida y no depende de ninguna entidad central. El protocolo no requiere de transmisión confiable de mensajes de control ya que se envían periódicamente, y tampoco requiere la entrega secuenciada de mensajes debido a que cada mensaje de control contiene un número de secuencia que se incrementa para cada mensaje permitiendo de esta manera que el nodo destino identifique información reciente. OLSR no requiere ningún cambio en el formato de los paquetes IP, solo interactúa con la gestión de la tabla de enrutamiento. OLSR funciona en tres pasos

- **Detección de vecinos:** La detección de vecinos se realiza mediante el intercambio de mensajes HELLO, cada nodo emite periódicamente y por broadcast estos mensajes entre los vecinos a un salto. A través de estos mensajes los nodos pueden seleccionar los MPR y también cada nodo establece una base de datos de estado de enlace y de vecinos.
- **Difusión de mensajes TC.** Los mensajes TC sirven para mantener las bases de datos de los nodos actualizadas, cada nodo a través de sus MPR, anuncia periódicamente su información de enlace a todos los otros nodos dentro de la red.
- **Cálculo de la tabla de enrutamiento:** En función de los mensajes TC recibidos de otros nodos, un nodo puede calcular la ruta más corta a todos los nodos accesibles en la red, utilizando un algoritmo similar al algoritmo de Dijkstra. De acuerdo con RFC 3626 (Clausen & Jacquet, 2003), el camino más corto en términos de la cantidad de saltos se usa para calcular la ruta en OLSR. (Misra, Misra, & Woungang, 2003)

4.5.1 MPR (MultiPoint Relays)

OLSR es un protocolo proactivo de estado de enlace, en donde los nodos intercambian periódicamente mensajes de control permitiéndoles conocer la topología de la red. No es aconsejable en redes inalámbricas inundar mensajes en la red debido a los limitados recursos de las mismas, por eso OLSR utiliza los Relés Multipunto, MPR (del inglés Multipoint Relays) que permiten disminuir la sobrecarga de tráfico de control en la red, ya que solo estos nodos retransmiten los mensajes de control; los demás nodos reciben y procesan pero no los retransmiten. Para la selección de los MPRs cada nodo selecciona de entre sus vecinos a un salto, un conjunto de nodos que puede retransmitir los

mensajes de control a los vecinos de dos saltos; este conjunto de nodos vecinos seleccionados son los MPR. Cada nodo mantiene información sobre el conjunto de vecinos que lo han seleccionado como MPR, este conjunto se denomina Conjunto Selector de Relés Multipunto (del inglés Multipoint Relay Selector Set). (Clausen & Jacquet, 2003), esta información se obtiene a través de los mensajes HELLO En la Figura 5 se puede observar cómo se realizaría la elección de los MPRs en una red.

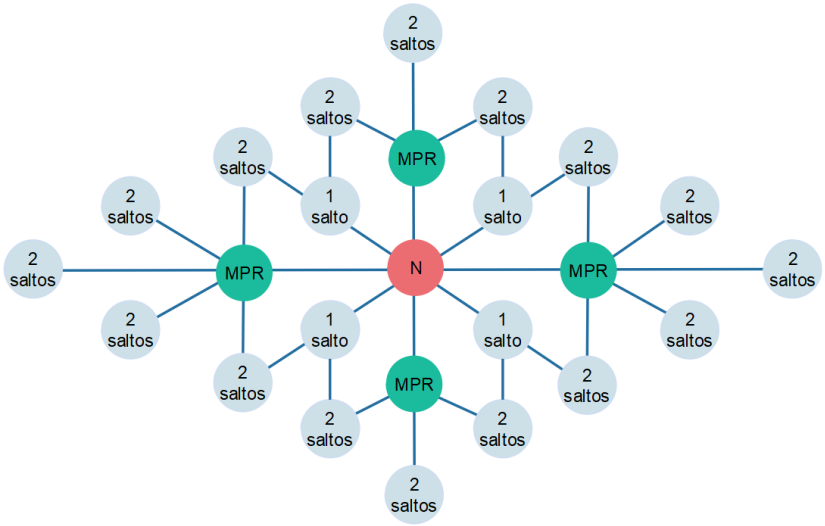

Figura 5. Selección de los MPRs en OLSR

Fuente: Recuperado de Wireless Mesh Networking por Yan Zhang, Jijun Luo, Honglin Hu

4.5.2 Mensajes y Formato de Paquete

OLSR utiliza el puerto UDP 698 el mismo que ha sido asignado por la IANA (Autoridad de Números Asignados en Internet, del inglés Internet Assigned Numbers Authority) para su uso exclusivo, los mensajes que utiliza OLSR están presentados de forma sucinta en la Al recibir un paquete un nodo examina el encabezado del mensaje y determinar de esta manera el destino del mensaje, además puede recibir un mensaje varias veces por lo que cada nodo mantiene un "Conjunto Duplicado" que evitan el re-procesamiento de información, este conjunto duplicado contiene información de la dirección principal y del número de secuencia del mensaje.

Tabla 4 con sus respectivos intervalos y el "tipo de mensaje" que es un campo del formato del paquete OLSR que consiste un número utilizado por OLSR para conocer de qué mensaje se trata, puesto que OLSR utiliza un formato de paquete unificado para todo el tráfico de control, facilitando de esta manera que cada paquete encapsule uno o

más mensajes. Los paquetes están integrados en datagramas UDP para su transmisión. El formato de paquete que utiliza OLSR se muestra en la Figura 6 obviando la cabecera UDP. (Clausen & Jacquet, 2003)

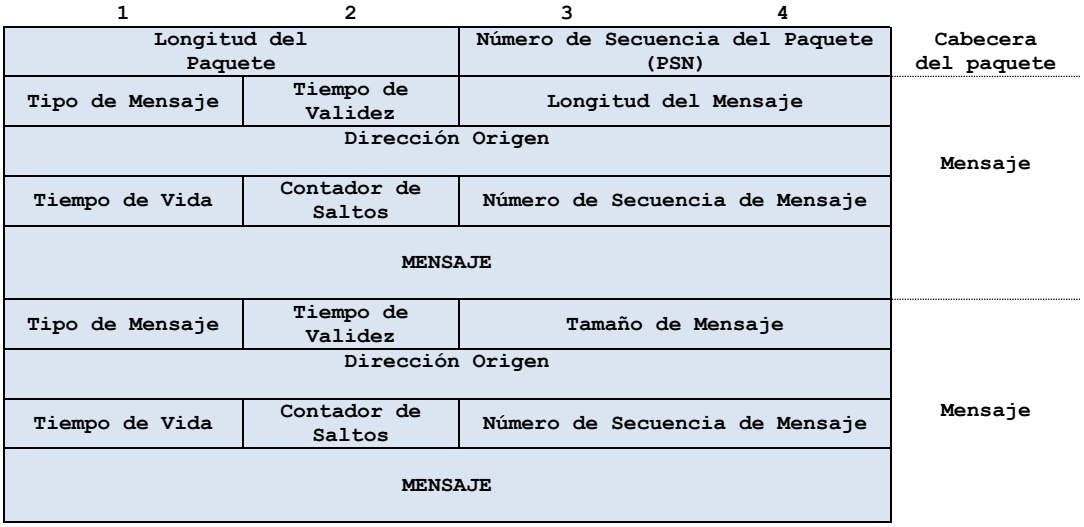

Figura 6. Formato de paquete en OLSR

Fuente: Recuperado de Optimized Link State Routing Protocol (OLSR) por Clausen, T; Jacquet, P

Los campos del paquete son:

- **Longitud del Paquete**: indica en bytes la longitud del paquete, cuando esta es igual a 0, el paquete se descarta.
- **Número de Secuencia del Paquete (PSN, del inglés Packet Sequence Number),** se incrementa cada vez que el nodo transmita un paquete OLSR, el PSN se maneja por separado para cada interfaz
- **Tipo de Mensaje**, indica de qué mensaje se trata.
- **Tiempo de Validez (Vtime)**, Indica por cuanto tiempo un nodo debe considera la información emitida por otro valida.
- **Longitud del mensaje**, indica el tamaño del mensaje desde el campo Tipo de mensaje.
- **Dirección Origen**, contiene la dirección principal del nodo que género el mensaje.
- **Tiempo de Vida (TTL, del inglés Time To Live),** este campo contiene la cantidad de saltos que deberá ser transmitido el mensaje, un TTL= 0 o 1 indica que l mensaje no debe ser transmitido. El valor máximo es 255
- **Contador de saltos** o número de saltos, este campo contiene la cantidad de saltos que ha alcanzado el mensaje.

- **Número de secuencia del mensaje,** es un número de identificación única dado a cada mensaje por el nodo originador.

Al recibir un paquete un nodo examina el encabezado del mensaje y determinar de esta manera el destino del mensaje, además puede recibir un mensaje varias veces por lo que cada nodo mantiene un "Conjunto Duplicado" que evitan el re-procesamiento de información, este conjunto duplicado contiene información de la dirección principal y del número de secuencia del mensaje.

Tabla 4. Mensajes OLSR con su intervalo

MENSAJE	FUNCIÓN	INTERVALOS (segundos)	TIPO DE MENSAJE
HELLO	Realizar la detección de enlaces, descubrimiento de vecinos y la elección de MPRs	2	1
TC (Toplogy Control)	Realizan la declaración de topología (anuncio de estados de enlace).	5	2
MID (Multiple Interface Declaration)	Declaran la presencia de múltiples interfaces en un nodo OLSR.	5	3
HNA (Host and Network Association)	Permitir la comunicación de redes o hosts NO-OLSR en la red	5	4

Fuente: Elaboración propia

1. *Mensaje HELLO*

Este tipo de mensaje realiza las siguientes funciones:

- *Detección de enlace:* cada nodo realiza la detección del enlace mediante la emisión periódica de mensajes HELLO, con la finalidad de detectar enlaces entre su interfaz y las del nodo vecino.
- *Detección de vecinos:* cada nodo anuncia a todos sus vecinos a un salto en cada una de sus interfaces con la finalidad de realizar esta tarea, el nodo deduce el conjunto de nodo vecinos gracias a la información intercambiada de la detección de enlace.
- *Elección de MPRs:* este tipo de mensajes sirve para la elección de los MPRs, cada nodo elige un subconjunto de sus vecinos para que puedan recibir mensajes los nodos a dos saltos, como se explicó en la sección anterior. (Clausen & Jacquet, 2003)

En la Figura 7 se puede observar cómo se realiza la detección de vecinos

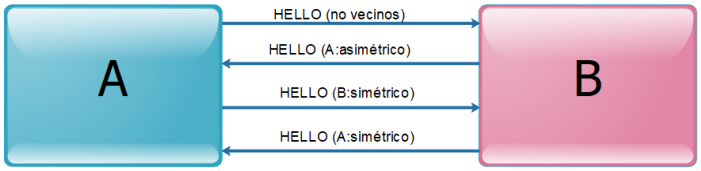

Figura 7. Detección de vecinos

Fuente: Recuperado de Mobile Ad-Hoc Network por http://www.olsr.org/docs/wos3-olsr.pdf

El intercambio de este mensaje sirve para un propósito común, el "descubrimiento de topología local", por esta razón el mensaje HELLO se genera en base a la lista de enlaces y de vecinos de una interfaz. Su formato se muestra en la Figura 8

1	2	3	4
Reservado		Intervalo de Tiempo	Disposición
Código de enlace	Reservado	Tamaño del mensaje de enlace	
Dirección de la interfaz de vecino			
Dirección de la interfaz de vecino			
...			
Código de enlace	Reservado	Tamaño del mensaje de enlace	
Dirección de la interfaz de vecino			
Dirección de la interfaz de vecino			
...			

Figura 8. Formato de mensaje HELLO

Fuente: Recuperado de Optimized Link State Routing Protocol (OLSR) por Clausen, T; Jacquet, P

En este mensaje el campo TTL está establecido en 1.

- **Reservado,** este campo está compuesto por una serie de ceros con fines de compatibilidad con el borrador de la rfc.
- **Intervalo de Tiempo (Htime),** este campo indica el intervalo de emisión del mensaje HELLO
- **Disposición:** Indica la disposición de un nodo para enviar y re-enviar tráfico de otros nodos
- **Código de enlace,** el código del enlace especifica información sobre el enlace y sobre el estado del vecino, entre la información que presenta tenemos:

  Tipos de enlace

  UNSPEC_LINK: Indica que no existe información sobre el enlace

  ASYM_LINK: Indica que el enlace es asimétrico, es decir solo existe comunicación en un solo sentido

  SYM_LINK: Indica que el mensaje es simétrico, es decir que los dos nodos se oyen.

LOST_LINK: Indica que el enlace se ha perdido.

Tipos de vecinos

SYM-NEIGH: Indica la existencia de un enlace simétrico entre el nodo origen y destino.

MPR_NEIGH: Indica la existencia de un enlace simétrico entre el nodo origen y destino y que ha sido seleccionado como MPR.

NOT_NEIGH: Indica que todavía no llegan a ser vecinos simétricos.

- **Tamaño del mensaje de enlace:** Tamaño del mensaje, se empieza a calcular desde el campo "Link Code"
- **Direciion de la interfaz del vecino,** indica la dirección IP principal de los nodos vecinos. (Clausen & Jacquet, 2003)

2. *Mensaje TC (Toplogy Control)*

Estos mensajes son enviados para declarar un conjunto de enlaces, denominado "advertised link set" (conjunto de enlaces anunciados) y que incluyen los enlaces de todos los nodos del Multipoint Relay Selector Set, es decir, los vecinos que han seleccionado al nodo emisor como una MPR. Su objetivo es proporcionar a cada nodo la información de estado enlace para permitir el cálculo de la ruta. Su formato se muestra en la Figura 9

1	2	3	4
Número de secuencia del vecino anunciado		Reservado	
Dirección principal del vecino anunciado			
Dirección principal del vecino anunciado			
...			

Figura 9. Formato de mensaje TC

Fuente: Recuperado de Optimized Link State Routing Protocol (OLSR) por Clausen, T; Jacquet, P

- **Número de secuencia de vecino anunciado (ANSN, del inglés Advertised Neighbor Sequence Number),** el número de secuencia del vecino anunciado es un número que incrementa un cambio en la información de los vecinos anunciados.
- **Dirección principal del vecino anunciado,** la dirección principal del vecino anunciado contiene la direcciones principales de todos los nodos vecinos (Clausen & Jacquet, 2003)

3. *Mensaje MID (Multiple Interface Declaration)*

Un nodo con múltiples interfaces que participen en la red OLSR, debe anunciar a través de menajes MID la configuración de estas. Este mensaje permite asociar varias interfaces a un nodo. Su formato se muestra en la Figura 10. El campo "OLSR Interface Address", contiene la dirección de una interfaz OLSR del nodo, sin tomar en cuenta la dirección principal que es la dirección originen. (Clausen & Jacquet, 2003)

1	2	3	4
Dirección de la interfaz OLSR			
Dirección de la interfaz OLSR			
…			

Figura 10. Formato del mensaje MID

Fuente: Recuperado de Optimized Link State Routing Protocol (OLSR) por Clausen, T; Jacquet, P

4. *Mensaje HNA (Host and Network Association)*

Un nodo puede poseer varias interfaces, algunas que no participen en el enrutamiento OLSR y otro si, los mensajes HNA proporcionan conectividad entre la interfaz OLSR y la interfaz NO-OLSR. (Clausen & Jacquet, 2003) Su formato se muestra en la Figura 11. En el campo "Netwok Address" indica la dirección de la red anunciada, y el campo "Netmask" su respectiva máscara de red. (Clausen & Jacquet, 2003)

1	2	3	4
Dirección de red			
Máscara			
Dirección de red			
Máscara			
…			

Figura 11. Formato de paquetes HNA

Fuente: Recuperado de Optimized Link State Routing Protocol (OLSR) por Clausen, T; Jacquet, P

4.5.3 Repositorios de información y cálculo de la tabla de enrutamiento

A través del intercambio de información realizada a través de los mensajes de control, cada nodo acumula información acerca de la red:

- ***Base de información de asociación de interfaces múltiples***: almacena información sobre las múltiples interfaces que puede tener un nodo y en las que corre OLSR, a través de la interfaz principal del nodo y de la interfaz del nodo, esta información se denomina "Conjunto de asociación de interfaz"

- ***Detección de enlaces- Base de información del enlace local***: almacena la interfaz local del nodo y la interfaz del nodo vecino así también información sobre los enlaces, este conjunto de información se denomina "Conjunto de Enlace"

- ***Detección de vecinos:*** Base de información del vecindario: Se almacena información sobre:

- Vecinos: en la cual se especifica si un nodo es simétrico o asimétrico y se denomina "Conjunto de vecinos"
- Vecinos de 2 saltos, se almacena la dirección principal del vecino y la del vecino a dos saltos a esta información se la denomina "Conjunto de vecinos a dos saltos"
- MPR: se almacena información sobre los nodos que fueron seleccionados como MPR esta información almacenada se la denomina como "Conjunto de MPR":
- Selectores de MPR: se almacena información sobre los vecinos que han seleccionado a un nodo como MPR, esta información se almacena bajo el nombre de "Conjunto Selector de MPRs":

- ***Base de información topológica:*** Esta información se obtiene de los mensajes TC y se utiliza para el cálculo de la tabla de enrutamiento, y se denomina "Conjunto de topología"

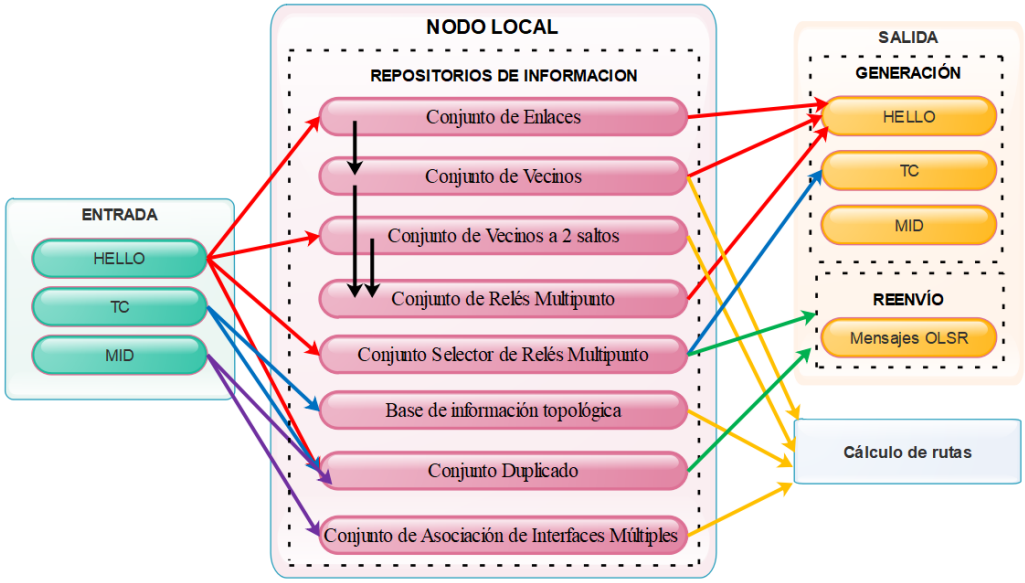

Figura 12. Flujo de Datos en OLSR

Fuente: Recuperado de Data Flow Diagrams por Commotion- https://wiki.commotionwireless.net/

Cada nodo calcula y mantiene una tabla de enrutamiento, la misma que se basa en la información contenida en el conjunto de enlaces, conjunto de vecinos, conjunto de vecinos a dos saltos, conjunto de topología y la base de información de asociación de interfaz múltiple. Por lo tanto, si se cambia cualquiera de estos conjuntos, la tabla de enrutamiento se vuelve a calcular para actualizar la información de ruta sobre cada destino. (Clausen & Jacquet, 2003)

En la Figura 12 se presenta un diagrama de flujo de la información contenida en los mensajes de control de OLSR y los repositorios de información para luego realizar el cálculo de la tabla de enrutamiento.

4.5.4 Métrica

La RFC 3626 indica que su métrica es el número de saltos pero deja abierta la posibilidad de que se implemente otras métricas, por lo que el protocolo fue evolucionando a través del tiempo para hacerse más eficiente y aumentar su rendimiento (GitHub, 2012). Se han presentado varias alternativas, como:

- **OLSR-ML** (del inglés, Minimum Loss), eligen los MPR dependiendo de perdida mínima de paquetes, la mejor ruta es la que tenga mayor probabilidad de transmisiones exitosas entre el nodo origen y nodo destino.
- **OLSR-MD** (del inglés, Minimum Delay) realiza la selección de la ruta entre un nodo y los otros nodos que tienen la suma de retardos más bajos
- **OLSR-ETX** (del inglés, Expected Transmission Count) realiza la elección de la ruta mediante la cantidad de retransmisiones necesarias para entregar el paquete al destino, siendo esta última la que presenta mejores prestaciones que los anteriores (Pandey & Manoj, 2004). Cada nodo proporciona un valor LQ (del inglés, Link Quality) que describe la calidad del enlace hacia el enrutador y el valor NLQ (del inglés, Neighbor Link Quality) que describe la calidad del enlace hacia el vecino, con estos parámetros se calcula el valor ETX como se indica en la Ecuación 1. El valor ETX de un enlace puede indicar la cantidad de retransmisiones necesarias para entregar el paquete al nodo destino o en otras palabra es una medida que indica la calidad del enlace, un valor de ETX=1 significa que es un enlace perfecto sin pérdida de paquetes. Desde la versión 0.6.0 del protocolo se usa ETX.

$$\mathbf{ETX} = \frac{\mathbf{1}}{\mathbf{NLQ} \times \mathbf{LQ}} \qquad \textbf{Ecuación 1}$$

Existen varios algoritmos para calcular la calidad del enlace:

- **etx_ff** (ETX Funkfeuer / Freifunk) es el algoritmo predeterminado para OLSRd. Utiliza el número de secuencia de los paquetes OLSR para determinar la tasa de perdida de paquetes.
- **etx_float** calcula el valor ETX mediante el uso de pérdida en los mensajes Hello entrantes, pero no presenta buenos resultados como el algoritmo etx_ff ya que no usan los mensajes TC para el cálculo de la calidad del enlace.
- **etx_fmp**, es similar a etx-float pero fue diseñado para ser utilizado en dispositivos de memoria reducida. (GitHub, 2012)

4.6 LINKSYS WRT54G/GS/GL

Linksys, es una división de Cisco Systems, Inc., este dispositivo es destinado para uso doméstico y de pequeñas empresas, admite la instalación de firmwares de terceros en casi todas sus versiones. A finales del año 2002 aparece en el mercado el WRT54G V1.0 de Linksys, teniendo buena aceptación por lo que desde el año de su lanzamiento se han realizado varias mejoras en la placa del dispositivo creando así varios modelos; esta versión incluía un puerto WAN, un conmutador Etherneth 10/100 de cuatro puertos, soporte para 802.11b, interfaz web para configuración. En el año 2003 Linksys lanzo su código bajo licencia GPL[3](del inglés, GNU General Public License), desde entonces varias personas han creado su propio firmware en base a este. (Asadoorian & Pesce, 2017)

4.6.1 Características comunes de la serie WRT54G

Aunque existen varios modelos en la serie WRT54G todos presentan algunas características comunes, entre las cuales podemos encontrar:

[3] Licencia General de GNU, permite que los usuarios finales tengan la libertad de usar, estudiar, compartir y modificar software

- **Alimentación:** Todas las versiones utilizan 12VDC y 1 A
- **Botón RESET:** Este botón sirve para reestablecer al dispositivo a sus valores de fábrica, pero también tienen otras funcionalidades dependiendo del tiempo que se presione y el firmware que se ejecute.
- **Indicadores LED:** Los indicadores LED muestran diferentes condiciones y difieren dependiendo del modelo. La luz POWER es de color verde, si es fija indica que el dispositivo está encendido, pero si es intermitente indica que está iniciando o se está aplicando otro firmware. La luz DMZ depende del firmware, en OpenWRT indica el proceso de arranque. Los indicadores WAN y Ethernet enumerados del 1 al 4 son de color verde, este color es fijo solo cuando existe conexión, y parpadea cuando hay actividad. Si la luz POWER y DMZ están encendidas y se encuentran parpadeando indica que la imagen del firmware instalado está dañada.
- **Arquitectura del procesador:** Todos los modelos utilizan un procesado Broadcom MIPS (del inglés, Microprocessor without Interlocked Pipeline Stages) que se basa en un conjunto reducido de instrucciones RISC (del inglés, Reduced Instruction Set Computer), este procesador varía dependiendo del modelo. Se utilizan dos familias de procesadores Bradcom en todos los modelos: BM47XX y BCM5352

  - BM47XX: existen dos modelos, la serie BCM4704 y la BCM4712. La primera proporciona solamente las funciones de CPU y se utilizaban chipset separados para control de acceso Etherneth y Wireless. La serie BCM4712 utiliza la tecnología SoC (del inglés, System on a Chip) para integrar la funcionalidad del CPU y Wireless MAC (mediante la integración del radio inalámbrico BCM2050) y utiliza un chipset separado para Ethernet MAC mediante los procesadores ADM6996 y BCM5325

  -BCM5352: Utiliza la tecnología SoC para integrar la función del CPU, Ethernet y Wireless MAC en un solo chip.
- **Almacenamiento:** El almacenamiento proviene de la memoria flash y varía dependiendo del modelo, el WRT54GS versión de 1.0 a 3.0 y la serie WRTSL54GS contienen mayor capacidad (8MB).
- **Memoria RAM:** Se utiliza la Memoria de Acceso Aleatorio Dinámica Síncrona SDRAM (del inglés, Synchronous Dynamic Random Access Memory) y está soldada directamente en la placa.

