

DESARROLLO DE UN PROTOTIPO DE RED MESH BASADA EN  
DISPOSITIVOS RASPBERRY PI PARA LA COMUNICACIÓN EN SITUACIONES  
DE DESASTRES

LIZETH JULIANA PEREZ CHAVES  
SONIA YULIANA RODRIGUEZ AMORTEGUI

UNIVERSIDAD ANTONIO NARIÑO  
FACULTAD DE INGENIERÍA DE SISTEMAS  
PROGRAMA INGENIERÍA DE SISTEMAS  
BOGOTÁ  
2020

DESARROLLO DE UN PROTOTIPO DE RED MESH BASADA EN  
DISPOSITIVOS RASPBERRY PI PARA LA COMUNICACIÓN EN SITUACIONES  
DE DESASTRES

LIZETH JULIANA PEREZ CHAVES

SONIA YULIANA RODRIGUEZ AMORTEGUI

Proyecto de trabajo de grado presentado como requisito para optar al

Título de:

Ingeniero de Sistemas y Computación

Director(a):

MARIA DEL PILAR SALAMANCA

INGENIERA ELECTRICISTA, M.Sc., Ph.D.

UNIVERSIDAD ANTONIO NARIÑO

FACULTAD DE INGENIERÍA DE SISTEMAS

PROGRAMA INGENIERÍA DE SISTEMAS

BOGOTÁ

2020

Nota de Aceptación

---

---

---

Presidente del Jurado

---

Jurado

---

Jurado

---

Bogotá D.C, 08 junio 2020

## ÍNDICE GENERAL

INDICE DE FIGURAS .....	6
ÍNDICE DE TABLAS .....	9
RESUMEN .....	10
INTRODUCCIÓN .....	12
1. PLANTEAMIENTO DEL PROBLEMA .....	14
1.1. DESCRIPCIÓN DEL PROBLEMA .....	14
1.2. FORMULACIÓN DEL PROBLEMA.....	20
1.3. JUSTIFICACIÓN .....	21
1.4 OBJETIVOS.....	23
1.4.1 Objetivo General.....	23
1.4.2 Objetivos Específicos .....	23
1.5 ALCANCE Y LIMITACIONES DEL PROYECTO .....	24
1.5.1 Alcance.....	24
1.5.2 Limitaciones.....	25
2. MARCO DE REFERENCIA .....	27
2.1. MARCO TEÓRICO .....	27
2.1.1. Redes Mesh .....	27
2.1.2. Redes inalámbricas .....	29
2.1.3. Redes WiFi.....	30
2.1.4. Microcomputador Raspberry Pi .....	31
2.1.5. Módulo WiPi .....	34
2.1.6. Linux LEDE.....	35
2.2. ANTECEDENTES Y ESTADO DEL ARTE .....	35
2.2.1. Diseño de un sistema de telecomunicaciones con redes ad hoc de drones como alternativa de medio de comunicación para hacer frente a desastres naturales .....	36
2.2.2. Diseño y despliegue de una red para atención post-desastre basada en UAVs .....	37
2.2.3. Diseño de una red mesh de UAVs para proporcionar servicios de comunicaciones.....	38

2.2.4. Diseño de la red de nueva generación tipo mesh, para la atención y prevención de emergencias en Bogotá, en la banda de 4,9 Ghz, bajo la resolución 1661 de 18 de julio de 2006 para la coordinación de acciones de socorro y salvamento.....	39
3. METODOLOGÍA.....	44
4. DESARROLLO.....	46
4.1.2. Instalación del sistema operativo.....	47
4.2. Configuración de la red mesh.....	51
4.2.1. Descripción de la red.....	51
4.2.2. Configuración de interfaces - Enrutamiento estático.....	52
4.2.2.1. Interfaz LAN.....	52
4.2.2.2. Interfaz DISTRIBUCIÓN.....	54
4.2.2.3. Interfaz EMERGENCIA.....	55
4.2.3. Servicio DHCP- Enrutamiento estático.....	55
4.2.4. Configuración de redes WiFi - Enrutamiento estático.....	56
4.2.5. Conexión entre las Raspberrys.....	60
4.2.6. Configuración de rutas estáticas.....	67
4.2.7. Diagrama de la Red - Enrutamiento estático.....	69
5. RESULTADOS.....	85
5.1. Pruebas de conexión – Enrutamiento estático.....	85
5.2. Pruebas de alcance.....	89
5.3. Pruebas de Conexión – Enrutamiento Dinámico.....	96
5.3.1. Red de Distribución.....	96
5.3.2. Red de Emergencia.....	98
6. CONCLUSIONES.....	103
7. BIBLIOGRAFÍA.....	105

## INDICE DE FIGURAS

<b>Figura 1</b> Porcentaje de área expuesta a movimientos en masa, sismos e inundaciones en Colombia en el año 2010. ....	15
<b>Figura 2</b> Porcentaje de población expuesta a movimientos en masa, sismos e inundaciones en Colombia en el año 2010. ....	16
<b>Figura 3</b> Distribución porcentual de las pérdidas humanas en Colombia, según el tipo de evento que las ocasiona (1970-2011) .....	17
<b>Figura 4</b> Distribución porcentual de las viviendas destruidas en Colombia, según el tipo de evento que las ocasiona (1970-2011). ....	18
<b>Figura 5</b> Redes Mesh.....	28
<b>Figura 6</b> Clasificación de las redes inalámbricas según el área de cobertura.....	30
<b>Figura 7</b> Microcomputador Raspberry Pi.....	34
<b>Figura 8</b> Módulo WiPi.....	35
<b>Figura 9</b> Arquitectura del sistema de comunicación inalámbrico propuesto en (Marocho, 2017).....	37
<b>Figura 10</b> Modelo de red basada en UAVs propuesta en (Panda, Das, Sen, & Arif I, 2019).....	38
<b>Figura 11</b> Red mesh de UAVs propuesta en (Rosa, 2017). ....	39
<b>Figura 12</b> Propuesta de diseño de red Ad Hoc Mesh para su aplicación en Telemedicina y Desastres Naturales en zona rural de Nicaragua (Morales, Matus, & Arias, 2016) .....	41
<b>Figura 13</b> Kit de Raspberry PI 3 modelo B+ marca Canakit.....	46
<b>Figura 14</b> Terminal del sistema operativo OpenWRT .....	49
<b>Figura 15</b> Interfaz de búsqueda de controladores.....	50
<b>Figura 16</b> Controladores Ralink para instalar.....	50
<b>Figura 17</b> Interconexión de dispositivos Raspberry.....	51
<b>Figura 18</b> Interfaz LuCI .....	53
<b>Figura 19</b> Configuración interfaz LAN en RaspberryA, RaspberryB y RaspberryC .....	54
<b>Figura 20</b> Configuración DHCP desde LuCI en red EMERGENCIA .....	55
<b>Figura 21</b> Configuración de redes WiFi.....	56
<b>Figura 22</b> Configuración redes WiFi desde LuCI, (a) RaspberryA, (b) RaspberryB y (c) RaspberryC.....	59
<b>Figura 23</b> Conexión entre Raspberry A y B.....	61
<b>Figura 24</b> Redes disponibles Raspberry A.....	61
<b>Figura 25</b> Configuración IP fija interfaz WWAN.....	62

<b>Figura 26</b> Conexión entre Raspberry A y C .....	62
<b>Figura 27</b> Redes disponibles RaspberryC.....	62
<b>Figura 28</b> Configuración IP fija interfaz WWAN2.....	63
<b>Figura 29</b> Conexión entre Raspberry B y C .....	63
<b>Figura 30</b> Redes disponibles RaspberryC.....	64
<b>Figura 31</b> Configuración de interfaz fija interfaz WWAN .....	64
<b>Figura 32</b> Estaciones asociadas en (a) RaspberryA, (b) RaspberryB y (c)RaspberryC.....	64
<b>Figura 33</b> Interfaces en (a) RaspberryA, (b) RaspberryB y (c) RaspberryC. ....	65
<b>Figura 34</b> Diagrama de la Red Mesh – Enrutamiento estático.....	69
<b>Figura 35</b> Pestaña Services en LuCI.....	71
<b>Figura 36</b> Habilitación de OLSR desde la interfaz web .....	71
<b>Figura 37</b> Wireless de RaspberryA desde LuCI .....	73
<b>Figura 38</b> Network de RaspberryA desde LuCI.....	75
<b>Figura 39</b> Wireless de RaspberryB desde LuCI .....	77
<b>Figura 40</b> Network de RaspberryB desde LuCI.....	78
<b>Figura 41</b> Wireless de RaspberryC desde LuCI.....	80
<b>Figura 42</b> Network de RaspberryC desde LuCI.....	81
<b>Figura 43</b> Configuración interfaces OLSR.....	82
<b>Figura 44</b> Interface mesh .....	83
<b>Figura 45</b> Conexión entre RaspberryA, RaspberryB y RaspberryC, red de emergencia y red de distribución .....	84
<b>Figura 46</b> Ping y Traza desde un dispositivo conectado a la RaspberryA a un dispositivo conectado a la red de acceso de la RaspberryB y RaspberryC. ....	86
<b>Figura 47</b> Ping y Traza desde un dispositivo conectado a la RaspberryB a un dispositivo conectado a la red de acceso de la RaspberryA y RaspberryC. ....	87
<b>Figura 48</b> Ping y Traza desde un dispositivo conectado a la RaspberryC a un dispositivo conectado a la red de acceso de la RaspberryA y RaspberryB. ....	88
<b>Figura 49</b> Espacio de pruebas de alcance .....	90
<b>Figura 50</b> Diagrama del promedio de paquetes perdidos variando distancia y tamaño en bytes .....	92
<b>Figura 51</b> Diagrama de cajas y bigotes del RTT en función del tamaño de paquetes en bytes a una distancia de separación de (a) 10 metros, (b) 15 metros y (c) 20 metros.....	93
<b>Figura 54</b> Pings desde la RaspberryA hacían las interfaces mesh de la RaspberryB y RaspberryC .....	96
<b>Figura 55</b> Pings desde la RaspberryB hacían las interfaces mesh de la RaspberryA y RaspberryC .....	97

<b>Figura 56</b> Pings desde la RaspberryC hacían las interfaces mesh de la RaspberryA y RaspberryB .....	97
<b>Figura 57</b> Ping desde un dispositivo conectado a la RaspberryA hacia un dispositivo conectado a la RaspberryB (izquierda) y RaspberryC (derecha) .....	98
<b>Figura 58</b> Ping desde de la RaspberryB hacia un dispositivo conectado a la RaspberryA y un dispositivo conectado a la RaspberryC .....	99
<b>Figura 59</b> Ping desde RaspberryC hacia un dispositivo conectado a la RaspberryA y un dispositivo conectado a la RaspberryB .....	100
<b>Figura 60</b> Rutas con las 3 Raspberrys encendidas.....	101
<b>Figura 61</b> Apagando una de las Raspberrys.....	102
<b>Figura 62</b> Rutas con dos Raspberrys encendidas.....	102

## ÍNDICE DE TABLAS

<b>Tabla 1</b> Comparación entre los modelos de Raspberry Pi 3B y 3B+ .....	33
<b>Tabla 2</b> Comparativo entre proyectos similares.....	42
<b>Tabla 3</b> Configuración DHCP de la red de EMERGENCIA .....	56
<b>Tabla 4</b> Configuración de rutas estáticas RaspberryA.....	67
<b>Tabla 5</b> Configuración de rutas estáticas RaspberryB.....	68
<b>Tabla 6</b> Configuración rutas estáticas RaspberryB.....	68
<b>Tabla 7</b> Promedio de paquetes perdidos variando distancia y tamaño en bytes ...	91
<b>Tabla 8</b> Calculo del Jitter .....	95
<b>Tabla 9</b> Tasa de Transmisión .....	95

## RESUMEN

El proyecto de investigación “Sistema de comunicación para sobrevivientes de un desastre basado en una red ad hoc de teléfonos inteligentes” tiene como uno de sus objetivos el desarrollo de una aplicación prototipo para dispositivos móviles con sistema operativo Android, que permita intercambiar información entre sobrevivientes de un desastre (Castañeda, 2019). El escenario para el cual se ha previsto el uso de esta aplicación es aquel en el cual, después de una catástrofe, las redes móviles han quedado fuera de servicio, ya sea porque colapsaron debido a la congestión generada por los usuarios que desean comunicarse a la mayor brevedad con sus seres queridos, o por la destrucción de la infraestructura de la red.

Aunque la aplicación podría ser el primer recurso (y posiblemente, el único) del que disponen los sobrevivientes inmediatamente después del desastre, es posible ofrecer una alternativa de conectividad de mayor alcance que también emplee los teléfonos inteligentes de los sobrevivientes. Esta alternativa consiste en el despliegue de una red *mesh* (en malla) de infraestructura sencilla y económica a la que se conectarán los teléfonos inteligentes de los sobrevivientes como si fuera una red WiFi común. La red mesh utiliza microcomputadores Raspberry Pi, y la conectividad entre estos se establece mediante módulos WiFi de tipo WiPi, que son adaptadores inalámbricos para los dispositivos Raspberry Pi, compatibles con el estándar IEEE 802.11n, por lo cual soportan tasas de transmisión de información hasta de 150 Mbps.

Con esta red mesh, y dependiendo de la cantidad de dispositivos Raspberry Pi que la conformen, se podría ampliar significativamente el cubrimiento de la zona afectada por un desastre, comparada con la red generada únicamente por teléfonos inteligentes. Inclusive, los miembros de los equipos de rescate se podrían conectar a ella para localizar y establecer contacto con los sobrevivientes de una manera más ágil. La red puede desplegarse sobre la superficie terrestre (siempre que las condiciones después del desastre así lo permitan), o pueden utilizarse drones que carguen a los dispositivos Raspberry Pi cuando sea imposible instalarla directamente en terreno.

Este trabajo de grado propone el desarrollo de un prototipo de la red mesh descrita, conformada por 3 dispositivos Raspberry Pi 3B+. Aunque ya se han implementado redes mesh semejantes, también con propósitos humanitarios, el conocimiento y la experiencia que se adquieran en este proceso serán de gran valor para el equipo de investigación y para la formulación de una propuesta para la segunda fase del proyecto de investigación.

## INTRODUCCIÓN

Existe una amplia variedad de desastres naturales ocasionados por la naturaleza, entre los cuales se pueden reconocer algunos como: terremotos, inundaciones, incendios, derrumbes, avalanchas, tifones, erupciones de volcanes, entre otros. En el caso de los fenómenos atmosféricos, por ejemplo, los estudios de meteorología pueden llegar a predecir un evento con un alto grado de precisión, sin embargo, el fenómeno puede ser de tal magnitud que los esfuerzos para mitigar las consecuencias podrían resultar insuficientes. En cuanto a terremotos, la predicción es aún muy difícil de lograr, lo cual deja aún menos posibilidad para actuar en el momento de la catástrofe.

En la actualidad, y siempre que se cuente con acceso a Internet y a las redes de telefonía, mediante los teléfonos inteligentes se hace más fácil la búsqueda de los sobrevivientes cuando ocurre un desastre, pues estos cuentan con aplicaciones que permiten enviar mensajes de texto, realizar llamadas, video llamadas o enviar la ubicación actual del móvil. No obstante, es claro que no siempre se va a contar con acceso a las redes de comunicaciones debido a la pérdida de la infraestructura o a la congestión de las redes, lo que dificulta la actividad de búsqueda de las personas de las que no se tiene conocimiento del estado en el que se encuentran. Estos fenómenos se presentan en cualquier momento, sin permitir una reacción oportuna para reunirse con las personas que se encuentren más cerca o con los familiares.

Las pérdidas materiales y humanas que se presentan en estos hechos pueden ser bastantes grandes.

Con este proyecto de investigación se busca aumentar la posibilidad de salvar vidas. Sabemos que en una primera instancia se podría hacer uso de una aplicación móvil que permita intercambiar información entre los sobrevivientes. Aunque la aplicación puede ser el primer recurso, es posible ofrecer una alternativa de conectividad de mayor alcance como la red mesh que se propone en este trabajo de grado, a la cual se podrían conectar los teléfonos inteligentes de los sobrevivientes, e inclusive de los miembros de los equipos de rescate, para poder localizar y establecer contacto con las personas que se encuentren dentro del área de cobertura.

## 1. PLANTEAMIENTO DEL PROBLEMA

### 1.1. DESCRIPCIÓN DEL PROBLEMA

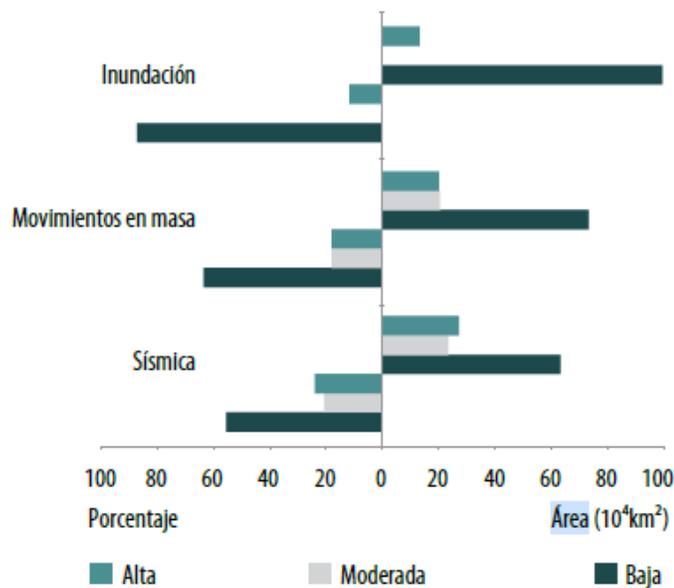
Cuando ocurre un desastre natural, es posible que la infraestructura de comunicaciones se pierda totalmente o que colapse debido a la congestión causada por las múltiples llamadas o mensajes de los sobrevivientes hacia sus seres queridos. Además, es conocido que la probabilidad de encontrar personas vivas se reduce drásticamente después de las 72 horas posteriores a la ocurrencia de una catástrofe (más conocidas como *72 horas doradas*). En el periódico Mexicano Milenio Digital, el doctor Manuel Bárcenas menciona que después de una persona encontrarse atrapada, debe afrontar situaciones como aplastamiento de órganos y es aún menos probable que resista si no es rescatada con prontitud (Milenio digital, 2017).

Según el Departamento Nacional de Planeación (DNP), entre los años 2006 y 2014 se presentaron 3.181 muertos y 12,3 millones de afectados en desastres naturales, información que fue obtenida de los reportes dados por Emergencias de la Unidad Nacional para la Gestión del Riesgo de Desastres (UNGRD) (DNP, 2015). Los tipos de eventos que se presentan en las distintas regiones del país son: avalancha, creciente súbita, deslizamiento, granizada, inundación, marejada, tormenta eléctrica, vendaval, entre otros.

En la **Figura 1** se muestra el porcentaje del área expuesta en Colombia según tres eventos: inundación, sismos y movimientos en masa. Para cada tipo de evento, se

puede observar el nivel de amenaza, que se clasifica en escala alta, media y baja, donde se evidencia que existe un mayor porcentaje de área expuesta a actividad sísmica de amenaza alta y moderada, en tanto que más del 80% del área está expuesta a inundaciones de baja amenaza.

**Figura 1** Porcentaje de área expuesta a movimientos en masa, sismos e inundaciones en Colombia en el año 2010.

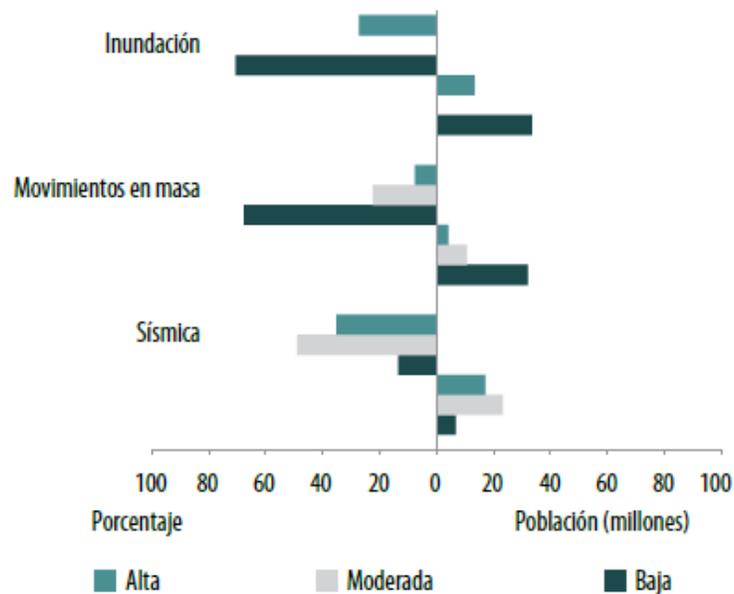


Banco Mundial. (2012). Recuperado de:  
<http://gestiondelriesgo.gov.co/sigpad/archivos/GESTIONDELRIESGOWEB.pdf>

La **Figura 2** representa el porcentaje de la población expuesta a los tres eventos mencionados anteriormente, donde se observa que en cuanto a las inundaciones y movimientos en masa en nivel de amenaza bajo, la población expuesta es mayor que la correspondiente a movimientos sísmicos de esta misma escala. Por otro lado,

el mayor porcentaje de población expuesta a amenazas de alto nivel es por actividad sísmica. De acuerdo con estas figuras, se puede concluir que el 86% de la población colombiana está expuesta a una amenaza sísmica alta y media, el 28% a un alto potencial de inundación y el 31% a una amenaza alta y media por movimientos en masa.

**Figura 2** Porcentaje de población expuesta a movimientos en masa, sismos e inundaciones en Colombia en el año 2010.

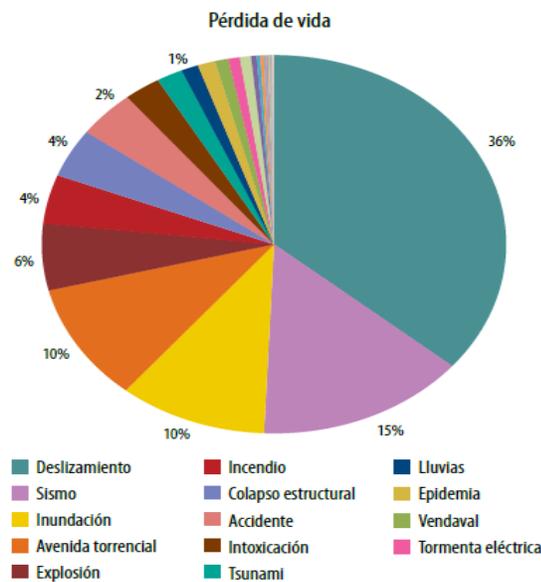


Banco Mundial. (2012). Recuperado de:  
<http://gestiondelriesgo.gov.co/sigpad/archivos/GESTIONDELRIESGOWEB.pdf>

Los eventos traen consigo muchos afectados, como se puede observar en la **Figura 3**. En esta se muestra la distribución porcentual de las pérdidas humanas en Colombia entre 1970 y 2011, con un 36% de las pérdidas debidas a los deslizamientos, seguido por los sismos con un 15%. La **Figura 4** muestra el

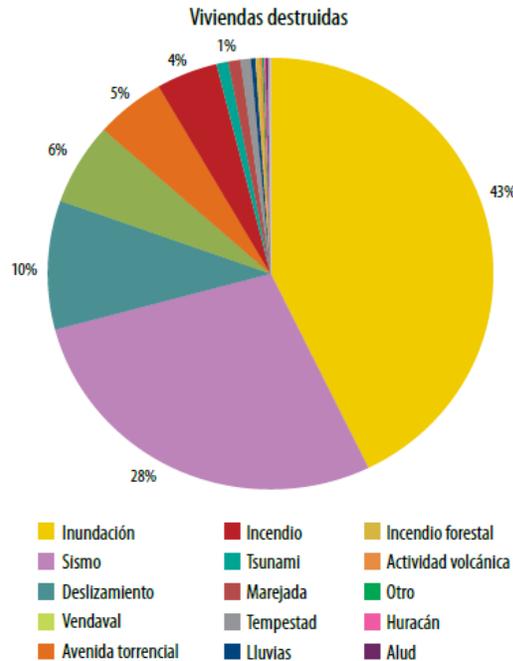
porcentaje de viviendas destruidas en el mismo periodo de tiempo, con un 43% debido a las inundaciones y un 28% ocasionada por los sismos. De acuerdo al tipo de evento, los fenómenos de tipo hidrometeorológico (inundaciones, lluvias, tormentas eléctricas, etc.) se caracterizan por generar impactos más localizados, pero de mayor frecuencia, los cuales en el transcurso del tiempo generan pérdidas mayores a las ocasionadas por eventos geológicos. En síntesis, las inundaciones han ocasionado el 43% de las viviendas destruidas y el 10% de las pérdidas de vidas, en tanto los deslizamientos han causado el 10% de las casas destruidas y el 36% de las muertes durante el lapso de tiempo estudiado (Banco Mundial, Colombia 2012).

