



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería Mecánica Eléctrica

**CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA
HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLEOPERADORES
EN EL ÁREA RURAL DE GUATEMALA**

Victor Alexander Figueroa Oliva

Asesorado por el Ing. José Anibal Silva de los Ángeles

Guatemala, septiembre de 2019

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK
PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA
LOS CABLEOPERADORES EN EL ÁREA RURAL DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

VICTOR ALEXANDER FIGUEROA OLIVA
ASESORADO POR EL ING. JOSÉ ANIBAL SILVA DE LOS ÁNGELES

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO ELECTRÓNICO

GUATEMALA, SEPTIEMBRE DE 2019

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Luis Diego Aguilar Ralón
VOCAL V	Br. Christian Daniel Estrada Santizo
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADOR	Ing. Carlos Eduardo Guzmán Salazar
EXAMINADOR	Ing. Walter Giovanni Álvarez Marroquín
EXAMINADOR	Ing. José Aníbal Silva de los Ángeles
SECRETARIA	Inga. Lesbia Magalí Herrera López

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLEOPERADORES EN EL ÁREA RURAL DE GUATEMALA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica, con fecha 9 de mayo de 2019.



Victor Alexander Figueroa Oliva

Guatemala 9 de mayo de 2019

Ingeniero
Julio Cesar Solares Peñate
Coordinador del Área de Electrónica
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC

Apreciable Ingeniero Solares.

Me permito dar aprobación al trabajo de graduación del título **"CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLE-OPERADORES EN EL ÁREA RURAL DE GUATEMALA"**, del señor **Victor Alexander Figueroa Oliva**, por considerar que cumple con los requisitos establecidos.

Por tanto, el autor de este trabajo de graduación y, yo, como su asesor, nos hacemos responsables del contenido y conclusiones del mismo.

Sin otro particular, me es grato saludarle.

Atentamente,


JOSE ANIBAL SILVA DE LOS ANGELES
ING. ELECTRONICO
Ing. José Anibal Silva de los Angeles
COLEGIADO 5067

Colegiado 5067

Asesor

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA

REF. EIME 30. 2019.

14 de mayo 2019.

Señor Director


Ing. Otto Fernando Andrino González
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC.

Señor Director:

Me permito dar aprobación al trabajo de Graduación titulado: **CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLE-OPERADORES EN EL ÁREA RURAL DE GUATEMALA**, del estudiante; Victor Alexander Figueroa Oliva, que cumple con los requisitos establecidos para tal fin.

Sin otro particular, aprovecho la oportunidad para saludarle.

Atentamente,
ID Y ENSEÑAD A TODOS

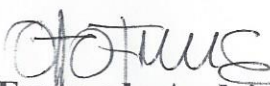

Ing. Julio César Solares Peñate
Coordinador de Electrónica





REF. EIME 31. 2019.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto bueno del Coordinador de Área, al trabajo de Graduación de el estudiante: VICTOR ALEXANDER FIGUEROA OLIVA titulado: CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLE-OPERADORES EN EL ÁREA RURAL DE GUATEMALA, procede a la autorización del mismo.


Ing. Otto Fernando Andriño González



GUATEMALA, 3 DE JUNIO 2019.



DTG. 384.2019

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica, al Trabajo de Graduación titulado: **CONSOLIDACIÓN DE TRÁFICO MULTICAST CON EQUIPO MIKROTIK PARA LA HOMOLOGACIÓN DE SEÑAL DE TELEVISIÓN DIGITAL PARA LOS CABLEOPERADORES EN EL ÁREA RURAL DE GUATEMALA**, presentado por el estudiante universitario: **Victor Alexander Figueroa Oliva**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:



Inga Anabela Cordova Estrada
Decana

Guatemala, septiembre de 2019

/gdech

ACTO QUE DEDICO A:

Dios	Por darme la fuerza, así como la voluntad para levantarme en los momentos más difíciles.
Mi padre	Victor Hugo Figueroa, por ser un ejemplo de lucha y perseverancia en mi vida.
Mi madre	Sonia Eugenia Oliva, por ser mi fuerza para seguir adelante cada día.
Mis hermanas	Ana María Figueroa y Yenifer Figueroa, por inspirarme a superarme como profesional.
Mi tío	Miguel Ángel Oliva, por ser la mayor influencia en mí e inspiración en mi carrera.
Mi novia	Jennifer Quiñonez, por apoyarme incondicionalmente durante mis estudios y ser mi compañera de vida durante más de 10 años.
Mi prima	Ana Luisa Palacios, porque tu recuerdo siempre vive en mí a pesar de no estar físicamente.
Mis amigos	Felix Concohá, Carlos Maldonado, Balam Lol, Joshua Castillo, Kevin Rodas, Ricardo Sontay, Oscar Cahueque, José del Cid, Enrique Sontay

Cristian Ramírez, Luis Garci Aguirre, Julio Cristales, Carlos Pineda y todos aquellos con quienes compartí experiencias.

Los ingenieros

De EIME por transmitirme sus conocimientos tan útiles, también por orientarme cuando fue necesario y en especial al Ing. José Anibal Silva por ser el asesor de este trabajo.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser mi casa de estudios, lugar donde crecí como persona y estudiante.
Facultad de Ingeniería	Porque siempre sacó lo mejor de mí, lugar donde viví las mejores experiencias de mi vida.
Mis amigos de la Facultad	Dado que sin ellos mis proyectos nunca hubiesen funcionado, con quienes compartí desvelos y experiencias.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
LISTA DE SÍMBOLOS	XI
GLOSARIO	XIII
RESUMEN.....	XIX
OBJETIVOS.....	XXI
INTRODUCCIÓN.....	XXIII
1. DISEÑO DE TOPOLOGÍA DE RED PARA EL TRANSPORTE DE TRÁFICO MULTICAST	1
1.1. Modelo OSI.....	1
1.2. Análisis de las 7 capas el modelo OSI.....	2
1.2.1. Capa 1: medios de transmisión de datos.....	2
1.2.1.1. Ethernet.....	2
1.2.1.2. Medios eléctricos.....	3
1.2.1.3. Medios ópticos.....	4
1.2.2. Capa 2: análisis de los dispositivos de <i>switching</i> , su funcionamiento y direccionamiento físico.....	6
1.2.2.1. Dominios de colisión.....	6
1.2.2.2. <i>Hub</i>	7
1.2.2.3. <i>Switch</i>	7
1.2.3. Capa 3: direccionamiento lógico a través de subredes.....	9
1.2.3.1. Router.....	9
1.2.3.2. Tipos de enrutamiento	10

1.2.4.	Capa 4: protocolos de comunicación y transmisión de datos	11
1.2.4.1.	TCP	11
1.2.4.2.	UDP	12
1.2.5.	Capa 5: conservación, apertura y cierre de sesiones	12
1.2.6.	Capa 6: presentación de datos.....	13
1.2.7.	Capa 7: aplicaciones y su interacción con el usuario.....	13
1.3.	TCP/IP	14
1.3.1.	IPv4	14
1.3.2.	Segmentación IPv4	16
1.4.	Máscara de subred.....	17
1.5.	Diseño preliminar de red de transporte	18
1.5.1.	Aspectos importantes.....	18
1.5.2.	Análisis de tráfico <i>multicast</i>	18
1.5.3.	Equipos a utilizar	20
1.5.4.	Diagrama de topología lógica.....	20
2.	UNIFICACIÓN DE TRÁFICO MULTICAST EN UN SOLO NODO DE DISTRIBUCIÓN	23
2.1.	Transporte y equipos.....	23
2.1.1.	FTTx.....	23
2.1.2.	Aspectos a tomar en cuenta sobre los equipos.....	24
2.1.2.1.	Justificación de equipos	24
2.1.3.	Ancho de banda	25
2.2.	Diseño de nodo principal de distribución.....	26
2.2.1.	Integración de equipos	26
2.2.1.1.	Enlace punto a punto	28