- **Arquitectura de Red:** Incluye un conmutador Ethernet de 5 puertos dividido en dos VLANs y una interfaz inalámbrica.
- **Conectores:** Todos los modelos (excepto el WRT54SLGS) cuentan con antenas extraíbles y con conectores de antena externos RP-TNC (del inglés Reverse Polarity–Threaded Neill-Concelman). (Asadoorian & Pesce, 2017)

4.6.2 WRT54GS v 3.0

La serie de enrutadores WRT54GS tienen muchas características comunes a la serie WRT54G, esta versión suele ser mucho más utilizada para la instalación de firmwares ya que contiene el doble de RAM y flash. La versión utilizada en el proyecto fue WRT54GS v 3.0; a continuación se detallan algunas de sus características:

- Alimentación de 12 Vcc, 1A
- Esta versión introduce un cambio significativo en el diseño de routers WRT54G, se remplaza el procesador Broadcom BCM4712 por el Procesador Broadcom BCM5352, este procesador incorpora la función del CPU, MAC Ethernet y Wireless MAC en un solo chip gracias a la arquitectura SoC.
- Velocidad de procesamiento de 200 MHz
- Capacidad de almacenamiento de 8MB y 32 MB de memoria RAM.
- Una interfaz WLAN 802.11g y una interfaz Ethernet de 802.3 que junto a una VLAN dan lugar a un switch de 5 puertos (4 puertos LAN y un WAN)
- Valor por defecto de potencia es de 18dBm.
- Prefijo del número de serie es CGN5
- Contiene dos conectores RP-TNC, conectados a dos antenas Rubber Ducky, estas antenas son omnidireccionales, tienen un patrón de radiación de 360 grados en el plano horizontal y de 75 grados en el plano vertical y una ganancia de 2,14 dBi cada una. (Linksys Group, 2002)

En la Figura 13 se puede observar la vista frontal del router Linksys WRT54gs, mientras que en la Figura 14 se indica la arquitectura interna del mismo y en la Tabla 5 se presenta la configuración por defecto de las interfaces de red del enrutador

Figura 13. Router Linksys WRT54gs V3.0
Fuente: Recuperado de Wireless-G Broadband Router: User guide

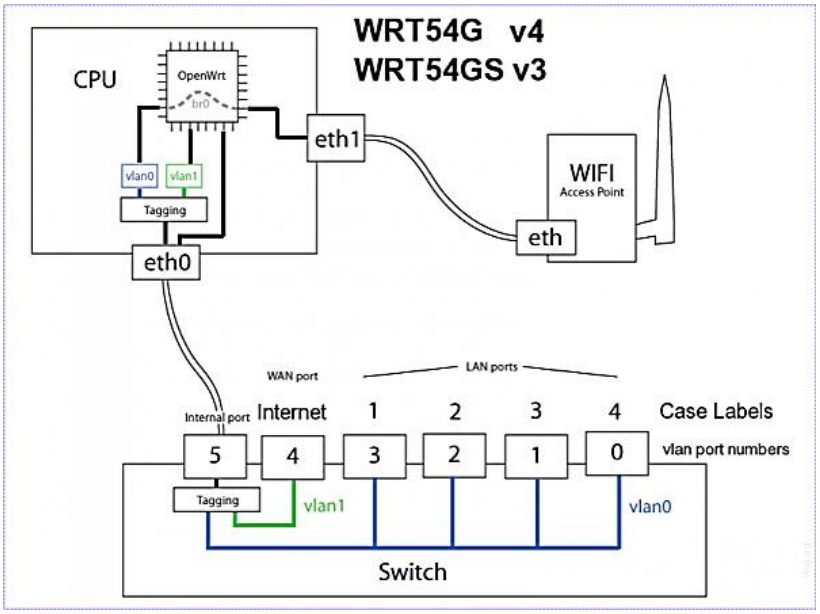

Figura 14. Arquitectura interna del router Linksys WRT54GS v3.0
Fuente: Recuperado de OpenWRT por openwrt.org

Tabla 5. Configuración por defecto de las interfaces del Linksys WRT54gs v3.0

Interfaz		Configuración por defecto
br-lan	LAN & WiFi	192.168.1.1/24
vlan0 (eth0.0)	Puertos LAN (1 to 4)	Ninguna
vlan1 (eth0.1)	Puerto WAN	DHCP
wl0	WiFi	Deshabilitado

Fuente: Recuperado de OpenWRT por openwrt.org

4.7 FIRMWARES OPENSOURCE

A continuación se detalla algunos de los firmwares disponibles para los routers Linksys, ya que este dispositivo será utilizado para el despliegue de la red, además que permitan la instalación del protocolo OLSR.

4.7.1 Firmware OEM

El firmware OEM (del inglés, Original Equipment Manufacturer), o firmware original del enrutador Linksys wrt54gs v 3.0 se basa en Linux que se distribuye bajo licencia GPL que permite que el código este siempre disponible y pueda ser modificado, lo que ha conllevado a que se desarrollen varios firmwares en base a este. Este firmware permite la creación de puntos de acceso de manera fácil. (Linksys Group, 2002) (Asadoorian & Pesce, 2017)

4.7.2 DD-WRT

Firmware creado por BrainSlayer, está basado en Linux y en la versión original del firmware de los enrutadores con chip Broadcom y Atheros, es apropiado para una gran variedad de routers inalámbricos IEEE 802.11 a / b / g / h / n. Permite la configuración del dispositivo a través de una interfaz gráfica web amigable e intuitiva con el usuario. Está soportado por una gran cantidad de desarrolladores permitiendo corregir problemas rápidamente, además ofrece información a través de los foros de usuarios, wiki y guías prácticas. Par fines privados DD-WRT es gratuito pero para fines comerciales requiere pagar una licencia, esta versión a diferencia de la gratuita permite una amplia configuración de los parámetros inalámbricos del dispositivo. Es compatible con más de 200 dispositivos diferentes, permite acceder al router por línea de comandos pero presenta ciertas limitaciones ya que posee características de seguridad en algunos archivos, permite el monitoreo remoto, aunque presenta un gran abanico de funcionalidades DD-WRT es el firmware que más recursos consume. (BrainSlayer, 2005) (Asadoorian & Pesce, 2017)

4.7.3 Freifunk

Freifunk nace como una iniciativa no comercial para apoyar redes inalámbricas en la región Alemana, en la actualidad cuenta con 400 comunidades y con más de 41 000 puntos de acceso. Este firmware está basado en OpenWRT y trae ya instalado el protocolo OLSR, aunque permite mediante la instalación de paquetes el uso del protocolo BATMAN. Freifunk tiene como objetivo implantar redes gratuitas, independientes y no comerciales mediante la configuración fácil y rápida de los

enrutadores que conforman la red. El movimiento Freifunk tiene sus inicios en el año 2000, aunque el sitio web se creó en el año 2003 en Berlín. (Freifunk, 2003)

4.7.4 OpenWRT

OpenWRT es una distribución de Linux, este firmware permite la instalación de diversos paquetes y varios protocolos de enrutamiento. Aunque en sus inicios solo estaba disponible para routers Linksys WRT54g en la actualidad es compatible con una gran cantidad de dispositivos. Existen varios lanzamientos o releases en torno a este firmware, en cada lanzamiento se realizan mejoras o corrigen errores que presentaba su antecesor. (OpenWRT, 2004) (Asadoorian & Pesce, 2017)

Tabla 6. Comparación de firmwares OpenSource para los enrutadores Linsksys

Características	Firmware Original	DD-WRT	FREIFUNK	OPENWRT
Interfaz web	Si	Si	Si	Si
SSH	No	Limitada	Si	Si
Consumo de recursos	Normal	Alta	Normal	Normal
Documentación disponible	Se limita al manual de usuario	Buena, incluye wiki, foros de usuarios, guías prácticas	Buena, incluye wiki, foros de usuarios, guías prácticas	Buena, incluye wiki, foros de usuarios, guías prácticas
Repositorios	No	Buenos	Buenos	Buenos
OLSR	No	Si	Si	Si
BATMAN	No	No	Corre sobre el protocolo OLSR	Si
BABEL	No	No	No	Si
802.11s	No	No	No	Si

Fuente: Elaboración propia

En la Tabla 6 se presenta una comparativa de los firmwares disponibles para los routers Linksys y que permitan la implementación del protocolo OLSR. Como se puede observar OpenWRT en comparación del firmware original, DD-WRT y Freifunk, permite la instalación de varios protocolos usados en redes mesh, gracias a la amplia cantidad de paquetes disponibles en los repositorios para cada versión del firmware. Otra de las características que hace a OpenWRT interesante es la documentación que presenta, en su página oficial podemos encontrar desde características técnicas de los dispositivos compatibles, complementos que deben ser instalados para el funcionamiento de los protocolos, ejemplos de configuración y archivos donde se alojan, comandos, entre otros. También es la base de varios firmwares y proyectos de

comunidades libres como Freifunk, Guifi-net y LED Project, por estas razones será utilizado en el proyecto. En la siguiente sección se explicará a detalle este firmware.

Para la instalación de estos firmwares de terceros existen varios métodos dependiendo del equipo, a continuación en la Tabla 7 se indican los métodos que se pueden utilizar para el Linksys WRT54gs v 3.0

Tabla 7. Métodos para la instalación de firmwares

MÉTODO	DESCRIPCIÓN	HERRAMIENTAS REQUERIDAS
Interfaz Web	Es el método más sencillo, se utiliza cuando se quiere actualizar el firmware de fábrica o de un tercero o cuando se instala un firmware por primera vez	PC, explorador web, cable de red
TFTP (Protocolo de transferencia de archivos trivial, del inglés Trivial File Transfer Protocol)	Este método se suele utilizar de mejor manera cuando ya se haya instalado firmwares de terceros	PC, cliente TFTP, cable de red
JTAG (Grupo de Acción Conjunta de Prueba, del inglés Joint Test Action Group)	Este método solo se utiliza para fines de recuperación y solo se debe usar si se tiene "ladrillo", es decir que no hay otra manera de recuperar el firmware. Este método puede tardar varias horas.	PC con puerto paralelo, cable JTAG, software especifico

Fuente: Recuperado de Linksys WRT54G. Ultimate Hacking por Asadoorian, P; Pesce, L.

4.8 OpenWRT

OpenWRT es una distribución de Linux para dispositivos integrados, especialmente para routers inalámbricos. OpenWRT nace a inicios del año 2004 como un proyecto de desarrollo cerrado el cual se basaba en el código fuente de los routers Linksys WRT54G, a mediados del siguiente año y con la colaboración de nuevos desarrolladores se decide publicar las primeras versiones del firmware, las cuales utilizan fuentes oficiales del kernel GNU/Linux junto a parches para el sistema en el chip y de los controladores de las interfaces de red de los enrutadores inalámbricos, desde entonces OpenWRT es un proyecto libre y de código abierto, de acceso fácil y gratuito e impulsado por la comunidad. OpenWRT permite personalizar el sistema operativo del enrutador agregándole características mediante la instalación de paquetes de software. (Asadoorian & Pesce, 2017) (OpenWRT, 2004)

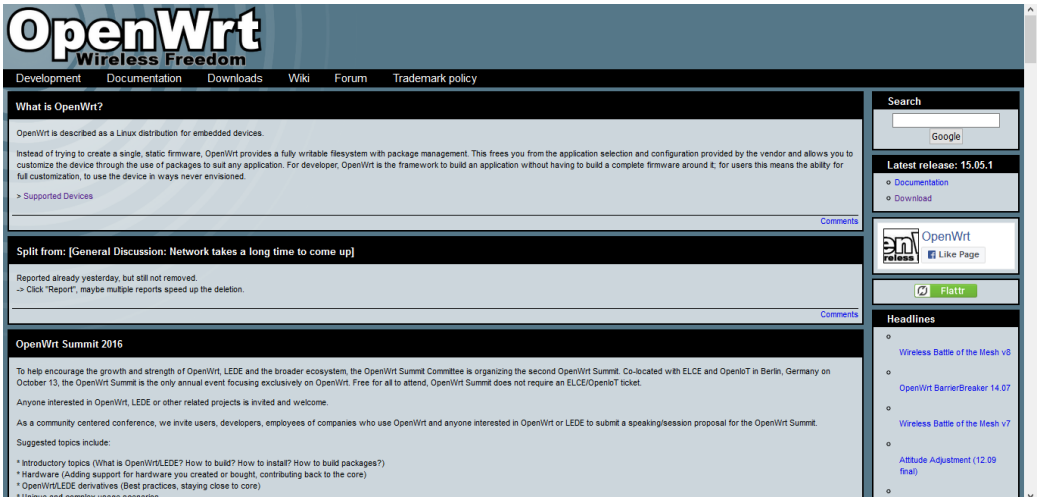

Figura 15. Página web de OpenWRT
Fuente: Elaboración propia

Existen varios release de OpenWRT, el seudónimo de estos se compone de dos partes, un nombre de un coctel y un numero de versión. El número de versión se compone de dos números separados por un punto, el primero indica el año de creación y el segundo el mes. Se adiciona un tercer número si existe un lanzamiento provisional del mismo, en la Tabla 8 se muestran los release y el año de su lanzamiento.

Tabla 8. Versiones de OpenWRT

Nombre del Release	Fecha de Lanzamiento	
	Año	Mes
Chaos Calmer 15.05.1	2016	Marzo
Chaos Calmer 15.05	2015	Septiembre
Barrier Breaker 14.07	2014	Octubre
Attitude Adjustment 12.09	2013	Abril
Backfire 10.03.1	2011	Diciembre
Backfire 10.03	2010	Abril
Kamikaze 8.09.2	2010	Enero
Kamikaze 8.09.1	2009	Junio
Kamikaze 8.09	2008	Septiembre
Kamikaze 7.09	2007	Septiembre
Kamikaze 7.07	2007	Julio
Kamikaze 7.06	2007	Junio
Whiterussian 0.9	2007	Enero

Fuente: Recuperado de OpenWrt

Para elegir la versión correcta para un dispositivo OpenWRT ofrece una lista denominada "Table of Hardware" en donde se puede consultar los dispositivos compatibles con el firmware y la versión que debe ser utilizada. (OpenWRT, 2004) Además se deberá conocer la versión y el nombre de los controladores de red del dispositivo a utilizarse, de esta manera se escogerá la versión apropiada para el

dispositivo. OpenWRT permite la configuración de los dispositivos a través de interfaz Web LuCI, que es una interfaz gráfica de fácil uso y a través de la interfaz de línea comandos. Para obtener acceso a través de la interfaz de línea de comandos cuando se trabaja en sistemas operativo Windows se necesita de algún programa como PuTTY, que es un programa cliente de SSH y telnet, desarrollado originalmente por Simon Tatham para la plataforma de Windows. OpenWRT utiliza el editor "vi" por defecto, aunque permite la instalación de otros editores como vim, joe, nano y zile. Este firmware utiliza OPKG (del inglés, Open PacKaGe) como administrador de paquetes, que permite descargar e instalar paquetes desde repositorios locales o ubicados en Internet. Este comando se utiliza de la siguiente manera: opkg [opciones...] subcomando [argumentos...]. Entre las opciones que se puede usar están: update, upgrade, install, configures y remove. La versión del protocolo es OLSRd versión 0.6.4, y vienen configurado por defecto con el algoritmo etx_ff para calcular la calidad del enlace. (GitHub, 2012) (OpenWRT, 2004)

Para el protocolo OLSR, se debe tener en cuenta algunos criterios en cuanto a la calidad del enlace y a la métrica ETX, cuando se analiza las tablas y las gráficas proporcionadas por las diferentes versiones del firmware, dese la Backfire 10.03 se maneja 4 colores, el VERDE indica un enlace "muy bueno" cuando el valor ETX es menor a 2 (ETX < 2), el color AMARILLO indica un enlace "bueno" cuando el valor de ETX se encuentra entre 2 y 4 (2 < ETX < 4), NARANJA indica un enlace es "todavía utilizable" cuando el valor de ETX se encuentra entre 4 y 10 (4 < ETX < 10), y el color ROJO indica que un enlace es malo cuando su ETX es mayor a 10 (ETX > 10).

4.9 HERRAMIENTAS DE ANÁSLISIS

A continuación se realizará una descripción breve de las herramientas utilizadas en el presente trabajo para el análisis del prototipo de red.

4.9.1 Wireshark

Wireshark es el analizador de protocolo de red ampliamente utilizado, desarrollado en 1997 por Gerald Combs con el nombre de Ethereal y publicado en 1998, desde entonces

un gran número de personas han contribuido al desarrollo de este proyecto. Entre sus características tenemos:

- Multiplataforma: se ejecuta en Windows, Linux, macOS, Solaris, FreeBSD, NetBSD y muchos otros
- Se distribuye bajo la licencia de código abierto General License GNU (del inglés, GNU's Not Unix)
- Funciona en modo promiscuo y no promiscuo.
- Tiene una GUI amigable con el usuario.
- La salida se puede exportar a XML, PostScript®, CSV o texto sin formato (Wireshark, 1999) (Orebaugh, 2007)

4.9.2 Iperf

Iperf es una herramienta para mediciones activas del ancho de banda máximo alcanzable en redes IP, permite ajustar varios parámetros relacionados con temporización, búferes y protocolos (TCP, UDP, SCTP con IPv4 e IPv6). Por cada prueba realizada, iperf permite obtener información del ancho de banda, la pérdida de paquetes entre otros parámetros. Además existe la versión gráfica de iperf basada en java denominada Iperf y una versión ya compilada en OpenWRT

- TCP y SCTP
  - Medición del ancho de banda
  - Informe el tamaño de MSS / MTU y los tamaños de lectura observados.
  - Soporte para el tamaño de ventana TCP a través de buffers de socket.
- UDP
  - Creación de flujos UDP de ancho de banda especificado.
  - Medición de la pérdida de paquetes
  - Medición de la fluctuación de retardo
- Multiplataforma: Windows, Linux, Android, MacOS X, FreeBSD, OpenBSD y otros.
- El cliente y el servidor pueden tener múltiples conexiones simultáneas
- El servidor maneja conexiones múltiples.

- Puede ejecutarse durante el tiempo especificado (opción -t), en lugar de una cantidad de datos establecida para transferir (opción -n o -k).
- Imprime informes periódicos, de ancho de banda intermedio, jitter e informes de pérdidas a intervalos específicos (opción -i). (The Iperf team, 2003)

4.9.3 Ping

La herramienta de diagnóstico PING (del inglés, Packet Internet Groper) permite verificar el estado de una conexión entre un host local con uno o varios equipos remotos mediante el envío de paquetes ICMP (del inglés, Internet Control Message Protocol), esta herramienta es utilizada para medir el retardo o tiempo en que tarda en comunicarse con puntos remotos.

4.9.4 Elastix

Elastix es un PBX [4](del inglés, Private Branch Exchange) basado en software de estándares abiertos, fácil de instalar y administrar. Considerado como un sistema telefónico apropiado para cualquier empresa, y pudiendo ser instalado on premise como máquina virtual o en la nube (en Google Cloud, OVH, entre otros) dependiendo de las necesidades. Es gratis para hasta 8 llamadas simultáneas con 25 usuarios aproximadamente. (Elastix.org, 2006)

4.9.5 Wampserver

WampServer (acrónimo utilizado para describir un entorno de desarrollo que utiliza **W**indows como sistema operativo, **A**pache como servidor web**, M**ySQL como gestor de base de datos y **P**HP como lenguaje de programación.), es un entorno de desarrollo web de Windows que permite crear aplicaciones web con Apache2, PHP y una base de datos MySQL. WampServer es muy fácil de instalar y de usar, entre las funcionalidades que presenta tenemos: (Bourdon, 2004)

- Administrar los servicios Apache y MySQL
- Instalar e intercambiar versiones de Apache, MySQL y PHP

[4] Un PBX es una central telefonica, que permite realizar llamadas. En el caso de las IP PBX, las llamadas son realizadas utilizando el protocolo Ip como protocolo de transporte.

- Administrar la configuración del servidor
- Acceder a los registros
- Acceder a los archivos de configuración
- Crear alias

4.9.6 Softphone

Un softphone es un software que permite realizar llamadas VoIP, se puede ejecutar en diversas plataformas y proveen de una interfaz gráfica que simula un teléfono. Se utilizó PortGo, que es un softphone SIP y una de las aplicaciones más confiables y diversas en funciones para Apple iOS, Android y Windows Desktop, es una aplicación sencilla y de fácil uso. (PortSip Solutions, 2010)

4.9.7 Wifi Analizer

Wifi Analyzer es una aplicación para dispositivos Android que permite analizar las redes WiFi en nuestro entorno, en la primera pestaña nos permite apreciar una gráfica de la calidad de la señal de todas las redes wifi cercanas y en una segunda pestaña la saturación de cada red. (López, 2013)

5. MATERIALES Y MÉTODOS

Para el desarrollo del presente trabajo de investigación se realizó una indagación sobre WMNs con software libre, el protocolo OLSR en diversas fuentes bibliográficas, los conocimientos adquiridos coadyuvaron al desarrollo y cumplimiento de los objetivos planteados. Por las diversas características del firmware OpenWRT frente a otros y por la extensa cantidad de información acerca del mismo, fue el elegido para la instalación en los routers Linksys WRT54GS v 3.0. En la Figura 16 se muestra el diagrama de trabajo realizado para para el cumplimiento de los objetivos.

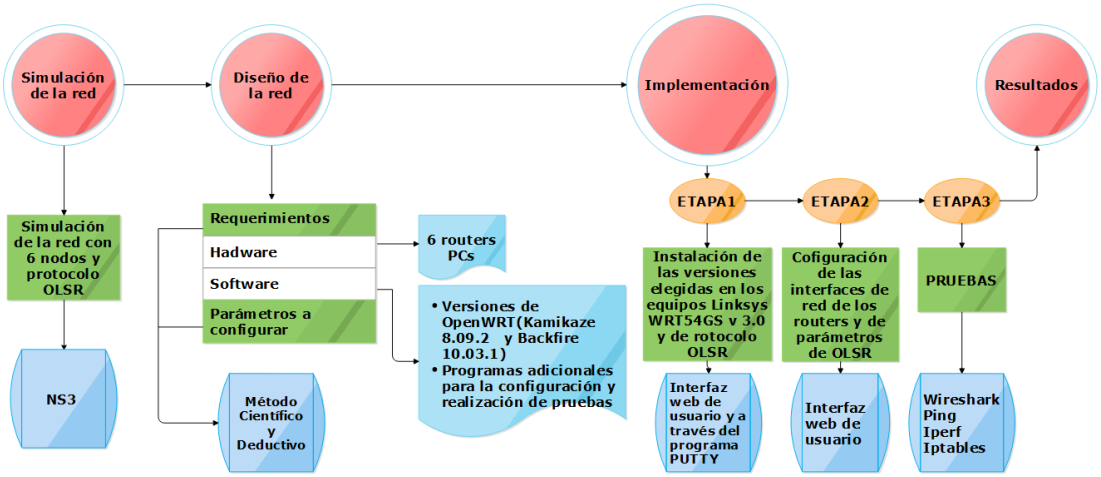

Figura 16. Diagrama de trabajo
Fuente: Elaboración propia

5.1 Simulación de la Red Mesh

En la actualidad existe una gran variedad de plataformas para la simulación de redes inalámbricas, sin embargo son pocas las que permite evaluar protocolos de enrutamiento como OLSR. NS-3 (del inglés, Network Simulator 3) es un simulador de redes inalámbricas que proporciona una plataforma de simulación destinada para investigación y uso educativo, es un software gratuito, licenciado bajo la licencia GNU GPL, y está a disposición del público para investigación, desarrollo y uso. El objetivo de NS-3 es desarrollar un entorno de simulación para la investigación de redes, bien documentado y de fácil uso. NS-3 incluye modelos para Wi-Fi, WiMaX, LTE, OLSR y AODV. (NS3: Network Simulator, 2010).

Es por esta razón que la simulación de red se realizó en NS3, el mismo que fue instalado en máquina virtual en el sistema Ubuntu 16.04 LTS, para su instalación se siguió la guía que presenta la página web oficial de NS3 (NS3: Network Simulator, 2010), y se procedió a la modificación del archivo "olsr-hna.cc" La topología que presenta este archivo es la que se observa en la Figura 17, donde hay dos nodos OLSR, los nodos A y B, y un nodo non-OLSR , el nodo C. El nodo B está conectado mediante cable al nodo C, y el nodo A y B de manera inalámbrica. Para permitir la comunicación entre el nodo C y A, el nodo B deberá anunciar su dirección de red del nodo C a través de los menajes HNA.

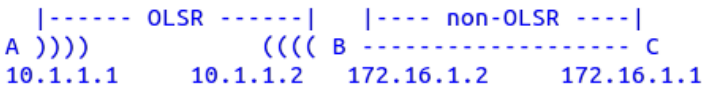

Figura 17. Topología de red que presenta el archivo OLSR-HNA de ns3.
Fuente: Recuperado de NS3 por https://www.nsnam.org/ archivo olsr-hna.cc

Existen 3 maneras de ejecutar este archivo:

- ./waf –-run olsr-hna. El nodo OLSR conectado al nodo HNA no generará mensajes HNA, por lo que solo existirá comunicación entre los nodos OLSR

Para la generación de mensajes HNA se podrá ejecutar el archivo a través de los comandos mostrados a continuación, lo que permitirá la comunicación entre el nodo A y C, estos métodos son similares y se pueden utilizar cualquiera.

- ./waf –-run "olsr-hna --assocMethod1=1" En este método la generación de los mensajes HNA se hace a través de la agregación de las rutas en la instancia Ipv4StaticRouting de manera que el nodo genere mensajes HNA para las rutas que estén asociadas con la interfaz noOLSR especificada al inicio del archivo.
- ./waf –-run "olsr-hna --assocMethod2=1" En este método la asociación se realiza directamente, es decir utiliza las líneas de simulación en donde se especifica las direcciones de red del nodo noOLSR para realizar la comunicación entre los nodos. (NS3: Network Simulator, 2010)

Se configuró el archivo de manera que la simulación se realice con 6 nodos OLSR y se analizaran los archivos .pcap generados para cada uno de ellos. En el Anexo 2 se indica el archivo "olsr-hna" modificado.

5.2 Diseño de la Red

Para el diseño del prototipo se deben tomar en cuenta algunas pautas, como el estudio del hardware (Linksys Wrt54gs v 3.0) y del firmware (OpenWRT) utilizados, de esta manera se adquirió conocimientos que fueron aplicados en cada una de las fases del proyecto, siguiendo las recomendaciones de la página oficial del firmware (OpenWRT, 2004) se pudo elegir el release del firmware correcto de acuerdo a la versión de nuestro router. La instalación del firmware se la realizo mediante la interfaz web de usuario, de igual manera las configuraciones; aunque también se la puede realizar mediante línea de comandos, la configuración de cada parámetro se las realizo siguiendo algunos ejemplos que plantea la página oficial de OpenWRT (OpenWRT, 2004), los archivos que corresponden a las configuraciones realizadas son sytem, network, Wireless, olsrd y firewall.

5.2.1 Requerimientos de Hardware y Software

Para el despliegue de la Red Inalámbrica Mallada con OpenWrt y protocolo OLSR, se debe tener en cuenta algunos requerimientos tanto de software como de hardware previos a la implementación de la red. A continuación se detalla cada uno de los elementos utilizados:

5.2.1.1 Requerimientos de Hardware

- Routers Linksys WRT54gs v 3.0: la red está compuesta por 6 routers que conformarán la red troncal, para dar acceso se tendrá dos opciones: acceder al router conectándose a través de un cable o inalámbricamente a través de un AP.
- Dos PC o Laptops con tarjeta Ethernet, tendrán el seudónimo de PC1 y PC2, fueron necesarias para la configuración de los routers, instalación del servidor Elastix y las pruebas.
- Un Smartphone, que será utilizado para la instalacion del softphone necesario para realizar las pruebas de VoIP.

5.2.1.2 Requerimientos de Software

- Firmware OpenWRT, release “Kamikaze 8.09.2” y “Backfire 10.03.1” para el kernel 2.4, que son los release compatibles con la versión de nuestro router, se

utilizó estos archivos porque contiene los drives del controlador original del router, para versiones de Kernel 2.6 o superiores los fabricantes aún no han liberado estos archivos. Versiones superiores se pueden instalar, pero presentan fallas al momento de su funcionamiento.

- Programa PUTTY, utilizado para la configuración e instalación de paquetes en los routers Linksys a través de línea de comandos en Windows.
- Wireshark, utilizado para la captura de tráfico y analizar el transporte de paquetes y manejo de protocolos, instalado en PC1 y PC2.
- Elastix, que es servidor de VoIP., instalado en máquina virtual en PC1
- Softphone, PortGo instalado en un Smartphone y en PC1 y PC2 para el desarrollo de pruebas.
- WifiAnalizer, permitió elegir el canal en el cual opera la red, a través de la información que brinda sobre las redes que se encuentran funcionando en el lugar.
- Iperf para el análisis de la tasa de transmisión y retardo, esta herramienta fue instalada en cada uno de los routers mediante el sistema de gestión de paquetes opkg.
- Ping para conocer el retardo en diferentes escenarios.

A continuación, en la Tabla 9 se presenta un listado de la versión instalada de OpenWRT en cada uno de los routers que conforman la red.