**Figura 3** Distribución porcentual de las pérdidas humanas en Colombia, según el tipo de evento que las ocasiona (1970-2011)



Banco Mundial. (2012). Recuperado de: <http://gestiondelriesgo.gov.co/sigpad/archivos/GESTIONDELRIESGOWEB.pdf>

**Figura 4** Distribución porcentual de las viviendas destruidas en Colombia, según el tipo de evento que las ocasiona (1970-2011).



**Banco Mundial. (2012). Recuperado de:**  
<http://gestiondelriesgo.gov.co/sigpad/archivos/GESTIONDELRIESGOWEB.pdf>

Por supuesto, los eventos desastrosos también pueden afectar significativamente toda la infraestructura de la zona afectada, no solo las obras civiles como puentes o carreteras, también las antenas y estaciones base que hacen parte de las redes móviles pueden quedar fuera de servicio. En la ciudad de Nepal (India) en 2015, por ejemplo, ocurrió un terremoto con intensidad 7,8 en la escala de Richter, donde hubo aproximadamente 4300 muertos y 8000 heridos. En el diario El Mundo de España, se reportó lo siguiente:

“Paralelamente, la réplica de 6,7 grados en la escala de Richter del domingo, una de las más de 40 que se han producido desde el sábado, terminaron de tumbar gran parte de las ya golpeadas comunicaciones en el país. En este momento los servicios de internet y telefonía móvil están caídos en la nación asiática, sólo algunas líneas de telefonía fija están activas” (El mundo, 2015, p.1).

Otro caso semejante es el del terremoto de intensidad 8 ocurrido en China en el año 2008, donde el número de muertos se aproximó a 8.500 y hubo más de 10.000 heridos (El Mundo, 2008). Con relación a esta catástrofe, el periódico El Mundo comunicó:

“El sismo se produjo a las 06:28 horas GMT y tuvo una fuerza de 8,0 grados en la escala de Richter, se sintió hasta en Bangkok y Pekín. Cientos de miles de habitantes de la provincia de Sichuan, la más afectada, se encuentran incomunicados. El ejército ha enviado a miles de soldados, helicópteros y equipos de rescate con perros rastreadores a la zona afectada. Se teme que decenas de miles de personas se hayan quedado sin hogar. Además, las lluvias dificultan las tareas de rescate y las comunicaciones telefónicas han quedado destruidas” (El Mundo, 2008).

A pesar de las fallas y de la congestión en las redes de comunicaciones que pueden ocurrir después de una catástrofe, los teléfonos inteligentes y dispositivos como portátiles o tabletas, ofrecen otras alternativas de conexión, por ejemplo, a redes WiFi. Esta tecnología podría ser aprovechada para lograr una comunicación de

alcance local que permita la búsqueda de sobrevivientes. En el caso de los teléfonos inteligentes, una opción así aprovecharía la enorme popularidad que han alcanzado en todo el mundo. Según el informe The Mobile Economy GSMA (GSMA, 2019), en 2018 se superaron los 5.100 millones de personas conectadas a los servicios móviles, esto significa que alrededor de un 67% de las personas en el mundo cuentan con un teléfono inteligente. Se estima que en el 2025 los suscriptores llegarán a 5.900 millones de personas, equivalentes a un 71% de la población mundial.

Considerando el impacto que puede tener un desastre en la infraestructura de telecomunicaciones y teniendo en cuenta la gran popularidad que han alcanzado los teléfonos inteligentes en los últimos años, en este trabajo de grado se propone diseñar un prototipo de red de comunicaciones que pueda desplegarse con rapidez y que permita la conexión entre los sobrevivientes y rescatistas a través de sus teléfonos inteligentes para facilitar la tarea de rescatar personas vivas.

## **1.2. FORMULACIÓN DEL PROBLEMA**

Ante un desastre natural donde las redes móviles se han quedado fuera de servicio o se encuentran congestionadas por la cantidad de tráfico en ellas, ¿cómo se podría implementar una red alternativa de comunicación entre los sobrevivientes, mediante la cual se puedan aprovechar los teléfonos inteligentes que estén disponibles en la zona afectada?

### 1.3. JUSTIFICACIÓN

Los desastres naturales son fenómenos que pueden ocurrir en cualquier parte del mundo, generalmente de gran impacto en la población y de difícil predicción. En Colombia, específicamente, los desastres naturales han significado numerosas pérdidas humanas y materiales. Según las estadísticas disponibles, entre 1970 y 2011, los desastres que más pérdidas humanas han ocasionado son los deslizamientos, con el 36% del total, en tanto que las inundaciones han producido el mayor porcentaje de viviendas destruidas, equivalente al 48% del total. El problema es tan complejo que el 61% de la población colombiana está expuesta a desastres como deslizamientos, sismos e inundaciones.

Los desastres naturales además pueden generar afectaciones en la infraestructura de comunicaciones, debido a daños físicos y falta de suministro de energía. Inclusive, la congestión ocasionada por el afán de los sobrevivientes por comunicarse con sus seres queridos, trae como consecuencia la indisponibilidad de la red. Sin embargo, los sobrevivientes que tienen a mano un teléfono inteligente con batería disponible, tienen un medio de comunicación que les podría facilitar la colaboración entre sí.

A pesar de no estar conectados a una red de telefonía celular, los teléfonos inteligentes pueden vincularse a redes locales que faciliten la comunicación entre sus usuarios. Una red mesh es una opción de bajo costo que puede ofrecer esa posibilidad. La red mesh que se propone en este trabajo de grado está conformada por un conjunto de dispositivos Raspberry Pi conectados mediante enlaces WiFi, y

podría ser desplegada por los equipos de rescate. Mediante la red mesh se conformaría una especie de *backbone* o espina dorsal de comunicaciones a la que se vincularían los teléfonos inteligentes de los sobrevivientes en el área afectada. De esta manera, se genera una red de bajo costo dado que cada Raspberry Pi cuesta aproximadamente US\$40, y se aprovecharía la gran popularidad de los teléfonos inteligentes: en el año 2018, en Colombia, el 55% de la población poseía un teléfono de estas características (GSMA, 2019), mientras que a nivel mundial alcanzó el 67%. El alcance de la red dependerá del número de dispositivos Raspberry que la conformen, y por limitaciones en el presupuesto, en este trabajo de grado serán tres.

De esta manera, mediante la red mesh se espera aumentar las probabilidades de salvar vidas, utilizando una red de bajo costo, de fácil despliegue y aprovechando los teléfonos inteligentes que los sobrevivientes tengan disponibles.

Si bien la cobertura de la red varía dependiendo del número de nodos que conforman la red mesh, por limitaciones de este trabajo de grado, no es posible conocer la distancia máxima con la que se contaría para rescatar y comunicarse con personas, sin embargo, se evaluaría el alcance en un espacio limitado. Por supuesto, al ser una red terrestre, su alcance está limitada por los obstáculos que se encuentren entre los nodos que hacen parte de ella.

Es importante destacar que se propone el uso de dispositivos Raspberry Pi para interconectar la red mesh en lugar de puntos de acceso inalámbricos, por ser más económicas que estos y para poder mitigar problemas de configuración, ya que

cada una se puede configurar con varios adaptadores WiFi, y se puede evaluar su desempeño con diferentes tipos de enrutamiento. En los puntos de acceso, estos aspectos son limitados y pueden implicar un aumento en su costo.

Otro de los beneficios de este trabajo de grado, es el aporte que se genera al grupo de Investigación LACSER de la Facultad de Ingeniería de Sistemas de la Universidad Antonio Nariño, ya que este es el primer trabajo de grado en la facultad que hace uso de dispositivos Raspberry Pi y redes mesh, fortaleciendo así la línea de investigación en redes y seguridad.

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo General**

Desarrollar un prototipo de una red mesh basada en dispositivos Raspberry Pi, que conformen una red de fácil implementación y bajo costo, útil para comunicar a sobrevivientes de un desastre mediante sus teléfonos inteligentes.

### **1.4.2 Objetivos Específicos**

- Conocer los aspectos básicos de configuración de los dispositivos Raspberry mediante la puesta en marcha del sistema operativo Raspbian, con el fin de familiarizarse con la conexión de periféricos, el proceso de instalación del sistema operativo y los diferentes puertos que ofrecen estos dispositivos.

- Instalar el sistema operativo Linux LEDE en los dispositivos Raspberry, con el propósito de facilitar la interconexión de estos en la red mesh y ampliar las alternativas en cuanto a protocolos de enrutamiento disponibles.
- Configurar la red mesh utilizando inicialmente enrutamiento estático y posteriormente enrutamiento dinámico, con el propósito de identificar el esquema de enrutamiento que se puede implementar con mayor facilidad y que ofrezca mayor versatilidad al personal que vaya a desplegar la red.

## **1.5 ALCANCE Y LIMITACIONES DEL PROYECTO**

### **1.5.1 Alcance**

La red mesh propuesta en este trabajo de grado tendrá las siguientes características:

- Se utilizarán 3 dispositivos Raspberry Pi 3B+ conectados mediante enlaces WiFi comunes.
- Cada dispositivo Raspberry Pi, a su vez, funcionará como un punto de acceso al que se podrán conectar los teléfonos inteligentes que se encuentren dentro de su radio de cobertura.
- La red mesh permitirá la comunicación entre los teléfonos inteligentes que se encuentren conectados a cada uno de los tres puntos de acceso que la conforman. También se podrán conectar a la red otros dispositivos como computadores portátiles o tabletas, sin embargo, se hace énfasis en los teléfonos inteligentes dada su gran popularidad.

- El enrutamiento en la red mesh podrá hacerse estáticamente o mediante protocolos de enrutamiento compatibles con el sistema operativo que se decida instalar en los dispositivos Raspberry Pi.
- El hardware necesario para adelantar este proyecto será provisto por el proyecto de investigación.
- El prototipo permitirá el intercambio de tráfico entre los móviles conectados a la red mesh y no requiere de acceso a internet para funcionar.

### **1.5.2 Limitaciones**

La red mesh propuesta en este trabajo de grado tendrá las siguientes limitaciones:

- Se utilizarán los modelos 3B+ de Raspberry Pi. Aunque la versión 4 fue liberada recientemente, aún no es clara su compatibilidad con el sistema operativo que se va a implementar en estos dispositivos.
- Los obstáculos entre dispositivos pueden reducir el alcance de la red y su desempeño.
- Debido a las restricciones para la libre circulación que se decretaron a nivel nacional a causa de la pandemia, las pruebas de la red se limitan al interior de una casa, en la que la máxima distancia de separación posible entre Raspberrys es de 20 metros.
- La cantidad máxima de dispositivos móviles que se puedan conectar a la red mesh no se podrá establecer ya que no se cuenta con un campo abierto disponible donde realizar las pruebas ni es posible convocar al equipo del

proyecto de investigación para llevarlas a cabo, debido a las restricciones establecidas por causa de la cuarentena.

- Debido a las limitaciones del presupuesto disponible, únicamente se utilizarán 3 Raspberry Pi.
- Este trabajo de grado no cubre el desarrollo de la aplicación para comunicar a los teléfonos inteligentes sobre la red mesh.

## 2. MARCO DE REFERENCIA

### 2.1. MARCO TEÓRICO

A continuación, se explican los conceptos generales que se deben conocer para contextualizar este trabajo de grado.

#### 2.1.1. Redes Mesh

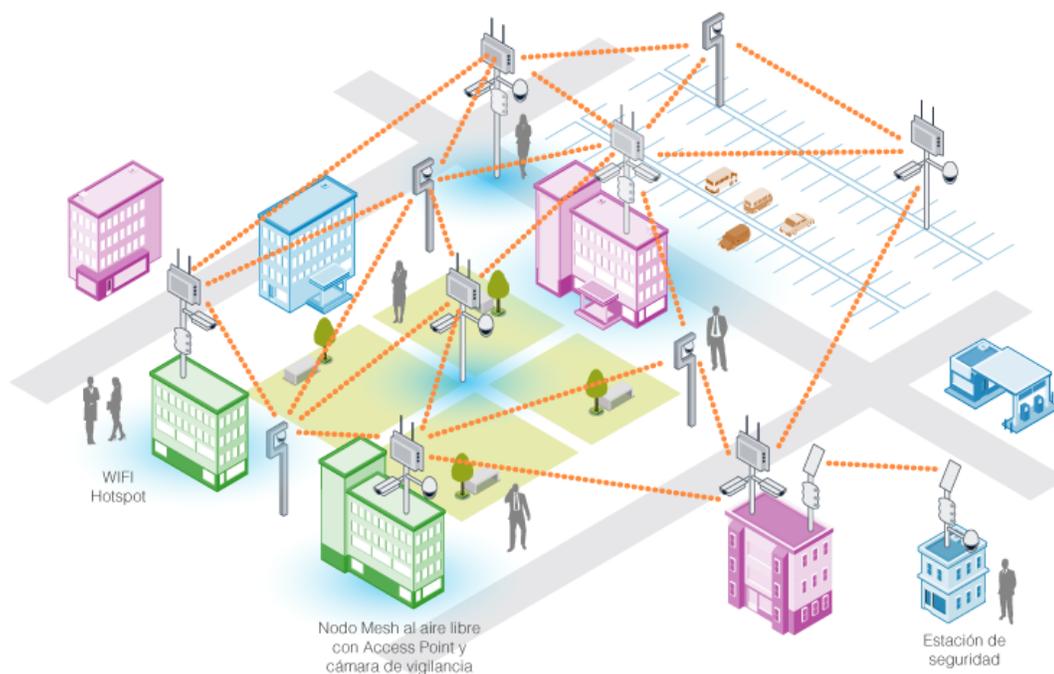
Para definir una red *mesh* es necesario partir del concepto de red *ad hoc*. Una red *ad hoc* es una agrupación de nodos auto-organizados que se comunican entre sí sin requerir de infraestructura. Las redes *ad hoc* de un único salto son la forma más simple y se generan interconectando dispositivos que se encuentran dentro de su mismo radio de transmisión. En las redes *ad hoc* multisalto, en cambio, los nodos deben cooperar entre sí para lograr las funcionalidades que usualmente provee la infraestructura, principalmente el enrutamiento. Sin embargo, la operación de las redes *ad hoc* puras involucra una gran complejidad, debido al movimiento de los nodos sin una trayectoria conocida, las restricciones de energía en las baterías de los nodos, la desaparición inesperada de un dispositivo, etc. Esto ha traído como consecuencia que las redes *ad hoc* puras se hayan desarrollado principalmente en el ámbito académico y su desempeño se ha evaluado especialmente a partir de simulaciones por computador (Basagni, 2013).

Las redes *mesh* se apoyan en el concepto de red *ad hoc* pero, a diferencia de estas, poseen una columna vertebral fija (más conocida como *backbone*) que reduce su complejidad de implementación. El *backbone* provee una infraestructura de

enrutamiento que no requiere la cooperación entre los nodos de los usuarios, reduce el impacto producido por la movilidad de estos y hace que las restricciones de energía en el diseño de los protocolos sean menos exigentes.

Al contrario de las redes ad hoc, desde un comienzo la investigación en redes mesh se enfocó en aspectos prácticos: implementación, integración y experimentación. Las redes mesh se desarrollaron para comunicar comunidades mediante enlaces del tipo IEEE 802.11 con el fin de compartir archivos, establecer conexiones telefónicas entre sí mediante Voz sobre IP y para compartir el acceso a Internet. (Basagni, 2013). En la **Figura 5**, podemos ver un ejemplo de un despliegue de una red mesh.

**Figura 5** Redes Mesh



DMS (2016). Red Mesh. Figura 5. Recuperado de: <https://dms.com.pe>

### 2.1.2. Redes inalámbricas

Las redes inalámbricas son redes que utilizan ondas de radio para conectar los dispositivos, sin la necesidad de usar cables de ningún tipo (Salazar, s.f). Hoy en día, este tipo de red se ha popularizado por el gran uso que tienen en los hogares, empresas y lugares públicos. Los dispositivos que comúnmente utilizan esta tecnología son los teléfonos inteligentes, computadores portátiles, tabletas, entre otros.

Las redes inalámbricas permiten a los dispositivos remotos que se conecten sin dificultad, independientemente de que estos dispositivos estén a unos metros o a varios kilómetros de distancia, todo ello sin necesidad de romper paredes para pasar cables o instalar conectores. Esto ha hecho que el uso de esta tecnología sea muy popular, extendiéndose muy rápidamente. (Salazar, s.f).

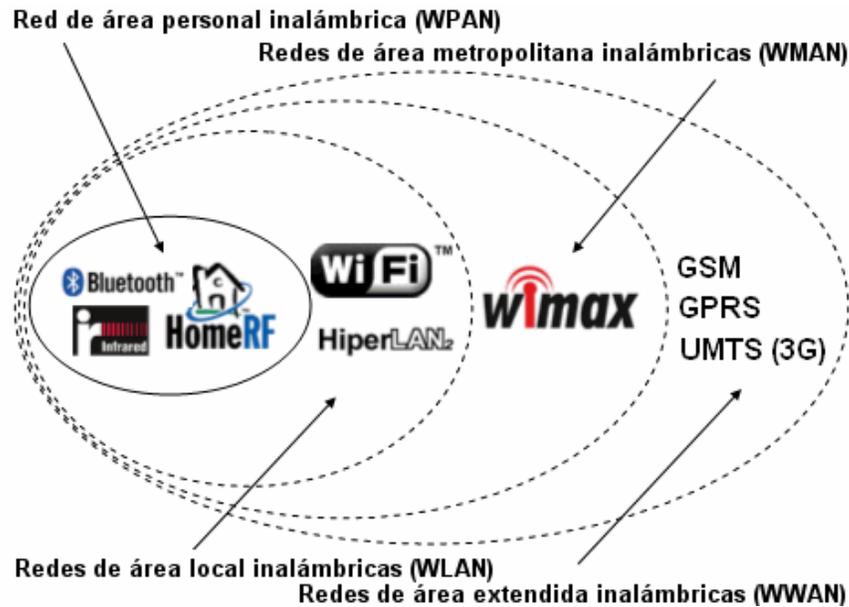
#### **Clasificación de las Redes Inalámbricas según el área de cobertura:**

- **WPAN:** Wireless Personal Area Network - Redes inalámbricas de área de personal.
- **WLAN:** Wireless Local Area Network - Redes inalámbricas de área local.
- **WMAN:** Wireless Metropolitan Area Network - Redes inalámbricas de área metropolitana.
- **WWAN:** Wireless Wide Area Network - Redes Inalámbricas de área extensa.

(Coñapes, 2015)

En la **Figura 6** podemos observar un esquema que muestra esta clasificación.

**Figura 6** Clasificación de las redes inalámbricas según el área de cobertura.



**Coñapes, S.(2015).Clasificación de redes inalámbricas .Figura 6.Recuperado de: <https://sites.google.com/site/redesinalambricas3/tipos-de-redes-inalambricas>**

### 2.1.3. Redes WiFi

WiFi es una tecnología de comunicación inalámbrica que nos permite conectar diferentes dispositivos a internet mediante el uso de radiofrecuencias. “La tecnología WiFi nació de la necesidad de establecer un mecanismo de conexión inalámbrica que fuese compatible entre distintos dispositivos” (Hernández s.f.). Las redes WiFi operan con el estándar IEEE 802.11 el cual codifica las mejoras que aumentan el rendimiento y el alcance inalámbrico. A continuación, se explican los

diferentes estándares de comunicación que son comúnmente usados (Hernández s.f., p.2):

- 802.11a: red inalámbrica con portadora en la banda ISM (banda de radio industrial, científica y médica) de 5 GHz y una tasa de transferencia de datos de hasta 54 Mbps.
- 802.11b: red inalámbrica con portadora en la banda ISM de 2,4 GHz y una tasa de transferencia de datos de hasta 11 Mbps.
- 802.11g: red inalámbrica con portadora en la banda ISM de 2,4 GHz y una tasa de transferencia de datos de hasta 54 Mbps.
- 802.11i: autenticación y encriptación.
- 802.11n: red inalámbrica con portadora en la banda ISM de 2,4 GHz y 5 GHz, con tasas de transferencia de datos de hasta 600 Mbps.
- 802.11ac: red inalámbrica con portadora debajo de 6 GHz, con tasas de transferencia de datos de al menos 1 Gbps en operación multiestación y 500 Mbps en un solo enlace.

#### **2.1.4. Microcomputador Raspberry Pi**

Los dispositivos Raspberry Pi son placas reducidas semejantes a un computador. Su nombre proviene de una fundación en Reino Unido donde creyeron que era necesario ofrecer alternativas de aprendizaje para los niños con un costo reducido, teniendo en cuenta que no siempre era posible adquirir un computador debido a su alto precio. Sin embargo, se requirieron 6 años aproximadamente para que el

mercado lo conociera, pues sus creadores llegaron a pensar que su idea de ayudar a fomentar el aprendizaje nunca iba a ser vista de manera novedosa (Hernández, Díaz, González, Durán, & Sánchez, 2018).

Raspberry Pi es un microcomputador cuyas partes internas son procesadoras, puertos USB, salidas de video, tarjeta de red, entre otras. También existieron dos modelos que se identificaron como A y B con características similares, pero para el caso del modelo B fue una actualización y contaba con mejores alternativas (Buettrich, 2007).

Una de las principales mejoras del modelo B+ fue el aumento de la frecuencia de su CPU, que pasa de los 1,2 GHz del modelo B a los 1,4 GHz en el modelo B+. Otro de los grandes cambios está en la conectividad WiFi, ya que ofrece doble banda (2.4 y 5 GHz) y la conectividad Ethernet se triplica, pasando de los 100 Mbits/s del modelo anterior a los 300 Mbits/s en modelo 3B+.

En la **Tabla 1** se describen las características técnicas de las Raspberry Pi 3B + versus Raspberry Pi 3B.

**Tabla 1** Comparación entre los modelos de Raspberry Pi 3B y 3B+

ESPECIFICACIONES	RASPBERRY Pi 3 MODEL B+	RASPBERRY Pi 3 MODEL B
Procesador	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz	Broadcom BCM2837, Cortex-A53 (ARMv8) 64-bit SoC @ 1.2GHz
RAM	1GB RAM	1GB RAM
Conectividad	WiFi 802.11.b/g/n/ac de doble banda 2.4GHz y 5GHz. Bluetooth 4.2 Puerto Ethernet de hasta 300Mbps	WiFi 802.11 b/g/n (2.4GHz) Bluetooth 4.1 Puerto Ethernet de hasta 100Mbps
Puertos	HDMI completo, 4 USB 2.0, Micro SD, CSI camera, DSI display	HDMI completo, 4 USB 2.0, Micro SD, CSI camera, DSI display

**Fuente:** <https://www.xataka.com/ordenadores/llega-la-nueva-raspberry-pi-3-model-b-mismo-precio-pero-mas-velocidad-y-wifi-de-doble-banda>

Algunas de sus funciones pueden ser similares a las de un computador dependiendo el uso que se le quiera dar, pues se puede conectar a una pantalla, un teclado y realizar diversas actividades, inclusive elegir el sistema operativo en el cual se quiera trabajar; algunas de las opciones son Linux, Windows, Kali, Firefox, OS x, que serán elegidas a necesidad del usuario (Bautista, et al., 2014). En la **Figura 7**, observamos una fotografía de un microcomputador Raspberry Pi.

**Figura 7** Microcomputador Raspberry Pi



**Microcomputador Raspberry Pi, (s.f). Recuperado: <https://market.samm.com>**

### **2.1.5. Módulo WiPi**

WiPi corresponde a las iniciales de *Wireless Internet Platform for Interoperability* (plataforma de internet inalámbrica para interoperabilidad). Es un adaptador de red USB inalámbrico, compatible con el estándar IEEE 802.11n, y algunas versiones anteriores como IEEE 802.11g, IEEE 802.11b, lo que permitirá que pueda establecer todas las conexiones inalámbricas que demanda la configuración en red de tipo mesh.