	2.2.1.2.	Enlaces secundarios.....	28
	2.2.2.	Pruebas de conectividad	28
2.3.		Problemática de integración <i>headends</i> secundarios	30
	2.3.1.	Solución al problema	30
	2.3.2.	Herramientas a utilizar	31
3.		DESARROLLO E IMPLEMENTACIÓN DE CONFIGURACIÓN EN ROUTER OS DE MIKROTIK PARA TRANSPORTE DE TRÁFICO MULTICAST	33
3.1.		Introducción a Mikrotik.....	33
	3.1.1.	Reseña histórica	33
	3.1.2.	Introducción a RouterOS	34
	3.1.2.1.	Características.....	34
	3.1.2.2.	RouterBOARD y arquitecturas.....	34
	3.1.2.3.	Nomenclatura RouterBOARD	35
	3.1.2.4.	Winbox.....	38
	3.1.2.5.	WebFig	40
	3.1.2.6.	Otras formas de acceso y servicios	41
	3.1.3.	Conceptos básicos	42
	3.1.3.1.	Direccionamiento	42
	3.1.3.2.	Enrutamiento	44
	3.1.3.3.	Puente	47
	3.1.3.4.	<i>Firewall</i>	49
	3.1.3.5.	Copia de respaldo.....	51
3.2.		Requerimientos de la configuración	52
	3.2.1.	Planteamiento de la configuración.....	53
	3.2.2.	Máxima unidad de transferencia (MTU).....	53
	3.2.3.	Implementación de puente	54
	3.2.4.	Posibles dificultades	55

3.3.	Desarrollo de configuración.....	56
3.3.1.	Configuración <i>headend</i>	56
3.3.2.	Configuración nodo de distribución	57
3.3.3.	Configuración cliente final	58
4.	IMPLEMENTACIÓN DE RED Y APLICACIÓN DE FILTROS PARA ENLACES REDUNDANTES	59
4.1.	Problemática de integración <i>headend</i> secundario	59
4.1.1.	Implementación de <i>firewall</i>	60
4.1.1.1.	Filtros a nivel de capa 3 del modelo OSI	60
4.1.2.	<i>Address list</i>	63
4.1.3.	Diseño e implementación de filtros.....	64
4.2.	Implementación de red de gestión	66
4.3.	Implementación de seguridad en la red	68
4.3.1.	Políticas de <i>firewall</i>	68
4.3.2.	Descubrimiento de vecinos	69
4.3.3.	Restricción de ARP	72
4.3.4.	Homologación de servicios y puertos.....	73
4.3.5.	Sistemas de protección	77
4.3.5.1.	<i>Intrusion detection system (IDS)</i>	77
4.3.5.2.	<i>Intrusion protection system (IPS)</i>	77
4.3.6.	Implementación de IDS	78
4.3.6.1.	Snort.....	78
4.3.6.2.	Integración de snort con mikrotik.....	79
4.3.7.	Instalación de Snort.....	82
4.3.7.1.	Windows.....	82
4.3.7.2.	Linux.....	89
4.3.8.	Radius	96

4.4.	Diseño final de la red.....	100
4.5.	Aspectos y configuraciones finales.....	102
4.6.	Ventajas y desventajas de la implementación final de red	106
5.	ANÁLISIS TÉCNICO FINANCIERA DE IMPLEMENTACIÓN	109
5.1.	<i>Headend</i>	109
5.2.	Nodo central	112
5.3.	Transporte de datos	113
5.4.	Cliente final.....	114
5.5.	Análisis financiero.....	115
5.5.1.	Rentabilidad.....	115
5.5.2.	Solvencia	119
5.5.2.1.	Activo.....	119
5.5.2.2.	Pasivo.....	119
5.5.3.	Liquidez	121
5.5.4.	TIR VAN	122
	CONCLUSIONES	127
	RECOMENDACIONES.....	129
	BIBLIOGRAFÍA.....	131
	APÉNDICE.....	133

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Estándar TIA/EIA T568-A y T568-B	3
2.	Conectores de fibra óptica	5
3.	Terminaciones APC/UPC	6
4.	Simbología del <i>switch</i>	8
5.	Simbología del <i>router</i>	10
6.	Red preliminar de distribución.....	21
7.	Mikrotik CCR 1036-12G-4S.....	25
8.	Diseño de red con múltiples señales de origen	26
9.	Red de distribución.....	27
10.	Prueba de conectividad a nivel de capa 2.....	29
11.	<i>Firewall</i> mikrotik	31
12.	Primera vista de Winbox	39
13.	Primera vista WebFig	40
14.	Manejo de direcciones RouterOS	43
15.	Agregando una dirección IP	44
16.	Tabla de rutas	45
17.	Agregando una ruta.....	46
18.	Puente.....	48
19.	Agregando puertos a un puente.....	49
20.	Primera vista <i>firewall</i>	51
21.	Copia de respaldo	52
22.	Código <i>headend</i>	56
23.	Código nodo de distribución.....	57