Tabla 9. Versión instalada en cada router

Nº	NOMBRE DEL ROUTER	DIRECCIÓN MAC	VERSIÓN INSTALADA	HARDWARE
1	NODO1	00:1C:10:0D:89:55	Kamikaze 8.09.2	Linksys WRT54GS v 3.0
2	NODO2	00:1C:10:0D:89:1C	Kamikaze 8.09.2	
3	NODO3	00:1C:10:0D:89:19	Kamikaze 8.09.2	
4	NODO4	00:1C:10:0D:89:31	Backfire 10.03.1	
5	NODO5	00:1C:10:0D:89:13	Kamikaze 8.09.2	
6	NODO6	00:1C:10:0D:89:01	Kamikaze 8.09.2	

Fuente: Elaboracion propia

5.2.2 Parámetros a configurar

Para la implementación de la WMN hay que tener en cuenta algunos parámetros como el rango de direcciones IP utilizadas, el canal en el cual opera los nodos y otros; a continuación se detalla cada uno de ellos:

5.2.2.1 Direcciones IP

Las direcciones IPs fueron asignadas según la Tabla 10, además se indica el parámetro HNA que deberá ir configurado en cada router

Tabla 10. Direcciones asignadas a los nodos.

NOMBRE DEL NODO	DIRECCIONES IPS		HNA (Host and Network Association)
	Interfaz Inalámbrica	Interfaz Ethernet	
NODO1	10.0.0.1 / 24	192.168.1.1 / 24	192.168.1.0/24
NODO2	10.0.0.2 / 24	192.168.2.1 / 24	192.168.2.0/24
NODO3	10.0.0.3 / 24	192.168.3.1 / 24	192.168.3.0/24
NODO4	10.0.0.4 / 24	192.168.4.1 / 24	192.168.4.0/24
NODO5	10.0.0.5 / 24	192.168.5.1 / 24	192.168.5.0/24
NODO6	10.0.0.6 / 24	192.168.6.1 / 24	192.168.6.0/24

Fuente: Elaboración propia

A continuación se detalla las direcciones IP configuradas en el servidor Elastix y WAMPserver, como se mencionó, para tener acceso a la red existen dos maneras, conexión a través de cable al router e inalámbricamente a través de un AP, este estará conectado a cualquier nodo de la red por lo que la X corresponde al tercer octeto de la dirección de la interfaz LAN del router que esté conectado, es decir si está conectada al NODO1 la IP será 192.168.1.2, si estuviera conectado al NODO3 será 192.168.3.2 y así sucesivamente.

Tabla 11. Direcciones Ip del servidor y del AP

	DIRECCIONES
Elastix	192.168.1.10/24
WAMPserver	192.168.1.5/24
Ap	192.168.X.2/24

Fuente: Elaboración propia

5.2.2.2 Canal, Modo de operación y nombre de la red

Como se manifestó en secciones anteriores, los nodos deberán operar en el mismo canal, tener el mismo nombre de red o SSID y operar en modo Ad-Hoc. Las redes inalámbricas que operan en la banda ISM de 2,4 GHz, está dividida en 11 canales utilizables, sin embargo solo 3 de ellos no se solapan como se puede observar en la Figura 18, por lo que hay que tomar en cuenta que solo se podrá elegir entre el canal 1, 6 y 11. Para elegir el canal se utilizó el programa Wifi Analizer, el cual realizó un sondeo en el lugar a desplegarse la red, permitiéndonos conocer los canales utilizados por otras redes. Como se puede observar en la Figura 19, el canal menos utilizado es el 6, por esta razón este será el elegido para que opere nuestra red.

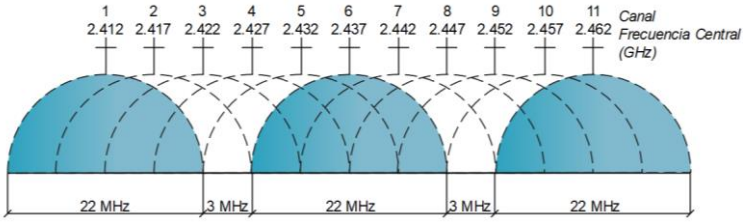

Figura 18. Canales no solapados en la banda ISM 2,4 GHz

Fuente: Recuperado de https://revistas.unimilitar.edu.co/index.php/rcin/article/view/220/1854

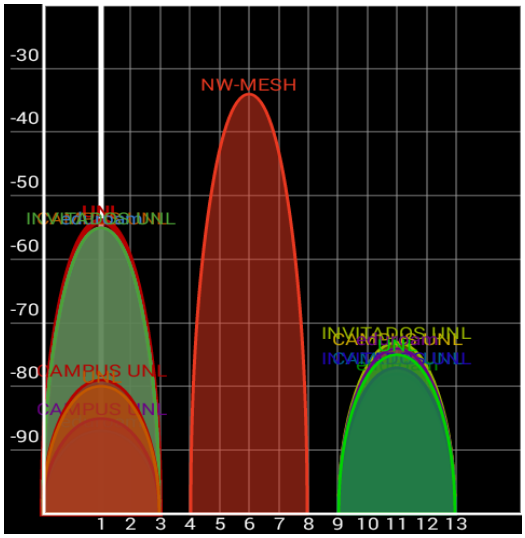

Figura 19. Sondeo de canales

Fuente: Elaboración propia

El nombre elegido para nuestra red es NW-MESH. En la Tabla 12 se presenta los parámetros que deben ir configurados en todos los routers.

Tabla 12. ESSID y modo de operación de la red.

PARÁMETROS	
Canal	6
ESSID	NW-MESH
Modo	Ad-Hoc

Fuente: Elaboración propia

5.2.2.3 Potencia

OpenWRT permite elevar la potencia de transmisión del Linksys WRT54GS v3.0, el release utilizado en este proyecto permite llegar hasta 32dBm, pero debemos tomar en cuenta que el límite permitido para redes de corto alcance es de 100 mW o 20dBm, además no es aconsejable aumentar este parámetro ya que estaríamos forzando al dispositivo a operar en niveles de potencia que no está diseñado provocando que el dispositivo se apague repentinamente o en el peor de los casos daños en la placa que conllevarían a averías irreversibles y además de la generación de señales espurias, por

eso aconsejable que opere a 20dBm (100mW) según Redes Inalámbricas en los Países en Desarrollo (WNDW, 2013).

5.2.2.4 Ubicación

La red mesh estará implementada en el laboratorio de Telecomunicaciones ubicado en la segunda planta del edificio del laboratorio del Área (hoy Facultad) de la Energía, las Industrias y los Recursos Naturales no Renovables. El laboratorio mide aproximadamente 6,45 m de ancho y 13,7 metros de largo como se pude observar en la Figura 20.

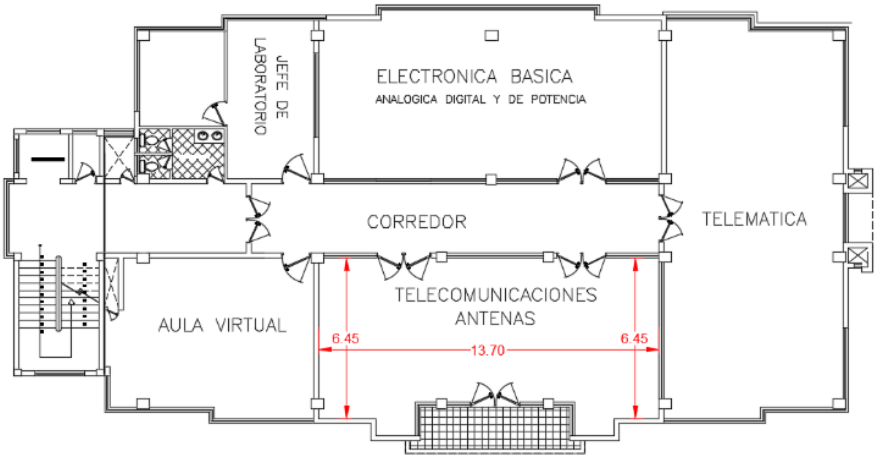

Figura 20. Plano de la segunda panta del edifico de laboratorios.
Fuente: Elaboración propia

Los equipos serán ubicados en los escritorios del laboratorio, de esta manera nos permitirá realizar pruebas en diferentes escenarios. Los equipos estarán distribuidos en el Laboratorio de Telecomunicaciones según la Figura 21

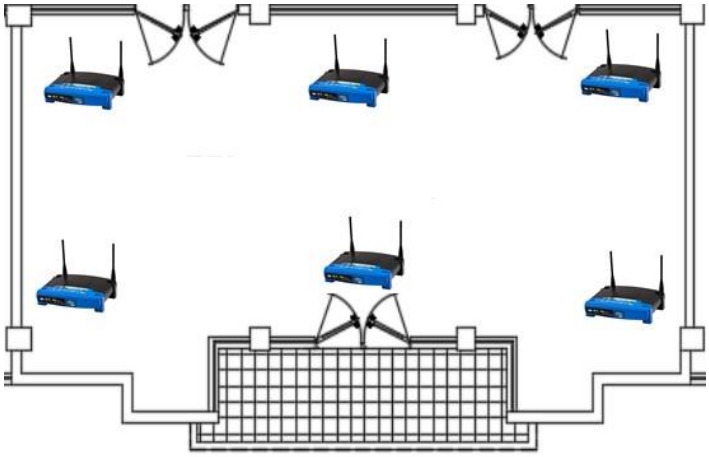

Figura 21. Distribución de los equipos en el Laboratorio de Telecomunicaciones.
Fuente: Elaboración propia

5.3 Implementación de la Red Mesh Inalámbrica

La implementación del prototipo de Red se realizó en varias etapas, que se las detalla en la Figura 22

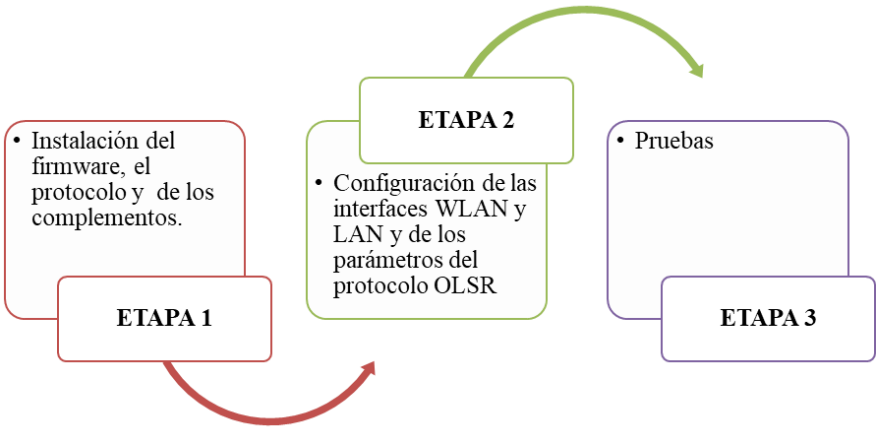

Figura 22. Etapas de la implementacion de la red
Fuente: Elaboración propia

Como requisito previo se deberá conocer la dirección IP del router, si no se tiene conocimiento de este parámetro, se puede resetear el router, de esta manera se podrá ingresar al dispositivo con la IP de fábrica, en el caso del Linksys WRT54GS v3.0 es la 192.168.1.1, además se deberá configurar la computadora con la que se realizará el proceso de instalación del firmware con una IP estática que se encuentre dentro del rango de la IP del router.

5.3.1 Etapa 1: Instalación

En esta primera etapa se realizó la instalación del firmware OpenWrt en los routers Linksys WRT54gs v 3.0, la instalación se puede realizar de diversas maneras, pero la utilizada fue a través de interfaz web, a continuación se detalla de manera breve el proceso:

- Ingresar a un explorador web
- Escribir la dirección IP del dispositivo en una pestaña de navegador.
- Escribir el username y el password
- Ingresar a “Administration/ Firmware Upgrade”

- Cargar el archivo .bin y esperar hasta que el equipo realice un reboot que indica que el proceso se ha realizado correctamente.

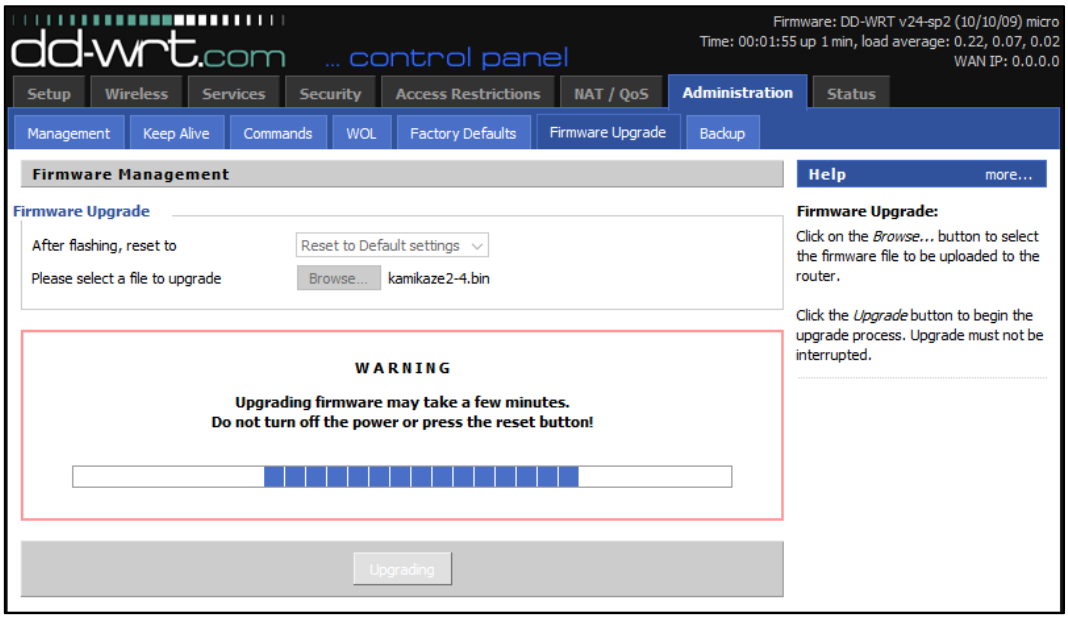

Figura 23. Instalación del firmware a través de Interfaz web
Fuente: Elaboración propia

En caso de que ocurra una mala configuración o haya ocurrido un mal flasheo del dispositivo, se puede seguir los siguientes pasos:

- Encender el router y presionar el botón RESET por unos segundos hasta que los leds POWER y DMZ titilen, una vez conseguido esto se deberá apagar el router.
- Escribir en el cmd del computador el siguiente comando sin ejecutarlo:

```
tftp -i 192.168.1.1 put C:\firmware.bin
```

- Al momento de encender el router se debe ejecutar el comando, y se espera unos minutos hasta que aparezca un mensaje indicándonos que nuestra transferencia de datos ha sido exitosa.
- Esperar a que el router realice un reboot.

Una vez instalado OpenWRT se procedió a la instalación de los paquetes necesarios para el funcionamiento del protocolo mediante el sistema de gestión de paquetes opkg, a continuación se describen los paquetes instalados en la Tabla 13

Tabla 13. Paquetes instalados en OpenWRT.

NOMBRE DE PAQUETE		DESCRIPCION
Protocolo OLSR	olsrd	Demonio del protocolo OLSR
Complementos para OLSR	olsrd-mod-dyn-gw-plain	Permite activar o desactivar automáticamente direcciones HNA, en función a la conectividad, es decir dependiendo de la existencia o ausencia de la ruta en el nodo.
	olsrd-mod-nameservice	Permite distribuir información acerca del nombre del nodo, es decir a momento de ejecutarse OLSR cada nodo anuncia su nombre a los otros nodos con los que está asociado. Utilizan los archivos /etc/hosts y /etc/resolv.conf para guardar su información.
	olsrd-mod-txtinfo	Este complemento acepta cualquier conexión en un socket, y devolverá tablas de información de estado, puede servir como base para muchos otros complementos de visualización.
Paquetes adicionales	nano	Editor de texto utilizado en el proyecto.
	luci-app-firewall	Permite realizar cambios en los parámetros de Firewall a través de la interfaz grafica
	luci-app-olsr	Permite realizar cambios en los parámetros de OLSR a través de la interfaz grafica
Paquetes para visualización de la red	olsrd-mod-dot-draw	Produce la topología en un archivo de formato dot en el puerto TCP 2004, la que posteriormente se puede visualizar a través de programas adicionales
	luci-app-olsr-viz	Genera una gráfica de la red que puede ser visualizada a través de la interfaz web.

Fuente: Recuperado de OpenWRT

5.3.2 Etapa 2: Configuración

Esta etapa consiste en la configuración de cada interfaz de red del router, y de la instalación de paquetes, OpenWRT permite la configuración a través de línea de comandos y de interfaz gráfica. A continuación se detallará de manera breve los parámetros configurados, en el Anexo 4 y se tendrá información más minuciosa sobre la configuración realizada y el resultado final de los archivos de configuración del router.

1. ***Cambio de password y nombre del router***
   a. Ingresar a “System/Admin Password”
   b. Cambiar la contraseña del equipo
2. ***Nombre de host***
   a. Ingresar a “System”
   b. Ingresar en Hostname el nombre del router, y elegir en Time Zone la zona horario adecuada en nuestro caso fue elegida “América/Guayaquil”

3. ***Interfaz Inalámbrica***
   a. Ingresar a “Network/Interfaces”
   b. Crear la interfaz “wl0”
   c. Crear zona “wl0”
   d. Creada la interfaz se llenó los campos correspondientes a dirección Ip y mascara de red
   e. Ingresar a “Network/Wifi/WL0”
   f. Escoger el canal en el cual trabajará la red inalámbrica.
   g. Escribir la potencia de operación del equipo, sino se realiza el cambio automáticamente se configura en 32dBm.
   h. Elegir en Network la opción “wl0”
   i. Escribir en ESSID el nombre de la red.
   j. Elegir el modo de operación.
4. ***Interfaz LAN***
   a. Ingresar a “Network/Interfaces/LAN”
   b. Desmarcar la opción “Bridge interfaces”
   c. Ingresar la dirección Ip y la máscara de red
   d. Crear la zona “lan”
5. ***OLSR***
   a. Ingresar a “Sevices/OLSR”
   b. Elegir en Network “wl0”
   c. Ingresar a “Services/OLSR/Plugins”
   d. Habilitar los plugins instalados marcando la casilla Enable
   e. Ingresar a “Services/OLSR/HNA Announcements”
   f. Añadir la dirección Ip y la máscara en el campo HNA
6. ***Firewall***
   a. Ingresar a “Network/Firewall”
   b. Dirigirse a la seccion “Zone”
   c. Asegurarse que en la columna “Name” se encuentren el nombre de las zonas creadas (lan y wlan) y que a su vez correspondan con las redes las cuales se indican en la columna “Networks”.

d. Elegir “accept” en las columnas de “Imcoming Traffic”, “Outgoing Traffic” y “Forwarded Trafic”

7. ***Editar el script “firewall.user”***
   a. Ingresar al router a través del programa PUTTY
   b. Ingresar al archivo firewall.user con el siguiente comando “nano /etc./firewall.user”
   c. Colocar las líneas que indica la página oficial (OpenWRT, 2004)
8. ***Visualización de la red***

En la versión Backfire 10.03.1 y superiores se puede instalar un pluging denominado luci-app-olsr-viz el cual permite visualizar de manera gráfica como está formada la red, además el gráfico se genera en una pestaña de la interfaz web del nodo, la misma que indica el nombre del nodo, la dirección IP de la interfaz inalámbrica y el valor de ETX de cada enlace. En cambio en el release Kamikaze 8.09.2 es posible generar gráficos de la red a través del pluging olsrd-mod-dot-draw, para su instalación se siguió la guía que se presenta el Capítulo 3 de Redes Inalámbricas de Países en Desarrollo (WNDW, 2013) La generación de gráficos a través del plugin dot-draw necesita de programas adicionales, por lo que en el Anexo 6 se detalla el proceso de generación.

Los campos correspondientes a: nombre del equipo, nombre de la red, dirección Ip y máscara de red, modo de operación, canal, potencia de transmisión, y campo HNA se configuraron según la Tabla 10 y Tabla 12

5.3.3 Etapa 3: Pruebas

En esta etapa se realizó las pruebas para comprobar el funcionamiento de la red y del protocolo, para lo cual se plantearon algunos escenarios. Para obtener algunas de las topologías utilizadas se editó el script “firewall.user”, a través de la herramienta iptables; aceptando los mensajes de control que lleguen al puerto 698 de los nodos que se quieran tener como vecinos directos, mientras que los otros se rechazan, el orden en que se colocan las reglas es un aspecto determinante ya que se van leyendo y aplicando en el orden colocado. La Figura 24 muestra un ejemplo de cómo deben colocarse las reglas, si se quiere tener al NODO2 como vecino directo, mientras que en la Figura 25 se muestra un esquema de las pruebas realizadas para el análisis del protocolo como para el análisis de la red.

```
iptables -A input_rule -s 10.0.0.2 -p udp --dport 698 -j ACCEPT
iptables -A input rule -p udp --dport 698 -j DROP
```

Figura 24. Reglas de firewall utilizadas para la creación de las topologías.
Fuente: Elaboración propia

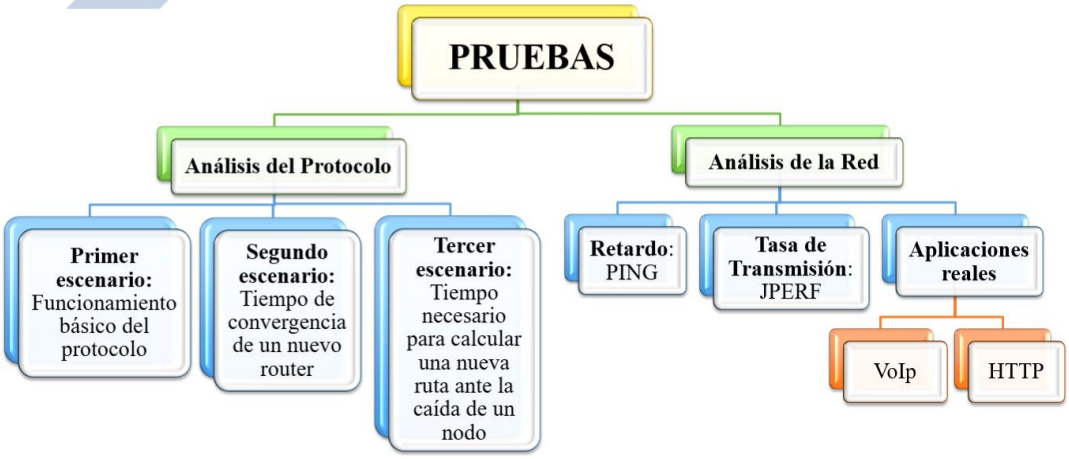

Figura 25. Pruebas realizadas para el análisis del protocolo y la red
Fuente: Elaboración propia

5.3.3.1 Análisis del protocolo.

Para el análisis de protocolo se propusieron tres escenarios que a continuación se detallan:

- ✓ **Primer Escenario: Funcionamiento básico del protocolo**

Una vez configurados los equipos, se los ubico como se indica en la Figura 21, la configuración realizada en la Etapa 2, permite que todos los nodos se conecten entre sí como muestra la Figura 26, luego se procede a realizar la captura de paquetes a través de Wireshark, para después analizarlos. Este escenario permitió obtener información acerca de los paquetes de control del protocolo OLSR. Para conectarse a la red y capturar el tráfico de control, se utilizó el programa *Red Ad-Hoc Viewer* que nos permite detectar y conectarnos a una red ad hoc (Neyra, 2014).

Para comprobar que los nodos estén conectados de una manera total se procederá a visualizar la tabla y la gráfica generada a través de la interfaz web del NODO4. En la Figura 27 se puede apreciar los enlaces entre los nodos, mientras que en la Figura 28 se muestra de manera gráfica la conexión de los nodos.

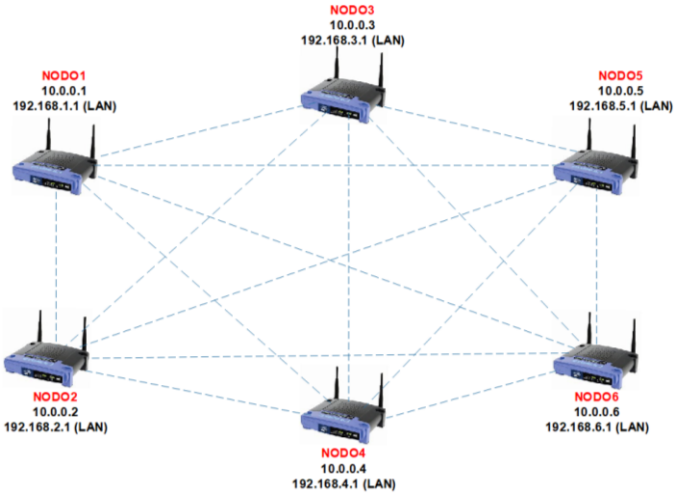

Figura 26. Topología utilizada en el primer escenario

Fuente: Elaboración propia

OLSR node	Last hop
10.0.0.1	10.0.0.5
10.0.0.1	10.0.0.4
10.0.0.1	10.0.0.2
10.0.0.1	10.0.0.3
10.0.0.2	10.0.0.1
10.0.0.2	10.0.0.5
10.0.0.2	10.0.0.3
10.0.0.2	10.0.0.6
10.0.0.2	10.0.0.4
10.0.0.3	10.0.0.4
10.0.0.3	10.0.0.5
10.0.0.3	10.0.0.6
10.0.0.3	10.0.0.2
10.0.0.3	10.0.0.1
10.0.0.4	10.0.0.3
10.0.0.4	10.0.0.5
10.0.0.4	10.0.0.6
10.0.0.4	10.0.0.2
10.0.0.4	10.0.0.1
10.0.0.5	10.0.0.6
10.0.0.5	10.0.0.4
10.0.0.5	10.0.0.1
10.0.0.5	10.0.0.3
10.0.0.5	10.0.0.2
10.0.0.6	10.0.0.5
10.0.0.6	10.0.0.2
10.0.0.6	10.0.0.4
10.0.0.6	10.0.0.3

Figura 27. Topología de la red / 1er escenario

Fuente: Elaboración propia

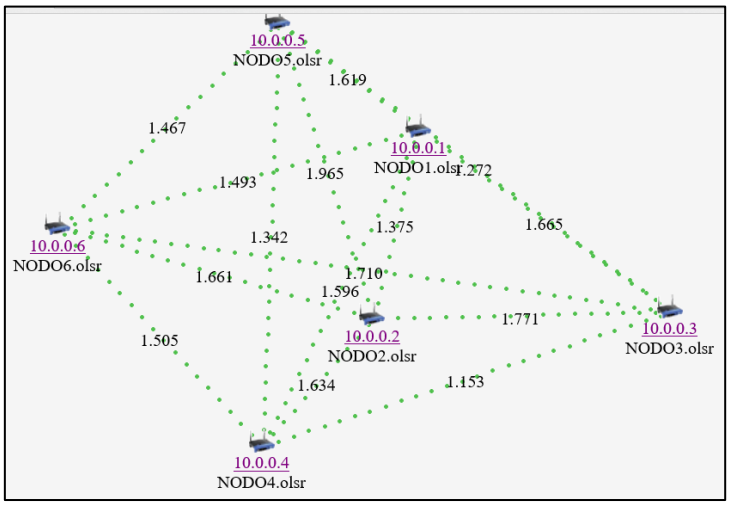

Figura 28. Grafica de la red/ 1er escenario

Fuente: Elaboración propia

- ✓ **Segundo Escenario: Tiempo de convergencia de un nuevo nodo**

Este escenario permitió conocer el tiempo de convergencia de un nuevo nodo, por lo que se creó un archivo ejecutable .bat (Nagashagui, 2015), el cual brindo información del tiempo en que tarda un nuevo router en establecer comunicación con otro mediante la herramienta ping. El NODO1 y el NODO2 estarán encendidos, PC1 estará conectado a NODO1 en donde se alojará y se ejecutará el archivo .bat. Se procede a ejecutar el archivo .bat luego se reinició el demonio OLSR en el NODO1 aplicando el siguiente comando *"/etc./init.d/olsrd restart"*, de manera que los dos nodos estén encendidos pero solo en uno de ellos esté corriendo el protocolo. Se calculó el tiempo desde el primer ping fallido y el primer ping exitoso.