En la **Figura 8** podemos ver un ejemplo de un módulo WiPi.

**Figura 8 Módulo WiPi**



**Módulo WiPi. (2012). Figura 8. Recuperado de:**  
[https://www.amazon.com/gp/product/B00BDW6D7I/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o00\\_s01?ie=UTF8&psc=1](https://www.amazon.com/gp/product/B00BDW6D7I/ref=ppx_yo_dt_b_asin_title_o00_s01?ie=UTF8&psc=1)

### **2.1.6. Linux LEDE**

LEDE corresponde a las iniciales *Linux Embedded Development Environment*, es un sistema Operativo de Linux basado en OpenWRT diseñado exclusivamente para enrutadores y minicomputadores. Esta distribución de Linux integrada, facilita a los desarrolladores y administradores de sistemas la creación y personalización de software para dispositivos integrados, especialmente enrutadores inalámbricos (About the LEDE project, 2017).

## **2.2. ANTECEDENTES Y ESTADO DEL ARTE**

En esta sección se reseñan algunos casos publicados en los cuales las redes mesh han sido utilizadas como un medio de comunicación útil después de la ocurrencia de un desastre.

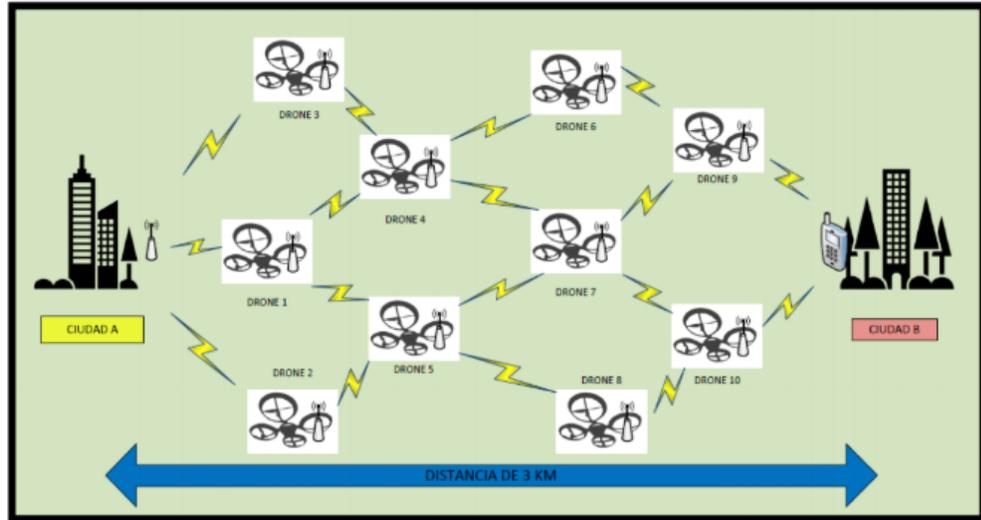
### **2.2.1. Diseño de un sistema de telecomunicaciones con redes ad hoc de drones como alternativa de medio de comunicación para hacer frente a desastres naturales**

En 2017, un estudiante de la Pontificia Universidad Católica del Perú realizó una investigación sobre la utilización de las redes en el mundo, el objetivo era diseñar un sistema de comunicaciones con redes ad hoc de drones para contrarrestar algún tipo de caos telefónico durante una situación de emergencia después de la ocurrencia de un desastre natural (Marocho, 2017).

Durante el desarrollo del sistema, hizo un reconocimiento de la zona en la cual se iba a implementar, luego procedió al estudio de las diferentes herramientas de simulación de redes inalámbricas, donde encontró la herramienta OPNET como la más adecuada, y mediante ella fue posible evidenciar que con la red se podían comunicar dos ciudades que se encontraban a 3 km de distancia. Finalmente realizó un estudio comercial sobre los dispositivos utilizados en la elaboración del proyecto en donde se procedió a realizar un cálculo total del costo del mismo. En esta tesis se realizaron las simulaciones en 4 protocolos de enrutamiento para redes ad hoc: AODV, DSR, OSLR, TORA, donde utilizaron 10 nodos en la construcción de la red (Marocho, 2017, p.4).

En la **Figura 9** se representa la arquitectura de comunicación creada por el autor para la elaboración de su tesis:

**Figura 9** Arquitectura del sistema de comunicación inalámbrico propuesto en (Marocho, 2017)



### 2.2.2. Diseño y despliegue de una red para atención post-desastre basada en UAVs

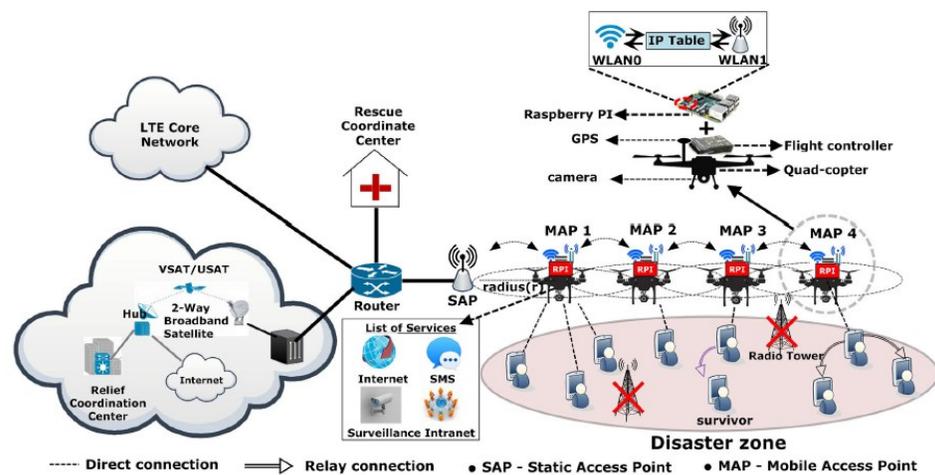
Panda, Das, Sen, & Arif (2019) realizaron un artículo llamado “Design and Deployment of UAV-Aided Post-Disaster Emergency Network” donde habla sobre el diseño de una red de comunicaciones de emergencia fiable, resistente y de despliegue rápido para la posterior gestión de un desastre. En este trabajo se propone una red WiFi de emergencia, donde las placas de Raspberry PI van encima de los UAV (vehículo aéreo no tripulado, conocido también como *drone*) para formar una red de cadenas de WiFi sobre la región de desastres para así agilizar las operaciones de rescate, guiando a los supervivientes a la ubicación del campamento de rescate más cercano. El diseño de red UAV es capaz de hacer vigilancia en el lugar y transmitir datos al centro de ayuda para una mejor

planificación de rescate. Un desafío importante es alertar a un sobreviviente de la red de emergencia, que se aborda mediante el diseño de un portal cautivo.

También se observa que a partir del experimento la distancia máxima de saltos para el prototipo es de 280 a 290 metros para una red WiFi configurada con los protocolos IEEE 802.11n e IEEE 802.11ac.

En la **Figura 10** podemos observar el prototipo de la red que fue construida en este estudio.

**Figura 10** Modelo de red basada en UAVs propuesta en (Panda, Das, Sen, & Arif I, 2019)

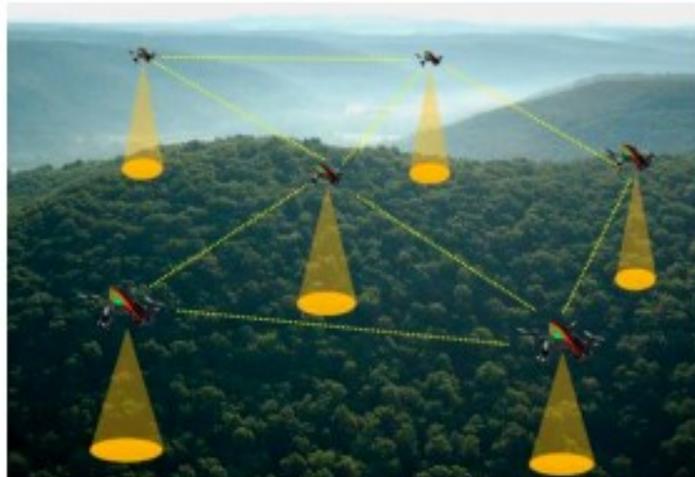


### 2.2.3. Diseño de una red mesh de UAVs para proporcionar servicios de comunicaciones

El objetivo primordial de este proyecto es el diseño y posterior evaluación de una red mesh compuesta por un conjunto de dispositivos Raspberry Pi que funcionarán como routers y puntos de acceso conectados entre sí, con el propósito de

proporcionar un servicio de comunicaciones inalámbrico a los distintos equipos que se encuentren a su alcance. Estos dispositivos Raspberry Pi serán los que se sitúen sobre los UAVs (Rosa, 2017). A pesar que el autor enfatiza sobre el diseño de la red mesh para UAVs, en la implementación real no incluyeron el uso de los drones. En **Figura 11** se muestra un ejemplo de la red ad hoc inalámbrica de múltiples UAVs.

**Figura 11** Red mesh de UAVs propuesta en (Rosa, 2017).



#### **2.2.4. Diseño de la red de nueva generación tipo mesh, para la atención y prevención de emergencias en Bogotá, en la banda de 4,9 Ghz, bajo la resolución 1661 de 18 de julio de 2006 para la coordinación de acciones de socorro y salvamento**

Este proyecto consiste en el diseño de una red mesh de banda ancha que ofrece diferentes tipos de servicios a los rescatistas. Los autores se basaron en la población de Bogotá, la geografía y topología, pues las respuestas a emergencias no eran suficientes para la densidad poblacional. Es por ello que en este trabajo se

propone el diseño de una red inalámbrica de banda ancha operando sobre la banda de 4,9 GHz, ofreciendo anchos de banda de 1 a 20 MHz para suplir todas las necesidades de comunicación de forma redundante y auto regenerable en el suceso de un evento de gran magnitud (terremoto). Esta banda está destinada para las acciones de socorro y salvamento bajo la Resolución 1661 de 2006.

Eligieron el software para simulación NS2 (Network Simulator 2) pues era la herramienta más eficiente para evaluar la red mesh. Realizaron varias pruebas comprobando la funcionalidad para el caso de una forma de controlar la red, el flujo de tráfico, el número de personas que la utilizan, de una manera dinámica y auto organizable (Ceballos, 2013).

#### **2.2.5. Diseño de red mesh ad hoc para telemedicina y Aplicaciones de desastres naturales en áreas rurales de Nicaragua**

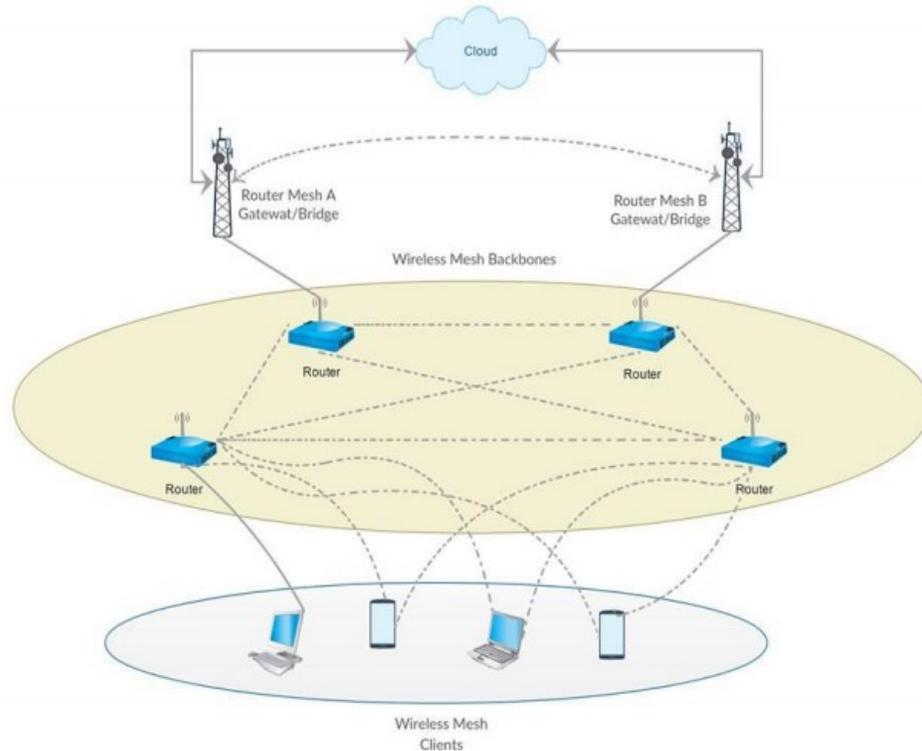
Morales, Matus, & Arias, 2016 diseñaron una red inalámbrica mesh, con la que querían ofrecer servicios de telemedicina en los sectores de salud y monitorear situaciones de emergencias naturales en tiempo real de las áreas rurales de Nicaragua, pues su objetivo era poder mitigar el impacto de los desastres naturales y que las personas pudieran mejorar los estilos de vida.

Los software que fueron utilizados para este proyecto fueron OpenWRT y radio Mobile, como medio de comunicación y consideraron este tipo de red porque les ofrecía ventajas como arquitectura descentralizada, lo que permitía que

cualquier nodo de la red pudiera brindar los servicios, la red propuesta por ellos es la que se puede evidenciar en la **Figura 12**.

Realizaron evaluaciones del comportamiento de la red, aunque no mencionan aspectos técnicos usados como dispositivos.

**Figura 12** Propuesta de diseño de red Ad Hoc Mesh para su aplicación en Telemedicina y Desastres Naturales en zona rural de Nicaragua (Morales, Matus, & Arias, 2016)



En la **Tabla 2** se presenta una comparación de los estudios presentados. Se incluye también el prototipo de esta tesis (última línea de la tabla).

**Tabla 2** Comparativo entre proyectos similares

Referencia del estudio	Tipo de red	Cobertura de la red	Dispositivos usados y cantidad	Banda de frecuencia
(Marocho, 2017)	Ad hoc	3 Km	10 Drones y D-LINK 3690 Realizado mediante simulación	2.4 GHz y 5 GHz.
(Gopal, 2019)	Ad hoc en cadena de nodos	280 a 290 m	Drones y Raspberry Pi 3B+	No se menciona
(Rosa, 2017).	Mesh	No se menciona	3 Raspberry Pi 3B+	No se menciona
(Ceballos, 2013)	Mesh	No se menciona	No se menciona. Realizado mediante simulación.	4.9 GHz
(Morales, Matus, & Arias, 2016)	Mesh	No se menciona	No se menciona	No se menciona
Desarrollo de un prototipo de red mesh basada en dispositivos Raspberry Pi para la comunicación en situaciones de desastres.	Mesh	Por determinar	3 Raspberry Pi 3B+	2.4 GHz

En el prototipo que se desarrolló en este trabajo de grado no se utilizaron drones, pero el uso de Raspberry Pi, por su diseño liviano, permite que en caso de requerirse, estos dispositivos sean fácilmente instalados en drones, lo cual aprovecharía la mayor cobertura de una red aérea para conectar a los sobrevivientes. Además, este tipo de red aérea es muy versátil en caso de desastres en los que es difícil el acceso a la zona afectada. Como se pudo evidenciar en los diferentes trabajos realizados, no se tiene una demostración real de la distancia que puede soportar el prototipo de red diseñado y la gran mayoría se basa en simulaciones. Uno de los aportes de nuestro trabajo de grado es la evaluación del protocolo de enrutamiento dinámico, el cual se va a configurar e implementar en un escenario real. Como se puede observar, el tema de este trabajo de grado es de actualidad y de gran pertinencia para un entorno como el colombiano.

### 3. METODOLOGÍA

A continuación, se describe la metodología que se utilizó para el desarrollo del prototipo de red mesh:

#### 1. Construir el prototipo de la red mesh:

**1.1.** Estudio inicial sobre el uso, configuración y pruebas de los dispositivos Raspberry Pi: conexión, instalación de periféricos (monitor, teclado, mouse) uso del instalador NOOBS para instalar el sistema operativo Raspbian, que viene en una tarjeta micro SD precargada.

**1.2.** Instalación de la distribución de Linux conocida como LEDE basada en OpenWRT, con el fin de tener más flexibilidad en cuanto a la configuración de aspectos de conectividad y enrutamiento de red.

**1.3.** Configuración de cada Raspberry Pi con: nombre, instalación y configuración de los módulos WiPi y de las tarjetas WiFi, asignación de direcciones IP en cada subred, creación y configuración de las redes WiFi, instalación y configuración del servidor DHCP que va a servir la red.

#### 2. Implementar esquema de enrutamiento para que el tráfico de los dispositivos móviles conectados a la red, pueda llegar a su destino:

**2.1.** Configuración y prueba de enrutamiento estático básico, inicialmente con dos Raspberrys, y luego extendiéndolo al conjunto de tres Raspberrys.

**2.2.** Selección, configuración y prueba de enrutamiento dinámico disponible para LEDE.

3. Realizar pruebas de cobertura, con teléfonos inteligentes conectados a los puntos de acceso:

**3.1.** Realizar pruebas de alcance variando la distancia entre los dispositivos Raspberry Pi. Las pruebas se realizarán dentro de una casa que ofrece una distancia máxima de 20 metros.

**3.2.** Realizar pruebas de conexión entre los teléfonos móviles conectados a los puntos de acceso, mediante el envío de paquetes por medio de la utilidad ping.

## 4. DESARROLLO

En este capítulo se explican las dos primeras actividades que se propusieron en la metodología: la construcción del prototipo de la red mesh y la implementación del esquema de enrutamiento.

### 4.1. Construcción del prototipo de la red mesh

Para la implementación de la red mesh se adquirieron tres kits denominados Canakits, cada uno de los cuales incluía: Raspberry Pi 3 Modelo B+, tarjeta MicroSD de 32 GB, lector de tarjeta MicroSD USB, adaptador de corriente para Raspberry pi de 2.5 A, cable HDMI, PiSwitch y un Premium case para Raspberry, como se evidencia en la **Figura 13**.

**Figura 13** Kit de Raspberry PI 3 modelo B+ marca Canakit



**Fuente:** *Elaboración propia*

Adicionalmente, se compraron tres adaptadores de red inalámbrica de marca WiPi. A continuación, se describen los pasos que se siguieron para lograr la configuración y puesta en marcha de una de las Raspberry Pi. Posteriormente, conociendo cómo se debía hacer el proceso, los mismos pasos se replicaron para configurar las otras Raspberry.

#### **4.1.1. Conexión de la Raspberry Pi a los periféricos**

Inicialmente, se realizó la conexión e instalación de los periféricos (monitor, teclado y mouse), mediante los puertos USB y HDMI que trae disponible cada Raspberry Pi, lo cual no requirió realizar una configuración específica para su funcionamiento.

#### **4.1.2. Instalación del sistema operativo**

Primero se instaló el sistema operativo Raspbian, el cual venía precargado en la memoria MicroSD. El proceso de instalación solamente requirió aceptar los términos solicitados desde el instalador NOOBS.

Una vez finalizada esta instalación, se descargó la imagen más reciente de OpenWRT con la versión 19.07.2 y, utilizando el software gratuito Etcher, se procedió a cargar el sistema operativo en la memoria MicroSD. Este sistema operativo es un *firmware* basado en una distribución de Linux que nos ofrece mayor flexibilidad en cuanto a la configuración de aspectos de conectividad y enrutamiento de red.

### 4.1.3. Configuración de OpenWRT en la Raspberry Pi 3B+

La configuración de la Raspberry Pi se inició conectándola mediante cable a un switch externo. Como se observa en la **Figura 14**, después de instalar el sistema operativo OpenWRT, aparece el terminal de línea de comandos, donde se realizaron las siguientes configuraciones:

- Cambio de clave del usuario root para acceder al terminal.
- Ejecución de los siguientes comandos para obtener una dirección IP dinámica que proveería el servidor DHCP del switch al que se encontraba conectada la Raspberry:

```
uci set network.lan.proto=dhcp
```

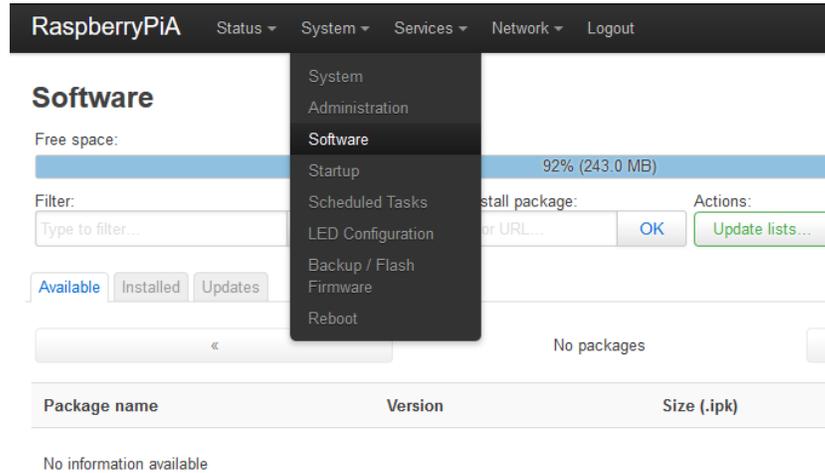
```
uci commit
```

```
/etc/init.d/network restart
```

- Actualización de los paquetes del sistema operativo a través del comando **opkg update**.



**Figura 15** Interfaz de búsqueda de controladores



**Fuente: Elaboración propia**

**Figura 16** Controladores Ralink para instalar

Filter:  Clear    Download and install package:  OK    Actions:

«    Displaying 1-10 of 10    »

Package name	Version	Size (.ipk)	Description	
kmod-rt2500-usb	4.14.171+4.19.98-1-1	9.6 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (RT2500 USB)	<input type="button" value="Install..."/>
kmod-rt2800-lib	4.14.171+4.19.98-1-1	50.2 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (rt2800 LIB)	<input type="button" value="Install..."/>
kmod-rt2800-usb	4.14.171+4.19.98-1-1	7.7 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (RT2870 USB)	<input type="button" value="Install..."/>
kmod-rt2x00-lib	4.14.171+4.19.98-1-1	22.2 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (LIB)	<input type="button" value="Install..."/>
kmod-rt2x00-usb	4.14.171+4.19.98-1-1	7.2 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (USB)	<input type="button" value="Install..."/>
kmod-rt73-usb	4.14.171+4.19.98-1-1	12.2 KB	<a href="#">Ralink</a> Drivers for RT2x00 cards (RT73 USB)	<input type="button" value="Install..."/>
rt2800-pci-firmware	20190416-1	4.7 KB	<a href="#">Ralink</a> RT28xx/3xxx PCI/SoC firmware	<input type="button" value="Install..."/>
rt2800-usb-firmware	20190416-1	3.6 KB	<a href="#">Ralink</a> RT28xx/3xxx USB firmware	<input type="button" value="Install..."/>
rt61-pci-firmware	20190416-1	7.2 KB	<a href="#">Ralink</a> RT2561 firmware	<input type="button" value="Install..."/>
rt73-usb-firmware	20190416-1	2.0 KB	<a href="#">Ralink</a> RT2573 firmware	<input type="button" value="Install..."/>

**Fuente: Elaboración propia**

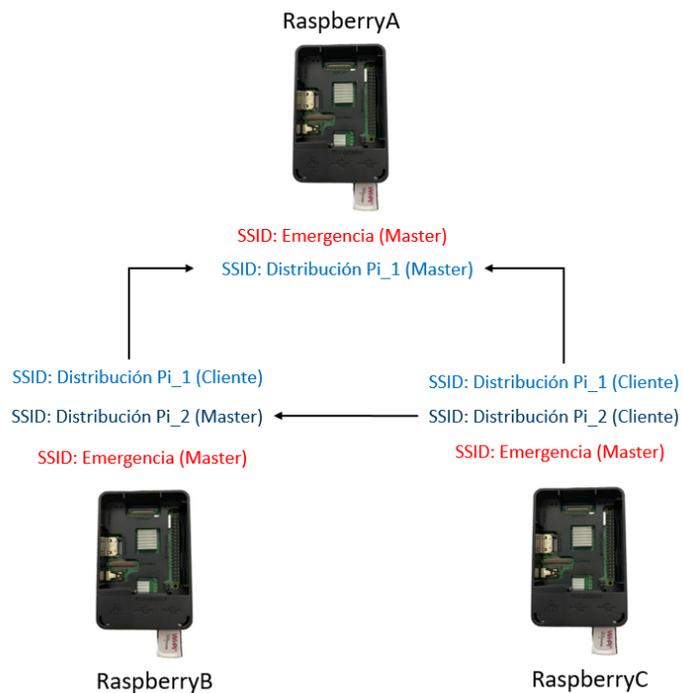
Una vez se terminó con éxito la instalación del sistema operativo y de los adaptadores WiPi, se procedió con la interconexión de las Raspberrys y la configuración del enrutamiento de la red.