24.	Código cliente final.....	58
25.	Filtro <i>firewall</i>	62
26.	Parámetros finales filtro <i>firewall</i>	63
27.	<i>Address list</i>	64
28.	Filtro simple	65
29.	IP Firewall Bridge.....	66
30.	Red de gestión.....	67
31.	Deshabilitando comunicación entre clientes	69
32.	Restricción de Neighbor Discovery vía CLI.....	70
33.	Restricción de Neighbor Discovery vía GUI.....	71
34.	Deshabilitación ARP vía GUI	72
35.	Deshabilitación ARP vía CLI.....	73
36.	Listado de servicios vía GUI	74
37.	Habilitación y des habilitación de servicios vía CLI.....	74
38.	Listado de servicios vía CLI	75
39.	Editando servicios vía GUI.....	76
40.	Editando servicios vía CLI	76
41.	Integración de snort en la red	79
42.	Configuración Switch Mikrotik.....	80
43.	Espejo entre puertos.....	81
44.	Instalación Snort en Windows.....	83
45.	Primera vista Snort en Windows	84
46.	Directorio de Snort en Windows	84
47.	Descomprimiendo las reglas en Windows	85
48.	Comando de prueba en Windows.....	88
49.	Ejecutando Snort sobre una interfaz en Windows	88
50.	Comando de prueba con log en consola en Windows.....	89
51.	Upgrade	89
52.	Instalaciones previas en Linux.....	90

53.	Instalación Daq en Linux	90
54.	Instalación Snort en Linux	91
55.	Comando de prueba en Linux	91
56.	Prueba Snort en Linux.....	92
57.	Creación de directorios en Linux.....	92
58.	Comentando reglas del archivo snort.conf en Linux	94
59.	Comando de prueba sobre una interface en Linux	95
60.	Ejecutando Snort sobre una interfaz en Linux.....	95
61.	Instalación User Manager Mikrotik	96
62.	Gestión de usuarios interfaz web	97
63.	Agregando IP y contraseña de conexión.....	97
64.	Perfiles de usuarios.....	98
65.	Agregando usuarios	99
66.	Configuración CPE.....	99
67.	Prueba radius.....	100
68.	Diseño final de red	101
69.	Configuración final <i>headend 1</i>	102
70.	Configuración final <i>headend 2</i>	103
71.	Configuración final <i>headend 3</i>	103
72.	Configuración final nodo de distribución.....	104
73.	Configuración final cliente	105
74.	Red de gestión	106
75.	Estructura <i>headend</i>	110
76.	Fórmula rentabilidad.....	116
77.	Fórmula precio de venta.....	117
78.	Ecuación de solvencia.....	120
79.	Fórmula, liquidez.....	121
80.	Formula VAN.....	122
81.	Gráfica VAN múltiples <i>headends</i>	124

82.	Gráfica VAN, topología final.....	125
-----	-----------------------------------	-----

TABLAS

I.	Clasificación IPv4.....	15
II.	Segmentos de IP privados.....	16
III.	Máscaras de subred	17
IV.	Resultados análisis tráfico <i>multicast</i>	19
V.	Arquitecturas soportadas por RouterBOARD	35
VI.	Estándar de nomenclatura RouterBOARD	36
VII.	Puertos por defecto servicios Mikrotik	41
VIII.	Costos equipos <i>headend</i>	111
IX.	Costos equipos nodo de distribución	112
X.	Costo de transporte	114
XI.	Costo entrega de servicio	115
XII.	Rentabilidades múltiples <i>headends</i>	117
XIII.	Rentabilidad topología final.....	118
XIV.	Activos mínimos múltiples <i>headends</i>	121
XV.	Activos corrientes mínimos topología final.....	121
XVI.	Cálculo VAN/TIR múltiples Headends	123
XVII.	Cálculo VAN/TIR, topología final.....	124