- ✓ **Tercer escenario: Cálculo de una nueva ruta**

Este escenario permitió conocer el tiempo que tarda el protocolo en calcular rutas alternativas cuando cae un nodo, se utilizó la topología implementada que se indica en la Figura 29, y se empleó el archivo .bat utilizado en los escenarios anteriores.

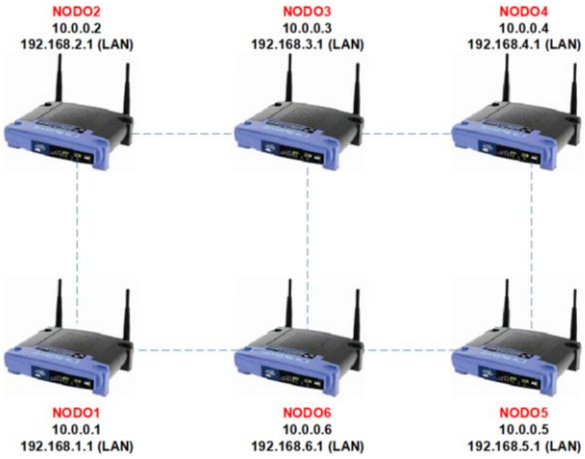

Figura 29. Topología utilizada para las pruebas del tercer escenario.
Fuente: Elaboración propia

La comunicación entre el NODO1 y NODO5 se realiza a través del NODO6 (camino elegido por el protocolo), como se puede observar en la Figura 30 a través del comando *traceroute*. Se ejecutó el archivo .bat desde PC1 que está conectado a NODO1, y luego se apagó el NODO6, y se tomó el tiempo que tarda en llegar el primer ping al NODO5 a través de la nueva ruta. En la Figura 31 se puede apreciar los enlaces entre los nodos, mientras que en la Figura 32 se muestra de manera gráfica la conexión de los nodos.

```
root@NODO1:~# traceroute 10.0.0.5
traceroute to 10.0.0.5 (10.0.0.5), 30 hops max, 38 byte packets
 1  10.0.0.6 (10.0.0.6)  2.125 ms  1.881 ms  5.982 ms
 2  10.0.0.5 (10.0.0.5)  6.796 ms  3.328 ms  3.267 ms
```

Figura 30. Resultado del comando Tracert/ Tercer escenario

Fuente: Elaboración propia

OLSR node	Last hop
10.0.0.1	10.0.0.2
10.0.0.1	10.0.0.6
10.0.0.2	10.0.0.1
10.0.0.2	10.0.0.3
10.0.0.3	10.0.0.4
10.0.0.3	10.0.0.2
10.0.0.3	10.0.0.6
10.0.0.4	10.0.0.3
10.0.0.4	10.0.0.5
10.0.0.5	10.0.0.6
10.0.0.5	10.0.0.4
10.0.0.6	10.0.0.5
10.0.0.6	10.0.0.1
10.0.0.6	10.0.0.3

Figura 31. Topología utilizada /3er escenario

Fuente: Elaboracion propia

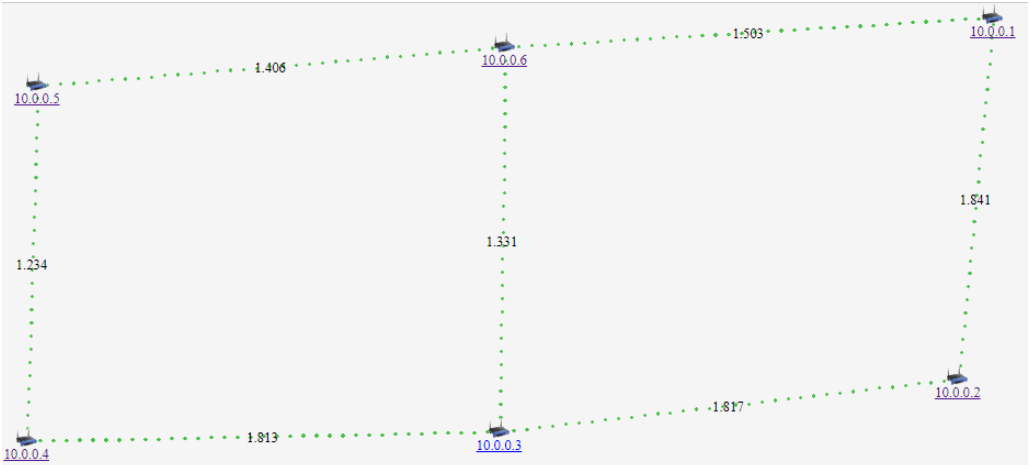

Figura 32. Gráfica de la red/ 3er escenario

Fuente: Elaboracion propia

5.3.3.2 Análisis de la Red.

Para el análisis de la red se propuso un escenario que permitió evaluar el impacto del aumento de número de saltos en parámetros como el retardo y la tasa de transmisión de la red, para lo cual se utilizó las herramientas ping, iperf y Wireshark para el análisis, se utilizó la topología que se indica en la Figura 33.

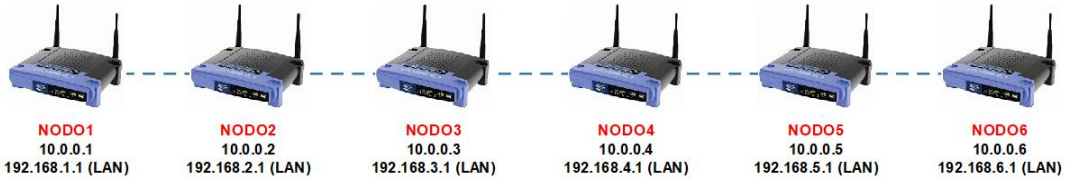

Figura 33. Escenario utilizado para el análisis de la red.
Fuente: Elaboración propia

Una vez configurados los equipos se procedió a comprobar la topología propuesta a través del comando traceroute como se observa en la Figura 34 y a través de las tablas y gráficos generados en la interfaz web del NODO4 como se observa en la Figura 35 y Figura 36 respetivamente.

```
root@NODO1:~# traceroute 10.0.0.6
traceroute to 10.0.0.6 (10.0.0.6), 30 hops max, 38 porte packets
 1  10.0.0.2 (10.0.0.2)  2.429 ms  3.521 ms  1.976 ms
 2  10.0.0.3 (10.0.0.3)  3.291 ms  4.231 ms  12.780 ms
 3  10.0.0.4 (10.0.0.4)  7.959 ms  9.401 ms  12.628 ms
 4  10.0.0.5 (10.0.0.5)  24.745 ms  6.232 ms  5.617 ms
 5  10.0.0.6 (10.0.0.6)  25.714 ms  7.008 ms  10.793 ms
```

Figura 34. Resultado del comando Tracert/Topología para al análisis de la red
Fuente: Elaboración propia

OLSR node	Last hop
10.0.0.1	10.0.0.2
10.0.0.2	10.0.0.1
10.0.0.2	10.0.0.3
10.0.0.3	10.0.0.2
10.0.0.3	10.0.0.4
10.0.0.4	10.0.0.5
10.0.0.4	10.0.0.3
10.0.0.5	10.0.0.6
10.0.0.5	10.0.0.4
10.0.0.6	10.0.0.5

Figura 35. Topología de la red/ Análisis de la red
Fuente: Elaboración propia

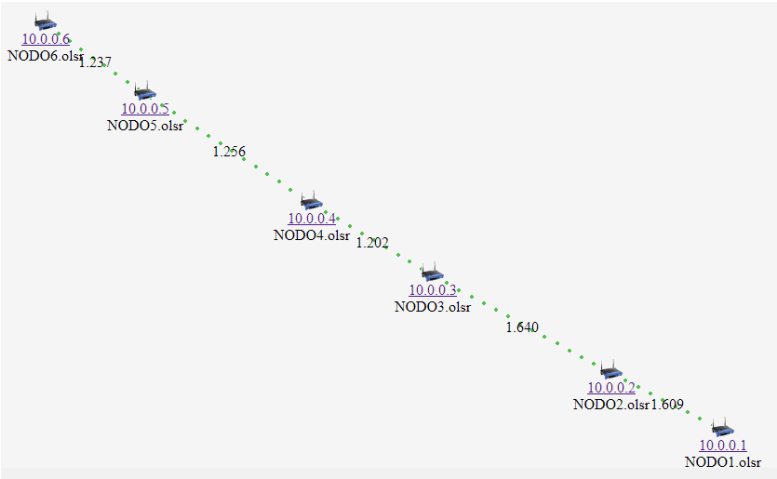

Figura 36. Gráfica obtenida/ Topología utilizada para el análisis de la red
Fuente: Elaboracion propia

A continuación se detalla las pruebas realizadas:

- ✓ **PING:** Para esta prueba se realizarán 3 muestras de 100 solicitudes cada una, como recomienda en la RFC 889[5]. Las solicitudes se enviarán desde el NODO1; hasta cada uno de los routers que conforman la topología de la Figura 33. El comando utilizado se muestra en la Figura 37, se utilizará 88 bytes.

```
ping 10.0.0.2 -w 100 -s 88 -q
```

Figura 37. Comando PING utilizado en las pruebas
Fuente: Elaboración propia

- ✓ **IPERF:** Para esta prueba iperf se instaló directamente en los routers mediante la versión compilada que ofrece OpenWRT y se configuro el NODO1 como servidor y los nodos restantes se configurarán como clientes como se indica en la RFC 2544[6] y como se indica en la página web de Network Bogotá (NetworkBogota, 2010). Se procedió a la creación de flujos TCP y UDP, de esta manera se obtuvieron reportes sobre la tasa de transmisión para TCP y UDP. En la Figura 38 se indica los comandos para la instalación de iperf en los routers y la configuración del cliente y del servidor.

COMANDOS	**TCP**	**Cliente**	iperf –c 10.0.0.1 –t 100s
		Servidor	iperf -s
	UDP	**Cliente**	iperf –c 10.0.0.1 –u –b 2m –t 100s
		Servidor	iperf –s -u

Figura 38. Comandos para generación de tráfico en iperf.
Fuente: Elaboración propia

- ✓ **PRUEBAS CON APLICACIONES REALES.** Además de los experimentos antes mencionados se realizaron dos pruebas con aplicaciones reales (VoIP y HTTP), para lo cual se colocaron tres routers distribuidos en cada uno de las plantas del edificio del Laboratorios del Área de la Energía, las Industrias y los Recursos Naturales No Renovables como se indica en la Figura 39

[5] RFC 889 "Internet Delay Experiments" plantea que se realicen pruebas usando los mensajes ICMP Echo/Reply.
[6] RFC 2544 "Benchmarking Methodology for Network Interconnect Devices" propone la topología a utilizarse para la realizacion de pruebas, es ampliamente utilizada en performance de redes WLAN.

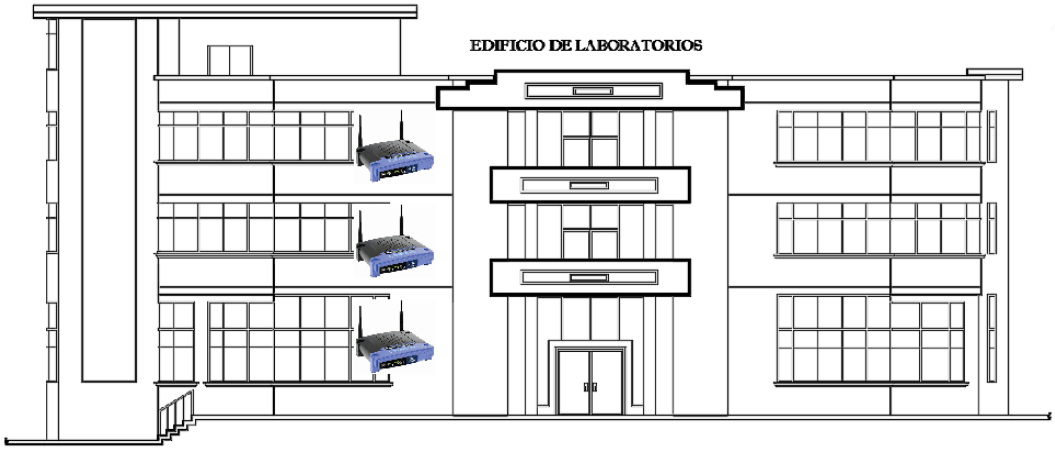

Figura 39. Ubicación de los routers en cada piso del edificio de Laboratorios.
Fuente: Elaboración propia

- **Voz sobre IP:** Para esta prueba se configuro un servidor de telefonía IP Elastix, corriendo en PC1 a través de máquina virtual, PC1 estará conectada al NODO1 y proveerá de servicio a los nodos restantes. Se configuraron 15 extensiones, desde la 101 a la 115 para poder usarlas en los dispositivos que tendrán instalado el softphone PortGo, como se indica en el Anexo 8. Se realizó una llamada del dispositivo móvil a PC1. Para la evaluación de la comunicación se procedió a capturar tráfico a través de Wireshark para después analizarlo. La disposición de los equipos se muestra en la Figura 40.

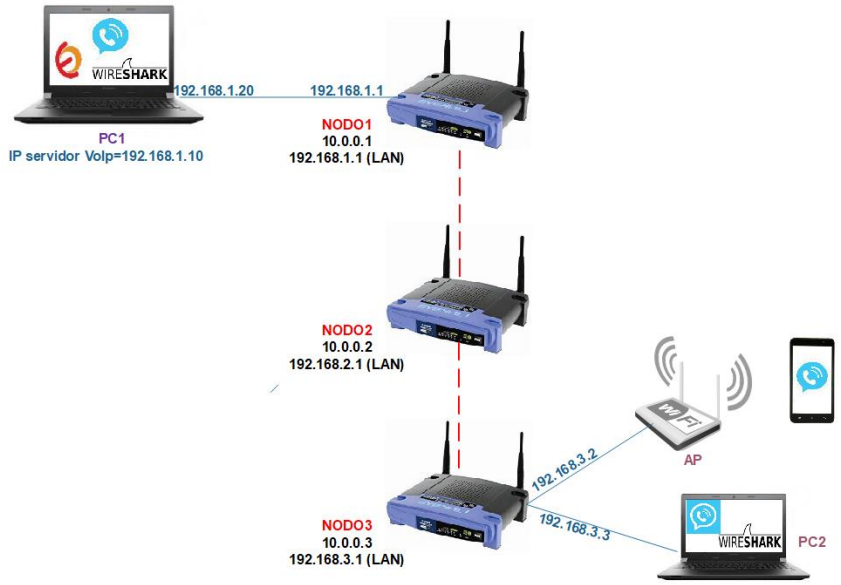

Figura 40. Disposición de los dispositivos para la realización de la prueba VoIP
Fuente: Elaboración propia

- **HTTP (Protocolo de Transferencia de Hiper Texto, del inglés HyperText Transfer Protocol):** Para esta prueba se configuró un servidor WAMPserver, corriendo en PC1 la misma estará conectada al NODO1 y proveerá de servicio a los nodos restantes. Se asignó la dirección IP 192.168.1.5 para el servidor, la misma que se utilizará para acceder a través de un navegador web desde los otros nodos. Se procederá a descargar un archivo cuyo tamaño es de 59MB, y se analizará a través de Wireshark. La disposición de los equipos se muestra en la Figura 41

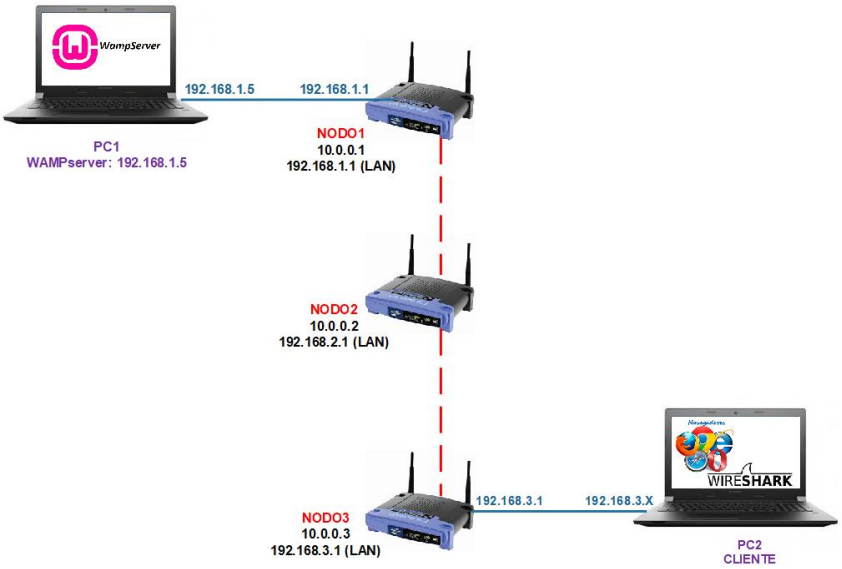

Figura 41. Disposición de los dispositivos para la realización de la prueba HTTP
Fuente: Elaboración propia

Para poder acceder a los archivos que se alojan en el servidor se debe agregar los dominios creados a la computadora que quiera acceder remotamente por lo que se debe seguir los siguientes pasos:

- Ingresar a la siguiente dirección de Windows C:\Windows\System32\drivers\etc
- Ingresar al archivo *host* y colocamos las siguientes líneas

```
192.168.1.5 www.tesisproyecto.cof
```

5.4 Parámetros a evaluar

Mediante las pruebas establecidas se procedió a medir dos parámetros:

- **Retardo**: Cuando un paquete viaja de un nodo origen a un nodo destino atraviesa varios nodos, por lo que el paquete sufre varios retardos a lo largo de una ruta. El

retardo total entre el origen y el destino o retardo terminal a terminal es la suma de estos retardos, es decir el tiempo que introduce cada router al procesar, transmitir y propagar los bits:

$$d_{terminal-terminal} = N\left(d_{proc} + d_{trans} + d_{prop}\right) \qquad \textbf{Ecuación 2}$$

Donde:

- *N:* Número de routers
- d_{proc}=*Retardo de procesamiento:* Se define con el tiempo necesario para examinar la cabecera de paquete y determinar a donde hay que enviarlo.
-d_{trans}=*Retardo de transmisión:* O también llamado retardo de almacenamiento y reenvío, es el tiempo necesario para transmitir todos los bits de un determinado paquete a través del enlace de comunicación. Este retardo será igual a L/R donde L es la longitud del paquete y R es la velocidad de transmisión del enlace.
- d_{prop}=*Retardo de propagación:* Una vez que un bit ha entrado en el enlace, tiene que propagarse el origen al destino. El bit se propaga a la velocidad de propagación del enlace, y esta depende del medio físico del enlace. (Kurose & Roos, 2010)

Además de los retardos de procesamiento, de transmisión y de propagación, en los sistemas terminales pueden existir retardos adicionales significativos, como el retardo de cola, retardo de empaquetamiento y otros. Para las aplicaciones como VoIP, los retardos terminal a terminal inferiores a 150 milisegundos no son perceptibles, los retardos entre 150 y 400 milisegundos son aceptables, y los retardos mayores que 400 milisegundos pueden afectar seriamente a la interactividad de la conversación. Otro componente crucial en VoIP son los retardos aleatorios de puesta en cola dentro de los routers ya que el tiempo el tiempo que transcurre desde que se origina hasta que se recibe un paquete puede variar, y a esta fluctuación se la conoce como jitter, para VoIP el jitter no debe superar los 20ms (Kurose & Roos, 2010)

El RTT (del inglés, Round Trip Delay Time) o tiempo de ida y vuelta descrito en la RFC 2681 incluye los retardos de propagación, los retardos de cola y los retardos de procesamiento; este se define como el tiempo medido en segundos, que transcurre desde que se envía el primer porte hasta que regresa al emisor pasando por el receptor, se lo puede realizar mediante la herramienta ping. También existe otra herramienta que permite obtener la magnitud del retardo terminal a terminal de una red de computadoras,

Traceroute, que está descrita en la RFC 1393; se trata de un programa simple que se puede ejecutar en cualquier host de Internet, en el cual, el host de origen envía varios paquetes especiales al destino y devuelve al origen un mensaje corto que contiene el nombre y la dirección del router. (Kurose & Roos, 2010) Se han tomado medidas de este parámetro en función al número de saltos (5 saltos máximo) mediante la herramienta ping y también se utilizó la herramienta traceroute. Además se procedió a realizar una llamada VoIP (con dos saltos como máximo) en la cual se obtuvo a través de Wireshark el retardo y el jitter.

-Tasa de transmisión: se define como una medida de cantidad de datos enviados entre un origen y un destino, su unidad de medida son los bits por segundo (bps); para obtener mediciones de esta métrica se utilizó el programa iperf, mediante la versión ya compilada que tiene OpenWRT, este se instaló en cada uno de los routers, aunque también se puede instalar en computadores. A demás se procedió a obtener la tasa de transferencia instantánea que es la velocidad (en bits/segundo) a la que el host destino recibe el archivo en cualquier instante de tiempo y la tasa media de transferencia es igual al cociente entre la longitud del archivo (F) y el tiempo de transmisión o transferencia (T) del mismo. (Kurose & Roos, 2010)

$$Tasa\ media\ de\ transferencia = F/T\ (bits/segundos) \qquad \boldsymbol{Ecuacion\ 3}$$

Para obtener estas métricas se procedió a descargar un archivo (con 2 saltos como máximo) a través de WAMPserver y analizarlo mediante Wireshark.

6. RESULTADOS

6.1 Simulación

La simulación permite obtener archivos .pcap, en la Figura 42 se muestra los resultados obtenidos. El ejemplo que fue modificado está destinado a la generación de mensajes Hello y HNA. Como se puede observar en "Source" se indica todos los nodos que han generado tráfico de control, seguidamente tenemos la dirección broadcast y el la versión del protocolo OLSR y su longitud

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.4	10.0.0.255	OLSR v1	108	OLSR (IPv4) Packet, Length: 20 Bytes
2	0.075647	10.0.0.1	10.0.0.255	OLSR v1	108	OLSR (IPv4) Packet, Length: 20 Bytes
3	0.090785	10.0.0.3	10.0.0.255	OLSR v1	108	OLSR (IPv4) Packet, Length: 20 Bytes
4	0.193120	10.0.0.5	10.0.0.255	OLSR v1	108	OLSR (IPv4) Packet, Length: 20 Bytes
5	0.227498	10.0.0.6	10.0.0.255	OLSR v1	128	OLSR (IPv4) Packet, Length: 40 Bytes
6	0.361820	10.0.0.2	10.0.0.255	OLSR v1	106	OLSR (IPv4) Packet, Length: 20 Bytes
7	2.083677	10.0.0.6	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
8	2.256553	10.0.0.3	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
9	2.294533	10.0.0.2	10.0.0.255	OLSR v1	146	OLSR (IPv4) Packet, Length: 60 Bytes
10	2.340849	10.0.0.1	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
11	2.408416	10.0.0.5	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
12	2.417101	10.0.0.4	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
13	4.010919	10.0.0.6	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
14	4.108854	10.0.0.5	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
15	4.159415	10.0.0.4	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
16	4.164363	10.0.0.2	10.0.0.255	OLSR v1	146	OLSR (IPv4) Packet, Length: 60 Bytes
17	4.227620	10.0.0.1	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
18	4.254147	10.0.0.3	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
19	5.409633	10.0.0.6	10.0.0.255	OLSR v1	112	OLSR (IPv4) Packet, Length: 24 Bytes
20	6.026646	10.0.0.6	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
21	6.033153	10.0.0.5	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
22	6.327766	10.0.0.2	10.0.0.255	OLSR v1	146	OLSR (IPv4) Packet, Length: 60 Bytes
23	6.335072	10.0.0.1	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
24	6.366567	10.0.0.3	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
25	6.376805	10.0.0.4	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
26	8.048648	10.0.0.6	10.0.0.255	OLSR v1	148	OLSR (IPv4) Packet, Length: 60 Bytes
27	8.110298	10.0.0.2	10.0.0.255	OLSR v1	146	OLSR (IPv4) Packet, Length: 60 Bytes

Figura 42. Archivo .pcap obtenido de la simulación
Fuente: Elaboración propia

Abrimos el primer paquete y detalla la siguiente información acerca del protocolo. Se utiliza el protocolo UDP y el puerto 698 para el transporte de los mensajes, en el frame 1 indica que se ha enviado un mensaje HELLO, con el tipo de mensaje que en este caso corresponde el número 1, la dirección del originador que corresponde al nodo con la dirección 10.1.1.4, el tiempo de valides del mensaje, el HopCount indica cero que quiere decir que el archivo se generó desde el NODO4.

```
▸ Frame 1: 108 bytes on wire (864 bits), 108 bytes captured (864 bits)
▸ Radiotap Header v0, Length 24
▸ 802.11 radio information
▸ IEEE 802.11 Data, Flags: o........
▸ Logical-Link Control
▸ Internet Protocol Version 4, Src: 10.0.0.4, Dst: 10.0.0.255
▸ User Datagram Protocol, Src Port: 698, Dst Port: 698
▾ Optimized Link State Routing Protocol
    Packet Length: 20
    Packet Sequence Number: 0
  ▾ Message: HELLO (1)
      Message Type: HELLO (1)
      Validity Time: 6,000 (in seconds)
      Message: 16
      Originator Address: 10.0.0.4
      TTL: 1
      Hop Count: 0
      Message Sequence Number: 0
      Hello Emission Interval: 2,000 (in seconds)
      Willingness to forward messages: Unknown (3)
```

Figura 43. Detalles del frame 1 de la captura de paquetes
Fuente: Elaboración propia

En el frame 5 se puede observar que se ha enviado un mensaje HELLO y un mensaje HNA, en los campos del mensaje HNA indica el tipo de mensaje, en este caso es un mensaje 4, su respectivo tiempo de valides, la dirección del nodo origen que el NODO6 que está anunciando su red, como indica la Figura 44

```
▸ Frame 5: 128 bytes on wire (1024 bits), 128 bytes captured (1024 bits)
▸ Radiotap Header v0, Length 24
▸ 802.11 radio information
▸ IEEE 802.11 Data, Flags: o.......
▸ Logical-Link Control
▸ Internet Protocol Version 4, Src: 10.0.0.6, Dst: 10.0.0.255
▸ User Datagram Protocol, Src Port: 698, Dst Port: 698
▾ Optimized Link State Routing Protocol
    Packet Length: 40
    Packet Sequence Number: 0
  ▾ Message: HELLO (1)
      Message Type: HELLO (1)
      Validity Time: 6,000 (in seconds)
      Message: 16
      Originator Address: 10.0.0.6
      TTL: 1
      Hop Count: 0
      Message Sequence Number: 0
      Hello Emission Interval: 2,000 (in seconds)
      Willingness to forward messages: Unknown (3)
  ▾ Message: HNA (4)
      Message Type: HNA (4)
      Validity Time: 15,000 (in seconds)
      Message: 20
      Originator Address: 10.0.0.6
      TTL: 255
      Hop Count: 0
      Message Sequence Number: 1
      Network Address: 192.168.6.0
      Netmask: 255.255.255.0
```