## 4.2. Configuración de la red mesh

### 4.2.1. Descripción de la red

La red configurada por los dispositivos Raspberry se dividió en 2 partes, Red de Distribución y Red de Acceso, como se observa en la **Figura 17**.

**Figura 17** Interconexión de dispositivos Raspberry



**Fuente: Elaboración propia**

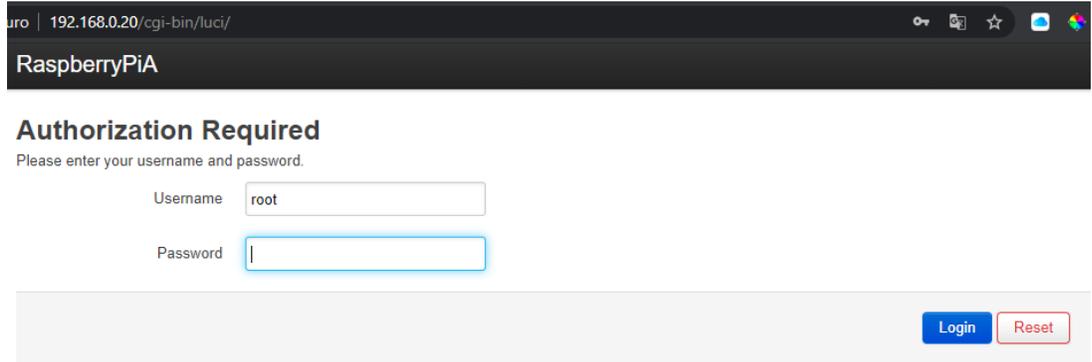
- Red de Distribución: es la encargada de enlazar los 3 dispositivos Raspberry que conforman la red mesh (se muestra en color azul en la **Figura 17**). También se encarga de realizar el enrutamiento del tráfico entre la red de distribución y la red de acceso. Para esta red se usaron los SSID: Distribucion\_1 y Distribucion\_2.
- Red de Acceso: es la encargada de conectar a los usuarios a la red mesh. Los sobrevivientes que se encuentren en el área de cobertura de la red mesh se conectarán mediante sus teléfonos inteligentes a la red WiFi con SSID: Emergencia (se muestra en color rojo en la **Figura 17**).

#### **4.2.2. Configuración de interfaces - Enrutamiento estático**

##### **4.2.2.1. Interfaz LAN**

Como se mencionó anteriormente, cada una de las Raspberry se conectó a un switch local mediante un cable conectado al puerto LAN (Ethernet) de la Raspberry. Luego se accedió a la interfaz LuCI que ofrece el sistema operativo OpenWRT, como se muestra en la **Figura 18**, a través de la IP dinámica que asignó el switch a la interfaz LAN de cada Raspberry.

**Figura 18** Interfaz LuCI



uro | 192.168.0.20/cgi-bin/luci/

RaspberryPiA

**Authorization Required**  
Please enter your username and password.

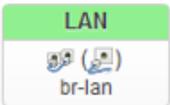
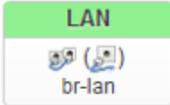
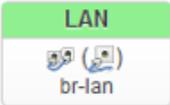
Username

Password

**Fuente:** *Elaboración propia*

Para evitar que la dirección IP cambie cada vez que se necesite reiniciarlas, se asignó un direccionamiento IP estático a la interfaz LAN de cada una de ellas. La función de esta interfaz es únicamente permitir acceder a LuCI para realizar las configuraciones correspondientes, sin necesidad de hacerlas por medio de línea de comandos. Se ingresó a la opción Network → Interfaces y se modificó la interfaz LAN, quedando configurada de la siguiente manera en cada dispositivo, como se observa en la **Figura 19**.

**Figura 19** Configuración interfaz LAN en RaspberryA, RaspberryB y RaspberryC

	<p>Protocol: Static address Uptime: 2h 40m 51s MAC: B8:27:EB:85:A4:0A RX: 4.05 MB (34922 Pkts.) TX: 5.86 MB (7512 Pkts.) IPv4: 192.168.0.20/24 IPv6: fd7b:d1d6:fb75::1/60</p>
	<p>Protocol: Static address Uptime: 0h 23m 44s MAC: B8:27:EB:6E:A0:87 RX: 1.24 MB (11784 Pkts.) TX: 2.82 MB (4707 Pkts.) IPv4: 192.168.0.30/24 IPv6: fddb:ce18:3fa3::1/60</p>
	<p>Protocol: Static address Uptime: 1h 57m 37s MAC: B8:27:EB:95:12:25 RX: 3.86 MB (29363 Pkts.) TX: 10.99 MB (13308 Pkts.) IPv4: 192.168.0.40/24 IPv6: fd88:3c72:60c4::1/60</p>

**Fuente: Elaboración Propia**

#### 4.2.2.2. Interfaz DISTRIBUCIÓN

Para la configuración de la red de distribución se utilizó el siguiente direccionamiento

IP estático:

RaspberryA: 192.168.2.20 / 255.255.255.0 (SSID: Distribucion Pi\_1)

RaspberryB: 192.168.4.30 /255.255.255.0 (SSID: Distribucion Pi\_2)

RaspberryC: no es necesaria la asignación de una IP estática a la interfaz de distribución ya que ese dispositivo se conectará como cliente a cada uno de los SSID, recibiendo una dirección dinámica.

### 4.2.2.3. Interfaz EMERGENCIA

Para la configuración de la red de Emergencia se utilizó el siguiente direccionamiento IP estático:

RaspberryA: 192.168.1.20 /255.255.255.0

RaspberryB: 192.168.3.30/ 255.255.255.0

RaspberryC: 192.168.5.40/ 255.255.255.0

### 4.2.3. Servicio DHCP- Enrutamiento estático

El servicio DHCP es el encargado de asignar direccionamiento IP a cada dispositivo que se conecte a la red de Emergencia. En cada interfaz creada en esa red, se habilitó este servicio. Podemos observar la configuración desde LuCI en la **Figura 20** y en la **Tabla 3** la configuración en las RaspberryA, RaspberryB y RaspberryC.

**Figura 20** Configuración DHCP desde LuCI en red EMERGENCIA

General Setup | Advanced Settings | IPv6 Settings

Ignore interface

Disable DHCP for this interface.

Start: 192.168.1.100

Lowest leased address as offset from the network address.

Limit: 150

Maximum number of leased addresses.

Lease time: 12h

Expiry time of leased addresses, minimum is 2 minutes (2m).

**Fuente: Elaboración propia**

**Tabla 3 Configuración DHCP de la red de EMERGENCIA**

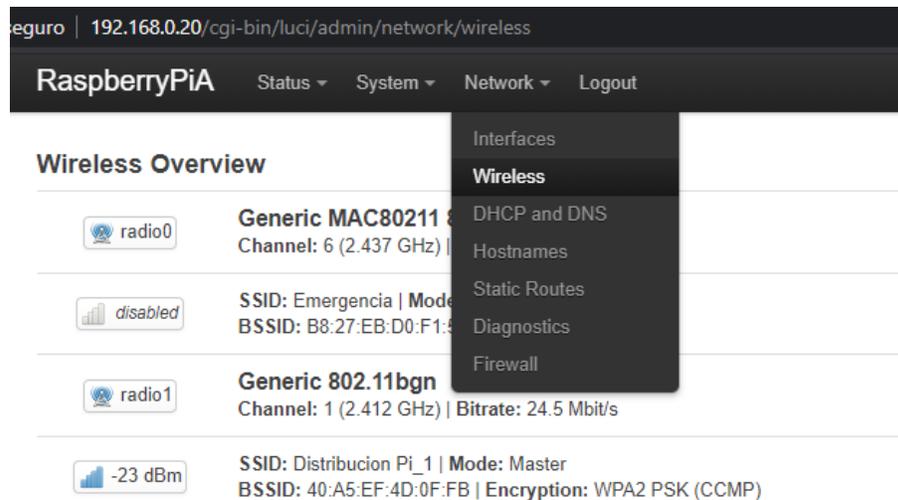
Raspberry	Start DHCP	Limit
RaspberryA	192.168.1.100	150
RaspberryB	192.168.3.100	150
RaspberryC	192.168.5.100	150

**Fuente: Elaboración propia**

#### 4.2.4. Configuración de redes WiFi - Enrutamiento estático

La configuración de las redes WiFi se realizó en la sección de Network → Wireless como se evidencia en la **Figura 21**.

**Figura 21 Configuración de redes WiFi**



**Fuente: Elaboración propia**

Cada una de las Raspberry tiene 2 adaptadores de red, con los cuales se realizaron las configuraciones correspondientes de las redes WiFi:

- Generic MAC80211 802.11acbg: adaptador de red embebido.
- Generic 802.11bgn: adaptador de red USB (módulo WiPi).

### **RaspberryA:**

Una señal WiFi tipo Master en el adaptador embebido:

- Tipo de señal: Master
- SSID: Emergencia
- Canal de radiofrecuencia: 6
- Encriptación: Ninguna

Una señal WiFi tipo Master en el adaptador WiPi:

- Tipo de señal: Cliente
- SSID: Distribucion\_1
- Canal de radiofrecuencia: 1
- Encriptación: WPA2 PSK (CCMP)

### **RaspberryB:**

Una señal WiFi tipo Cliente en el adaptador embebido, que recibirá la señal que provee la RaspberryA:

- Tipo de señal: Cliente
- SSID: Distribucion\_1

- Canal de radiofrecuencia: 1
- Encriptación: WPA2 PSK (CCMP)

Una señal WiFi tipo Master en el adaptador WiPi, que conectará con la RaspberryB:

- Tipo de señal: Master
- SSID: Distribucion\_2
- Canal de radiofrecuencia: 1
- Encriptación: WPA2 PSK (CCMP)

Una señal WiFi tipo Master en el adaptador WiPi:

- Tipo de señal: Master
- SSID: Emergencia
- Canal de radiofrecuencia: 1
- Encriptación: Ninguna

### **RaspberryC:**

Una señal WiFi tipo Cliente en el adaptador embebido, que recibirá la señal que provee la RaspberryB:

- Tipo de señal: Cliente
- SSID: Distribucion\_2
- Canal de radiofrecuencia: 1
- Encriptación: WPA2 PSK (CCMP)

Una señal WiFi tipo Cliente en el adaptador WiPi, que recibirá la señal que provee la RaspberryA:

- Tipo de señal: Cliente

- SSID: Distribucion\_1
- Canal de radiofrecuencia: 1
- Encriptación: WPA2 PSK (CCMP)

Una señal WiFi tipo Master en el adaptador WiPi:

- Tipo de señal: Master
- SSID: Emergencia
- Canal de radiofrecuencia: 1
- Encriptación: Ninguna

Se recomienda tener la red de distribución en un mismo canal de radiofrecuencia y la red de acceso en 3 canales diferentes, esto con el fin de evitar ruido e interferencias en las comunicaciones si la red llegara a ser más grande. Esta configuración no fue posible realizarla en nuestro prototipo porque en un mismo adaptador de red estamos manejando dos redes y la configuración no permitía asignar canales diferentes a cada una de ellas.

En la **Figura 22**, podemos observar la configuración de las redes WiFi desde LuCI.

**Figura 22** Configuración redes WiFi desde LuCI, (a) RaspberryA, (b) RaspberryB y (c) RaspberryC.

RaspberryPiA Status System Services Network Logout

**Wireless Overview**

radio0	<b>Generic MAC80211 802.11bgnac</b> Channel: 6 (2.437 GHz)   Bitrate: ? Mbit/s
disabled	SSID: Emergencia_   Mode: Master BSSID: B8:27:EB:D0:F1:5F   Encryption: None
radio1	<b>Generic 802.11bgn</b> Channel: 1 (2.412 GHz)   Bitrate: 54.1 Mbit/s
-27 dBm	SSID: Distribucion Pi_1   Mode: Master BSSID: 40:A5:EF:4D:0B:93   Encryption: WPA2 PSK (CCMP)

(a)

RaspberryPiB Status System Services Network Logout

Wireless Overview

radio0	<b>Generic MAC80211 802.11bgnac</b> Channel: 1 (2.412 GHz)   Bitrate: 65 Mbit/s
-34 dBm	SSID: Distribucion Pi_1   Mode: Client BSSID: B8:27:EB:3B:F5:D2   Encryption: WPA2 PSK (CCMP)
radio1	<b>Generic 802.11bgn</b> Channel: 1 (2.412 GHz)   Bitrate: 14.4 Mbit/s
-69 dBm	SSID: Emergencia_B   Mode: Master BSSID: 40:A5:EF:4D:0F:FB   Encryption: None
-15 dBm	SSID: Distribucion Pi_2   Mode: Master BSSID: 40:A5:EF:4D:0F:FA   Encryption: WPA2 PSK (CCMP)

(b)

RaspberryPiC Status System Services Network Logout

Wireless Overview

radio0	<b>Generic MAC80211 802.11bgnac</b> Channel: 1 (2.412 GHz)   Bitrate: 58.5 Mbit/s
-26 dBm	SSID: Distribucion Pi_2   Mode: Client BSSID: B8:27:EB:C0:47:70   Encryption: WPA2 PSK (CCMP)
radio1	<b>Generic 802.11bgn</b> Channel: 1 (2.412 GHz)   Bitrate: 1 Mbit/s
disabled	SSID: Emergencia_C   Mode: Master BSSID: 40:A5:EF:4D:08:BF   Encryption: None
-35 dBm	SSID: Distribucion Pi_1   Mode: Client BSSID: 40:A5:EF:4D:08:BE   Encryption: WPA2 PSK (CCMP)

(c)

**Fuente: Elaboración propia**

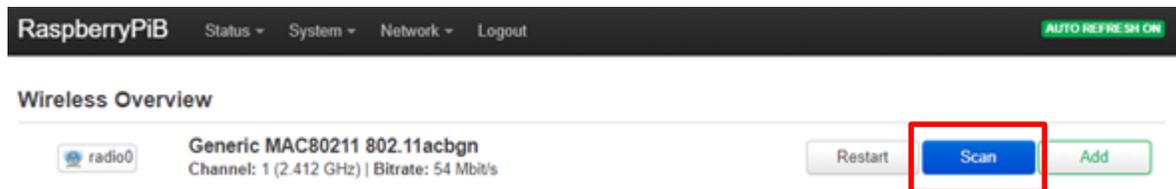
#### 4.2.5. Conexión entre las Raspberrys

Una vez creadas las interfaces y las señales WiFi en cada uno de los nodos, se realizó la conexión entre ellos. Para realizar estas conexiones, se asignaron direcciones IP fijas y se deshabilitó el firewall en las interfaces.

- **Conexión entre RaspberryA y RaspberryB:**

Desde la interfaz gráfica de la RaspberryB, como se observa en la **Figura 23**, en la opción Network → Wireless y desde el adaptador embebido se seleccionó el botón Scan para escanear las redes disponibles. Se eligió la red llamada Distribucion Pi\_1 que es la Master de la RaspberryA como se observa en la **Figura 24**.

**Figura 23** Conexión entre Raspberry A y B



**Fuente: Elaboración propia**

**Figura 24** Redes disponibles Raspberry A

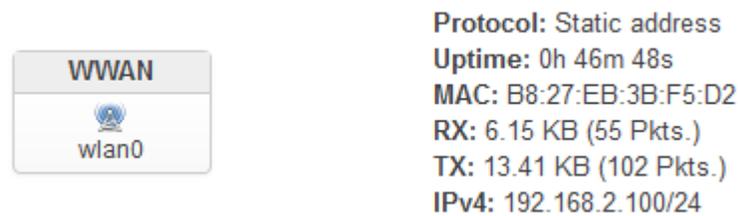
Join Network: Wireless Scan

Signal	SSID	Channel	Mode	BSSID	Encryption	
-32 dBm	Distribucion Pi_1	1	Master	40:A5:EF:4D:0F:FB	WPA2 PSK (CCMP)	Join Network
-17 dBm	Distribucion Pi_2	1	Master	40:A5:EF:4D:08:BF	WPA2 PSK (CCMP)	Join Network
-18 dBm	Emergencia	1	Master	40:A5:EF:4D:08:BE	None	Join Network
-28 dBm	Emergencia	1	Master	40:A5:EF:4D:0B:92	None	Join Network

**Fuente: Elaboración propia**

Al ingresar a la sección de interfaces en la RaspberryB, se debe evidenciar la interfaz creada. A esta interfaz se le asignó una IP fija, como se observa en la **Figura 25**.

**Figura 25 Configuración IP fija interfaz WWAN**



**WWAN**  
wlan0

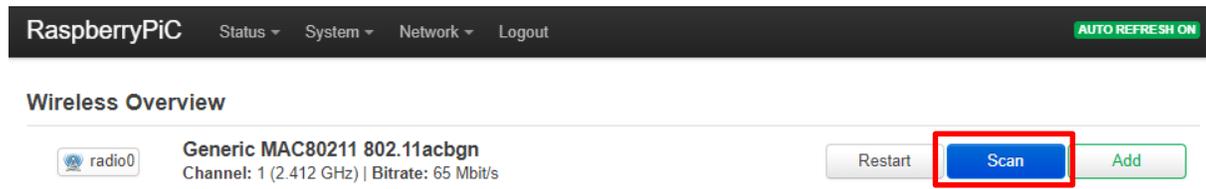
Protocol: Static address  
Uptime: 0h 46m 48s  
MAC: B8:27:EB:3B:F5:D2  
RX: 6.15 KB (55 Pkts.)  
TX: 13.41 KB (102 Pkts.)  
IPv4: 192.168.2.100/24

**Fuente: Elaboración propia**

- **Conexión entre RaspberryA y RaspberryC**

Desde la interfaz gráfica de la RaspberryC, como se observa en la **Figura 26**, en la opción Network → Wireless y desde el adaptador WiPi se seleccionó el botón Scan para escanear las redes disponibles. Se eligió la red la llamada Distribucion Pi\_1 que es la Master de la RaspberryA, como se observa en la **Figura 27**.

**Figura 26 Conexión entre Raspberry A y C**



RaspberryPiC Status System Network Logout AUTO REFRESH ON

**Wireless Overview**

radio0 Generic MAC80211 802.11acbgm  
Channel: 1 (2.412 GHz) | Bitrate: 65 Mbit/s

Restart **Scan** Add

**Fuente: Elaboración propia**

**Figura 27 Redes disponibles RaspberryC**

Join Network: Wireless Scan

Signal	SSID	Channel	Mode	BSSID	Encryption	
100%	Distribucion Pi_1	1	Master	40:A5:EF:4D:0F:FB	WPA2 PSK (CCMP)	<b>Join Network</b>
100%	Distribucion Pi_2	1	Master	40:A5:EF:4D:08:BF	WPA2 PSK (CCMP)	Join Network
100%	Emergencia	1	Master	40:A5:EF:4D:08:BE	None	Join Network
100%	Emergencia	1	Master	40:A5:EF:4D:0B:92	None	Join Network

**Fuente: Elaboración propia**

Al ingresar a la sección de interfaces en la RaspberryC, debe aparecer la interfaz creada. A esta interfaz se le asignó una IP fija, como se observa en la **Figura 28**.

**Figura 28** Configuración IP fija interfaz WWAN2

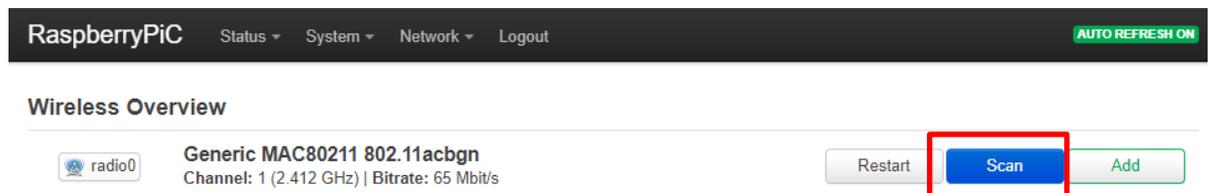
<b>WWAN2</b>  wlan1	<b>Protocol:</b> Static address <b>Uptime:</b> 2h 21m 35s <b>MAC:</b> 40:A5:EF:4D:08:BE <b>RX:</b> 10.65 KB (83 Pkts.) <b>TX:</b> 8.07 KB (65 Pkts.) <b>IPv4:</b> 192.168.2.101/24
--	---

**Fuente:** *Elaboración propia*

- **Conexión entre RaspberryB y RaspberryC**

Desde la interfaz gráfica de la RaspberryC, como se observa en la **Figura 29**, en la opción Network → Wireless y desde el adaptador embebido se seleccionó el botón Scan para escanear las redes disponibles. Se eligió la red la llamada Distribucion Pi\_2 que es la Master de la RaspberryB, como se observa en la **Figura 30**.

**Figura 29** Conexión entre Raspberry B y C



The screenshot shows the RaspberryPiC interface with a navigation bar at the top containing 'RaspberryPiC', 'Status', 'System', 'Network', and 'Logout'. A green 'AUTO REFRESH ON' button is in the top right. Below the navigation bar is the 'Wireless Overview' section. It displays 'radio0' with a signal strength icon, 'Generic MAC80211 802.11acbgm', and 'Channel: 1 (2.412 GHz) | Bitrate: 65 Mbit/s'. At the bottom of this section are three buttons: 'Restart', 'Scan' (highlighted with a red box), and 'Add'.

**Fuente:** *Elaboración propia*

**Figura 30** Redes disponibles RaspberryC

Join Network: Wireless Scan

Signal	SSID	Channel	Mode	BSSID	Encryption	
100%	Distribucion Pi_1	1	Master	40:A5:EF:4D:0F:FB	WPA2 PSK (CCMP)	Join Network
100%	Distribucion Pi_2	1	Master	40:A5:EF:4D:08:BF	WPA2 PSK (CCMP)	Join Network
100%	Emergencia	1	Master	40:A5:EF:4D:08:BE	None	Join Network
100%	Emergencia	1	Master	40:A5:EF:4D:0B:92	None	Join Network

**Fuente: Elaboración propia**

Al ingresar a la sección de interfaces en la RaspberryC, debe aparecer la interfaz creada, cuya dirección IP es fija, como se observa en la **Figura 31**.

**Figura 31** Configuración de interfaz fija interfaz WWAN

WWAN

  
wlan0

**Protocol:** Static address  
**Uptime:** 0h 51m 26s  
**MAC:** B8:27:EB:C0:47:70  
**RX:** 3.58 KB (27 Pkts.)  
**TX:** 8.43 KB (55 Pkts.)  
**IPv4:** 192.168.4.100/24

**Fuente: Elaboración propia**

En la **Figura 32**, las estaciones asociadas en cada Raspberry después de realizar la interconexión entre ellas desde la interfaz de LuCI.