LISTA DE SÍMBOLOS

Símbolo	Significado
A	Activos, representan todos los bienes que posee una entidad
C	Costo, costo de un producto
Q	Flujo de efectivo anual utilizado en la fórmula del VAN
r	Interés anual utilizado en la fórmula del VAN
I	Inversión inicial del proyecto
L	Liquidez, determina la capacidad de una empresa para afrontar sus obligaciones a corto plazo
P	Pasivos, representan todos los gastos y endeudamientos de una entidad
P	Precio de venta de un producto
R	Rentabilidad, determina si una entidad es capaz de generar ganancias cubriendo también sus costos de operación
S	Solvencia, determina la capacidad de una empresa para afrontar sus obligaciones a largo plazo
Σ	Sumatoria, indica la sumatoria de valores desde un valor inicial hasta un valor final, generalmente dados por un índice y un subíndice

GLOSARIO

API	Conjunto de funciones para ser utilizadas por otros softwares.
ASCII	Acrónimo en inglés de <i>american standard code for information interchange</i> , especifica un código estándar para el intercambio de información.
<i>Bandwidth test</i>	Herramienta disponible en RouterOS para saturar un enlace con tráfico en UDP o TCP.
<i>Bit</i>	Dígito binario, representa la unidad más básica de un sistema informático.
Bug	Utilizado en informática para hacer referencia a un error de un programa.
Cifrado	Alteración de la estructura de un mensaje por medio de algoritmos, que solo puede ser descifrada con la clave correspondiente.
Codificación	Método de conversión de un lenguaje natural a otro sistema o representación gráfica.
Conmutación	Establecer a conveniencia el enlace por el que debe transportarse el tráfico.

CPE	Siglas para <i>customer premises equipment</i> , hace referencia al equipo terminal que se encuentra en la sede de un cliente.
CSMA/CD	Acceso Múltiple con Escucha de Portadora y Detección de Colisiones, parte fundamental del funcionamiento del protocolo <i>Ethernet</i> .
Datagrama	Fracción de paquete que contiene la mínima información como para ser encaminado a través de la red.
Dominio de colisión	Segmento de red en donde las tramas pueden colisionar.
Enlace de datos	Enlace en el cual solamente se transportan datos, mas no se tiene salida hacia internet a través del mismo.
Enlace redundantes	Enlaces de datos o internet utilizados para casos de emergencia, realizando una función de <i>failover</i> .
Enlace punto a punto	Enlace de datos dedicado entre dos puntos específicos de una red.
Escalabilidad	Dimensionamiento de la red para soportar futuros crecimientos en la misma.

Ethernet	Estándar de redes basado en detección de colisiones y acceso múltiple con escucha de portadora CSMA/DC.
FTP	Siglas para <i>file transfer protocol</i> , es un protocolo utilizado para transferir archivos.
GUI	Siglas para <i>graphical user interface</i> , hace referencia a la interfaz gráfica para el usuario final.
HotSpot	Punto de red inalámbrica donde se ofrece acceso a internet a cambio de algún tipo de contraseña o registro.
IEEE	Instituto de Ingeniería Eléctrica y Electrónica, se encarga de la regulación y desarrollo sobre estas ramas de la ingeniería.
Impulso	Onda electromagnética que viaja a través de un conductor o medio óptico.
LAN	Red de área local por sus siglas en inglés <i>local area network</i> .
Mbps	Siglas para hace referencia a la velocidad de transmisión de datos en mega bits sobre unidad de tiempo.
MIT	Instituto Tecnológico de Massachusetts.

MPLS	Siglas para <i>multiprotocol label switching</i> , se trata de un estándar para transportar datos definido en el RFC 3031.
<i>Multicast</i>	Información que se envía a un grupo selecto de usuarios que se encuentra suscrito a un grupo de multidifusión.
NOC	Acrónimo para <i>network operation center</i> , es el departamento destinado al monitoreo y realizar acciones correctivas sobre la red.
OSI	<i>Open system interconnection</i> , modelo utilizado para que se comuniquen dos sistemas abiertos.
<i>Ping</i>	Herramienta utilizada para diagnóstico de problemas en una red, útil para comprobar conectividad con otro dispositivo.
PPP	Siglas para <i>point to point protocol</i> , utilizado para conectar enrutadores estableciendo túneles a través de la red.
Puerto	Interfaz física de un dispositivo de red, <i>switch</i> o <i>router</i> .
QoS	Siglas para <i>quality of service</i> , se utiliza para mejorar el rendimiento de la red.