Figura 44. Detalles del frame 5 de la captura de paquetes

Fuente: Elaboración propia

En el paquete 7 podemos observar que el NODO6 ha establecido un enlace asimétrico con los demás nodos es decir que la comunicación se realiza en un solo sentido, mientas que en el paquete 13 ya se establece una comunicación simétrica.

```
: · Packet 7 · olsr-hna-0-0
▸ Internet Protocol Version 4, Src: 10.0.0.6, Dst: 10.0.0.255
▸ User Datagram Protocol, Src Port: 698, Dst Port: 698
▾ Optimized Link State Routing Protocol
    Packet Length: 60
    Packet Sequence Number: 1
  ▾ Message: HELLO (1)
      Message Type: HELLO (1)
      Validity Time: 6,000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.6
      TTL: 1
      Hop Count: 0
      Message Sequence Number: 2
      Hello Emission Interval: 2,000 (in seconds)
      Willingness to forward messages: Unknown (3)
    ▾ Link Type: Asymmetric Link (1)
        Link Message Size: 8
        Neighbor Address: 10.0.0.4
    ▾ Link Type: Asymmetric Link (1)
        Link Message Size: 8
        Neighbor Address: 10.0.0.1
    ▾ Link Type: Asymmetric Link (1)
        Link Message Size: 8
        Neighbor Address: 10.0.0.3
    ▾ Link Type: Asymmetric Link (1)
        Link Message Size: 8
        Neighbor Address: 10.0.0.5
    ▾ Link Type: Asymmetric Link (1)
        Link Message Size: 8
        Neighbor Address: 10.0.0.2
```

Figura 45. Detalles del frame 7 de la captura de paquetes

Fuente: Elaboración propia

```
· Packet 13 · olsr-hna-0-0

▸ Internet Protocol Version 4, Src: 10.0.0.6, Dst: 10.0.0.255
▸ User Datagram Protocol, Src Port: 698, Dst Port: 698
▾ Optimized Link State Routing Protocol
    Packet Length: 60
    Packet Sequence Number: 2
  ▾ Message: HELLO (1)
      Message Type: HELLO (1)
      Validity Time: 6,000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.6
      TTL: 1
      Hop Count: 0
      Message Sequence Number: 3
      Hello Emission Interval: 2,000 (in seconds)
      Willingness to forward messages: Unknown (3)
    ▾ Link Type: Symmetric Link (6)
        Link Message Size: 8
        Neighbor Address: 10.0.0.4
    ▾ Link Type: Symmetric Link (6)
        Link Message Size: 8
        Neighbor Address: 10.0.0.1
    ▾ Link Type: Symmetric Link (6)
        Link Message Size: 8
        Neighbor Address: 10.0.0.3
    ▾ Link Type: Symmetric Link (6)
        Link Message Size: 8
        Neighbor Address: 10.0.0.5
    ▾ Link Type: Symmetric Link (6)
        Link Message Size: 8
        Neighbor Address: 10.0.0.2
```

Figura 46. Detalles del frame 13 de la captura de paquetes

Fuente: Elaboración propia

6.2 Configuración de la Red

Una vez realizada la configuración descrita en la sección anterior, la red se encuentra formada. OpenWRT permite visualizar a través de tablas, los vecinos, las redes HNA, y si un nodo tiene múltiples interfaces. Se tomó como referencia el NODO1, para poder analizar las tablas, en la Figura 47, "*OLSR Connections*" indica los vecinos que tiene el nodo, la primera columna indica la dirección de la interfaz inalámbrica de los nodos vecinos, la segunda columna la interfaz inalámbrica del nodo; (indica la interfaz inalámbrica, ya que sobre ella corre el protocolo), la tercera columna indica el valor LQ y NLQ mientras más se acerque a 1 indica una mejor calidad de enlace del nodo y del nodo vecino respectivamente, y en la última columna el valor ETX, indicándonos que el NODO4 presenta más perdida de paquetes mientras que el NODO6 presenta un valor ETX que se acerca a 1 indicando una mejor calidad del enlace.

Neighbour IP	Local interface IP	LQ	NLQ	ETX
10.0.0.2	10.0.0.1	0.952	0.748	1.400
10.0.0.6	10.0.0.1	0.753	0.921	1.440
10.0.0.5	10.0.0.1	0.717	0.917	1.518
10.0.0.3	10.0.0.1	0.646	0.901	1.713
10.0.0.4	10.0.0.1	0.521	1.000	1.917

Figura 47. Tabla de vecinos, interfaz gráfica OpenWRT

Fuente: Elaboración propia

La Figura 48 la información "*Know OLSR routers*" indica todas las redes anunciadas, ya sea porque corre sobre ellas el protocolo OLSR (interfaz inalámbrica) o porque fueron anunciadas a través de mensajes HNA (interfaz LAN). También indica la interfaz por la que corre OLSR y a través de que nodo fueron anunciadas.

Announced network	OLSR gateway	Interface
192.168.6.0/24	10.0.0.6	wl0
10.0.0.6/32	10.0.0.6	wl0
10.0.0.3/32	10.0.0.3	wl0
192.168.3.0/24	10.0.0.3	wl0
10.0.0.2/32	10.0.0.2	wl0
192.168.2.0/24	10.0.0.2	wl0
10.0.0.5/32	10.0.0.5	wl0
192.168.5.0/24	10.0.0.5	wl0
192.168.4.0/24	10.0.0.4	wl0
10.0.0.4/32	10.0.0.4	wl0

Figura 48. Direcciones de las redes LAN y WLAN de cada nodo.
Fuente: Elaboración propia

La Figura 49 en la pestaña "*Active OLSR nodes*" indica la topología de la red mesh, dando a conocer los enlaces que tiene cada nodo. La red fue implementada de tal manera que todos los nodos estén interconectados entre sí.

OLSR node	Last hop	LQ	ILQ	ETX
10.0.0.1	10.0.0.5	0.839	0.658	1.809
10.0.0.1	10.0.0.2	0.917	0.615	1.770
10.0.0.1	10.0.0.3	0.976	0.721	1.419
10.0.0.1	10.0.0.6	0.937	0.631	1.689
10.0.0.1	10.0.0.4	0.862	0.701	1.650
10.0.0.2	10.0.0.5	0.740	0.765	1.764
10.0.0.2	10.0.0.4	0.913	0.835	1.310
10.0.0.2	10.0.0.3	0.937	0.839	1.271
10.0.0.2	10.0.0.1	0.557	0.937	1.915
10.0.0.2	10.0.0.6	0.776	0.858	1.499
10.0.0.3	10.0.0.5	0.693	0.917	1.569
10.0.0.3	10.0.0.2	0.819	0.937	1.302
10.0.0.3	10.0.0.1	0.709	0.956	1.472
10.0.0.3	10.0.0.4	0.713	0.968	1.446
10.0.0.3	10.0.0.6	0.753	0.882	1.505
10.0.0.4	10.0.0.1	0.709	0.862	1.633
10.0.0.4	10.0.0.6	0.780	0.697	1.835
10.0.0.4	10.0.0.3	0.886	0.713	1.580
10.0.0.4	10.0.0.2	0.804	0.913	1.361
10.0.0.4	10.0.0.5	0.701	0.878	1.621
10.0.0.5	10.0.0.6	0.894	0.765	1.462
10.0.0.5	10.0.0.4	0.862	0.701	1.650
10.0.0.5	10.0.0.1	0.658	0.839	1.809
10.0.0.5	10.0.0.2	0.796	0.725	1.731
10.0.0.5	10.0.0.3	0.937	0.701	1.520
10.0.0.6	10.0.0.5	0.769	0.917	1.417
10.0.0.6	10.0.0.2	0.882	0.776	1.459
10.0.0.6	10.0.0.1	0.623	0.937	1.711
10.0.0.6	10.0.0.4	0.697	0.776	1.845
10.0.0.6	10.0.0.3	0.882	0.753	1.505

Figura 49. Topología de la red mesh, interfaz gráfica de OpenWRT
Fuente: Elaboración propia

La Figura 50 en la parte "*Active host net announcements*" indica las redes o hosts anunciadas a través de los mensajes HNA en cada nodo, como se puede observar se encuentran las 6 redes que fueron configuradas individualmente. La primera columna indica la red HNA anunciada, la segunda columna indica la dirección del nodo del cual fueron anunciadas.

Announced network	OLSR gateway
192.168.1.0/24	10.0.0.1
192.168.2.0/24	10.0.0.2
192.168.3.0/24	10.0.0.3
192.168.4.0/24	10.0.0.4
192.168.5.0/24	10.0.0.5
192.168.6.0/24	10.0.0.6

Figura 50. Redes anunciadas a través de los mensajes HNA

Fuente: Elaboracion propia

Como se mencionó anteriormente en versiones superiores a la utilizada en este proyecto es posible instalar un pluging denominado olrs-mod-viz, el NODO4 fue instalado el release Backfire 10.03.1, lo que nos permite visualizar la red como indica la Figura 51, mientras que en los otros dispositivos es posible generar una gráfica mediante el pluging olsr-dot-draw como se observa en la Figura 52.

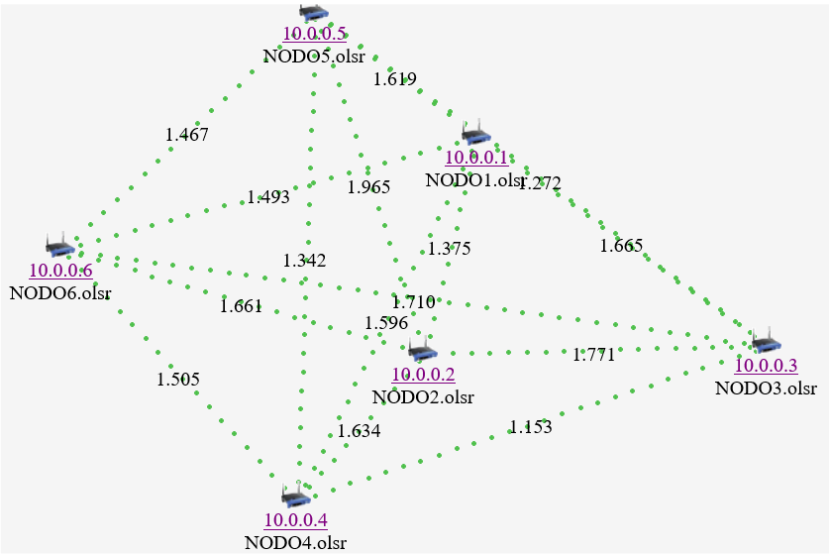

Figura 51. Visualización de la red, a través de la interfaz gráfica de OpenWRT

Fuente: Elaboración propia

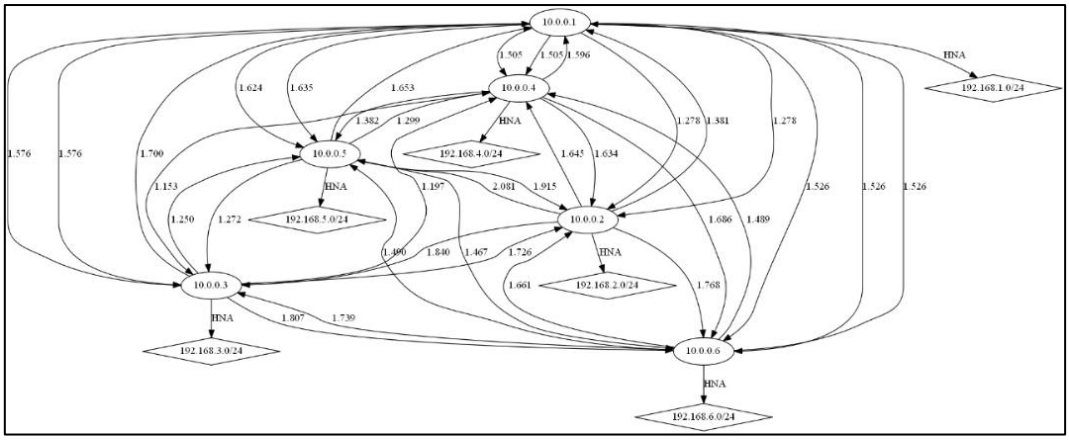

Figura 52. Visualizacion de la red mediante el plugin dot-draw

Fuente: Elaboracion propia

6.3 ANÁLISIS DEL PROTOCOLO.

6.3.1 Primer Escenario: Funcionamiento básico del protocolo.

Se realizó la captura de paquetes OLSR a través de Wireshark, para averiguar el comportamiento del protocolo sobre la red y como fueron enviados los diferentes mensajes de control. Como se puede observar en Figura 53 en la columna correspondiente a "Source" nos indica la dirección IP de los nodos que conforman la red, además de indicarnos el protocolo en este caso el OLSR v1 o conocido solamente como OLSR y la longitud de paquete, así también como el destino de los paquetes, recordemos que los mensajes de control se enviaban en broadcast por lo que el destino será 10.255.255.255.255, además se puede observar la dirección correspondiente a los seis nodos que conforman la red mesh.

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

olsr

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.5	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
2	0.203465	10.0.0.4	10.255.255.255	OLSR v1	162	OLSR (IPv4) Packet, Length: 120 Bytes
3	0.845248	10.0.0.6	10.255.255.255	OLSR v1	94	OLSR (IPv4) Packet, Length: 52 Bytes
4	1.032372	10.0.0.2	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
5	1.401682	10.0.0.1	10.255.255.255	OLSR v1	162	OLSR (IPv4) Packet, Length: 120 Bytes
6	1.558365	10.0.0.5	10.255.255.255	OLSR v1	166	OLSR (IPv4) Packet, Length: 124 Bytes
7	1.568087	10.0.0.1	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
9	1.837985	10.0.0.5	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
10	1.976428	10.0.0.4	10.255.255.255	OLSR v1	106	OLSR (IPv4) Packet, Length: 64 Bytes
12	2.380519	10.0.0.3	10.255.255.255	OLSR v1	162	OLSR (IPv4) Packet, Length: 120 Bytes
13	2.669678	10.0.0.2	10.255.255.255	OLSR v1	158	OLSR (IPv4) Packet, Length: 116 Bytes
14	2.805543	10.0.0.6	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
16	3.334273	10.0.0.1	10.255.255.255	OLSR v1	158	OLSR (IPv4) Packet, Length: 116 Bytes
17	3.483698	10.0.0.2	10.255.255.255	OLSR v1	110	OLSR (IPv4) Packet, Length: 68 Bytes
18	3.653358	10.0.0.4	10.255.255.255	OLSR v1	130	OLSR (IPv4) Packet, Length: 88 Bytes
19	3.758061	10.0.0.5	10.255.255.255	OLSR v1	122	OLSR (IPv4) Packet, Length: 80 Bytes
20	3.855843	10.0.0.6	10.255.255.255	OLSR v1	106	OLSR (IPv4) Packet, Length: 64 Bytes
21	3.983501	10.0.0.3	10.255.255.255	OLSR v1	234	OLSR (IPv4) Packet, Length: 192 Bytes
25	4.565646	10.0.0.6	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
26	4.668731	10.0.0.2	10.255.255.255	OLSR v1	102	OLSR (IPv4) Packet, Length: 60 Bytes
28	5.020940	10.0.0.1	10.255.255.255	OLSR v1	122	OLSR (IPv4) Packet, Length: 80 Bytes
30	5.501207	10.0.0.3	10.255.255.255	OLSR v1	214	OLSR (IPv4) Packet, Length: 172 Bytes
31	5.527316	10.0.0.4	10.255.255.255	OLSR v1	166	OLSR (IPv4) Packet, Length: 124 Bytes

Figura 53. Captura de tráfico en la red mallada a través de Wireshark.
Fuente: Elaboración propia

Tomamos algunos de los paquetes de la captura para analizarlos, al abrir el paquete 1 se puede observar el puerto utilizado por OLSR, que es el UDP 698 , en la parte del protocolo indica que contiene un solo mensaje. El NODO5 ha enviado un mensaje TC en el cual incluye información de las direcciones IPs de sus nodos vecinos además el número de secuencia que va aumentando por cada paquete, entonces este mensaje es el número 32560 que ha enviado el NODO5 desde que empezó a funcionar, el tiempo que

debe ser considerada esta informacion validad es hasta que se cumpla un tiempo de 256 segundos según el campo “Validity Time”, a un lado de la dirección de los nodos vecinos también se encuentra el valor LQ y NLQ (LQ/NLQ) , el hop count igual a 0 indica que el mensaje ha sido generado por el nodo como se muestra en la Figura 54

```
> Frame 1: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface 0
> Ethernet II, Src: Cisco-Li_0d:89:15 (00:1c:10:0d:89:15), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.0.0.5, Dst: 10.255.255.255
> User Datagram Protocol, Src Port: 698, Dst Port: 698
v Optimized Link State Routing Protocol
    Packet Length: 60
    Packet Sequence Number: 10546
  v Message: TC (LQ, olsr.org) (202)
      Message Type: TC (LQ, olsr.org) (202)
      Validity Time: 256.000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.5
      TTL: 2
      Hop Count: 0
      Message Sequence Number: 32560
      Advertised Neighbor Sequence Number (ANSN): 344
    > Neighbor Address: 10.0.0.1 (198/86)
    > Neighbor Address: 10.0.0.2 (188/93)
    > Neighbor Address: 10.0.0.3 (243/166)
    > Neighbor Address: 10.0.0.4 (176/88)
    > Neighbor Address: 10.0.0.6 (60/72)
```

Figura 54. Frame 1 de la captura de tráfico de control de la red implementada
Fuente: Elaboración propia

En el paquete 6 se puede visualizar que contiene dos tipos de mensajes, un mensaje HELLO y un TC. El NODO 5 ha recibido un mensaje TC del NODO4 indicando las direcciones IP de sus vecinos con sus respectivos valores de LQ y NLQ, así mismo el intervalo de validez del mensaje y a su vez que genera un mensaje HELLO indicando que el enlace es simétrico con los nodos 6, 1, 2 y 4 es decir que es un enlace bidireccional en la cual los nodos antes mencionados se oyen con el NODO5 adicionalmente indica el valor de LQ y NLQ de cada enlace , y que el NODO3 lo ha escogido como un MPR e indica los valores LQ y NLQ, el número de secuencia y el tiempo de validez del mensaje HELLO como se puede apreciar en la Figura 55.

En la Figura 56 se detalla el paquete 19, en donde el NODO5 recibe un mensaje HNA del NODO4 indicando la red anunciada a través de este mensaje, el número de mensaje (4), y este nodo ha generado un mensaje TC en el que incluye todas las direcciones IP de sus vecinos y sus valores LQ y NLQ que indican la calidad de enlace entre el nodo y el vecino.

```
> Frame 6: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits) on interface 0
> Ethernet II, Src: Cisco-Li_0d:89:15 (00:1c:10:0d:89:15), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.0.0.5, Dst: 10.255.255.255
> User Datagram Protocol, Src Port: 698, Dst Port: 698
v Optimized Link State Routing Protocol
    Packet Length: 124
    Packet Sequence Number: 10547
  v Message: TC (LQ, olsr.org) (202)
      Message Type: TC (LQ, olsr.org) (202)
      Validity Time: 288.000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.4
      TTL: 1
      Hop Count: 1
      Message Sequence Number: 53334
      Advertised Neighbor Sequence Number (ANSN): 3208
    > Neighbor Address: 10.0.0.1 (223/221)
    > Neighbor Address: 10.0.0.2 (203/221)
    > Neighbor Address: 10.0.0.3 (116/203)
    > Neighbor Address: 10.0.0.5 (95/165)
    > Neighbor Address: 10.0.0.6 (42/88)
  v Message: HELLO (LQ, olsr.org) (201)
      Message Type: HELLO (LQ, olsr.org) (201)
      Validity Time: 40.000 (in seconds)
      Message: 64
      Originator Address: 10.0.0.5
      TTL: 1
      Hop Count: 0
      Message Sequence Number: 32561
      Hello Emission Interval: 5.000 (in seconds)
      Willingness to forward messages: Unknown (3)
    v Link Type: Symmetric Link (6)
        Link Message Size: 36
      > Neighbor Address: 10.0.0.6 (60/72)
      > Neighbor Address: 10.0.0.1 (195/98)
      > Neighbor Address: 10.0.0.2 (186/93)
      > Neighbor Address: 10.0.0.4 (178/95)
    v Link Type: MPR Link (10)
        Link Message Size: 12
      > Neighbor Address: 10.0.0.3 (243/166)
```

Figura 55. Frame 6 de la captura de paquetes de la red

Fuente: Elaboración propia

```
> Frame 19: 122 bytes on wire (976 bits), 122 bytes captured (976 bits) on interface 0
> Ethernet II, Src: Cisco-Li_0d:89:15 (00:1c:10:0d:89:15), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.0.0.5, Dst: 10.255.255.255
> User Datagram Protocol, Src Port: 698, Dst Port: 698
v Optimized Link State Routing Protocol
    Packet Length: 80
    Packet Sequence Number: 10549
  v Message: HNA (4)
      Message Type: HNA (4)
      Validity Time: 288.000 (in seconds)
      Message: 20
      Originator Address: 10.0.0.4
      TTL: 254
      Hop Count: 1
      Message Sequence Number: 53337
      Network Address: 192.168.4.0
      Netmask: 255.255.255.0
  v Message: TC (LQ, olsr.org) (202)
      Message Type: TC (LQ, olsr.org) (202)
      Validity Time: 256.000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.5
      TTL: 2
      Hop Count: 0
      Message Sequence Number: 32563
      Advertised Neighbor Sequence Number (ANSN): 344
    > Neighbor Address: 10.0.0.1 (209/98)
    > Neighbor Address: 10.0.0.2 (184/110)
    > Neighbor Address: 10.0.0.3 (243/166)
    > Neighbor Address: 10.0.0.4 (187/94)
    > Neighbor Address: 10.0.0.6 (69/72)
```

Figura 56. Frame 19 de la captura de tráfico de la red

Fuente: Elaboración propia

En cambio en el paquete 50 mostrado en la Figura 57 se puede observar q ha recibido un mensaje HNA del NODO1 anunciando la red LAN y un mensaje TC del NODO4

anunciando todos sus vecinos, además se puede observar que el nodo ha generado un mensaje TC indicando sus nodos vecino.

```
> Frame 50: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface 0
> Ethernet II, Src: Cisco-Li_0d:89:15 (00:1c:10:0d:89:15), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.0.0.5, Dst: 10.255.255.255
> User Datagram Protocol, Src Port: 698, Dst Port: 698
v Optimized Link State Routing Protocol
    Packet Length: 136
    Packet Sequence Number: 10552
  v Message: TC (LQ, olsr.org) (202)
      Message Type: TC (LQ, olsr.org) (202)
      Validity Time: 288.000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.4
      TTL: 1
      Hop Count: 1
      Message Sequence Number: 53343
      Advertised Neighbor Sequence Number (ANSN): 3216
    > Neighbor Address: 10.0.0.1 (237/215)
    > Neighbor Address: 10.0.0.2 (218/220)
    > Neighbor Address: 10.0.0.3 (127/203)
    > Neighbor Address: 10.0.0.5 (109/165)
    > Neighbor Address: 10.0.0.6 (61/113)
  v Message: HNA (4)
      Message Type: HNA (4)
      Validity Time: 108.000 (in seconds)
      Message: 20
      Originator Address: 10.0.0.1
      TTL: 253
      Hop Count: 2
      Message Sequence Number: 55896
      Network Address: 192.168.1.0
      Netmask: 255.255.255.0
  v Message: TC (LQ, olsr.org) (202)
      Message Type: TC (LQ, olsr.org) (202)
      Validity Time: 256.000 (in seconds)
      Message: 56
      Originator Address: 10.0.0.5
      TTL: 8
      Hop Count: 0
      Message Sequence Number: 32568
      Advertised Neighbor Sequence Number (ANSN): 344
    > Neighbor Address: 10.0.0.1 (205/100)
    > Neighbor Address: 10.0.0.2 (178/113)
    > Neighbor Address: 10.0.0.3 (247/169)
    > Neighbor Address: 10.0.0.4 (187/109)
    > Neighbor Address: 10.0.0.6 (94/127)
```

Figura 57. Frame 50 de la captura de paquetes en la red

Fuente: Elaboracion propia

6.3.2 Segundo Escenario: Tiempo de convergencia de un nuevo nodo.

Para el segundo y tercer escenario se compararon los resultados obtenidos con varios estudios del protocolo BATMAN, se consideró este protocolo principalmente porque fue creado para reemplazar a OLSR, además es ampliamente utilizado en redes comunitarias conjuntamente con el protocolo OLSR.

Siguiendo los pasos indicados en la sección anterior, se obtuvieron los resultados que se muestran en la Tabla l4.

El tiempo de convergencia de un nodo nuevo es de 23 segundos, si lo comparamos con los resultados obtenidos por María Benito en su trabajo de fin de carrera (Benito, 2010) para el protocolo BATMAN obtiene un tiempo de 6 segundos, es decir el tiempo de convergencia de un nuevo nodo en el protocolo BATMAN es un 26% menor que el protocolo OLSR. Este comportamiento se debe a la simplicidad con la que fue creado BATMAN, además el protocolo OLSR maneja más tráfico de control que el protocolo BATMAN.

6.3.3 Tercer Escenario: Cálculo de una nueva ruta

El protocolo OLSR tarda 48 segundos en calcular una nueva ruta (3 saltos). Si comparamos este tiempo con los resultados obtenidos en el trabajo de titulación de Albert Batiste (Batiste, 2011) en el cual realiza un análisis teórico y experimental de protocolos como BATMAN, consiguiendo intervalos de tiempo de 17 segundo hay que tomar en cuenta que en este trabajo los intervalos de los mensajes de control de dichos protocolos fueron reducidos a la mitad, por lo que consecuentemente se ven reducidos los tiempos de convergencia de una nueva ruta, por otra parte en el trabajo de fin de carrera de María Benito (Benito, 2010) se obtiene un tiempo de 41 segundos para el protocolo BATMAN. Analizando los resultados podemos concluir que la diferencia entre OLSR y BATMAN es minina cuando se recalcula una nueva ruta ante la caída de un nodo. Este comportamiento se debe a que OLSR mantiene información actualizada de la tabla de enrutamiento.