**Figura 32** Estaciones asociadas en (a) RaspberryA, (b) RaspberryB y (c)RaspberryC.

Network	MAC-Address	Host
 Master "Distribucion Pi_1" (wlan1)	40:A5:EF:4D:0B:93	192.168.2.101
 Master "Distribucion Pi_1" (wlan1)	B8:27:EB:3B:F5:D2	RaspberryPiB.lan (192.168.2.100)

**(a)**

Network	MAC-Address	Host
 Client "Distribucion Pi_1" (wlan0)	40:A5:EF:4D:0F:FB	192.168.2.20
 Master "Distribucion Pi_2" (wlan1-1)	B8:27:EB:C0:47:70	192.168.4.100

(b)

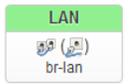
Network	MAC-Address	Host
 Client "Distribucion Pi_2" (wlan0)	40:A5:EF:4D:0F:FA	192.168.4.30
 Client "Distribucion Pi_1" (wlan1)	40:A5:EF:4D:0B:93	192.168.2.20

(c)

**Fuente: Elaboración propia**

En la **Figura 33**, veremos la sección de interfaces una vez realizadas todas las configuraciones anteriormente mencionadas:

**Figura 33** Interfaces en (a) RaspberryA, (b) RaspberryB y (c) RaspberryC.

	<p>Protocol: Static address Uptime: 2h 52m 12s MAC: 40:A5:EF:4D:0B:93 RX: 20.94 KB (236 Pkts.) TX: 12.58 KB (122 Pkts.) IPv4: 192.168.2.20/24</p>
	<p>Protocol: Static address Uptime: 2h 55m 8s MAC: B8:27:EB:D0:F1:5F RX: 0 B (0 Pkts.) TX: 1.09 KB (8 Pkts.) IPv4: 192.168.1.20/24</p>
	<p>Protocol: Static address Uptime: 2h 55m 8s MAC: B8:27:EB:85:A4:0A RX: 4.59 MB (39778 Pkts.) TX: 7.17 MB (8932 Pkts.) IPv4: 192.168.0.20/24 IPv6: fd7b:d1d6:fb75::1/60</p>

(a)

<p><b>DISTRIBUCION</b></p>  <p>wlan1-1</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 0h 40m 49s <b>MAC:</b> 40:A5:EF:4D:0F:FA <b>RX:</b> 3.70 KB (27 Pkts.) <b>TX:</b> 6.90 KB (42 Pkts.) <b>IPv4:</b> 192.168.4.30/24</p>
<p><b>EMERGENCIA</b></p>  <p>wlan1</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 0h 40m 49s <b>MAC:</b> 40:A5:EF:4D:0F:FB <b>RX:</b> 47.86 KB (275 Pkts.) <b>TX:</b> 29.84 KB (256 Pkts.) <b>IPv4:</b> 192.168.3.30/24</p>
<p><b>LAN</b></p>  <p>br-lan</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 0h 40m 50s <b>MAC:</b> B8:27:EB:6E:A0:87 <b>RX:</b> 1.87 MB (18112 Pkts.) <b>TX:</b> 3.57 MB (6329 Pkts.) <b>IPv4:</b> 192.168.0.30/24 <b>IPv6:</b> fd8b:ce18:3fa3::1/60</p>
<p><b>WWAN</b></p>  <p>wlan0</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 0h 35m 59s <b>MAC:</b> B8:27:EB:3B:F5:D2 <b>RX:</b> 5.39 KB (49 Pkts.) <b>TX:</b> 11.48 KB (90 Pkts.) <b>IPv4:</b> 192.168.2.100/24</p>

(b)

<p><b>EMERGENCIA</b></p>  <p>wlan1-1</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 2h 10m 15s <b>MAC:</b> 40:A5:EF:4D:08:BF <b>RX:</b> 156.41 KB (1187 Pkts.) <b>TX:</b> 112.23 KB (1002 Pkts.) <b>IPv4:</b> 192.168.5.40/24</p>
<p><b>LAN</b></p>  <p>br-lan</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 2h 14m 58s <b>MAC:</b> B8:27:EB:95:12:25 <b>RX:</b> 4.64 MB (36300 Pkts.) <b>TX:</b> 12.74 MB (15639 Pkts.) <b>IPv4:</b> 192.168.0.40/24 <b>IPv6:</b> fd88:3c72:60c4::1/60</p>
<p><b>WWAN</b></p>  <p>wlan0</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 0h 38m 13s <b>MAC:</b> B8:27:EB:C0:47:70 <b>RX:</b> 3.58 KB (27 Pkts.) <b>TX:</b> 8.43 KB (55 Pkts.) <b>IPv4:</b> 192.168.4.100/24</p>
<p><b>WWAN2</b></p>  <p>wlan1</p>	<p><b>Protocol:</b> Static address <b>Uptime:</b> 2h 9m 9s <b>MAC:</b> 40:A5:EF:4D:08:BE <b>RX:</b> 9.45 KB (75 Pkts.) <b>TX:</b> 8.07 KB (65 Pkts.) <b>IPv4:</b> 192.168.2.101/24</p>

(c)

**Fuente: Elaboración propia**

#### 4.2.6. Configuración de rutas estáticas

Una vez configuradas las interfaces y las redes WiFi en los dispositivos, se crearon las rutas estáticas para poder comunicar entre sí a los usuarios que se conecten a la red de acceso.

- **RaspberryA**

Se crearon 2 rutas estáticas. La primera ruta, es la que permitirá la comunicación con la subred 192.168.3.0/24 de la RaspberryB y su puerta de enlace será la IP de la RaspberryB en la red Distribucion Pi\_1. La segunda ruta, es la que permitirá la comunicación con la subred 192.168.5.0/24 de la RaspberryC y su puerta de enlace será la IP de la RaspberryC en la red Distribucion Pi\_1. Ambas rutas se asignaron a la interfaz de DISTRIBUCIÓN. En la **Tabla 4**, se puede observar el direccionamiento de las rutas estática.

**Tabla 4** Configuración de rutas estáticas RaspberryA

Interfaz	Subred de destino	Máscara	Puerta de enlace
DISTRIBUCIÓN	192.168.3.0	255.255.255.0	192.168.2.100
DISTRIBUCIÓN	192.168.5.0	255.255.255.0	192.168.2.101

**Fuente: Elaboración propia**

- **RaspberryB**

Se crearon 2 rutas estáticas. La primera ruta, está asignada a la interfaz wwan que pertenece a la interfaz creada entre la conexión de la RaspberryB con la RaspberryA y es la que permitirá la comunicación con la subred 192.168.1.0/24 de la RaspberryA. La segunda ruta, está asignada a la interfaz DISTRIBUCIÓN la cual

permitirá comunicarnos con la subred 192.168.5.0/24 de la RaspberryC y su puerta de enlace es la IP de la RaspberryC en la red Distribucion Pi\_2. En la **Tabla 5**, podemos observar la configuración de las rutas.

**Tabla 5 Configuración de rutas estáticas RaspberryB**

Interfaz	Subred de destino	Máscara	Puerta de enlace
wwan	192.168.1.0	255.255.255.0	192.168.2.20
DISTRIBUCIÓN	192.168.5.0	255.255.255.0	192.168.4.100

**Fuente: Elaboración propia**

- **RaspberryC**

Se crearon 2 rutas estáticas. La primera ruta, está asignada a la interface wwan2 que pertenece a la interfaz creada entre la conexión de la RaspberryC con la RaspberryA y es la que permitirá la comunicación a la subred 192.168.1.0/24 de la RaspberryA. La segunda ruta, está asignada a la interface wwan que pertenece a la interfaz creada entre la conexión de la RaspberryC con la RaspberryB y es la que permitirá la comunicación a la subred 192.168.3.0/24 de la RaspberryB. En la **Tabla 6**, observamos la configuración de las rutas.

**Tabla 6 Configuración rutas estáticas RaspberryB**

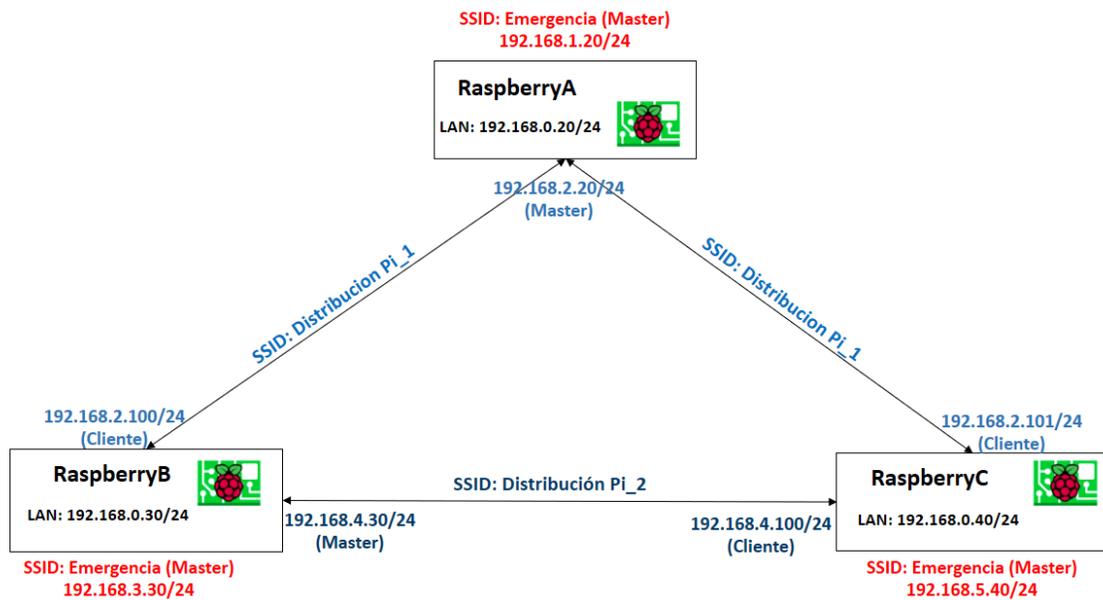
Interfaz	Subred de destino	Máscara	Puerta de enlace
wwan2	192.168.1.0	255.255.255.0	192.168.2.20
wwan	192.168.3.0	255.255.255.0	192.168.4.30

**Fuente: Elaboración propia**

#### 4.2.7. Diagrama de la Red - Enrutamiento estático

En la **Figura 34** podemos observar el diagrama final de la configuración de nuestra red mesh.

**Figura 34** Diagrama de la Red Mesh – Enrutamiento estático



**Fuente: Elaboración propia**

#### 4.2.8. Enrutamiento Dinámico OLSR

OLSR (Optimized Link-State Routing Protocol), o *protocolo de enrutamiento de estado de enlace optimizado*, es un protocolo pro-activo, es decir, que los nodos mantienen actualizada en todo momento la tabla de ruteo y es específicamente diseñado para redes ad-hoc móviles. Se basa en el uso de estaciones llamadas *multipoint relays*, quienes hacen el reenvío de paquetes o el tráfico de control, esto

permite reducir la sobrecarga de los mismos en comparación con otros protocolos donde el intercambio de muchos paquetes provoca congestión en la red.

El protocolo OLSR tiene una gran estabilidad debido a su naturaleza proactiva, una ventaja es que tiene rutas disponibles en cada momento y funciona con un alto número de nodos, cuanto mayor es la cantidad de nodos en la red, mayor es la optimización. Trabaja de manera distribuida y no requiere de una entidad central (Pastor, 2008).

#### 4.2.8.1. Instalación y configuración de OLSR

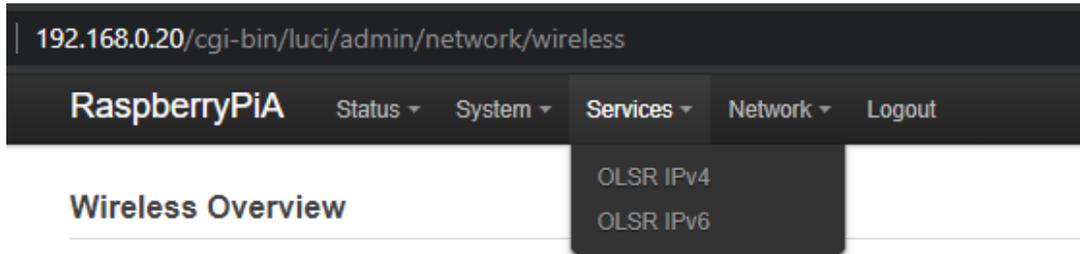
- **Instalación**

Desde la consola de comandos se instalaron los siguientes paquetes para agregar el protocolo OLSR a LEDE en cada Raspberry:

- `opkg update`
- `opkg install luci-app-olsr luci-app-olsr-services luci-app-olsr-viz olsrd olsrd-mod-arprefresh olsrd-mod-bmf olsrd-mod-dot-draw olsrd-mod-dyn-gw olsrd-mod-dyn-gw-plain olsrd-mod-httpinfo olsrd-mod-mdns olsrd-mod-nameservice olsrd-mod-p2pd olsrd-mod-pgraph olsrd-mod-secure olsrd-mod-txtinfo olsrd-mod-watchdog olsrd-mod-quagga wireless-tools luci-lib-json kmod-ipip`

Después de instalar los paquetes, aparece una pestaña llamada Services donde están las opciones para configurar OLSR, como se observa en la **Figura 35**.

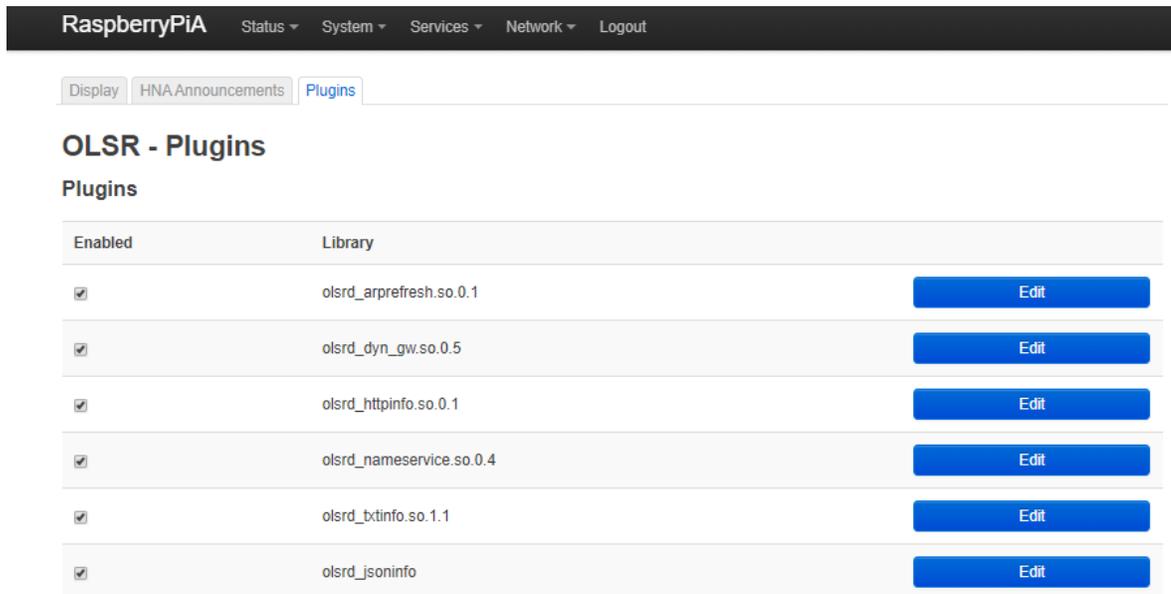
**Figura 35** Pestaña Services en LuCI



**Fuente: Elaboración propia**

Desde la opción Services → OLSR IPv4 → Plugins, se debe habilitar la opción `olsrd_jsoninfo`, para poder administrar OLSR desde la interfaz web, como se observa en la **Figura 36**.

**Figura 36** Habilitación de OLSR desde la interfaz web



**Fuente: Elaboración propia**

- **Configuración**

### RaspberryA

En el adaptador de red embebido se configuró la red de acceso llamada Emergencia tipo Access Point. En el módulo WiPi, se configuró la red llamada Distribución (de tipo AdHoc) que se encargará de aplicar el protocolo OLSR.

Se editó el archivo de configuración *wireless* ubicado en la ruta “/etc/config/wireless”. En la **Figura 37**, se observa la configuración desde LuCI.

- **Archivo *wireless***

```

config wifi-device 'radio0'
    option type 'mac80211'
    option path
'platform/soc/3f300000.mmc/mmc_host/mmc1/mmc1:0001/mmc1:0001:1'
    option hwmode '11g'
    option channel '11'

config wifi-iface 'wifinet3'
    option ssid 'Distribucion'
    option key 'password'
    option encryption 'none'
    option device 'radio1'
    option mode 'adhoc'
    option network 'mesh'

config wifi-iface 'wifinet1'
    option device 'radio0'
    option key 'password'
    option encryption 'none'
    option mode 'ap'
    option network 'EMERGENCIA'
    option ssid 'Emergencia_A'

config wifi-device 'radio2'
    option type 'mac80211'
    option channel '11'
    option hwmode '11g'
    option path 'platform/soc/3f980000.usb/usb1/1-1/1-1.3/1-1.3:1.0'

```

```
option htmode 'HT20'
```

```
config wifi-iface 'default_radio2'
option device 'radio2'
option encryption 'none'
option mode 'adhoc'
option ssid 'Distribucion'
option network 'mesh'
```

**Figura 37** Wireless de RaspberryA desde LuCI

The screenshot shows the LuCI interface for RaspberryPiA. At the top, there is a navigation bar with 'RaspberryPiA', 'Status', 'System', 'Services', 'Network', and 'Logout'. Below this is the 'Wireless Overview' section, which lists four wireless interfaces:

- radio0**: Generic MAC80211 802.11acbg, Channel: 11 (2.462 GHz) | Bitrate: ? Mbit/s
- disabled**: SSID: Emergencia\_A | Mode: Master, BSSID: B8:27:EB:D0:F1:5F | Encryption: None
- radio2**: Generic 802.11bgn, Channel: 11 (2.462 GHz) | Bitrate: 10.4 Mbit/s
- 27 dBm**: SSID: Distribucion | Mode: Ad-Hoc, BSSID: 40:A5:EF:4D:0B:93 | Encryption: None

**Fuente:** *Elaboración propia*

Por otra parte, se editó el archivo de configuración *network* ubicado en la ruta “/etc/config/network”.

- **Archivo *network***

```
config interface 'loopback'
option ifname 'lo'
option proto 'static'
option ipaddr '127.0.0.1'
option netmask '255.0.0.0'
```

```
config globals 'globals'
option ula_prefix 'fd7b:d1d6:fb75::/48'
```

```
config interface 'lan'
option type 'bridge'
option ifname 'eth0'
```

```
option netmask '255.255.255.0'  
option ip6assign '60'  
option delegate '0'  
option proto 'static'  
option ipaddr '192.168.0.20'  
list dns '8.8.8.8'  
option gateway '192.168.0.1'
```

```
config interface 'mesh'  
option proto 'static'  
option ipaddr '192.168.10.100'  
option netmask '255.255.255.0'
```

```
config interface 'EMERGENCIA'  
option proto 'static'  
option ipaddr '192.168.1.20'  
option netmask '255.255.255.0'
```

En la **Figura 38**, se observa la configuración desde LuCI. Se configuraron tres interfaces: LAN, MESH y EMERGENCIA. La interfaz LAN es la que permite el acceso a la interfaz web LuCI a través de la IP 192.168.0.20/24. La interfaz MESH es la que permite la conexión a la red de distribución, esta tiene la IP 192.168.10.100/24, y la interfaz EMERGENCIA es la que asignará direcciones IP a los dispositivos que se conecten a ella, por tal motivo es la única que tiene habilitado el servicio DHCP; esta tiene la IP 192.168.1.20/24 y asignará direcciones IP a partir de la 192.168.1.100.

**Figura 38** Network de RaspberryA desde LuCI

<p><b>LAN</b></p>  <p>br-lan</p>	<p>Protocol: Static address Uptime: 1h 48m 52s MAC: B8:27:EB:85:A4:0A RX: 2.71 MB (29178 Pkts.) TX: 1.29 MB (5616 Pkts.) IPv4: 192.168.0.20/24 IPv6: fd7b:d1d6:fb75::1/60</p>
<p><b>MESH</b></p>  <p>wlan1</p>	<p>Protocol: Static address Uptime: 1h 48m 51s MAC: 40:A5:EF:4D:0B:93 RX: 771.24 KB (6862 Pkts.) TX: 536.70 KB (4189 Pkts.) IPv4: 192.168.10.100/24</p>
<p><b>EMERGENCIA</b></p>  <p>wlan0</p>	<p>Protocol: Static address Uptime: 1h 48m 51s MAC: B8:27:EB:D0:F1:5F RX: 250.40 KB (3553 Pkts.) TX: 580.07 KB (4176 Pkts.) IPv4: 192.168.1.20/24</p>

**Fuente: Elaboración Propia**

### RaspberryB:

La configuración de esta Raspberry sigue los mismos lineamientos de la configuración de la Raspberry A: en el adaptador de red embebido se configuró la red de acceso llamada Emergencia tipo Access Point, y en el módulo WiPi, se configuró la red llamada Distribucion, de tipo AdHoc, que se encargará de aplicar el protocolo OLSR.

El archivo de configuración wireless, ubicado en la ruta “/etc/config/wireless”, quedó como se muestra a continuación.

- **Archivo *wireless***

```
config wifi-device 'radio0'
  option type 'mac80211'
  option path
'platform/soc/3f300000.mmc/mmc_host/mmc1/mmc1:0001/mmc1:0001:1'
  option hwmode '11g'
  option channel '11'
```

```
config wifi-iface 'wifinet3'
  option ssid 'Distribucion'
  option key 'password'
  option encryption 'none'
  option device 'radio1'
  option mode 'adhoc'
  option network 'mesh'
```

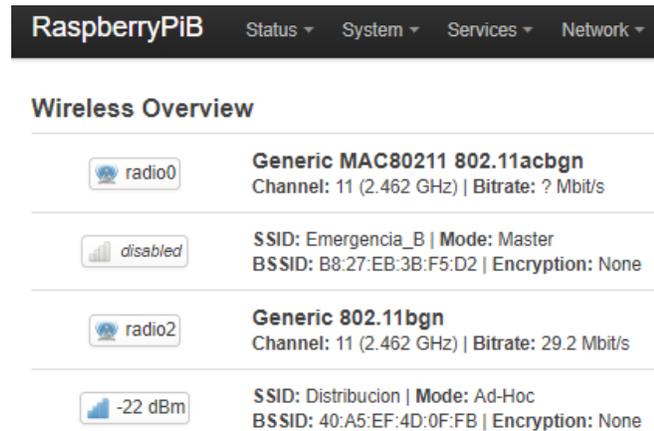
```
config wifi-iface 'wifinet1'
  option device 'radio0'
  option key 'password'
  option encryption 'none'
  option mode 'ap'
  option ssid 'Emergencia_B'
  option network 'EMERGENCIA'
```

```
config wifi-device 'radio2'
  option type 'mac80211'
  option channel '11'
  option hwmode '11g'
  option path 'platform/soc/3f980000.usb/usb1/1-1/1-1.3/1-1.3:1.0'
  option htmode 'HT20'
```

```
config wifi-iface 'default_radio2'
  option device 'radio2'
  option encryption 'none'
  option mode 'adhoc'
  option ssid 'Distribucion'
  option network 'mesh'
```

En la **Figura 39**, se observa la configuración desde LuCI.

**Figura 39** Wireless de RaspberryB desde LuCI



**Fuente: Elaboración propia**

El archivo de configuración network ubicado en la ruta “/etc/config/network”, se muestra en seguida. En la **Figura 40**, se observa la configuración desde LuCI.

- **Archivo network**

```

config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config globals 'globals'
    option ula_prefix 'fd7b:d1d6:fb75::/48'

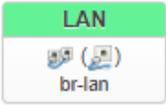
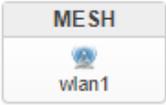
config interface 'lan'
    option type 'bridge'
    option ifname 'eth0'
    option netmask '255.255.255.0'
    option ip6assign '60'
    option delegate '0'
    option proto 'static'
    list dns '8.8.8.8'
    option gateway '192.168.0.1'
    option ipaddr '192.168.0.30'
    
```

```

config interface 'mesh'
  option proto 'static'
  option netmask '255.255.255.0'
  option ipaddr '192.168.10.101'

config interface 'EMERGENCIA'
  option proto 'static'
  option netmask '255.255.255.0'
  option ipaddr '192.168.2.30'
  
```

**Figura 40** Network de RaspberryB desde LuCI

 <p>LAN br-lan</p>	<p>Protocol: Static address            Uptime: 1h 46m 50s            MAC: B8:27:EB:6E:A0:87            RX: 2.49 MB (23684 Pkts.)            TX: 552.20 KB (3189 Pkts.)            IPv4: 192.168.0.30/24            IPv6: fd7b:d1d6:fb75::1/60</p>
 <p>MESH wlan1</p>	<p>Protocol: Static address            Uptime: 1h 46m 49s            MAC: 40:A5:EF:4D:0F:FB            RX: 752.14 KB (6644 Pkts.)            TX: 485.30 KB (3735 Pkts.)            IPv4: 192.168.10.101/24</p>
 <p>EMERGENCIA wlan0</p>	<p>Protocol: Static address            Uptime: 1h 46m 49s            MAC: B8:27:EB:3B:F5:D2            RX: 139.61 KB (1953 Pkts.)            TX: 533.24 KB (3851 Pkts.)            IPv4: 192.168.2.30/24</p>

**Fuente: Elaboración Propia**

Las tres interfaces LAN, MESH y EMERGENCIA, quedaron con las siguientes direcciones IP:

Interfaz LAN: 192.168.0.30/24

Interfaz MESH: 192.168.10.101/24

Interfaz EMERGENCIA: 192.168.2.30/24, con el servicio DHCP habilitado, asignando IP's a partir de la 192.168.2.100.