RAM	Siglas para <i>random access memory</i> , haciendo referencia a la memoria temporal de un dispositivo.
Reflexión	Fenómeno óptico que ocurre cuando la luz choca con un medio y esta se desvía regresando al medio del cual salió.
RFC	Siglas para <i>request for comments</i> , una serie de publicaciones en internet.
Segmentación	Subdivisión de la red a conveniencia, según los requerimientos de configuración.
Sniffer	Herramienta utilizada para analizar tráfico dentro de una red.
SSL	Siglas para <i>secure socket layer</i> , es un protocolo de seguridad generalmente utilizado en páginas web para proteger la integridad de la información.
SSH	Protocolo de comunicación que utiliza encriptación para proteger la información transmitida.
STP	Utilizado con el mismo objetivo que el UTP, pero con blindaje.
Telnet	Protocolo de comulación sobre texto plano.

<i>Throughput</i>	Tasa de transferencia máxima para una interfaz física de un equipo.
<i>Traceroute</i>	Herramienta que sirve para revisar todo el camino de un paquete desde su origen hasta su destino, brindando información como tiempos de respuesta e incluso la IP remota.
<i>Transceiver</i>	Dispositivo utilizado para transmitir y recibir información en forma eléctrica u óptica.
<i>Unicode</i>	Estándar de codificación diseñado para facilitar el tratamiento y presentación de información.
UTP	Par de cable trenzado sin blindaje, se utiliza en telecomunicaciones para interconectar dispositivos.
VPN	Siglas para <i>virtual private network</i> , utilizado para extender una red en forma segura.
www	Siglas para <i>world wide web</i> , utilizado para manejar el sistema de documentos de hiper texto.

RESUMEN

Desde los orígenes de las telecomunicaciones hasta hoy se han tenido avances sustancialmente importantes que permiten brindar mejores servicios en comparación del pasado. La introducción del protocolo TCP/IP marco un punto de inflexión muy significado, dando paso a lo que se conoce actualmente, un mundo íntimamente conectado a través de redes intercontinentales y proveedores de servicios impresionantemente grandes capaces de dar cobertura en diversos lugares; también, el crecimiento constante de las redes es un agente muy importante en cuanto al crecimiento de un país.

Actualmente, en nuestro país el ámbito de las telecomunicaciones ha tenido un crecimiento bastante importante; actualmente, en el mercado existen proveedores grandes y chicos los cuales generalmente se encuentran en el área rural de Guatemala, como los proveedores de servicio de televisión por cable e internet, dado que actualmente el mercado es muy competitivo, el presente proyecto busca incrementar el desarrollo de dichos proveedores para posicionar firmemente sus empresas frente a los más poderosos del país.

Lo anterior por medio de la unificación de una red de transporte de tráfico *Multicast*, que homologa la señal de televisión bajo demanda en el área rural de Guatemala y aprovecha la infraestructura que ya se encuentra en el país; reduce de esta manera los gastos de operación y mantenimiento del cable operador en el área rural; se aplica así los conocimientos obtenidos en el área de telecomunicaciones, especialmente en el área de ruteo.

OBJETIVOS

General

Diseñar una topología de red para transportar canales de televisión digital a diversos puntos en el país con la capacidad de brindar un servicio ininterrumpido y que favorezca los costos de operación de los cable-operadores guatemaltecos, a su vez, demuestra la rentabilidad del proyecto.

Específicos

1. Consolidar una programación de canales de televisión unificada.
2. Diseñar una estructura de red de transporte capaz de responder en caso de fallas para brindar un servicio estable.
3. Realizar la configuración de equipos de enrutamiento para que la red pueda responder conmutando el tráfico, para aplicar filtros a nivel de la capa 3 del modelo OSI.
4. Argumentar que la implementación del proyecto es viable, desde el punto de vista técnico.