Tabla 14. Resultados primer y segundo escenario

RESULTADOS	OLSR	BATMAN
Tiempo de convergencia de un nuevo router	23	6
Calculo de una nueva ruta	48	41

Fuente: Elaboración propia

6.4 ANÁLISIS DE LA RED

6.4.1 Ping

A continuación se presentan los resultados obtenidos en la Figura 58

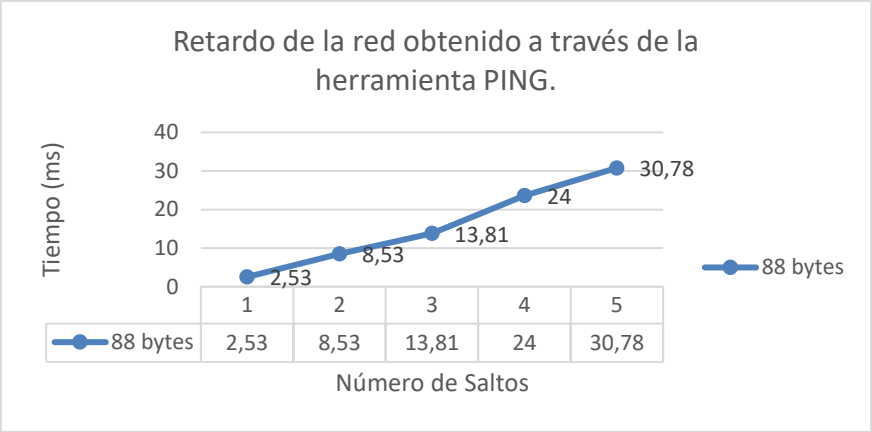

Figura 58. Datos obtenidos al realizar ping.
Fuente: Elaboración propia

Como se observa en la Figura 58 a medida que aumenta los saltos en la red aumenta el retardo en la misma, este parámetro es importante ya que el QoS o calidad de servicio de muchas aplicaciones o servicios dependen de esta métrica, por ejemplo para VoIP se establece un retardo menor a 150ms para que no sea vea comprometida la calidad de la llamada. A continuación se procedió a ejecutar la herramienta traceroute desde el NODO1 con destino al NODO6:

```
root@NODO1:~# traceroute 10.0.0.6
traceroute to 10.0.0.6 (10.0.0.6), 30 hops max, 38 porte packets
 1  10.0.0.2 (10.0.0.2)  2.429 ms  3.521 ms  1.976 ms
 2  10.0.0.3 (10.0.0.3)  3.291 ms  4.231 ms  12.780 ms
 3  10.0.0.4 (10.0.0.4)  7.959 ms  9.401 ms  12.628 ms
 4  10.0.0.5 (10.0.0.5)  24.745 ms  6.232 ms  5.617 ms
 5  10.0.0.6 (10.0.0.6)  25.714 ms  7.008 ms  10.793 ms
```

Figura 59. Resultado obtenido de traceroute para análisis de retardo
Fuente: Elaboración propia

Como se puede observar Figura 59 existen 4 nodos entre el NODO1 y NODO6, información que se encuentra en la primera columna, en la segunda y tercera se encuentra la dirección IP; los retardos se acercan a los valores obtenidos al utilizar la herramienta ping, además si ponemos atención en la línea número 4, en el segundo y tercer tiempo se obtienen retardos menores a las correspondientes de la línea 3, esto se debe a la variación del retardo de cola de cada nodo.

6.4.2 Iperf

A continuación se presenta en la Tabla 15 los resultados obtenidos con la herramienta iperf.

Tabla 15. Resultados obtenidos de las pruebas realizadas con la herramienta IPERF

		1	2	3	4	5
TCP	Tasa de Transmisión (Mbits/s)	11	6,05	5,10	2,24	0,781
UDP	Tasa de Transmisión (Mbits/s)	14	7,95	3,94	2,98	1,06

Fuente: Elaboración propia.

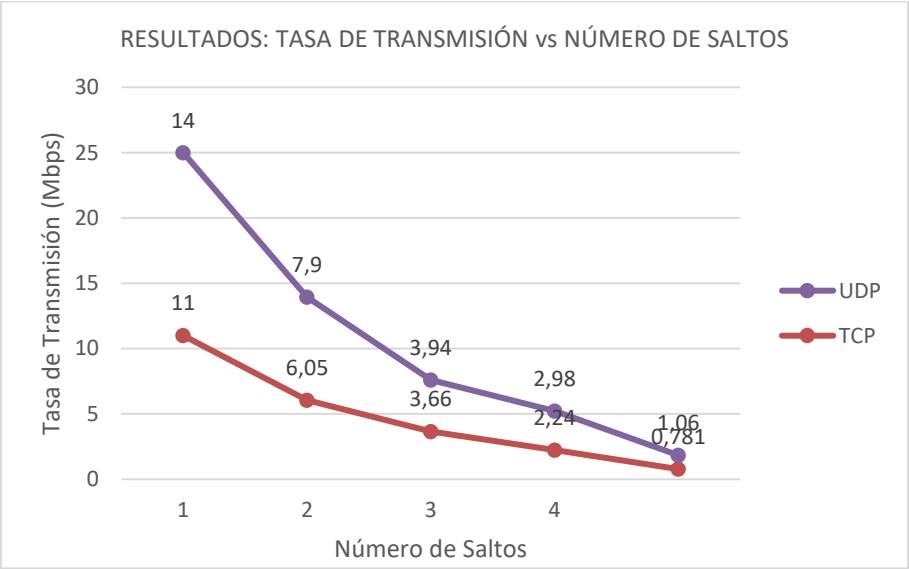

Figura 60. Gráfica de resultados de inyeccion de tráfico TCP/UDP

Fuente: Elaboración propia

Como se puede observar en la Figura 60 a medida que se aumentan los saltos la tasa de transmisión va disminuyendo. La reducción de la tasa de transmisión se debe a que todos los nodos comparten el mismo canal, por lo que cuando solo dos de los nodos están transmitiendo y recibiendo datos nos podemos acercar a la tasa de transmisión máximo de la red, mientras que cuando se aumenta el número de dispositivos que tienen que atravesar la información este disminuye, para dos saltos se obtendrá aproximadamente la mitad de la tasa de transmisión, mientras que para 3 saltos se obtendrá aproximadamente la tercera parte. De lo anterior se puede deducir que la tasa de transmisión tendrá el siguiente comportamiento: throughput máximo/ número de saltos o también expresado en throughput máximo/ (número de nodos – 1).

6.4.3 Prueba con Aplicaciones Reales.

Siguiendo la configuración descrita en la sección anterior, se obtuvieron los siguientes resultados para las pruebas con aplicaciones reales.

- **VoIP:** En la Figura 61 y Figura 62 se presenta la captura de paquetes a través de Wireshark, mientras que en la Tabla 16 se representan de manera resumida los resultados.

Wireshark · RTP Streams · LLAMADA1salto

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter
192.168.1.10	11094	192.168.2.100	10002	0x93de97d	g711U	3434	0 (0.0%)	22.216	0.639	0.309
192.168.1.10	12726	192.168.1.20	20002	0x43ec402c	g711U	3013	0 (0.0%)	1518.535	166.597	19.206

Figura 61. Captura de paquetes, llamada a 1 salto
Fuente: Elaboración propia

Wireshark · RTP Streams · 2 saltos 6 a1

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter
192.168.1.10	11984	192.168.6.118	10008	0x73a0e56f	g711U	2333	0 (0.0%)	24.283	0.869	0.493
192.168.1.10	16436	192.168.1.20	20004	0x238768a7	g711U	2248	0 (0.0%)	264.210	28.533	11.445

Figura 62. Captura de paquetes, llamada a 2 saltos
Fuente: Elaboración propia

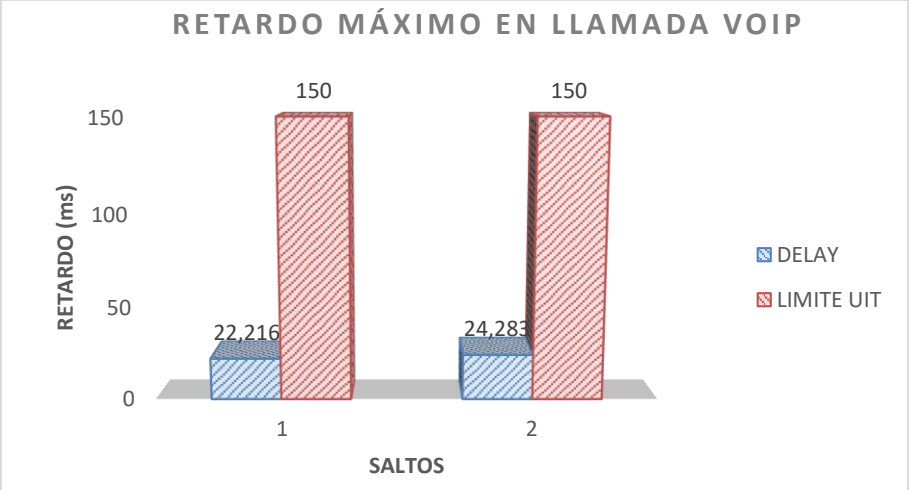

Figura 63. Retardo máximo obtenido en llamada VoIP
Fuente: Elaboración propia

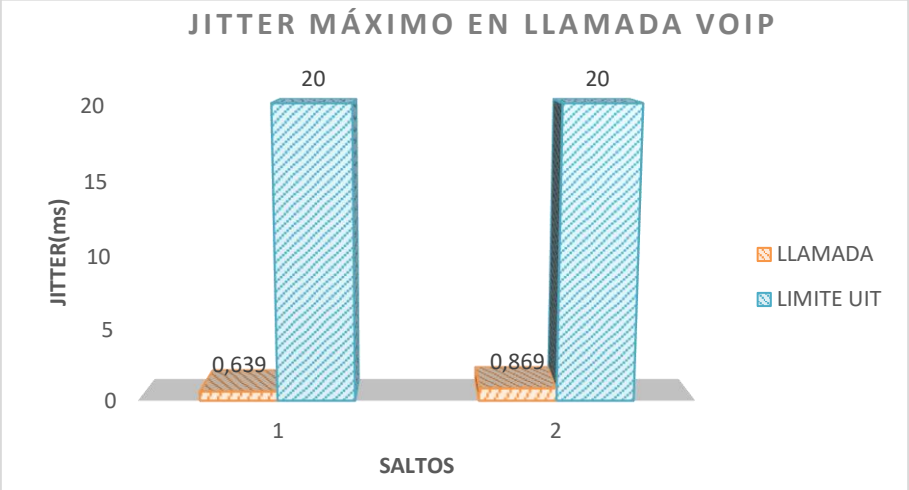

Figura 64. Jitter máximo obtenido en llamada VoIP
Fuente: Elaboración propia

Tabla 16. Resultados obtenidos de las pruebas VoIP mediante Wireshark.

PARÁMETRO	NÚMERO DE SALTOS	
	1	2
Delay	22.216 ms	24,283 ms
Jitter	0.639 ms	0,896 ms

Fuente: Elaboracion propia

Como se puede observar en la Figura 63 y en la Figura 64 los valores máximos obtenidos para el retardo y el jitter al realizar la prueba indica que a medida que aumentan los saltos, estos parámetros también incrementan pero sin llegar a sobrepasar los límites recomendados en la Rec. UIT-T G.114. En la Figura 65 y Figura 66 se muestran las gráficas de las métricas correspondientes para cada salto, como se puede observar para las gráficas de dos saltos existe más fluctuaciones en las métricas que en las gráficas de un salto, sin embargo como se mencionó anteriormente los valores máximos se encuentran por debajo de los valores recomendados, lo que se comprobó al tener una conversación clara y fluida. Indudablemente si se aumenta más terminales mesh el retardo sería mayor hasta podría llegar a superar los valores establecidos, perdiendo interactividad en la comunicación, produciendo eco, perdida de paquetes e incluso llegar a que la comunicación sea imposible. Hay que tener en cuenta que las pruebas se realizaron cuando existía poca afluencia de estudiantes en el edificio y la red funcionaba en el canal menos utilizado.

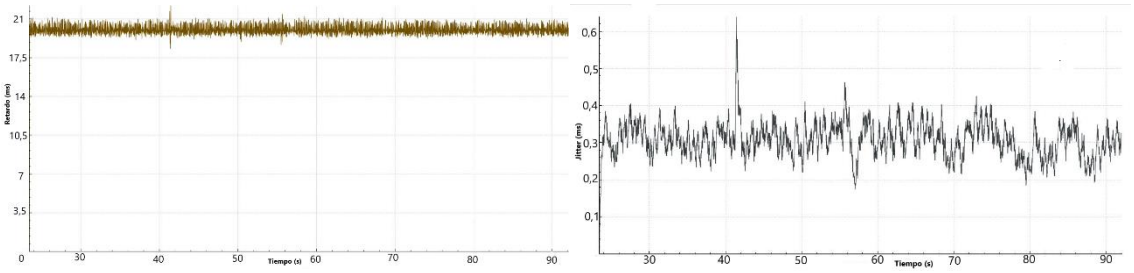

Figura 65. Gráfica RETARDO (izquierda) y JITTER (derecha)- llamada 1 salto

Fuente: Elaboración propia

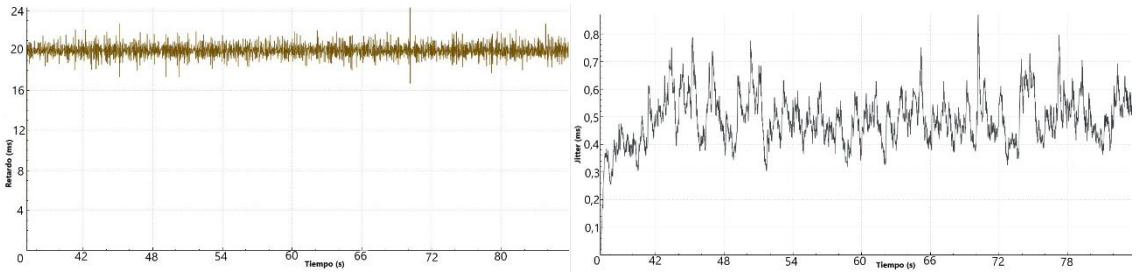

Figura 66. Gráfica RETARDO (izquierda) y JITTER (derecha)-llamada 2 saltos

Fuente: Elaboración propia

- **HTTP:**

Una vez configurado los terminales, se procedió a tomar el tiempo y velocidad de descarga para lo cual se obtuvieron los resultados descritos en la Tabla 17, además como se puede observar en la Figura 67 y Figura 68 la velocidad de transmisión del archivo no es constante, esto se debe a que TCP, que es el protocolo de transporte utilizado para HTTP, proporciona un servicio fiable de flujo, por lo que cuando se detecta problemas de congestión (perdida de paquetes) se reduce la tasa de envío hasta que se solucionen dichos problemas y después se aumenta paulatinamente.

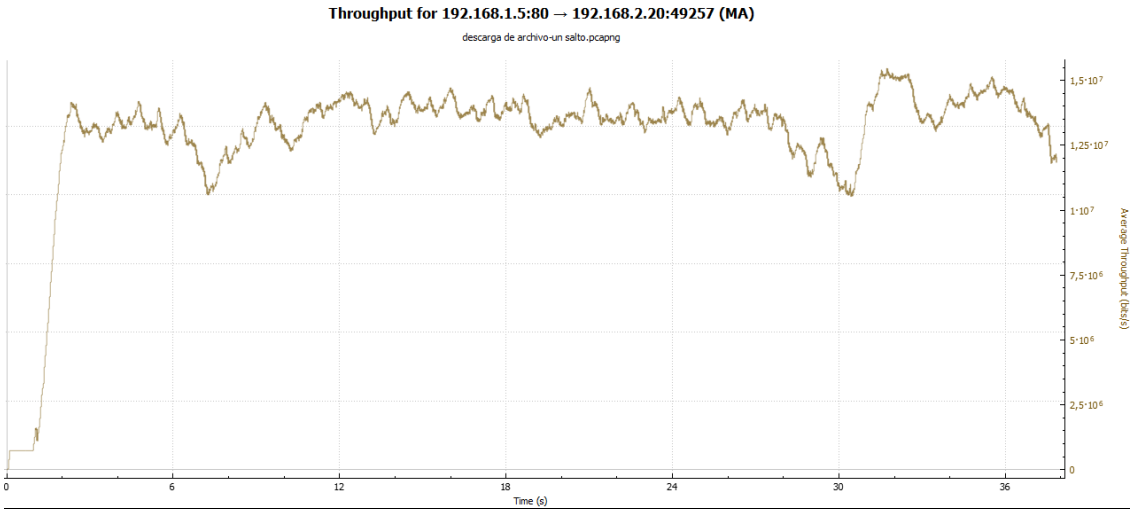

Figura 67. Throughput obtenido en la descarga del archivo. 1_SALTO
Fuente: Elaboración propia

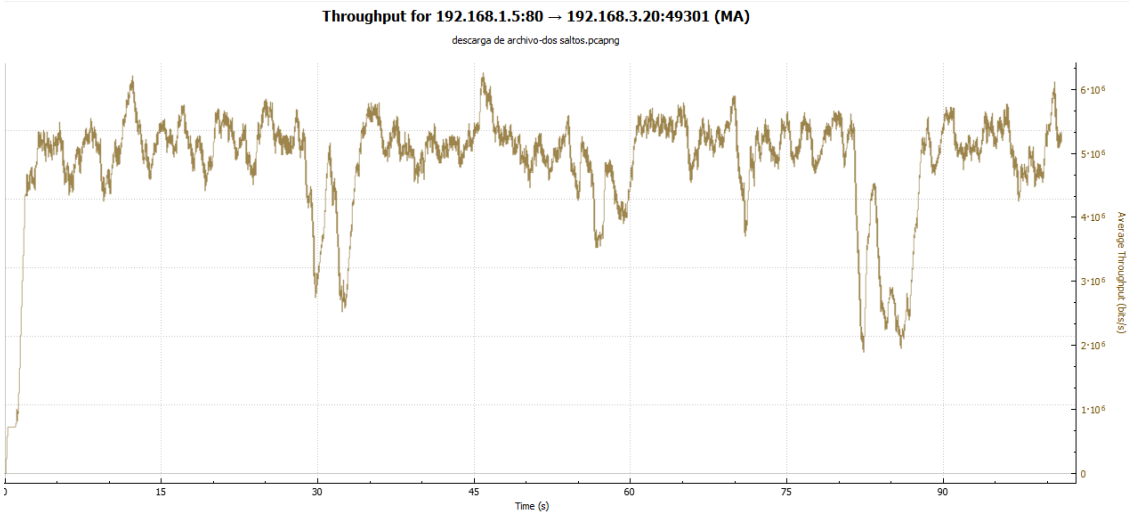

Figura 68. Throughput obtenido en la descarga del archivo. 2_SALTOS
Fuente: Elaboración propia

En la Figura 68 se observa una gráfica más variable de la tasa de transferencia instantánea que en la Figura 67, esto se debe a la interferencia co-canal y de canal adyacente debido a la presencia de redes que operan en el mismo edificio y al mayor número de nodos que debe atravesar la información. Una vez obtenidos el tiempo de transferencia a través de la herramienta Wireshark se procedió a obtener la tasa media de transmisión como se indicó en la Ecuacion 3, los resultados se observan en la Tabla 17.

Tabla 17. Resultados obtenidos en la prueba de HTTP

	NUMERO DE SALTOS	
	1	**2**
Tamaño de Archivo	59 MB	59MB
Tiempo de Descarga (segundos)	37,82	101,28
Velocidad de Transmisión	12 Mbps	5,5 Mbps

Fuente: Elaboración propia

Para un salto se obtiene una velocidad media de transmisión cerca de 12,5Mbps mientras que para dos saltos se obtienen menos del 50% por lo que podemos deducir que el número de dispositivos es un aspecto fundamental en las redes mesh, por lo que afecta negativamente al rendimiento y que el desempeño se ve afectado debido a que los equipos utilizan el mismo canal y a la presencia de otras redes las cuales causan interferencia.

6.5 COSTO DEL PROTOTIPO

Para poder realizar una estimación del costo del prototipo se tomó en cuenta los precios actuales de los equipos y materiales utilizados, y se incluyó equipos que permitan el acceso inalámbrico (Access Point). En la Tabla 18 se presenta el presupuesto del prototipo.

Tabla 18. Presupuesto del prototipo

Descripción	Cantidad	Valor Unitario	Valor Total
Linksys WRT54gs v 3.0	6	60.00	360.00
Linksys WRT54Gx	6	60.00	360.00
Otros	---	20.00	20.00
TOTAL			740.00

Fuente: Elaboración propia

También se creyó conveniente incluir materiales que permitan el funcionamiento de los nodos en ambientes outdoor, aunque no fueron utilizados en la realización de prototipo; para de esta manera poder realizar una comparación con otros nodos mesh disponibles en el mercado. En la Tabla 19 se detalla el presupuesto del prototipo con los materiales necesarios para que el nodo opere en ambientes outdoor.

Tabla 19. Presupuesto del prototipo para ambientes outdoor

Descripción	Cantidad	Valor Unitario	Valor Total
Linksys WRT54gs v 3.0	6	60.00	360.00
Linksys WRT54GX	6	60.00	360.00
Caja para exteriores	6	80.00	480.00
Otros	----	30.00	30.00
TOTAL			1 230.00

Fuente: Elaboración propia

Se puede observar que el precio unitario para la construcción de un nodo mesh es de un valor estimado de 200 dólares, un precio mucho menor a los equipos mesh que se encuentran en el mercado que varían entre los 400, 500 y 1000 dólares por lo que se puede observar la viabilidad al usar estos equipos para su modificación y creación de nodos mesh.

7. DISCUSIÓN

Las redes inalámbricas son una solución tecnológica a muchos problemas, entre ellos ofrecer servicios a lugares remotos o zonas afectadas por desastres naturales, por eso son varios los estudios que se han realizado en torno a esta tecnología, las potencialidades de las redes mesh inalámbricas ha conllevado a la creación de varios firmwares open source y protocolos de enrutamiento debido a sus grandes ventajas, siendo el protocolo OLSR el más utilizado en varias redes y OpenWrt el firmware más completo y utilizado como base para proyectos como Freifunk, LEDE Project, entre otros. Este proyecto tuvo como objetivo el diseño de un piloto de Red Wireless Mesh con OpenWrt y protocolo OLSR; y su implementación en un ambiente de Laboratorio, para luego a través de herramientas de análisis de red evaluar el desempeño del protocolo y de la red a través de métricas como la tasa de transmisión y retardo

Objetivo 1. Configurar los routers Linksys existentes en el Laboratorio de Telecomunicaciones de la A.E.I.R.N.N.R de la Universidad Nacional de Loja con el firmware OpenWrt y protocolo OLSR.

Par el cumplimiento de este objetivo se realizó un estudio previo de las diversas propiedades que posee OpenWrt, como las versiones disponibles, dispositivos soportados, archivos de configuración, comandos, editores de texto, administrador de paquetes, entre otras; así también se realizó una indagación sobre las características técnicas, como controladores inalámbricos, capacidad de memoria, puertos que posee, entre otras, de los equipos existentes en el Laboratorio de Telecomunicaciones del A.E.I.R.N.N.R permitiéndonos elegir al Linksys Wrt54gs versión 3.0 como dispositivo apropiado para la implementación del prototipo, de la misma manera se efectuó un análisis del protocolo OLSR y de las redes mesh; lo que nos permitió formar una base sólida para la configuración de los nodos que conforman el prototipo.

Los conocimientos adquiridos conllevaron a elegir las versiones Kamikaze 8.09.2 y Backfire 10.03.1 de OpenWRT para ser instalada en los Linksys WRT54gs versión 3.0, y a realizar la instalación de paquetes adecuados y necesarios para el correcto funcionamiento de OpenWrt y el protocolo OLSR. Para la configuración de los nodos se tuvo que tomar en cuenta que todos deben tener el mismo nombre de red y canal de

operación, así mismo se utilizó el parámetro HNA de OLSR para establecer comunicación entre la interfaz inalámbrica y la ethernet. Se pudo evidenciar que OpenWrt conjuntamente con el protocolo OLSR presentan una solución viable y sencilla a la hora de implementar redes mesh, gracias a la amplia documentación que presenta, gran cantidad de paquetes disponibles para cada versión y por poder instalarse en un amplia gama de dispositivos.

El presente objetivo se cumplió al instalar en los routers Linksys WRT54gs versión 3.0 el release Kamikaze 8.09.2 y Backfire 10.03.1 de OpenWRT y el demonio OLSR conjuntamente con complementos necesarios para su correcto funcionamiento, y permitiéndonos a través de la interfaz gráfica una configuración sencilla, obteniendo routers funcionales para la implementación de una red mesh inalámbrica con protocolo OLSR.

Objetivo 2. Implementar el prototipo de Red Inalámbrica Mallada en un ambiente de Laboratorio.

Para la implementación de la red, primero se efectuó un sondeo de las redes existentes en el laboratorio de Telecomunicaciones a través del software Wifi Analizer, lo que nos permitió elegir el canal que menos ocupado esté, luego se procedió a colocar los 6 nodos que conforman la red sobre los escritorios ya que cerca de ellos se encuentran las tomas de corriente para el suministro de energía, puntos de red en caso de que se requiera hacer descarga de paquetes en el router o tener acceso a internet y computadores que nos permiten visualizar el estado de los routers. Fue necesario definir políticas de filtrado a través del comando iptables, obteniendo de esta manera varias topologías de red que fueron utilizadas para la realización de pruebas.

El objetivo número dos se cumplió al implementar en el Laboratorio de Telecomunicaciones del A.E.I.R.N.N.R un prototipo de red mesh que consta de 6 nodos conectados de manera total, y, a través del comando iptables se pudo obtener varías topologías en donde los nodos están conectados de manera parcial y que se utilizaron para diferentes pruebas.

Objetivo 3. Analizar el comportamiento del protocolo OLSR sobre una red mallada configurada con firmware libre.

Una vez implementada el prototipo de red, se propusieron tres escenarios que nos permitió analizar el comportamiento del protocolo, el primer escenario nos permitió realizar el análisis del protocolo a través de la captura de tráfico de Wireshark, mediante este software se pudo obtener información de los paquetes de control que utiliza el protocolo, y mediante el análisis de se pudo verificar los conocimientos obtenidos través la indagación bibliográfica y la simulación con los obtenidos en la práctica. El segundo escenario consistió en obtener el tiempo de convergencia de un nuevo router a través del comando ping y fue comparado con el protocolo BATMAN obteniendo un tiempo mayor a su contendiente, y el tercer escenario nos permitió obtener el tiempo que tarda el protocolo en establecer una nueva ruta obteniendo tiempos similares con el protocolo BATMAN. A pesar de que en el segundo escenario se muestra un tiempo "desfavorable" para el protocolo OLSR hay que tomar en cuenta su complejidad, además que es más factible utilizarlo en redes de gran tamaño ya que trabaja a nivel de capa de enlace.

El tercer objetivo se efectuó mediante las múltiples pruebas realizadas en los escenarios, y a través de herramientas de análisis de red como Wireshark y ping, se pudieron obtener parámetros que nos permitieron evaluar el comportamiento del protocolo y poder analizarlo con resultados obtenidos de experimentos similares para el protocolo BATMAN.

Objetivo 4. Verificar el correcto funcionamiento y desempeño del piloto, mediante parámetros como la tasa de transmisión y retardo con ayuda de herramientas de análisis de red.

Para verificar el correcto funcionamiento y desempeño del piloto se estableció un escenario para la realización de las pruebas y conjuntamente con varias herramientas de análisis de red se obtuvieron las métricas: tasa de transmisión y retardo. Los escenarios se propusieron con el fin de analizar el impacto que tiene el número de saltos en las métricas ya mencionadas, para lo cual se utilizó la herramienta de diagnóstico ping permitiéndonos constatar que el retardo se ve afectado por el número de saltos, con iperf se comprobó que la tasa de transmisión decae. Se decidió realizar pruebas con servicios reales (VoIP y HTTP), para lo cual se colocó tres dispositivos en cada piso del Edificio de Laboratorios, para la prueba de telefonía se realizó una llamada y se capturo tráfico

a través de Wireshark, lo que nos permitió evidenciar que las métricas delay y jitter se encuentran por debajo de los límites establecidos por la UIT, mientras que para la pruebas HTTP se decidió descargar un archivo alojado en WAMPserver permitiéndonos corroborar los resultados obtenidos de las pruebas realizadas anteriormente y la capacidad de la red para la provisión de diferentes servicios .