### **RaspberryC:**

Se configuró de manera semejante a las otras dos Raspberrys.

El archivo de configuración wireless quedó como se muestra a continuación.

- **Archivo *wireless***

```
config wifi-device 'radio0'
  option type 'mac80211'
  option path
'platform/soc/3f300000.mmc/mmc_host/mmc1/mmc1:0001/mmc1:0001:1'
  option hwmode '11g'
  option channel '11'

config wifi-iface 'wifinet3'
  option ssid 'Distribucion'
  option key 'password'
  option encryption 'none'
  option device 'radio1'
  option mode 'adhoc'
  option network 'mesh'

config wifi-iface 'wifinet1'
  option device 'radio0'
  option key 'password'
  option encryption 'none'
  option mode 'ap'
  option network 'EMERGENCIA'
  option ssid 'Emergencia_C'

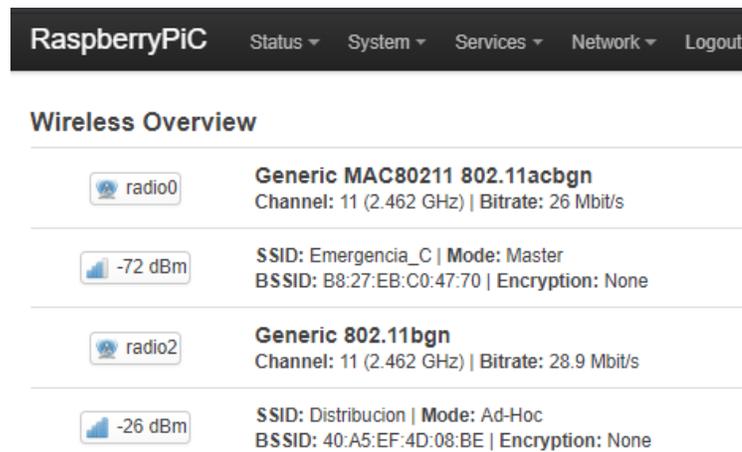
config wifi-device 'radio2'
  option type 'mac80211'
  option channel '11'
  option hwmode '11g'
  option path 'platform/soc/3f980000.usb/usb1/1-1/1-1.3/1-1.3:1.0'
  option htmode 'HT20'

config wifi-iface 'default_radio2'
```

```
option device 'radio2'
option encryption 'none'
option mode 'adhoc'
option ssid 'Distribucion'
option network 'mesh'
```

En la **Figura 41**, se observa la configuración desde LuCI.

**Figura 41** Wireless de RaspberryC desde LuCI



**Fuente: Elaboración propia**

A continuación, se muestra el contenido del archivo network.

- **Archivo network**

```
config interface 'loopback'
option ifname 'lo'
option proto 'static'
option ipaddr '127.0.0.1'
option netmask '255.0.0.0'

config globals 'globals'
option ula_prefix 'fd7b:d1d6:fb75::/48'

config interface 'lan'
option type 'bridge'
option ifname 'eth0'
```

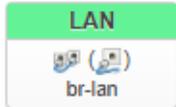
```
option netmask '255.255.255.0'
option ip6assign '60'
option delegate '0'
option proto 'static'
list dns '8.8.8.8'
option gateway '192.168.0.1'
option ipaddr '192.168.0.40'
```

```
config interface 'mesh'
option proto 'static'
option netmask '255.255.255.0'
option ipaddr '192.168.10.102'
```

```
config interface 'EMERGENCIA'
option proto 'static'
option netmask '255.255.255.0'
option ipaddr '192.168.3.40'
```

En la **Figura 42**, se observa la configuración desde LuCI.

**Figura 42** Network de RaspberryC desde LuCI

 <p>LAN br-lan</p>	<p>Protocol: Static address Uptime: 1h 34m 1s MAC: B8:27:EB:95:12:25 RX: 2.25 MB (21730 Pkts.) TX: 2.41 MB (7146 Pkts.) IPv4: 192.168.0.40/24 IPv6: fd7b:d1d6:fb75::1/60</p>
 <p>MESH wlan1</p>	<p>Protocol: Static address Uptime: 1h 34m 0s MAC: 40:A5:EF:4D:08:BE RX: 683.61 KB (6046 Pkts.) TX: 424.13 KB (3258 Pkts.) IPv4: 192.168.10.102/24</p>
 <p>EMERGENCIA wlan0</p>	<p>Protocol: Static address Uptime: 1h 34m 0s MAC: B8:27:EB:C0:47:70 RX: 362.98 KB (2772 Pkts.) TX: 483.63 KB (3521 Pkts.) IPv4: 192.168.3.40/24</p>

**Fuente: Elaboración propia**

Las tres interfaces LAN, MESH y EMERGENCIA, quedaron con las siguientes direcciones IP:

Interfaz LAN: 192.168.0.40/24

Interfaz MESH: 192.168.10.102/24

Interfaz EMERGENCIA: 192.168.3.40/24, con el servicio DHCP habilitado, asignando IP's a partir de la 192.168.3.100.

Una vez realizadas las configuraciones anteriores en cada una de las Raspberrys, se dirige a la opción Services → OLSR IPv4 → Interfaces, como se observa en la **Figura 43**. Las redes mesh y EMERGENCIA deben aparecer en esta sección, si alguna no aparece se deben agregar con el botón Add.

**Figura 43** Configuración interfaces OLSR

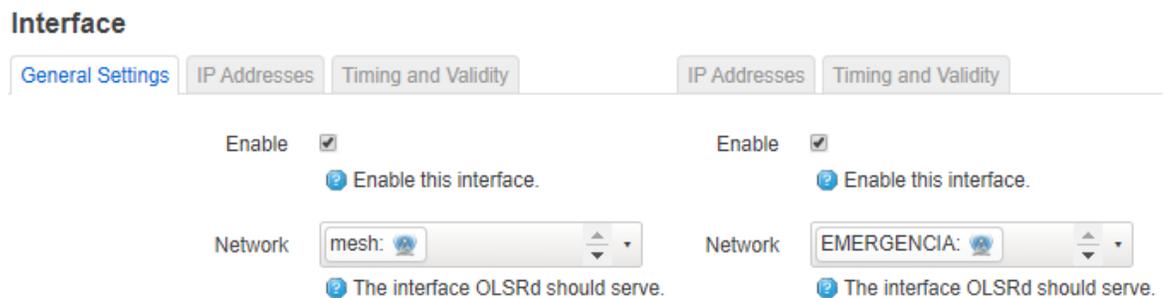
**Interfaces**

Enable	Network	Mode	Hello	TC	MID	HNA			
<input checked="" type="checkbox"/>	mesh: 	mesh	5.0s / 40.0s	2.0s / 256.0s	18.0s / 324.0s	18.0s / 108.0s	<a href="#">Edit</a>	<a href="#">Delete</a>	
<input checked="" type="checkbox"/>	EMERGENCIA: 	mesh	5.0s / 40.0s	2.0s / 256.0s	18.0s / 324.0s	18.0s / 108.0s	<a href="#">Edit</a>	<a href="#">Delete</a>	
<a href="#">Add</a>									
							<a href="#">Save &amp; Apply</a>	<a href="#">Save</a>	<a href="#">Reset</a>

**Fuente: Elaboración Propia**

Se debe verificar que la red correspondiente tenga vinculada la interface anteriormente creada. Para verificar esto, se da en la opción Edit y se agrega en la opción Network la interface correspondiente, como se observa en la **Figura 44**.

**Figura 44** Interface mesh



**Fuente: Elaboración propia**

Después de realizar toda la configuración anterior, ya se logra ver las Raspberrys interconectadas como se observa en la **Figura 45**. La métrica sobre cada enlace es conocida como ETX (Expected Transmission Number) o número esperado de transmisiones, que significa el número de transmisiones esperadas de un paquete para que se reciba sin error en su destino. Un ETX con valor uno indica un medio de transmisión en muy buenas condiciones, pero un ETX con valor infinito, representa un enlace no funcional (Couto & Morris, 2004). También hay autores que resaltan que:

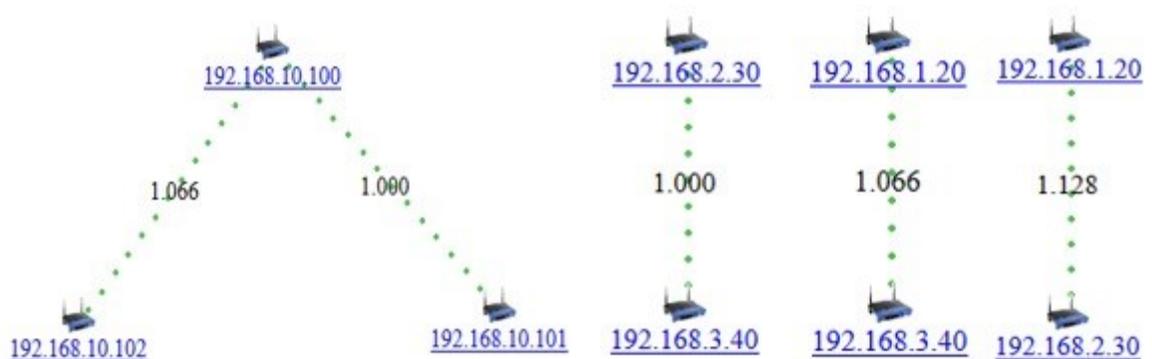
Cuando un protocolo de enrutamiento se encuentra con dos rutas distintas para llegar al mismo destino, debe ser capaz de diferenciar cuál es la más conveniente. Una métrica es una forma de evaluar esta situación basándose

en uno o varios parámetros. Cada protocolo de enrutamiento utiliza su propia métrica.

Tradicionalmente, la métrica más usada ha sido la del mínimo número de saltos hacia el nodo destino. Pero en redes inalámbricas, esta métrica puede no ser la más adecuada. El camino más corto no tiene por qué ser siempre ni el más rápido ni el más fiable para llegar a un destino, teniendo en cuenta que en las redes inalámbricas podemos encontrarnos con enlaces de diferente calidad. Una misma métrica no tiene por qué ser la mejor para todo tipo de entornos y aplicaciones. (Granados Bayona & Pérez, p.36)

En la **Figura 45**, también se pueden observar las redes de distribución y de emergencia desde cada una de las Rasperrys.

**Figura 45** Conexión entre RaspberryA, RaspberryB y RaspberryC, red de emergencia y red de distribución



**Fuente:** Elaboración propia

## 5. RESULTADOS

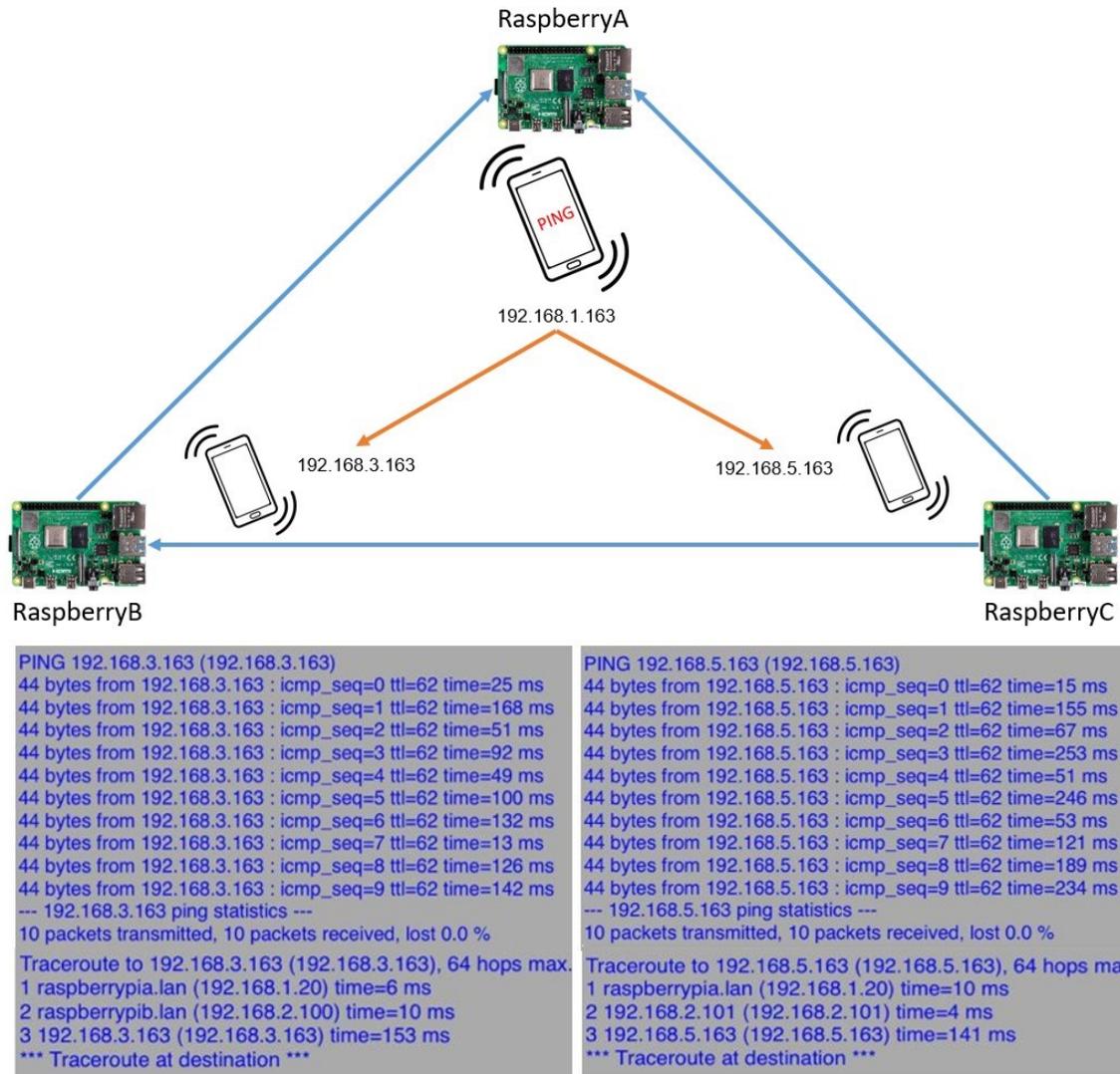
En este capítulo se presentan los resultados de las pruebas realizadas con la red mesh. En la primera parte, se realizan las pruebas de conexión entre los dispositivos Raspberry que conforman la red mesh y se verifica la conectividad entre dispositivos conectados a la red de acceso. En la segunda parte, se caracteriza el alcance de la red mesh.

### 5.1. Pruebas de conexión – Enrutamiento estático.

Para realizar pruebas de conexión entre la red de acceso en cada Raspberry, se hará uso de una aplicación móvil llamada Ping Lite, instalada en los teléfonos inteligentes, que permite utilizar las funciones Ping y Traceroute para verificar la comunicación y la ruta entre cada dispositivo.

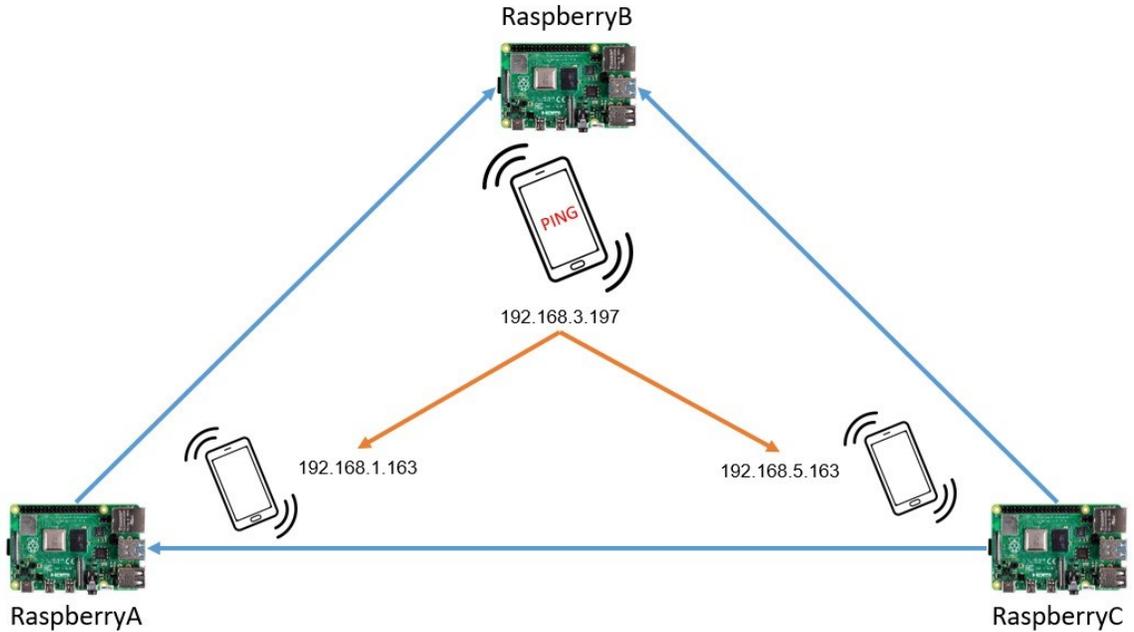
Se realizaron pings y trazas desde un dispositivo conectado a la RaspberryA hacia un dispositivo conectado a la RaspberryB y a la RaspberryC. Estas mismas pruebas se realizaron con las demás Raspberry. Como se observa en la **Figura 46**, **Figura 47** y **Figura 48**, todos los pings responden correctamente en cada uno de los dispositivos y las trazas nos demuestran el camino que un paquete realiza para llegar a su destino.

**Figura 46** Ping y traza desde un dispositivo conectado a la RaspberryA a un dispositivo conectado a la red de acceso de la RaspberryB y RaspberryC.



Fuente: Elaboración Propia

**Figura 47** Ping y traza desde un dispositivo conectado a la RaspberryB a un dispositivo conectado a la red de acceso de la RaspberryA y RaspberryC.



```
PING 192.168.1.163 (192.168.1.163)
44 bytes from 192.168.1.163 : icmp_seq=0 ttl=62 time=29 ms
44 bytes from 192.168.1.163 : icmp_seq=1 ttl=62 time=48 ms
44 bytes from 192.168.1.163 : icmp_seq=2 ttl=62 time=19 ms
44 bytes from 192.168.1.163 : icmp_seq=3 ttl=62 time=46 ms
44 bytes from 192.168.1.163 : icmp_seq=4 ttl=62 time=8 ms
44 bytes from 192.168.1.163 : icmp_seq=5 ttl=62 time=18 ms
44 bytes from 192.168.1.163 : icmp_seq=6 ttl=62 time=84 ms
44 bytes from 192.168.1.163 : icmp_seq=7 ttl=62 time=45 ms
44 bytes from 192.168.1.163 : icmp_seq=8 ttl=62 time=49 ms
44 bytes from 192.168.1.163 : icmp_seq=9 ttl=62 time=39 ms
--- 192.168.1.163 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

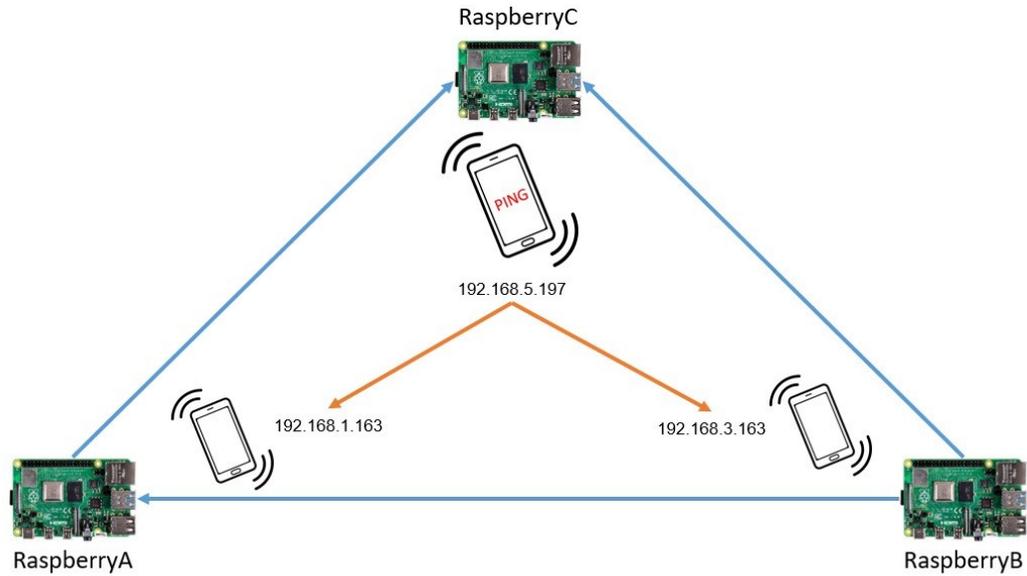
Traceroute to 192.168.1.163 (192.168.1.163), 64 hops max.
1 raspberrypi.ban (192.168.3.30) time=47 ms
2 192.168.2.20 (192.168.2.20) time=80 ms
3 192.168.1.163 (192.168.1.163) time=191 ms
*** Traceroute at destination ***
```

```
PING 192.168.5.163 (192.168.5.163)
44 bytes from 192.168.5.163 : icmp_seq=0 ttl=62 time=273 ms
Request timeout for icmp_seq 1
44 bytes from 192.168.5.163 : icmp_seq=2 ttl=62 time=232 ms
44 bytes from 192.168.5.163 : icmp_seq=3 ttl=62 time=223 ms
44 bytes from 192.168.5.163 : icmp_seq=4 ttl=62 time=328 ms
44 bytes from 192.168.5.163 : icmp_seq=5 ttl=62 time=273 ms
44 bytes from 192.168.5.163 : icmp_seq=6 ttl=62 time=217 ms
44 bytes from 192.168.5.163 : icmp_seq=7 ttl=62 time=165 ms
44 bytes from 192.168.5.163 : icmp_seq=8 ttl=62 time=210 ms
44 bytes from 192.168.5.163 : icmp_seq=9 ttl=62 time=257 ms
--- 192.168.5.163 ping statistics ---
10 packets transmitted, 9 packets received, lost 10.0 %

Traceroute to 192.168.5.163 (192.168.5.163), 64 hops max.
1 raspberrypi.ban (192.168.3.30) time=6 ms
2 192.168.4.100 (192.168.4.100) time=4 ms
3 192.168.5.163 (192.168.5.163) time=113 ms
*** Traceroute at destination ***
```

**Fuente: Elaboración propia**

**Figura 48** Ping y traza desde un dispositivo conectado a la RaspberryC a un dispositivo conectado a la red de acceso de la RaspberryA y RaspberryB.



```

PING 192.168.1.163 (192.168.1.163)
44 bytes from 192.168.1.163 : icmp_seq=0 ttl=62 time=236 ms
44 bytes from 192.168.1.163 : icmp_seq=1 ttl=62 time=14 ms
44 bytes from 192.168.1.163 : icmp_seq=2 ttl=62 time=122 ms
44 bytes from 192.168.1.163 : icmp_seq=3 ttl=62 time=169 ms
44 bytes from 192.168.1.163 : icmp_seq=4 ttl=62 time=16 ms
44 bytes from 192.168.1.163 : icmp_seq=5 ttl=62 time=21 ms
44 bytes from 192.168.1.163 : icmp_seq=6 ttl=62 time=64 ms
44 bytes from 192.168.1.163 : icmp_seq=7 ttl=62 time=14 ms
44 bytes from 192.168.1.163 : icmp_seq=8 ttl=62 time=101 ms
44 bytes from 192.168.1.163 : icmp_seq=9 ttl=62 time=149 ms
--- 192.168.1.163 ping statistics ---
10 packets transmitted, 11 packets received, lost -10.0 %

Traceroute to 192.168.1.163 (192.168.1.163), 64 hops max.
1 raspberrypic.lan (192.168.5.40) time=12 ms
2 192.168.2.20 (192.168.2.20) time=3 ms
3 192.168.1.163 (192.168.1.163) time=23 ms
*** Traceroute at destination ***
    
```

```

PING 192.168.3.163 (192.168.3.163)
44 bytes from 192.168.3.163 : icmp_seq=0 ttl=62 time=576 ms
44 bytes from 192.168.3.163 : icmp_seq=1 ttl=62 time=267 ms
44 bytes from 192.168.3.163 : icmp_seq=2 ttl=62 time=416 ms
Request timeout for icmp_seq 3
44 bytes from 192.168.3.163 : icmp_seq=3 ttl=62 time=12 ms
44 bytes from 192.168.3.163 : icmp_seq=4 ttl=62 time=1845 ms
44 bytes from 192.168.3.163 : icmp_seq=5 ttl=62 time=200 ms
44 bytes from 192.168.3.163 : icmp_seq=6 ttl=62 time=248 ms
44 bytes from 192.168.3.163 : icmp_seq=7 ttl=62 time=399 ms
44 bytes from 192.168.3.163 : icmp_seq=8 ttl=62 time=242 ms
44 bytes from 192.168.3.163 : icmp_seq=9 ttl=62 time=290 ms
--- 192.168.3.163 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

Traceroute to 192.168.3.163 (192.168.3.163), 64 hops max.
1 raspberrypic.lan (192.168.5.40) time=11 ms
2 192.168.4.30 (192.168.4.30) time=4 ms
3 192.168.3.163 (192.168.3.163) time=161 ms
*** Traceroute at destination ***
    
```

**Fuente: Elaboración propia**

Todas las pruebas realizadas demuestran que la red mesh está completamente interconectada utilizando la configuración realizada mediante rutas estáticas.