Con los valores obtenidos en las pruebas se logró verifica la viabilidad de un red mesh para el manejo de trafico sensible al retardo y a la perdida, sin embargo hay que tener en cuenta que el número de saltos influye fuertemente en la variación de métricas como la tasa de transmisión y el retardo, además que este comportamiento se presenta por la utilización del mismo canal; por lo que el desempeño de la red disminuye, convirtiéndose el número de dispositivos en un aspecto fundamental a la hora del diseño de una red mesh.

8. CONCLUSIONES

- Se determinó que OpenWRT es el firmware open-source más completo en comparación a otros como DD-WRT y Freifunk, debido a la gran cantidad de información detallada y actualizad alojada en sus repositorios y principalmente a la variedad de paquetes que pueden ser instalados para la creación de redes inalámbricas mesh con diferentes protocolos de enrutamiento, además el demonio OLSR está disponible como paquete ya compilado para este entorno, permitiéndonos crear de una manera sencilla y rápida redes inalámbricas mesh con OLSR.
- Se comprobó que el proceso de instalación y configuración de OpenWRT y OLSR es bastante sencillo y no presenta inconvenientes en su desarrollo, además existe la posibilidad de añadir funcionalidades extras sin alterar el funcionamiento básico del protocolo.
- Se diseñó e implementó un prototipo de red mesh en un ambiente de laboratorio que consta de 6 enrutadores Linksys WRT54gs v3.0, a través del filtrado de paquetes OLSR (mediante el comando iptables) se pudo crear varias topologías, que sirvieron para la realización de diferentes pruebas que nos permitieron caracterizar el desempeño del protocolo y de la red.
- A través de las pruebas se evidencio que OLSR tiene un retardo alto cuando se introduce un nuevo nodo, este comportamiento se debe a que OLSR maneja más tráfico de control que BATMAN. A demás se obtuvieron tiempos similares cuando se calcula una nueva ruta, dicho comportamiento se debe a que OLSR dispondrá de rutas alternativas ya que tiene conocimiento total de la topología de la red, mientras que BATMAN tendrá que calcularlas ya que no conoce todas las posibles rutas a través de la red.
- Las pruebas realizadas para analizar el protocolo OLSR nos permitieron inferir que OLSR ofrece mejores prestaciones en redes con topología estática como las redes MESH mientras que BATMAN es factible utilizarlo en redes que presenta una topología dinámica donde los cambios de topología son constantes y se requiere una rápida respuesta como las redes MANET.

- Las pruebas realizadas en los diferentes escenarios nos permitieron obtener mediciones de las métricas: delay, jitter y tasa de transmisión, con las pruebas se constató que el retardo y el jitter son proporcionales al número de saltos, mientras que la tasa de transmisión es inversamente proporcional, para tráfico sensible al retardo y a la perdida como VoIP estos parámetros son importantes ya que si se sobrepasa límites establecidos para diferentes servicios podrían afectar la QoS.

9. RECOMENDACIONES

- Conocer el nombre de los controladores inalámbricos del equipo a utilizar, de esta manera se podrá elegir el release del firmware apropiado que deberá ser instalado, además ser cuidadoso en el proceso de flasheo del router, una mala instalación del firmware o una mala elección conllevaría al bloqueo del router, teniendo que recurrir a resetear el nodo o a métodos tediosos para volver a cargar el firmware y en el peor de los casos un dispositivo inservible.
- Realizar un estudio previo de OpenWRT y el protocolo OLSR, esto nos permitirá conocer de manera detallada algunos parámetros y características importantes que servirán en el momento de la instalación, configuración y análisis de la red.
- Configurar la potencia de transmisión del equipo a 20 dBm como máximo, potencias más altas producen ruido en la transmisión y posibles daños en el equipo.
- Realizar el montaje de la red con otros dispositivos de diferentes marcas y con Linksys que contengan chipset BRC 47XX, y de esta manera poder corroborar la interoperabilidad entre dispositivos al instalar en ellos OpenWRT, también dando la oportunidad de instalar versiones más recientes del firmware.
- Realizar un estudio e implementación de una WMN en un escenario real con los dispositivos ya configurados, teniendo en cuenta el cambio de las antenas de fábrica por unas de mayor ganancia además de la posibilidad de hacer que los nodos reciban energía a través de paneles solares o baterías para brindar servicio a zonas de difícil acceso.
- Utilizar el prototipo como escenario para el estudio de otros protocolos para redes mesh que pueden ser implementadas con Open-WRT como BATMAN y 802.11s.
- Realizar un estudio sobre el impacto en el retardo y la tasa de transmisión cuando se realiza el acceso a la red a través de Access Point, y de esta manera determinar la variación de las métricas con los resultados obtenidos en el presente trabajo.
- Realizar un estudio sobre el impacto del número de saltos con las métricas ya mencionadas utilizando equipos 802.11 con dos radios en diferentes bandas (2,4GHz y 5GHz) o mediante la construcción de nodos mesh utilizando dos equipos 802.11 que cumplan con las características ya mencionadas.

10. BIBLIOGRAFÍA

Asadoorian, P., & Pesce, L. (2017). *Linksys WRT54G. Ultimate Hacking.* Burlington USA: Syngress Publishing,Inc.

Batiste, A. (19 de Junio de 2011). *Protocolos de encaminamiento en redes inalámbricas mesh: un estudio teorico y experimental.* Obtenido de http://hdl.handle.net/10609/8164

Benito, M. (2010). *Evaluacion experimental de redes malladas basadas en el protocolo B.A.T.M.A.N.* Obtenido de https://e-archivo.uc3m.es/handle/10016/7365

Bourdon, R. (13 de Julio de 2004). *WampServer: APACHE, PHP, MySQL on Windows*. Obtenido de http://www.wampserver.com/en/

BrainSlayer. (22 de Enero de 2005). *dd-wrt.com Your connection is over IPv4*. Obtenido de https://www.dd-wrt.com

Büttrich, S. (Octubre de 2007). *Unidad 13: Mesh.* Obtenido de http://docplayer.es

Carpenter, T. (2008). Chapter 5: Wireless Desing Models, Topologies, and Infraestructure . En *CWNA Certified Wireless Network Administrator Official Study Guide* (Cuarta ed., págs. 216-218). USA: Mc Graw Hill.

Clausen, T., & Jacquet, P. (Octubre de 2003). *Optimized Link State Routing Protocol (OLSR)*. (IETF RFC 3626) Obtenido de https://tools.ietf.org/html/rfc3626

Elastix.org. (01 de Marzo de 2006). *Elastix: Freedom to comunicate.* Obtenido de https://www.elastix.org/es/

Forouzan, B. (2002). Conceptos básicos. En *Transmision de datos y redes de comunicaciones* (Segunda ed.). Madrid, España: McGRAW-HILL.

Freifunk. (2003). *freifunk.net*. Obtenido de https://freifunk.net/

GitHub. (6 de Agosto de 2012). *OLSRd (version 0.6.0) protocol extensions*. (GitHub.Inc) Obtenido de https://github.com/servalproject/olsr/blob/master/README-Olsr-Extensions

Grupo NAP del COIT. (2004). *La situacion de las Tecnologías WLAN basadas en el estándar IEEE 802.11 y sus variantes ("Wi-Fi").* Obtenido de Colegio Oficial de Ingenieros de Telecomunicaciones: https://www.coit.es/informes/la-situacion-de-las-tecnologias-wlan-basadas-en-el-estandar-ieee-80211-y-sus-variantes-wi-0

Guerrero, A. (Julio de 2010). *Estudio experimental del funcionamiento de OSPF-MANET y OLSR en redes malladas multi-salto inalámbricas.* Obtenido de http://hdl.handle.net/10016/11094

Guifi.net. (2004). Obtenido de https://guifi.net

Held, G. (2005). *Wireless Mesh Networks.* New York: Auerbach Publications.

Ishfaq, M. (2014). Comparative Analysis of Ad Hoc Routing Protocols in. *World Applied Sciences Journal, 9*(23), 408-414.

Johnson, D., Maltz, D., & Hu, Y.-C. (2007). *The Dynamic Source Routing Protocol (DSR)*. (IETF RFC 4728) Obtenido de https://www.ietf.org/rfc/rfc4728.txt

Kurose, J., & Roos, K. (2010). *Redes de computadoras: Un enfoque descendene* (Quinta ed.). Madrid, España: Pearson Educacion, S.A.

Linksys Group. (2002). *Wireless-G Broadband Router: User guide.* Recuperado el 18 de Febrero de 2017, de https://www.linksys.com/us/support-article?articleNum=139826

Lopez, A. (29 de Noviembre de 2013). *Google Play: Wifi Analizer.* Obtenido de https://wifi-analyzer.uptodown.com/android

Misra, S., Misra, S. C., & Woungang, I. (2003). *Guide to Wireless Mesh Networks.* Londres: Springer-Verlag.

Nagashagui (Dirección). (26 de Febrero de 2015). *Hacer ping con fecha hora - how to ping with time stamp* [Película]. Obtenido de https://www.youtube.com/watch?v=ot4kExUpCuc

NetworkBogota. (2010). Obtenido de https://networkbogota.org

Neumann, A., Aichele, E., Lindner, M., & Wunderlich, S. (9 de Octubre de 2008). *Better Approach To Mobile Ad-hoc Networking (B.A.T.M.A.N.)*. (Internet-Draft) Obtenido de https://tools.ietf.org/html/draft-wunderlich-openmesh-manet-routing-00

Neyra, D. (Dirección). (Junio de 2014). *Red Ad Hoc Viewer: Detectar y Conectar Ad-Hoc en Windows 8.1* [Película]. Obtenido de https://www.youtube.com/watch?v=30c7nQJwzxs

NS3: Network Simulator. (2010). Obtenido de https://www.nsnam.org/

open-mesh. (2006). *B.A.T.M.A.N. protocol concept*. Obtenido de Open-Mesh: https://www.open-mesh.org/projects/open-mesh/wiki/BATMANConcept

OpenWRT. (Enero de 2004). *OpenWRT: Wireless Freedom*. Recuperado el 14 de Marzo de 2017, de https://openwrt.org/

Orebaugh, A. (2007). *Wireshark & Ethereal Network Protocol Analyzer Toolkit* . Rockland,MA: Syngress Publishing,Inc. .

Pandey, A., & Manoj, B. (2004). *Performance Analysis of OLSR and Modified Version of OLSR-ETX/MD/ML in Mesh Networks.* Obtenido de International Journal of Computer Science and Communication Networks (IJCSCN): http://www.ijcscn.com/Documents/Volumes/vol2issue2/ijcscn2012020220.pdf

Perkins, C., Nokia, Belding-Royer, E., Universidad de California, Das, S., & Universidad de Cincinnati. (2003). *Ad-hoc On-demand Distance Vector (AOVD)*. (IETF RFC 3561) Obtenido de https://www.ietf.org/rfc/rfc3561.txt

PortSip Solutions. (25 de Enero de 2010). *PortGo: Connected Anywhere, and everywhere*. Obtenido de https://www.portsip.com/download-portsip-softphone/

Salazar, J. (2016). *Redes Inalámbricas.* Cataluña: European Virtual Learning Platform for Electrical and Information Engineering. Obtenido de http://hdl.handle.net/2117/100918

SoCalFreeNet.org Wireless User Group. (2004). *Wireless Hacking: Projects for Wi-Fi Enthusiasts.* (L. Barken, Ed.) USA: Syngress.

The Iperf team. (Marzo de 2003). *iPerf: la herramienta de prueba de velocidad máxima para TCP, UDP y SCTP.* Obtenido de https://iperf.fr/

VillageTelco. (2011). Obtenido de https://villagetelco.org

Wireshark. (1999). *Wireshark*. Recuperado el 16 de Febrero de 2017, de https://www.wireshark.org/

WNDW. (2013). *Wireless Networking in the Developing World.* Obtenido de http://wndw.net/

Zhang, Y., Luo, J., & Honglin, H. (2007). NETWORK ARCHITECTURAL DESING ISSUES y ROUTING IN WIRELESS MESH NETWORKS. En *WIRELESS MESH NETWORKING: Architectures, Protocols and Standards* (págs. 14, 114-127). USA: Auerbach Publications.

11. ANEXOS.

Anexo 1. Acrónimos

ACRÓNIMO	SIGNIFICADO
ARCOTEL	Agencia de Regulación y Control de las Telecomunicaciones
ANSN	Número de secuencia de vecino anunciado/ Advertised Neighbor Sequence Number
AODV	Vector Distancia para Redes Ad.hoc / Ad-hoc On-demand Distance Vector
AP	Punto de Acceso/ Acces Point
BATMAN	Mejor Enfoque para Redes Móviles Ad-hoc / Better Approach to Mobile Ad hoc Networking
BSS	Conjuntos de Servicios Básicos/ Basic Server Set
BSSID	Identificador del Conjunto de Servicios Básicos/ Basic Service Set Identifier
CCK	Modulación por Código Complementario/ Complementary Code Keying
DSR	Enrutamiento de Origne Dinámico/ Dynamic Source Routing
DHCP	Protocolo de Configuración Dinámica de Servidor /Dymanic Host Configuration Protocol
DSSS	Espectro ensanchado de secuencia directa/Direct Sequence Spread Spectrum
EDCA	Función Mejorada de Distribución de Acceso al Canal/ Enhanced Distributed Channel Access
ESS	Conjunto de Servicios Extendidos / Extended Service Set
ESSID	Identificador del Conjunto del Servicios Extendidos / Extended Service Set IDentifier

ETSI	Instituto Europeo de Estándares de Telecomunicaciones/ European Telecommunications Standards Institute
FCC	Comisión Federal de Comunicaciones/Federal Communications Commission
FHSS	Espectro Expandido por Salto de Frecuencia/ Frecuency Hopping Spread Spectrum
GNU	GNU's Not Unix/ GNU no es Unix
GNU GPL	GNU General Public License/ Licencia Pública General de GNU
HCCA	Función HCF Mejorada de Distribución de Acceso al Canal/ HCF Controlled Channel Access
HCF	Función de Coordinación Híbrida/ Hybrid Coordination Function
HiperLAN2	Red de Área Local de Radio de Alto Rendimiento 2.0/ High Performance Radio LAN 2.0
HNA	Asociación de Host y Red/ Host and Network Association
HTTP	Protocolo de Transferencia de Hiper Texto/ HyperText Transfer Protocol
IANA	Autoridad de Números Asignados en Internet/ Internet Assigned Numbers Authority
IBSS	Conjunto de Servicios Básicos Independientes/ Independent Basic Server Set
ICMP	Protocolo de mensajes de control de Internet / Internet Control Message Protocol
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos/ Institute of Electrical and Electronics Engineers
IETF	Grupo de trabajo de ingeniería de Internet/ Internet Engineering Task Force
INRIA	Instituto Nacional de Investigación en Informática y Automatización/ Institut National de Recherche en Informatique et en Automatique

ISM	Ciencia, Médica e Industrial/ Industrial, Scientific and Medical
LQ	Link Quality/ Calidad de Enlace
MAC	Método de acceso al medio/ Medium Access Mechanism
MID	Declaración de Interfaz Múltiple/ Multiple Interface Declaration
MIMO	Múltiples-Entradas, Múltiples-Salida /Multiple-Input Multiple-Output
MIPS	Microprocesador sin bloqueos en las etapas de segmentación/ Microprocessor without Interlocked Pipeline Stages
MPR	Relés Multipunto/ Multipoint Pelays
MPtMP	Multipunto a Multipunto/ Multipoint to Multipunto
NIC	Tarjeta de Interfaz de Red/ Network Interface Card
NLQ	Neighbor Link Quality/ Calidad del Enlace Vecino
NS-3	Simulador de Red version 3/ Network Simulator 3
OEM	Fabricante de Equipos Originales/ Original Equipment Manufacturer
OFDM	Multiplexación por División De Frecuencia Ortogonal/ Orthogonal Frecuency Division Multiplexing.
OGM	Mensaje Originador/ Originator Message
OLSR	Enrutamiento de estado enlace optimizado/ Optimized Link State Routing
OLSR-ETX	OLSR-Conteo de Transmisión Esperada/ OLSR-Expected Transmission Count
OLSR-MD	OLSR- Retardo Mínimo/ OLSR- Minimum Delay
OLSR-ML	OLSR-Perdidas Mínimas/ OLSR-Minimum Loss
OPKG	Gestor de Paquetes Open PacKaGe management
PBX	Central Privada Automática/ Private Branch Exchange
PING	Buscador o rastreador de paquetes en redes/ Packet Internet Groper

POS	Espacio Operativo Personal/Personal Operating Space
PSN	Número de Secuencia del Paquete/ Packet Sequence Number
PtMP	Punto a Multipunto/ Point to Multipointo
PtP	Punto a Punto/Point to Poiny
QoS	Calidad de Servicio/ Quality of Service
RFID	Identificación Por Radio Frecuencia/ Radio Frequency IDentification
RISC	Computador con Conjunto de Instrucciones Reducidas/ Reduced Instruction Set Computer
RP-TNC	Reverse Polarity–Threaded Neill-Concelman
RSL	Nivel Mínimo de Señal Recibida/ Received Signal Level)
RTT	Tiempo de Ida y Vuelta/ Round Trip Time
SDRAM	Memoria de Acceso Aleatorio Dinámica Síncrona/ Synchronous Dynamic Random Access Memory
SoC	Sistema en un chip/ System on a Chip
SSID	Identificador del Conjunto del Servicios/ Service Set Identifier
TC	Control de Topología/Toplogy Control
TCP	Protocolo de Control de Transmisión/ Transmission Control Protocol
TTL	Tiempo de Vida/ Time To Live
UDP	Protocolo de Datagrama de Usuario/ User Datagram Protocol
UIT	Unión Internacional de Telecomunicaciones
U-NII	Infraestructura Nacional de Información Sin Licencia/ Unlicensed National Information Infrastructure
VoIP	Voz sobre IP/ Voice-over-IP
WAMP	Windows, Apache, MySQL, PHP

WiMAX	Interoperabilidad Mundial para Acceso de Microondas/ Worldwide Interoperability for Microwave Access
WLAN	Red Inalámbrica de Área Local/Wireless Local Area Network
WLL	Wireless Local Loop/ Bucle Local Inalámbrico
WMAN	Red Inalámbrica de Área Metropolitana /Wireless Metropolitan Area Network
WMN	Red Inalámbrica Mallada/ Wireless Mesh Network
WN	Red Inalámbrica/ Wireless Network
WPAN	Red Inalámbrica de Área Personal/ Wireless Personal Area Network
WWAN	Red Inalámbrica de Área Amplia/ Wireless Wide Area Networks Area Network

Anexo 2. Simulación NS3

1. INSTALACIÓN DE NS3

- **Abrir un terminal e ingresar como super usuario**

  sudo -i

- **Instalación de pre-requisitos:** Seguir los pasos indicados en la página oficial de ns3 dependiendo de la versión de Linux que se utilice, para Ubuntu 16.04 LTS https://www.nsnam.org/wiki/index.php/Installation#Ubuntu.2FDebian.2FMint
- **Instalación de ns3**
  - ✓ mkdir ns3
  - ✓ cd ns3
  - ✓ wget http://www.nsnam.org/release/ns-allinone-3.27.tar.bz2
  - ✓ tar xjf ns-allinone-3.27.tar.bz2
- **Habilitación de los módulos, ejemplos y test**
  - ✓ ./build.py –enable-examples –enable-test

2. EDITAR EL ARCHIVO "olsr-hna"

Una vez instalado NS3 en nuestro computador se tendrá que ingresar a la carpeta donde está alojado el archivo para poderlo editar.

- cd ns3/ ns-allinone-3.27/ ns-3.27/src/olsr/examples
- **Creación de nodos OLSR :** Se colocará dentro de los corchetes el número de nodos, en nuestro caso 6

```
NodeContainer olsrNodes;
olsrNodes.Create (6);
```

- **Comunicación del nodo OLSR y el nodo noOLSR:** Este comando indica que el sexto nodo OLSR (el conteo de nodos comienza desde 0), estará conectado al nodo 1 no OLSR

```
NetDeviceContainer csmaDevices = csma.Install (NodeContainer (csmaNodes.Get
(0), olsrNodes.Get (5)));
```

- **Movilidad de nodos:** en estas líneas de comandos se asigna la posición de los nodos OLSR y del nodo no-OLSR.

```
MobilityHelper mobility;
Ptr<ListPositionAllocator> positionAlloc = CreateObject<ListPositionAllocator> ();
positionAlloc->Add (Vector (100.0, 100.0, 0.0));
positionAlloc->Add (Vector (200.0, 100.0, 0.0));
positionAlloc->Add (Vector (300.0, 100.0, 0.0));
positionAlloc->Add (Vector (100.0, 200.0, 0.0));
positionAlloc->Add (Vector (200.0, 200.0, 0.0));
positionAlloc->Add (Vector (300.0, 200.0, 0.0));
positionAlloc->Add (Vector (400.0, 200.0, 0.0));
mobility.SetPositionAllocator (positionAlloc);
mobility.SetMobilityModel ("ns3::ConstantPositionMobilityModel");
mobility.Install (olsrNodes);
mobility.Install (csmaNodes);
```

- **Exclusión de interfaz:** excluye la interfaz 2 del nodo OLSR 6 para que no participe en el enrutamiento OLSR.

```
olsr.ExcludeInterface (olsrNodes.Get (5), 2);
```

- **Asignación de dirección IP**: indica que es el nodo 6 al que se asignará la dirección IP de la asociación:

```
Ptr<Socket> source = Socket::CreateSocket (olsrNodes.Get (5), tid);
```

- **Asociacion:** Indica que el nodo 6 es el que participa en la asociacion

```
Ptr<Ipv4> stack = olsrNodes.Get (5)->GetObject<Ipv4> ();
```

3. CORRER EL SCRIPT.

Hay tres maneras de hacer correr el archivo, para poder observar la generación de mensajes HNA se puede utilizar el segundo o tercer comando

1. ./waf –run olsr-hna
2. ./waf –run “olsr-hna assocMethod1=1”

3. ./waf –run “olsr-hna assocMethod2=1”

Para poder visualizar de manera gráfica se agrega al final del comando anterior la palabra --vis

- ./waf –run “olsr-hna assocMethod2=1” --vis

Al correr la simulación se generan archivos .pcap que se alojan en ns3/ ns-allinone-3.27/ ns-3.27/ y se los analiza con el programa Wireshark.

Anexo 3. Instalación del Protocolo y de los Paquetes

La instalación del protocolo y de los paquetes adicionales se realizó mediante la herramienta PUTTY, a continuación se detalla el proceso seguido:

1. Ingresar al router a través de PUTTY

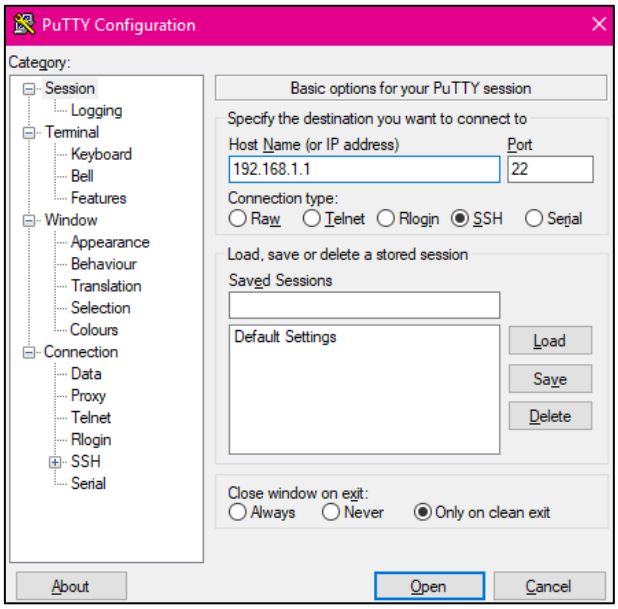

Figura 69. Ingreso al nodo a través de la herramienta Putty
Fuente: Elaboración propia

2. Escribir el username y password del router
3. Realizar un update

```
opkg update
```

4. Instalación de paquetes

```
opkg install nano luci-app-firewall luci-app-olsr  olsrd-luci-mod-dyn-gw-plain  olsrd-luci-mod-nameservice olsrd-luci-mod- txtinfo
```

5. Inicialización de OLSR

```
/etc/init.d/olsrd enable
/etc/init.d/olsrd start
```

Anexo 4. Configuración Mediante Interfaz Gráfica

Todo cambio realizado deberá ser guardado, para lo cual se deberá dar clic en la pestaña "Save & Apply" cada vez que se realicen, además todas las configuraciones se realizaran en la pestaña ADMINISTRATION

1. Contraseña y nombre

- Se deberá cambiar la contraseña del dispositivo, para poder ingresar a él a través de línea de comandos, para esto debemos ingresar a *System/Admin Password* y realizar el cambio

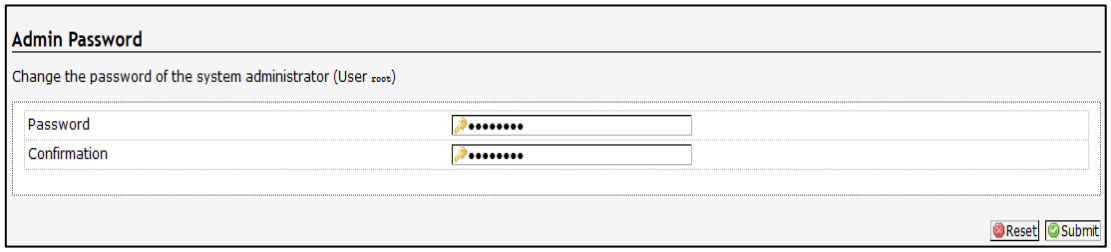

Figura 70. Cambio de contraseña.
Fuente: Elaboración propia

- Ingresar a "System", y escribir el nombre del router en el campo de "Hostname", además seleccionar "Timezone" correcta

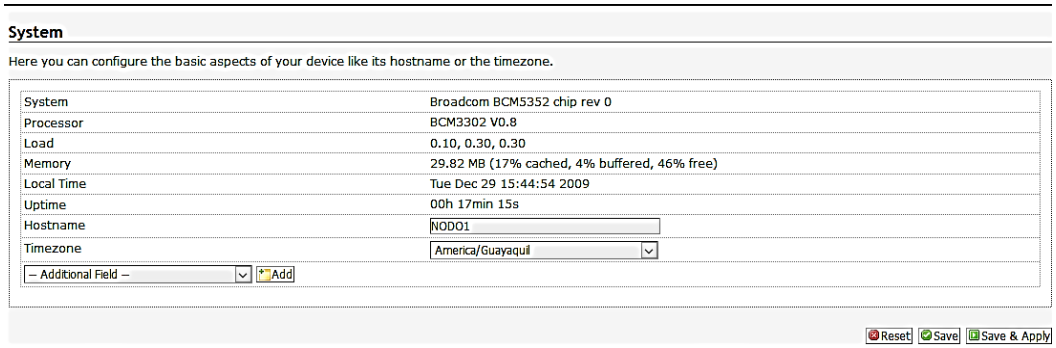

Figura 71. Nombre del router y Timezone
Fuente: Elaboración propia

2. Configuración de la interfaz LAN

- Ingresar a Network/Interfaces/LAN
- Desmarcar la casilla "Bridges Interfaces"
- Cambiar la Ip y la máscara de red

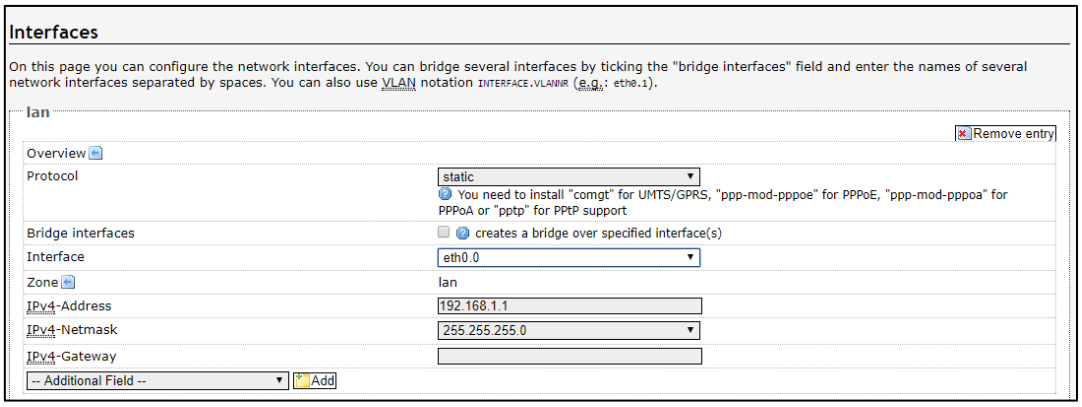

Figura 72. Interfaz LAN configurada
Fuente: Elaboración propia

3. Crear y configurar la interfaz inalámbrica

- Ingresar a Netwok/Interfaz
- Escribir el nombre que se le dará a la interfaz inalámbrica, en este caso "wl0" y dar clic en "Add entry".
- Una vez creada, automáticamente se redirige a una página en donde se deberá asignar la dirección IP y la máscara de red, según el rango de direcciones IP anteriormente mencionadas.