## 5.2. Pruebas de alcance

Debido a las restricciones para la libre circulación que se decretaron a nivel nacional a causa de la pandemia, las pruebas de alcance se hicieron al interior de una casa, en la que la máxima distancia de separación posible entre Raspberrys es de 20 metros. El objetivo de estas pruebas es evaluar el alcance entre Raspberrys a estas distancias, y conocer el impacto del tamaño de los paquetes en el alcance de la red.

Para realizar pruebas de alcance se utilizaron dos Raspberry, separándolas entre sí a 3 distancias diferentes: 10 metros, 15 metros y 20 metros. Se conectaron dispositivos muy próximos a cada una de ellas con el fin de evitar pérdidas en la red de acceso y por medio de la función ping se enviaron 100 paquetes, cambiando su tamaño en cada prueba: 100 bytes, 500 bytes, 1000 bytes y 1450 bytes, de manera unidireccional.

Las pruebas de alcance descritas anteriormente se realizaron en una zona dentro de una casa, como se muestra en la **Figura 49**.

**Figura 49** Espacio de pruebas de alcance



**Fuente:** *Elaboración propia*

Como se puede observar, la zona que se encuentra entre las distancias de 0 a 10 metros está rodeada de paredes y entre las distancias de 15 a 20 metros es un área más descubierta, factores que influyen en la propagación de la señal.

En la **Tabla 7**, se presenta la cantidad de paquetes perdidos como función de la distancia de separación y el tamaño de paquetes en bytes. Cuando se realizó la prueba a una distancia de 10 metros enviando paquetes de 1000 bytes y 1450 bytes se observa una pérdida significativa, a pesar de ser la menor distancia de separación. Este resultado puede deberse a la condición del espacio donde se realizó la prueba que, como se mencionó anteriormente, entre 0 y 10 metros tiene un corredor angosto que está rodeado de paredes.

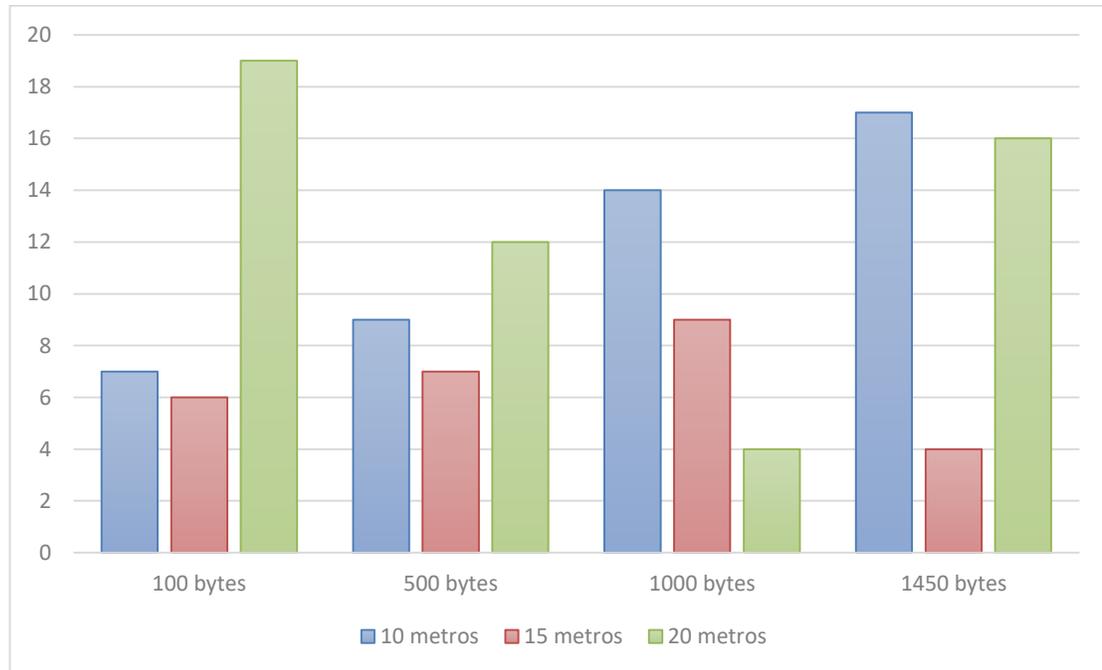
**Tabla 7 Paquetes perdidos variando distancia entre nodos y tamaño del paquete, al enviar 100 paquetes mediante el comando ping**

Tamaño de paquetes enviados	Paquetes Perdidos
Distancia 10 m	
100 bytes	7
500 bytes	9
1000 bytes	14
1450 bytes	17
Distancia 15 m	
100 bytes	6
500 bytes	7
1000 bytes	9
1450 bytes	4
Distancia 20 m	
100 bytes	19
500 bytes	12
1000 bytes	4
1450 bytes	16

**Fuente: Elaboración propia**

En la **Figura 50** se puede observar el diagrama de bloques correspondiente a los datos mostrados en la **Tabla 7**. Si bien al separar 10 metros a las dos Raspberry se puede concluir que las pérdidas se incrementan al aumentar el tamaño del paquete, esa tendencia no se observa al aumentar la distancia a 15 y a 20 metros. Estos resultados, en los que no hay una clara relación entre las pérdidas y la distancia de separación, y entre las pérdidas y el tamaño del paquete enviado, pueden deberse a que el escenario utilizado no es un campo abierto, libre de interferencia. Por el contrario, hay paredes que rodean el espacio de prueba, así como otras señales WiFi cercanas, que no permiten caracterizar con claridad el alcance de la señal.

**Figura 50** Diagrama de bloques del número de paquetes perdidos al enviar 100 paquetes mediante el comando ping

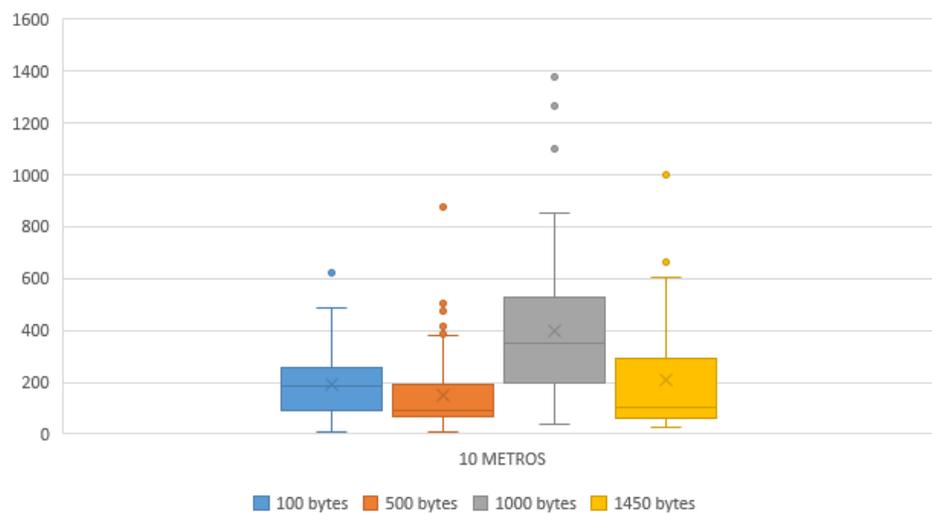


**Fuente: Elaboración propia**

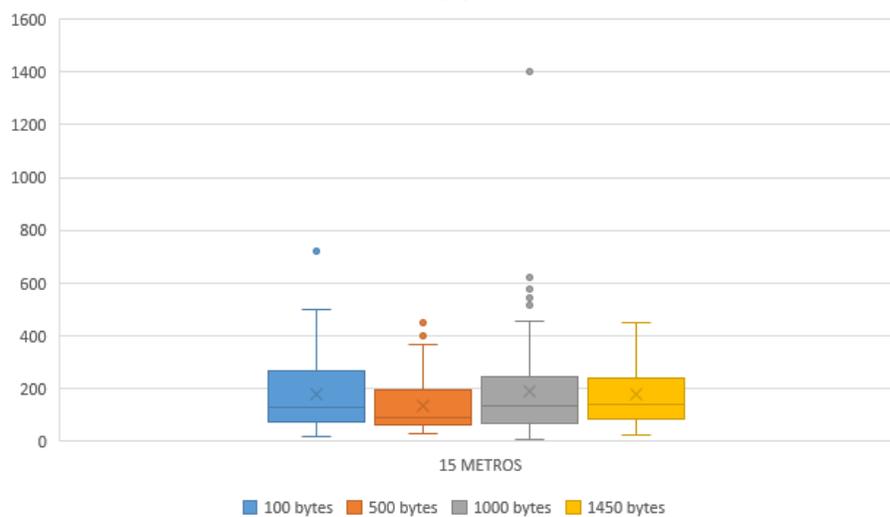
En la **Figura 51**, se observa un diagrama de cajas que se realizó con los RTT (tiempo de ida y vuelta) obtenidos mediante el comando ping en este mismo experimento. El RTT es el tiempo que tarda un paquete en llegar a su destino y volver a regresar a su punto inicial. Dentro de cada caja, la X corresponde al valor de la media del RTT, en tanto que la línea continua interior muestra la mediana. La dispersión más significativa se observa a 10 metros de separación con paquetes de 1000 bytes, donde además el retardo promedio es cercano a 400 ms, casi dos veces más que en el resto de pruebas. No obstante, al igual que ocurre con las pérdidas,

en el retardo tampoco se puede identificar una tendencia clara que permita concluir cuál es el impacto de la separación y del tamaño del paquete. Esto puede deberse, como se mencionó en el caso anterior, a la disposición del escenario y a la interferencia de otras señales cercanas.

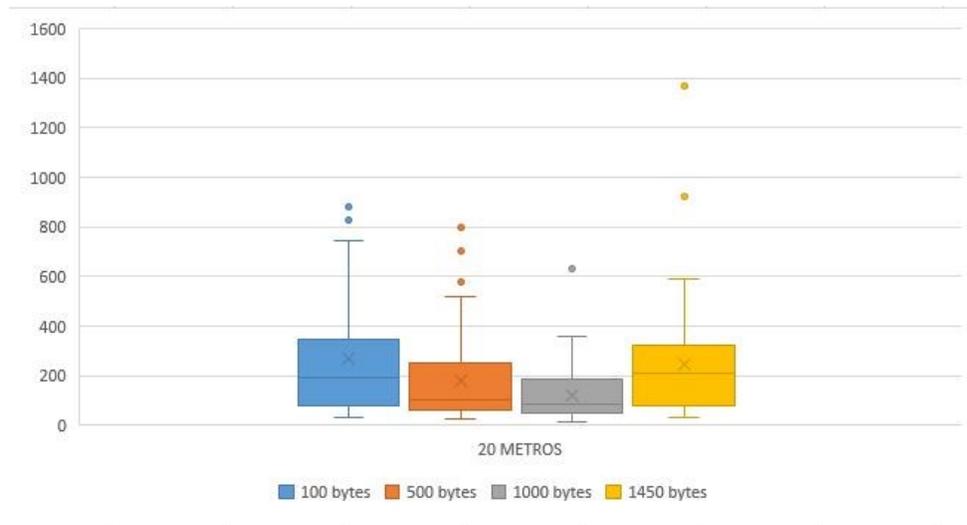
**Figura 51** Diagrama de cajas del RTT (en milisegundos) para diferentes tamaños de paquetes a una distancia de separación de (a) 10 metros, (b) 15 metros y (c) 20 metros



(a)



(b)



(c)

**Fuente: Elaboración propia**

Con base en el RTT obtenido a partir del comando ping, se realizó el cálculo del *jitter*, que mide cuanto varía el retardo de los paquetes enviados y es importante principalmente al transmitir voz. Específicamente para poder transmitir este tipo de tráfico, el jitter no debe sobrepasar 30 ms y, adicionalmente, el retardo debe ser máximo de 150 ms y las pérdidas menores al 5% de los paquetes (Keagy, 2001). Para el cálculo del jitter se utilizó el retardo en un solo sentido, asumiendo que el retardo es la mitad del RTT, y los resultados se muestran en la **Tabla 8**. A partir de los resultados obtenidos, se puede concluir que en este escenario no sería posible transmitir tráfico de voz con calidad aceptable; se requiere un escenario preferiblemente sin obstáculos, donde la señal se propague sin dificultad y en consecuencia las pérdidas y el jitter sean menores.

**Tabla 8** Cálculo del jitter

	100 bytes		1000 bytes	
	JITTER (ms)	RETARDO (ms)	JITTER (ms)	RETARDO (ms)
10 metros	63,94	95,84	115,20	197,95
15 metros	74,22	88,99	75,64	96,62
20 metros	103,53	133,60	50,43	62,26
	500 bytes		1450 bytes	
	JITTER (ms)	RETARDO (ms)	JITTER (ms)	RETARDO (ms)
10 metros	50,82	73,71	56,41	105,90
15 metros	36,88	66,26	44,70	89,44
20 metros	57,56	89,57	91,43	124,14

**Fuente:** *Elaboración propia*

Finalmente, como se observa en la **Tabla 9**, se realizó el cálculo de la Tasa de Transmisión que nos indica la velocidad de transferencia de datos en kilobits por segundo. Esta tasa se calculó a partir del retardo promedio de los paquetes más grandes (1450 bytes), adicionando al tamaño del paquete, 20 bytes del encabezado IP y 8 bytes del encabezado ICMP. Más que un valor específico, este cálculo muestra un rango de valores, producto de las características de la tecnología empleada y de las condiciones del escenario en el que se realizó el experimento.

**Tabla 9** Tasa de Transmisión

Tasa de Transmisión 10 Metros Kbits/seg	Tasa de Transmisión 15 Metros Kbits/seg	Tasa de Transmisión 20 Metros Kbits/seg
893.2	1057.6	762.0

**Fuente:** *Elaboración propia*

### 5.3. Pruebas de Conexión – Enrutamiento Dinámico

#### 5.3.1. Red de Distribución

Desde la consola de comandos se realizaron pings a las interfaces mesh de cada una de las Raspberrys para verificar su conexión. Como se observa en la **Figura 52**, **Figura 53**, **Figura 54**, los pings responden satisfactoriamente.

*Figura 52 Pings desde la RaspberryA hacia las interfaces mesh de la RaspberryB y RaspberryC*

```

root@RaspberryPiA: /
root@RaspberryPiA:/# ping 192.168.10.101
PING 192.168.10.101 (192.168.10.101): 56 data bytes
64 bytes from 192.168.10.101: seq=0 ttl=64 time=1.400 ms
64 bytes from 192.168.10.101: seq=1 ttl=64 time=1.218 ms
64 bytes from 192.168.10.101: seq=2 ttl=64 time=1.114 ms
64 bytes from 192.168.10.101: seq=3 ttl=64 time=1.081 ms
64 bytes from 192.168.10.101: seq=4 ttl=64 time=1.044 ms
^C
--- 192.168.10.101 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.044/1.171/1.400 ms
root@RaspberryPiA:/# ping 192.168.10.102
PING 192.168.10.102 (192.168.10.102): 56 data bytes
64 bytes from 192.168.10.102: seq=0 ttl=64 time=5.655 ms
64 bytes from 192.168.10.102: seq=1 ttl=64 time=3.002 ms
64 bytes from 192.168.10.102: seq=2 ttl=64 time=1.234 ms
64 bytes from 192.168.10.102: seq=3 ttl=64 time=2.136 ms
64 bytes from 192.168.10.102: seq=4 ttl=64 time=1.048 ms
^C
--- 192.168.10.102 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.048/2.615/5.655 ms
root@RaspberryPiA:/# █

```

**Fuente: Elaboración Propia**

**Figura 53 Pings desde la RaspberryB hacía las interfaces mesh de la RaspberryA y RaspberryC**

```

root@RaspberryPiB: ~
root@RaspberryPiB:~# ping 192.168.10.100
PING 192.168.10.100 (192.168.10.100): 56 data bytes
64 bytes from 192.168.10.100: seq=0 ttl=64 time=1.130 ms
64 bytes from 192.168.10.100: seq=1 ttl=64 time=1.862 ms
64 bytes from 192.168.10.100: seq=2 ttl=64 time=3.057 ms
64 bytes from 192.168.10.100: seq=3 ttl=64 time=1.018 ms
64 bytes from 192.168.10.100: seq=4 ttl=64 time=0.956 ms
^C
--- 192.168.10.100 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.956/1.604/3.057 ms
root@RaspberryPiB:~# ping 192.168.10.102
PING 192.168.10.102 (192.168.10.102): 56 data bytes
64 bytes from 192.168.10.102: seq=0 ttl=64 time=1.076 ms
64 bytes from 192.168.10.102: seq=1 ttl=64 time=1.036 ms
64 bytes from 192.168.10.102: seq=2 ttl=64 time=0.864 ms
64 bytes from 192.168.10.102: seq=3 ttl=64 time=1.014 ms
64 bytes from 192.168.10.102: seq=4 ttl=64 time=3.706 ms
64 bytes from 192.168.10.102: seq=5 ttl=64 time=0.816 ms
^C
--- 192.168.10.102 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.816/1.418/3.706 ms
root@RaspberryPiB:~# █

```

**Fuente: Elaboración Propia**

**Figura 54 Pings desde la RaspberryC hacía las interfaces mesh de la RaspberryA y RaspberryB**

```

root@RaspberryPiC: /
root@RaspberryPiC:/# ping 192.168.10.100
PING 192.168.10.100 (192.168.10.100): 56 data bytes
64 bytes from 192.168.10.100: seq=0 ttl=64 time=63.749 ms
64 bytes from 192.168.10.100: seq=1 ttl=64 time=14.558 ms
64 bytes from 192.168.10.100: seq=2 ttl=64 time=12.396 ms
64 bytes from 192.168.10.100: seq=3 ttl=64 time=0.972 ms
64 bytes from 192.168.10.100: seq=4 ttl=64 time=0.943 ms
64 bytes from 192.168.10.100: seq=5 ttl=64 time=0.916 ms
^C
--- 192.168.10.100 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0.916/15.589/63.749 ms
root@RaspberryPiC:/# ping 192.168.10.101
PING 192.168.10.101 (192.168.10.101): 56 data bytes
64 bytes from 192.168.10.101: seq=0 ttl=64 time=1.252 ms
64 bytes from 192.168.10.101: seq=1 ttl=64 time=8.945 ms
64 bytes from 192.168.10.101: seq=2 ttl=64 time=29.488 ms
64 bytes from 192.168.10.101: seq=3 ttl=64 time=15.967 ms
64 bytes from 192.168.10.101: seq=4 ttl=64 time=7.721 ms
64 bytes from 192.168.10.101: seq=5 ttl=64 time=3.787 ms
^C
--- 192.168.10.101 ping statistics ---
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 1.252/11.193/29.488 ms
root@RaspberryPiC:/# █

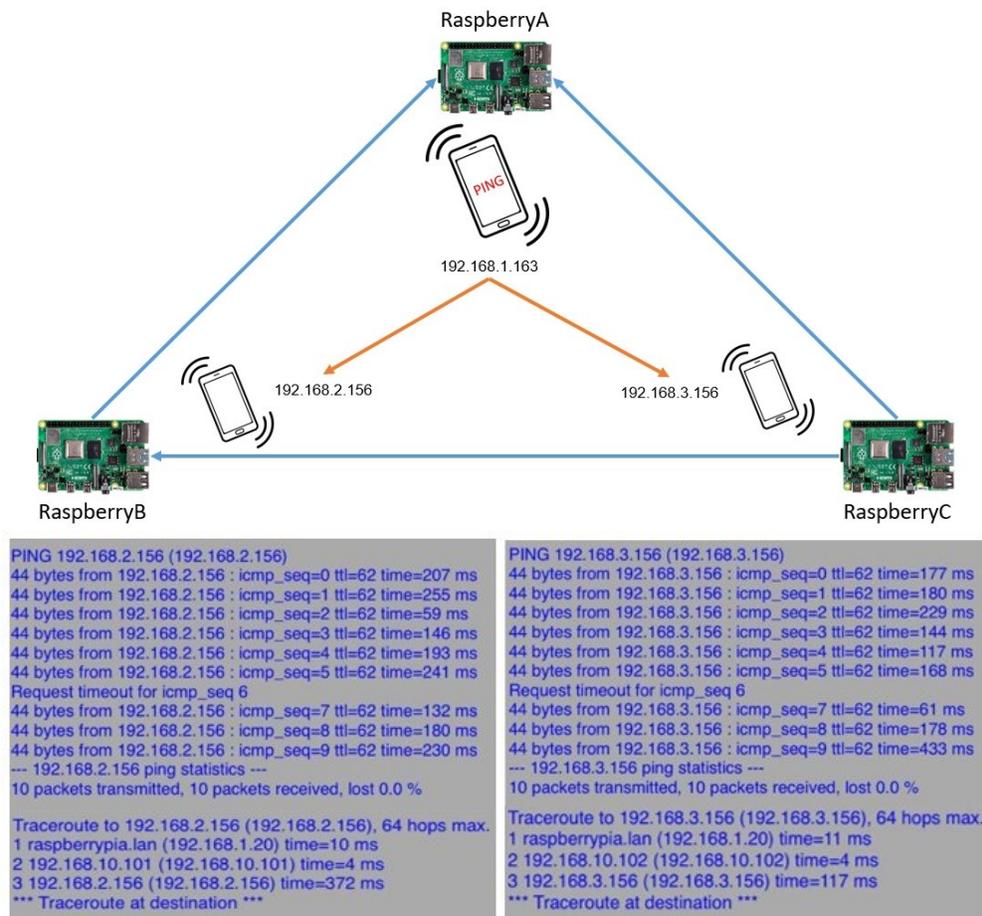
```

**Fuente: Elaboración Propia**

### 5.3.2. Red de Emergencia

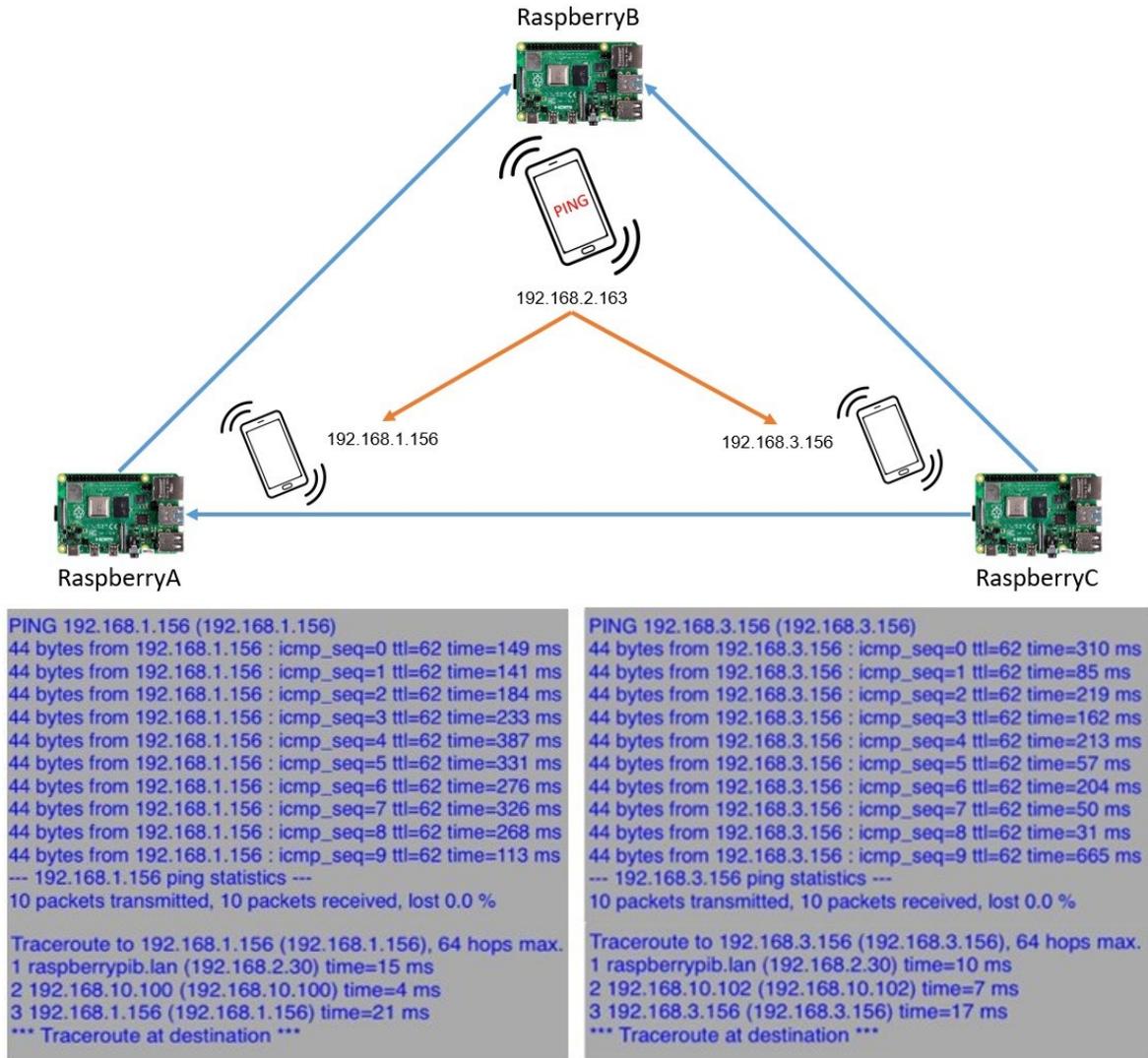
Por medio de la aplicación Ping Lite se realizaron pings y trazas desde un dispositivo conectado a la RaspberryA hacia un dispositivo conectado a la RaspberryB y a la RaspberryC. Estas mismas pruebas se realizaron con las demás Raspberrys. Como se observa en la **Figura 55**, **Figura 56** y **Figura 57** todos los ping responde correctamente en cada uno de los dispositivos y las trazas nos demuestran el camino que un paquete realiza para llegar a su destino.