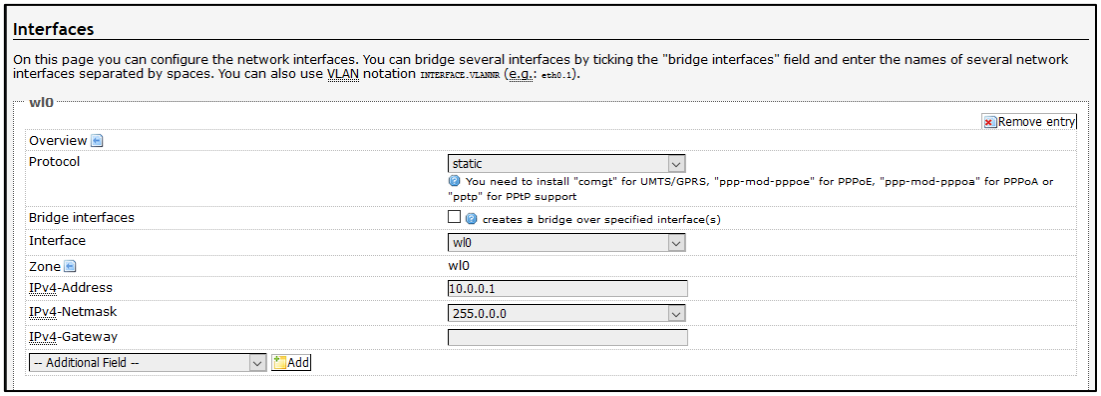

Figura 73. Creación y configuración de la interfaz wlo
Fuente: Elaboración propia

- Ingresar a Network/Interfaces/Wifi
- Llenar los campos como muestra la figura, todos los equipos deberán tener esta configuración

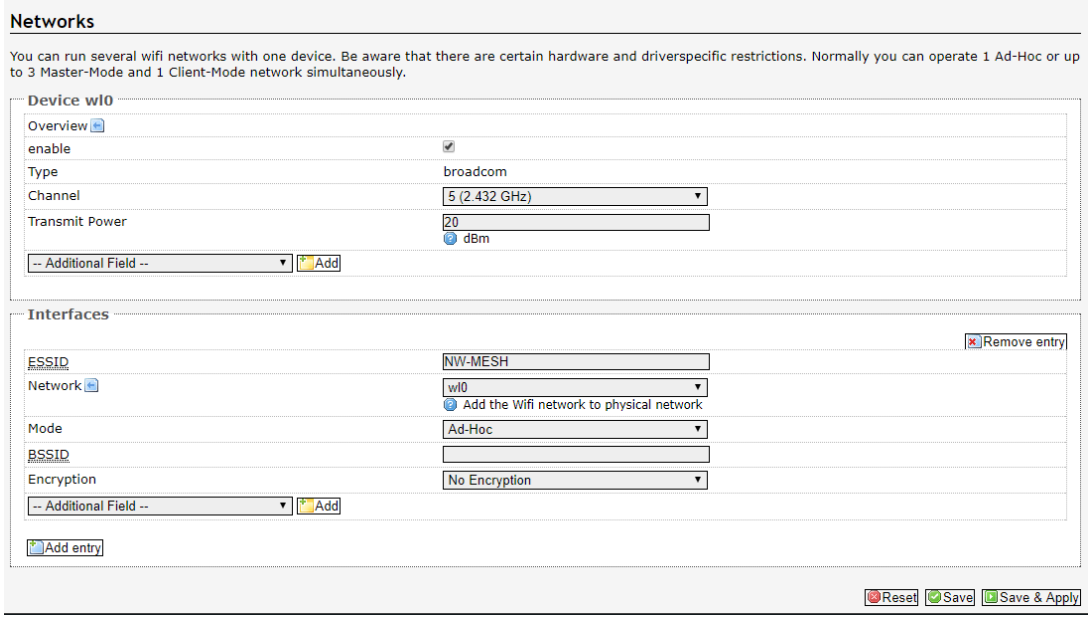

Figura 74. Configuración de la interfaz inalámbrica.
Fuente: Elaboración propia

4. Configuración de OLSR

- Ingresar a Services/OLSR
- Elegir la interfaz en la que correrá OLSR "wl0"

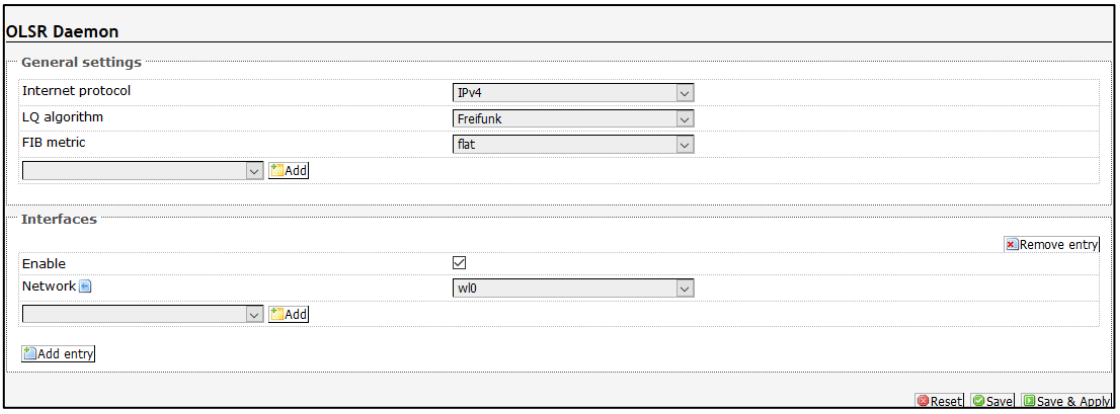

Figura 75. Elección de la interfaz en la que va a correr OLSR
Fuente: Elaboración propia

- Ingresar a Srvices/OLSR/Plugings
- Habilitar todos los plugins instalados marcando "Enable"

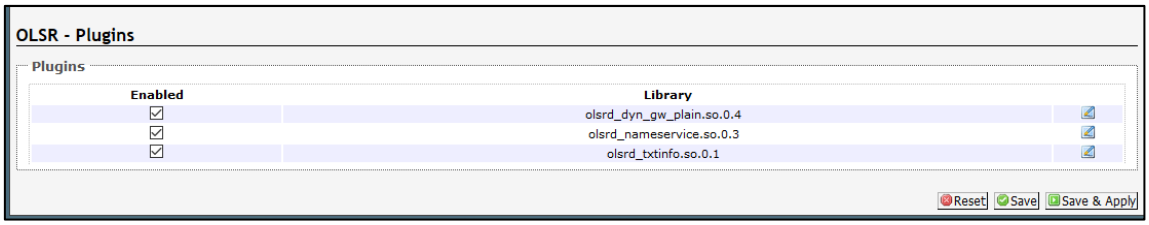

Figura 76. Complementos de OLSR habilitados.
Fuente: Elaboración propia

- Ingresar a Services/HNA-Announcements
- Escribir la dirección IP de la red a la que corresponde la interfaz LAN, de esta manera se permitirá la comunicación entre la interfaz WLAN y la LAN. En caso de que el router tenga acceso a internet se deberá colocar la dirección 0.0.0.0

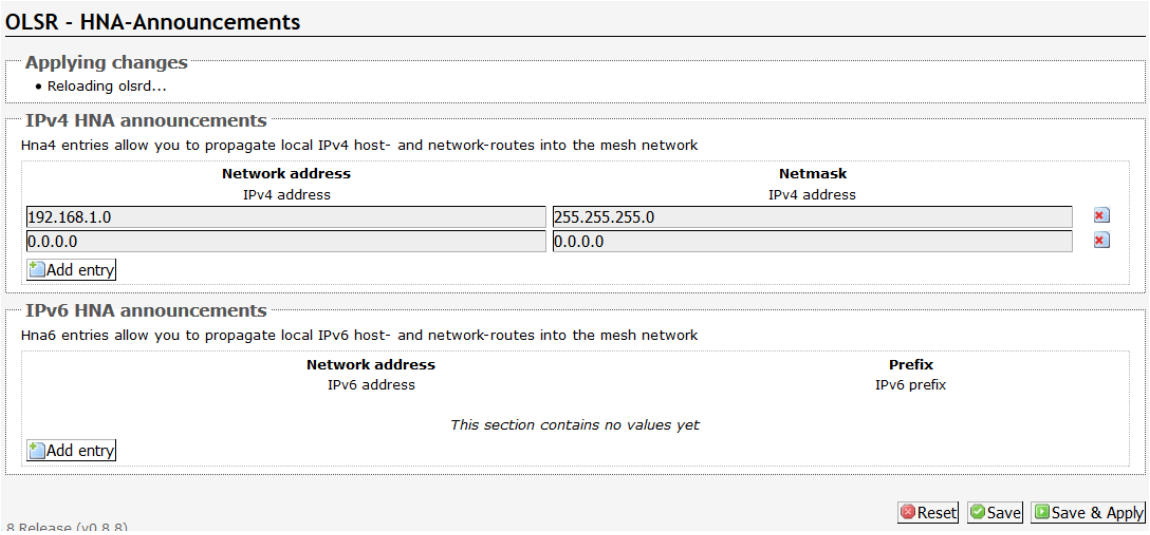

Figura 77. Configuración del parámetro HNA
Fuente: Elaboración propia

Anexo 5. Archivos de Configuración.

A continuación se presenta el resultado final de los archivos de configuración del dispositivo, después de cada cambio realizado

1. /etc./config/system

```
config 'system'
	option 'hostname' 'NODO1'
	option 'zonename' 'America/Guayaquil'
	option 'timezone' 'ECT5'
```

2. etc./config/network

```
config 'switch' 'eth0'
	option 'vlan0' '0 1 2 3 5*'
	option 'vlan1' '4 5'

config 'interface' 'loopback'
	option 'ifname' 'lo'
	option 'proto' 'static'
	option 'ipaddr' '127.0.0.1'
	option 'netmask' '255.255.255.0'

config 'interface' 'lan'
	option 'ifname' 'eth0.0'
	option 'proto' 'static'
	option 'ipaddr' '192.168.1.1'
	option 'netmask' '255.255.255.0'

config 'interface' 'wan'
	option 'ifname' 'eth0.1'
	option 'proto' 'dhcp'

config 'interface' 'wl0'
	option 'proto' 'static'
	option 'ifname' 'wl0'
	option 'ipaddr' '10.0.0.1'
	option 'netmask' '255.255.255.0'
	option 'defaultroute' '0'
	option 'peerdns' '0'
```

3. /etc./config/wireless

```
config 'wifi-device' 'wl0'
	option 'type' 'broadcom'
	option 'channel' '5'
	option 'disabled' '0'
	option 'txpower' '20'
	option 'hmode' 'l1g'

config 'wifi-iface'
	option 'device' 'wl0'
	option 'encryption' 'none'
	option 'ssid' 'NW-MESH'
	option 'network' 'wl0'
	option 'mode' 'adhoc'
```

4. /etc./config/olsrd

```
config 'olsrd'
	option 'IpVersion' '4'
	option 'FIBMetric' 'flat'
	option 'LinkQualityAlgorithm' 'etx_ff'

config 'Interface'
	option 'ignore' '0'
	option 'interface' 'wl0'

config 'LoadPlugin'
	option 'library' 'olsrd_dyn_gw_plain.so.0.4'

config 'LoadPlugin'
	option 'library' 'olsrd_nameservice.so.0.3'
	option 'suffix' '.olsr'
	option 'hosts_file' '/var/etc/hosts.olsr'

config 'LoadPlugin'
	option 'library' 'olsrd_txtinfo.so.0.1'
	option 'accept' '127.0.0.1'

config 'Hna4'
	option 'netaddr' '192.168.30.0'
	option 'netmask' '255.255.255.0'
```

5. /etc./config/firewall

```
config 'defaults'
	option 'syn_flood' '1'
```

```
        option 'input' 'ACCEPT'
        option 'output' 'ACCEPT'
        option 'forward' 'REJECT'

config 'zone'
        option 'name' 'lan'
        option 'input' 'ACCEPT'
        option 'output' 'ACCEPT'
        option 'forward' 'ACCEPT'

config 'zone'
        option 'name' 'wan'
        option 'input' 'REJECT'
        option 'output' 'ACCEPT'
        option 'forward' 'REJECT'
        option 'masq' 'l'
        option 'mtu_fix' 'l'

config 'forwarding'
        option 'src' 'lan'
        option 'dest' 'wan'

config 'rule'
            option 'src' 'wan'
            option 'proto' 'udp'
            option 'dest_port' '68'
            option 'target' 'ACCEPT'

config 'include'
        option 'path' '/etc/firewall.user'

config 'zone'
            ption 'name' 'wl0'
            option 'network' 'wl0'
            option 'input' 'ACCEPT'
            option 'output' 'ACCEPT'
```

6. **/etc./firewall.user**

```
#POLITICAS POR DEFECTO
iptables -F input_rule
iptables -F output_rule
iptables -F forwarding_rule
iptables -t nat -F prerouting_rule
iptables -t nat -F postrouting_rule
iptables -F input_wan
iptables -F forwarding_wan
iptables -t nat -F prerouting_wan
```

```
#ACEPTAR TODO EL TRAFICO TCP PORVENIENTE DEL PUERTO 22
iptables	-A input_wan	-p tcp --dport 22 -j ACCEPT

#
iptables	-A input_wan	-p tcp --dport 1979 -j ACCEPT

#ACEPTAR TODO EL TRAFICO UDP PROVENIENTE DEL PUERTO 698 (puerto
#utilizado por OLSR para el tráfico de control)
iptables -A input_rule -p udp --dport 698 -j ACCEPT

#Las siguientes líneas fueron utilizadas para obtener algunas de las topologías
#propuestas. En este ejemplo solo se aceptará el tráfico del NODO2 y el de los demás se
#rechazará
iptables -A input_rule –s 10.0.0.2 -p udp --dport 698 -j ACCEPT
iptables -A input_rule -p udp --dport 698 -j DROP


#PERMITIR EL REDIRECIIONAMIENTO DE TRÁFICO ENTRE LAS
#INTERFACES
iptables -A forwarding_rule -i $WAN -o $WIFI -j ACCEPT
iptables -A forwarding_rule -i $WIFI -o $WAN -j ACCEPT
iptables -A forwarding_rule -i $LAN -o $WIFI -j ACCEPT
```

Anexo 6. Visualización de la Red

El plugin dot-draw puede ser utilizado en todas las versiones de OpenWRT, para la generación de la imagen se sigue los siguientes pasos

1. **Pre-requisitos**

   1.1. Programas que deben ser instalados en la computadora (Windows)

   - Instalar el programa ImageMagic
   - Instalar el programa Graphviz

   1.2. Complemento instalado en el router

opkg update
opkg install olsrd-mod-dot-draw

   1.3. Asegurarse que el complemento dot-draw esté activado

2. **Configuración de la maquina Windows**

   - Dirigirse a Equipo/Propiedades/Configuración Avanzada del sistema/Variables de entorno
   - En la sección "Variables de usuario" crear una variable con los siguientes campos
     - ✓ Nombre de la variable: PATH
     - ✓ Valor de la variable: se deberá copiar la ruta de la carpeta "bin" generada en el disco local al momento que se instala Graphviz, y aumentar una coma al final.
   - Crear un archivo .txt y ubicarlo en cualquier parte del computador, en este caso se colocará en el escritorio, luego abrir el archivo y colocar las siguiente líneas en el archivo

```
digraph G {
}
```

3. Entrar al router y colocar la siguiente línea de comando:

telnet localhost 2004

4. Copiar la salida de texto que se genera (solo la que se encuentra dentro de los corchetes)

5. Pegar el texto generado dentro de los corchetes del archivo.txt
6. Dirigirse al cmd de Windows y colocar los siguientes comandos

cd Desktop
dot -Tpng archivo.txt -o archivo.png

7. En el escritorio se generará una imagen de topología de la red.

Anexo 7. Archivo .bat

Al ejecutar este archivo, se procederá a guardar en ping.txt el resultado de realizar un ping a una dirección IP. En nuestro caso se debe remplazar la **X.X.X.X** por la dirección IP del router. El archivo ping.txt se guardará en Escritorio. Para cambiar el tamaño de la trama se colocará antes de la dirección IP "**-l 64**" si se desea un trama de 64 portes.

```
@echo off
:bucle


set p1=.
set p2=.


ping -n 1 X.X.X.X | find "Respuesta" > p1.txt
ping -n 1 X.X.X.X | find "Tiempo de espera agotado" > p2.txt
set /p p1=< p1.txt
set /p p2=< p2.txt


echo %time% %p1% %p2%
echo %time% %p1% %p2% >> ping.txt


break >p1.txt
break >p2.txt


ping l27.0.0.1  -w 1000 -n 2 > NUL


GOTO bucle


pause
```

Anexo 8. Configuración de las Extensiones y del Softphone

CREACION DE EXTENSIONES EN ELASTIX.

- ✓ Escribir la dirección IP en una pestaña del navegador
- ✓ Ingresar usuario y contraseña

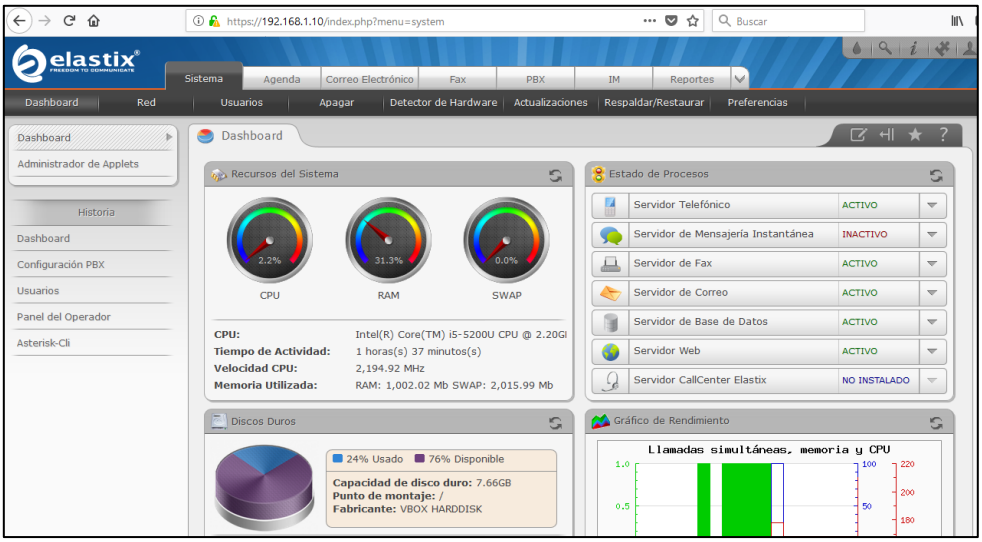

Figura 78. Interfaz gráfica del servidor Elastix.
Fuente: Elaboración propia

- ✓ Ingresar a PBX/Configuración PBX
- ✓ En la sección “Dispositivo” elegir “Dispositivo SIP genérico”
- ✓ Clic en el botón enviar
- ✓ Llenar los siguientes campos:
    - Extensión de usuario: 102
    - Nombre para mostrar: user2
    - Secret: contraseña
- ✓ Clic en enviar

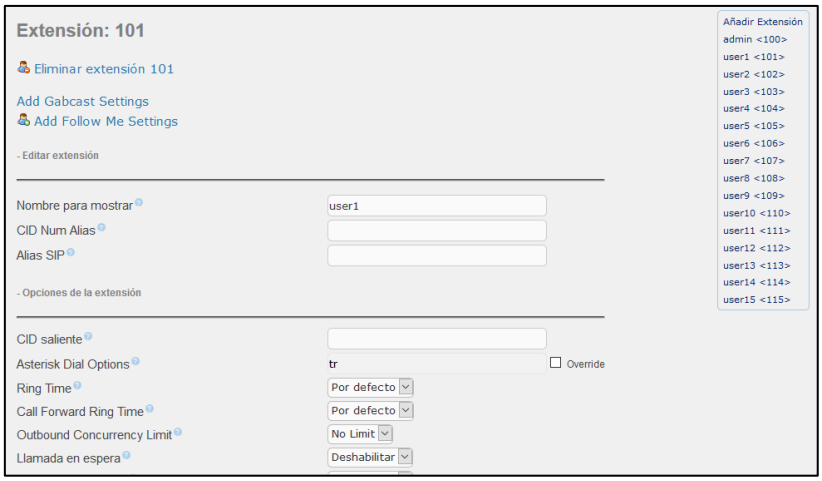

Figura 79. Extensión 101, creada a través de la interfaz web de Elastix
Fuente: Elaboración propia

CONFIGURACIÓN DEL SOFTPHONE PORTGO

Para poder utilizar el softphone debemos ingresar algunos parámetros, en la Figura 80 se indica cómo deben ser configurados los datos en el softphone, y en la Figura 81 se muestra la interfaz que se genere una vez ingresados y verificados los datos.

- ✓ USERNAME: extensión creada en Elastix (desde 102 hasta 115)
- ✓ PASSWORD: contraseña configurada para la extensión
- ✓ SERVER: dirección del servidor de telefonía IP (192.168.1.10)

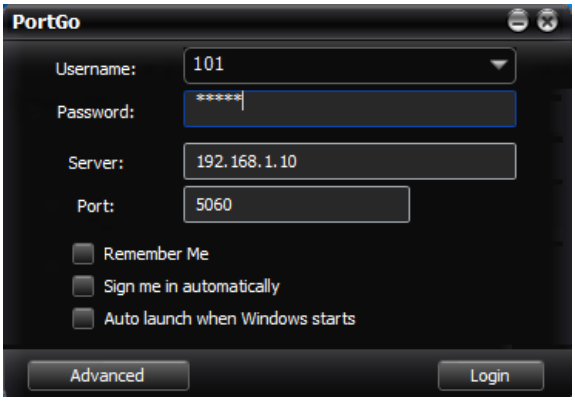

Figura 80. Parámetros configurados en el Softphone Port Go en un PC
Fuente: Elaboración propia

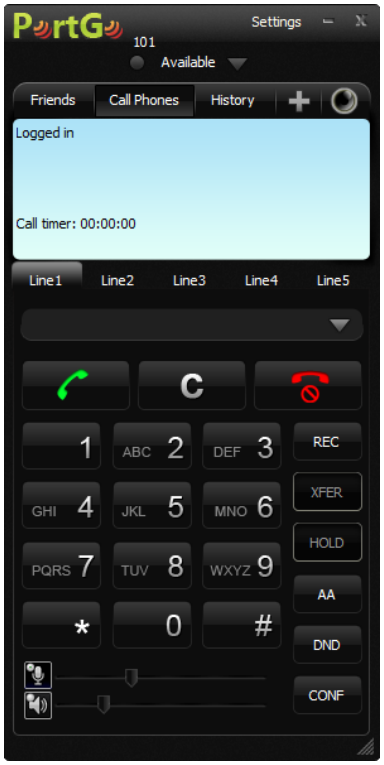

Figura 81. Habilitación del Softphone PortGo

Fuente: Elaboración propia

Anexo 9. Tasa de Transferencias de Diferentes Aplicaciones

Tabla 20. Aplicaciones con sus diferentes tasas de transmisión.

APLICACIONES	REQUISITOS	NOTAS
Mensajes de texto/ IM	< 1 kbps	Como el tráfico es infrecuente y asíncrono, IM tolera latencia alta.
Correo Electrónico	1 a 100 kbps	Igual que con IM, el email es asíncrono e intermitente así que va a tolerar latencia. Archivos anexos grandes, virus, y spam incrementan significativamente el uso de ancho de banda. Note que los servicios de email basados en web (como Yahoo y Hotmail) deben ser considerados como navegación en web y no como email.
Navegación Web	50 - 100+ kbps	Los navegadores web solo usan la red cuando los datos son solicitados. La comunicación es asíncrona, así que una buena cantidad de retardo puede tolerarse. A medida que los navegadores solicitan más datos (imágenes grandes, descargas largas, etc.) el uso de ancho de banda aumenta considerablemente.
Audio en tiempo real (streaming)	96 - 160 kbps	Cada usuario de audio en tiempo real va a usar una cantidad considerable y constante de ancho de banda por todo el tiempo en que esté escuchando. Puede tolerar latencia transitoria utilizando grandes cantidades de memoria del cliente. Pero períodos largos de retardo van a producir "saltos" en el audio o fallos completos de la sesión.
Voz sobre IP (VoIP)	24 - 100+ kbps	Igual que con el audio en tiempo real, la VoIP compromete un ancho de banda constante por cada usuario durante el tiempo de la llamada. Pero con la VoIP, el ancho de banda se usa aproximadamente de manera igual en ambas direcciones. La latencia en una conexión VoIP es una molestia inmediata para los usuarios. Los retrasos más largos de unos pocos milisegundos son inaceptables en VoIP.
Video en tiempo real (streaming)	64 - 200+ kbps	Igual que con el audio en tiempo real, alguna latencia intermitente se puede evitar usando memoria del cliente. El video en tiempo real necesita un caudal elevado y baja latencia para funcionar adecuadamente.
Aplicaciones de compartir archivos (peer-to -peer)	0- infinito Mbps	Mientras que las aplicaciones peer-to-peer toleran cierta latencia, tienden a usar todo el caudal disponible para transmitir datos al mayor número de clientes posible, en el menor tiempo posible. El uso de estas aplicaciones va a ocasionar problemas de latencia y caudal para el resto de los usuarios de la red a menos que usted use un control de tráfico muy cuidadoso

Fuente: Recuperado de Wireless Networking in the Developing World por WNDW Group