**Figura 55** Ping desde un dispositivo conectado a la RaspberryA hacia un dispositivo conectado a la RaspberryB (izquierda) y RaspberryC (derecha)



**Fuente: Elaboración Propia**

**Figura 56** Ping desde de la RaspberryB hacia un dispositivo conectado a la RaspberryA y un dispositivo conectado a la RaspberryC



```

PING 192.168.1.156 (192.168.1.156)
44 bytes from 192.168.1.156 : icmp_seq=0 ttl=62 time=149 ms
44 bytes from 192.168.1.156 : icmp_seq=1 ttl=62 time=141 ms
44 bytes from 192.168.1.156 : icmp_seq=2 ttl=62 time=184 ms
44 bytes from 192.168.1.156 : icmp_seq=3 ttl=62 time=233 ms
44 bytes from 192.168.1.156 : icmp_seq=4 ttl=62 time=387 ms
44 bytes from 192.168.1.156 : icmp_seq=5 ttl=62 time=331 ms
44 bytes from 192.168.1.156 : icmp_seq=6 ttl=62 time=276 ms
44 bytes from 192.168.1.156 : icmp_seq=7 ttl=62 time=326 ms
44 bytes from 192.168.1.156 : icmp_seq=8 ttl=62 time=268 ms
44 bytes from 192.168.1.156 : icmp_seq=9 ttl=62 time=113 ms
--- 192.168.1.156 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

Traceroute to 192.168.1.156 (192.168.1.156), 64 hops max.
1 raspberrypi.ban (192.168.2.30) time=15 ms
2 192.168.10.100 (192.168.10.100) time=4 ms
3 192.168.1.156 (192.168.1.156) time=21 ms
*** Traceroute at destination ***
  
```

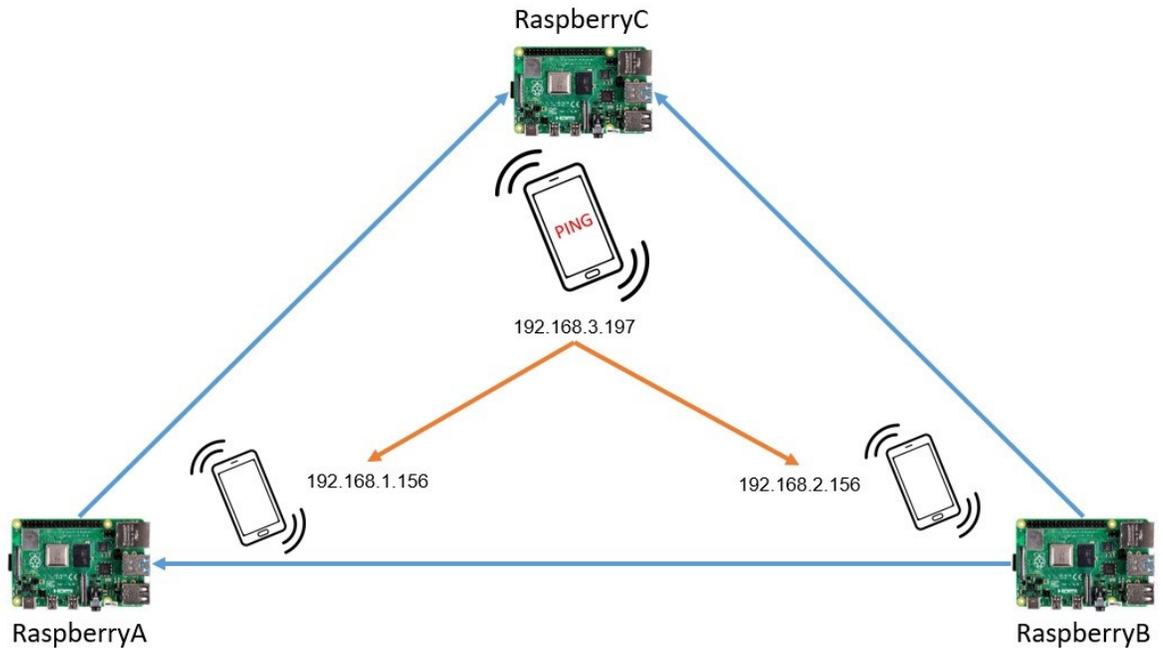
```

PING 192.168.3.156 (192.168.3.156)
44 bytes from 192.168.3.156 : icmp_seq=0 ttl=62 time=310 ms
44 bytes from 192.168.3.156 : icmp_seq=1 ttl=62 time=85 ms
44 bytes from 192.168.3.156 : icmp_seq=2 ttl=62 time=219 ms
44 bytes from 192.168.3.156 : icmp_seq=3 ttl=62 time=162 ms
44 bytes from 192.168.3.156 : icmp_seq=4 ttl=62 time=213 ms
44 bytes from 192.168.3.156 : icmp_seq=5 ttl=62 time=57 ms
44 bytes from 192.168.3.156 : icmp_seq=6 ttl=62 time=204 ms
44 bytes from 192.168.3.156 : icmp_seq=7 ttl=62 time=50 ms
44 bytes from 192.168.3.156 : icmp_seq=8 ttl=62 time=31 ms
44 bytes from 192.168.3.156 : icmp_seq=9 ttl=62 time=665 ms
--- 192.168.3.156 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

Traceroute to 192.168.3.156 (192.168.3.156), 64 hops max.
1 raspberrypi.ban (192.168.2.30) time=10 ms
2 192.168.10.102 (192.168.10.102) time=7 ms
3 192.168.3.156 (192.168.3.156) time=17 ms
*** Traceroute at destination ***
  
```

**Fuente: Elaboración Propia**

**Figura 57** Ping desde RaspberryC hacia un dispositivo conectado a la RaspberryA y un dispositivo conectado a la RaspberryB



```
PING 192.168.1.156 (192.168.1.156)
44 bytes from 192.168.1.156 : icmp_seq=0 ttl=62 time=127 ms
44 bytes from 192.168.1.156 : icmp_seq=1 ttl=62 time=25 ms
44 bytes from 192.168.1.156 : icmp_seq=2 ttl=62 time=122 ms
44 bytes from 192.168.1.156 : icmp_seq=3 ttl=62 time=16 ms
44 bytes from 192.168.1.156 : icmp_seq=4 ttl=62 time=153 ms
44 bytes from 192.168.1.156 : icmp_seq=5 ttl=62 time=130 ms
44 bytes from 192.168.1.156 : icmp_seq=6 ttl=62 time=366 ms
44 bytes from 192.168.1.156 : icmp_seq=7 ttl=62 time=29 ms
44 bytes from 192.168.1.156 : icmp_seq=8 ttl=62 time=51 ms
44 bytes from 192.168.1.156 : icmp_seq=9 ttl=62 time=32 ms
--- 192.168.1.156 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

Traceroute to 192.168.1.156 (192.168.1.156), 64 hops max.
1 raspberrypic.lan (192.168.3.40) time=12 ms
2 192.168.10.100 (192.168.10.100) time=4 ms
3 192.168.1.156 (192.168.1.156) time=15 ms
*** Traceroute at destination ***
```

```
PING 192.168.2.156 (192.168.2.156)
Request timeout for icmp_seq 0
44 bytes from 192.168.2.156 : icmp_seq=1 ttl=62 time=43 ms
44 bytes from 192.168.2.156 : icmp_seq=2 ttl=62 time=208 ms
44 bytes from 192.168.2.156 : icmp_seq=3 ttl=62 time=144 ms
44 bytes from 192.168.2.156 : icmp_seq=4 ttl=62 time=128 ms
44 bytes from 192.168.2.156 : icmp_seq=5 ttl=62 time=11 ms
44 bytes from 192.168.2.156 : icmp_seq=6 ttl=62 time=172 ms
44 bytes from 192.168.2.156 : icmp_seq=7 ttl=62 time=15 ms
44 bytes from 192.168.2.156 : icmp_seq=8 ttl=62 time=159 ms
44 bytes from 192.168.2.156 : icmp_seq=9 ttl=62 time=257 ms
--- 192.168.2.156 ping statistics ---
10 packets transmitted, 10 packets received, lost 0.0 %

Traceroute to 192.168.2.156 (192.168.2.156), 64 hops max.
1 raspberrypic.lan (192.168.3.40) time=6 ms
2 192.168.10.101 (192.168.10.101) time=7 ms
3 192.168.2.156 (192.168.2.156) time=115 ms
*** Traceroute at destination ***
```

**Fuente: Elaboración Propia**

- **Pruebas de funcionamiento de la red en caso de falla de una Raspberry**

Con las tres Raspberrys encendidas, como se observa en la **Figura 58**, todas las rutas se encuentran operativas. Al apagar una de las Raspberry, las rutas relacionadas con el dispositivo que está indisponible cambian de color, como se observa en la **Figura 59**, para finalmente quedar solo las rutas activas en ese momento, como se observa en la **Figura 60**. Con esta prueba se pudo confirmar que nuestra red con enrutamiento dinámico está operando satisfactoriamente ya que si se apaga o se vuelve a encender algún nodo, o inclusive si se adiciona uno nuevo, este automáticamente entrará a operar con la configuración que se tiene en la red. Esta es una ventaja muy significativa del enrutamiento dinámico frente al enrutamiento estático.

**Figura 58 Rutas con las 3 Raspberrys encendidas**

The screenshot shows the RaspberryPiA network management interface. At the top, there is a navigation bar with 'RaspberryPiA' and several menu items: Status, System, Services, Network, and Logout. An 'AUTO REFRESH ON' button is visible on the right. Below the navigation bar, there are several tabs: Neighbours, Routes (selected), Topology, HNA, MID, Interfaces, and OLSR-Viz. The main content area is titled 'Known OLSR routes' and 'Overview of currently known routes to other OLSR nodes'. It contains a table with the following data:

Announced network	OLSR gateway	Interface	Metric	ETX
192.168.2.0/24	192.168.10.101	wlan1	1	1.062
192.168.10.101/32	192.168.10.101	wlan1	1	1.062
192.168.2.30/32	192.168.10.101	wlan1	1	1.062
192.168.3.40/32	192.168.10.102	wlan1	1	1.188
192.168.3.0/24	192.168.10.102	wlan1	1	1.188
192.168.10.102/32	192.168.10.102	wlan1	1	1.188

**Fuente: Elaboración Propia**

**Figura 59** Estado de las rutas al apagar una de las Raspberrys

The screenshot shows the RaspberryPiA network status page. The 'Routes' tab is selected. The page title is 'Known OLSR routes' and the subtitle is 'Overview of currently known routes to other OLSR nodes'. The table below lists the routes with their respective metrics and ETX values.

Announced network	OLSR gateway	Interface	Metric	ETX
192.168.10.102/32	192.168.10.102	wlan1	1	1.062
192.168.3.40/32	192.168.10.102	wlan1	1	1.062
192.168.3.0/24	192.168.10.102	wlan1	1	1.062
192.168.2.30/32	192.168.10.101	wlan1	1	2.026
192.168.10.101/32	192.168.10.101	wlan1	1	2.026
192.168.2.0/24	192.168.10.101	wlan1	1	2.026

**Fuente: Elaboración Propia**

**Figura 60** Rutas con dos Raspberrys encendidas

The screenshot shows the RaspberryPiA network status page. The 'Routes' tab is selected. The page title is 'Known OLSR routes' and the subtitle is 'Overview of currently known routes to other OLSR nodes'. The table below lists the routes with their respective metrics and ETX values.

Announced network	OLSR gateway	Interface	Metric	ETX
192.168.3.0/24	192.168.10.102	wlan1	1	1.000
192.168.3.40/32	192.168.10.102	wlan1	1	1.000
192.168.10.102/32	192.168.10.102	wlan1	1	1.000

**Fuente: Elaboración Propia**

## 6. CONCLUSIONES

Como resultado de este trabajo de grado, se ha logrado poner en marcha un prototipo de una red mesh conformada por dispositivos Raspberry Pi, útil en escenarios de postdesastre cuando las redes de telefonía celular queden fuera de servicio. A la red de acceso de la red mesh, tal como se ha demostrado, se podrían conectar los teléfonos inteligentes de los sobrevivientes, lo cual facilitaría la colaboración entre estas personas y ayudaría a los rescatistas a establecer comunicación con quienes se encuentren en el área del desastre.

Con base en las dos configuraciones de la red que se evaluaron, una utilizando enrutamiento estático y otra con enrutamiento dinámico (OLSR), se demostró que el enrutamiento dinámico resultó ser más efectivo y fácil de configurar, el cual permitirá, además, añadir más nodos de conexión de una forma más sencilla.

Es importante destacar que en las pruebas iniciales utilizando enrutamiento estático, se pudo observar que las configuraciones tipo Bridge (puente) entre las interfaces no daban buenos resultados porque los tiempos de respuesta entre dispositivos eran altos, próximos o superiores a 1 segundo, inclusive estando muy cerca. Por tal motivo, esta configuración no se tuvo en cuenta y se eligió la configuración con enrutamiento estático que se explicó en este documento.

Por otra parte, el escenario en el que se realizaron las pruebas de alcance presentaba obstáculos, como paredes, y esto afectó la propagación de la señal. También se pudo observar cómo al encontrarse varias redes WiFi activas en las cercanías, y al tener varios usuarios conectados a la red WiFi de la casa, se

afectaban los resultados de las pruebas, alcanzando pérdidas hasta del 70% de los paquetes. A partir de los resultados, fue claro que esta red debe desplegarse en un campo abierto, sin obstáculos entre los dispositivos Raspberry.

El desarrollo del proyecto demuestra que los objetivos se cumplieron, por el nivel de conocimiento que se adquirió sobre los dispositivos Raspberry, sobre Linux LEDE y sus posibilidades, y por el proceso de configuración tanto del enrutamiento estático como dinámico, a partir del cual fue posible concluir que la mejor opción para la red propuesta es el enrutamiento con OLSR.

Como trabajo futuro se propone determinar el alcance máximo de la red en un campo abierto, evaluando el desempeño de la misma cuando se conecten más nodos. Adicionalmente, sería interesante desarrollar una aplicación que permita saber qué usuarios se encuentran conectados a la red mesh, para poder establecer comunicación, por ejemplo mediante chat, con algunos de ellos.

## 7. BIBLIOGRAFÍA

14, E. (n.d.). Wi-Pi – WLAN module for the Raspberry Pi.

*About the LEDE project.* (2017). Obtenido de A reboot of the OpenWrt community:  
<https://lede.readthedocs.io/en/latest/about.html>

GSMA, Association. (2019). *The Mobile Economy*. Retrieved from  
<https://www.gsmaintelligence.com/research/?file=b9a6e6202ee1d5f787cfeb-b95d3639c5&download>

Basagni, S., Conti, M., Giordano, S., & Stojmenovic, I. (2013). *Mobile Ad Hoc Networking: cutting edge directions (2da edition)*. IEEE Press.

BAUTISTA, G. A., G. A., & J. S. (2015). *TELEMETRÍA SOBRE RASPBERRY PI*. Bogotá.

Buettrich, S. (2007). *REDES MESH*.

Castañeda, C. (2019). *Aplicación móvil para la comunicación de sobrevivientes de un desastre*. Bogotá

Ceballos, H. Z. (2013). *Diseño de la Red De Nueva Generación Tipo Mesh, para la Atención Y Prevención De Emergencias En Bogotá, En La Banda De 4,9 Ghz, Bajo La Resolución 1661 de 18 De Julio De 2006 para la Coordinación De Acciones De Socorro Y Salvamento*. Bogotá.

Coñapes, S. (2015, Marzo. 31 ). *Tipos De Redes inalámbricas*. Retrieved from  
<https://sites.google.com/site/redesinalambricas3/tipos-de-redes-inalambricas>

Digital, M. (2017, 09 22). *¿Por qué son importantes las 72 horas luego de un sismo?*  
Retrieved from <https://www.milenio.com/estados/por-que-son-importantes-las-72-horas-luego-de-un-sismo>

DNP, P. W. ( 2015, Mayo 25). *3.181 muertos y 12,3 millones de afectados: las cifras de desastres naturales entre 2006 y 2014*. Retrieved from

<https://www.dnp.gov.co/Paginas/3-181-muertos,-21-594-emergencias-y-12,3-millones-de-afectados-las-cifras-de-los-desastres-naturales-entre-2006-y-2014-.aspx>

Douglas S. J. De Couto and Robert T. Morris. (2004). *High-Throughput Routing for Multi-Hop Wireless Networks*. Ph.D. Dissertation. Massachusetts Institute of Technology, USA.

Fernández, V. N. ( 2005 , septiembre 5 ). *Redes mesh basadas en puntos de acceso inteligentes 802.11 open source (II)* . Retrieved from <https://upcommons.upc.edu/bitstream/handle/2099.1/6829/mem%c3%b2ria.pdf?sequence=1&isAllowed=y>

Granados Bayona , G., & Pérez, T. (s.f.). REDES MESH. Obtenido de [http://www.adminso.es/recursos/Proyectos/PFM/2010\\_11/PFM\\_mesh.pdf](http://www.adminso.es/recursos/Proyectos/PFM/2010_11/PFM_mesh.pdf)

Hernández, M. F., J. L., R. D., J. D., & N. S. (2018). Raspberry PI 3 y pcDUINO. In *Tutorial de instalación y configuración*. Bogotá, Colombia.

Keagy, S. (2001). *Integración de redes de voz y datos*. Cisco Press. Pearson Educación.

Leonard, P. (2018, Junio 27 – 29.). *Situación de las estadísticas e indicadores de eventos extremos y desastres*. Retrieved from Estadísticas Ambientales, División de Estadísticas: [https://www.cepal.org/sites/default/files/presentations/2018-06-2areu-expertos-ea-4\\_2-cepal-pleonard.pdf](https://www.cepal.org/sites/default/files/presentations/2018-06-2areu-expertos-ea-4_2-cepal-pleonard.pdf)

MAROCHO, F. W. (2017). *DISEÑO DE UN SISTEMA DE TELECOMUNICACIONES CON REDES AD HOC DE DRONES COMO ALTERNATIVA DE MEDIO DE COMUNICACIÓN PARA HACER FRENTE A DESASTRES NATURALES*. In F. W. MAROCHO. San Miguel.

Morales, L., Matus, F., & Arias, M. R. (2016). *Ad hoc Mesh Network Design for telemedicine and Natural Disaster Applications in Rural Areas of Nicaragua*. Universidad Nacional de Ingeniería, Managua, Nicaragua

MUNDIAL, B. (2012). *Análisis de la gestión del riesgo de desastres en Colombia*. Retrieved from Un aporte para la construcción de políticas públicas:

- <http://gestiondelriesgo.gov.co/sigpad/archivos/GESTIONDELRIESGOWEB.pdf>
- mundo, E. (2008, 05 12). *Terremoto en China*. Retrieved from <https://www.dw.com/es/terremoto-en-china/a-3331283>
- mundo, E. (2015 , 04 27). *El terremoto de Nepal se cobra ya más de 4.300 vidas y deja sin comunicaciones al país*. Retrieved from <https://www.elmundo.es/internacional/2015/04/26/553c926922601d20568b4570.html>
- PANDA, K. G., DAS, S., SEN, D., & ARIF, W. (2019, Julio 29). Design and Deployment of UAV-Aided Post-Disaster Emergency Network. *IEEE Access*, p. 15.
- Pastor, F. J. (Diciembre de 2008). *Estudio de viabilidad de la utilización de redes inalámbricas Ad-Hoc en edificios departamentales*. Valencia. Obtenido de Estudio de viabilidad de: [https://riunet.upv.es/bitstream/handle/10251/13183/TesinaMaster\\_FcoJavierHidalgo.pdf?sequence=1](https://riunet.upv.es/bitstream/handle/10251/13183/TesinaMaster_FcoJavierHidalgo.pdf?sequence=1)
- Rosa, S. L. (2017). *Diseño de una red Mesh de UAVs para proporcionar servicios de comunicaciones*. Sevilla: Departamento de Ingeniería electrónica Escuela Técnica Superior de Ingeniería Universidad de Sevilla. Sevilla.
- Rozo, V. J., & J. C. (2016). *Diseño de un Modelo de Infraestructura para Redes Mesh en Entornos Comunitarios o Rurales de Colombia: Como Apoyo al Proyecto de Investigación“Formulación de una Metodología para Diseñar e Implementar Redes Mesh como Alternativa de Solución para Redes Comunit*. In V. J. Rozo, & J. C.. Bogotá.
- SALAZAR, J. (n.d.). *REDES INALÁMBRICAS*. Retrieved from [https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01\\_R\\_ES.pdf](https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf)
- 
- ZAMBRANO, M. K., & ZAMBRANO, M. D. (2017). *AFRONTAMIENTO DEL CUIDADOR FAMILIAR DE LOS NIÑOS Y NIÑAS MENORES DE 5 AÑOS DE EDAD POSTERIOR AL TERREMOTO DEL 16A EN LA PARROQUIA COLON DE OCTUBRE A DICIEMBRE 2016*. Retrieved from

<http://repositorio.puce.edu.ec/bitstream/handle/22000/14181/TESIS%20MARIUXI%20ZAMBRANO%20Y%20MARIA%20LÓPEZ.%201.pdf?sequence=1&isAllowed=y>