INTRODUCCIÓN

En el área rural de Guatemala existe una cantidad bastante considerable de cable-operadores que brindan servicios de televisión e internet; según datos brindados en la Expo Cable en mayo de 2018, la Gremial de Operadores de Televisión por Cable registra más de 424 compañías autorizadas para la venta de servicios de televisión, telefonía e internet las cuales forman parte del sector de la micro, mediana y macro empresa.

En sus inicios estos servicios llegaban hasta los hogares guatemaltecos por medios análogos utilizando tecnologías bastante antiguas; dichas tecnologías de transporte tenían diversos problemas, principalmente una calidad de sonido e imagen bastante precarias, limitadas por tecnología de su época. Posteriormente, a esto aparecieron las modulaciones digitales las cuales presentan muchas mejoras; son estas menos sensibles al ruido; reduce errores en la transmisión de datos, así como mayor capacidad de ancho de banda.

A pesar de tener ventaja sobre las modulaciones análogas, los sistemas de modulaciones digitales son más complejos debido a la lógica binaria que utilizan para transportar los datos; por otro lado, cabe resaltar que a las modulaciones digitales se le sumó la incorporación de la fibra óptica en el mundo de las telecomunicaciones, lo cual sumó un avance bastante importante en cuanto a transporte de datos se refiere, el desarrollo de esta tecnología permitió mayores alcances en distancia y ancho de banda.

Actualmente, los cableoperadores están incorporando dentro de su red metro fibra óptica hasta los hogares guatemaltecos (FTTH) utilizando métodos

de transporte digitales, lo cual representa una gran ventaja dado que se puede brindar al usuario una mejor experiencia sobre el servicio; coloca así a los cableoperadores en una mejor posición en el mercado actual que reduce los costos de operación y amplía el alcance de operación, pero sobre todo los ingresos monetarios.

1. DISEÑO DE TOPOLOGÍA DE RED PARA EL TRANSPORTE DE TRÁFICO MULTICAST

1.1. Modelo OSI

Hoy en día las telecomunicaciones han tenido un avance enorme en capacidad, hablamos de que en algunas décadas el crecimiento en cuanto a dispositivos y usuarios ha sido exponencial gracias a los avances tecnológicos.

El modelo OSI como tal especifica los protocolos y pasos a seguir para que la comunicación entre sistemas abiertos pueda llevarse a cabo, en telecomunicaciones han existido una gran variedad de fabricantes, en sus inicios estos crearon sus propios protocolos de comunicación específicamente para sus dispositivos por lo cual incluir dispositivos de otro fabricante era una tarea sumamente difícil si no es que imposible.

Fue por esto que a principios del año 1980 se creó este estándar por la Organización Internacional de Normalización ISO; este puede ser encontrado en la norma ISO 7498. El modelo OSI se compone de 7 niveles denominados capas en las cuales se subdivide todo el proceso a seguir para que dos sistemas puedan establecer la comunicación.

Básicamente, se describe el proceso completo desde que el mensaje se encuentra en forma de bits en lenguaje máquina, hasta que el mensaje es legible para un usuario final a través de caracteres en diferentes alfabetos. Dichas capas serán desglosadas y debidamente explicadas a continuación.

1.2. Análisis de las 7 capas el modelo OSI

A continuación, se muestra el análisis de las 7 capas del modelo OSI.

1.2.1. Capa 1: medios de transmisión de datos

La capa física del modelo OSI se refiere a todos los medios necesarios para establecer la comunicación, es esta la más cercana a la máquina ya que en ella se encuentran los datos forma de bits y se transmiten por medio de impulsos eléctricos u ópticos.

Siendo este el nivel más bajo, es donde se determina en que forma, así como el medio en que serán transmitidos los datos, cabe mencionar también que es en este punto donde los equipos se encargan de manejar la codificación de los bits provenientes desde la capa de enlace de datos, el medio a utilizar para la recepción y envío de datos, que tiene limitantes claramente en función del medio e incluso el equipo utilizado.

1.2.1.1. Ethernet

El estándar ethernet se encuentra fuerte ligado a la capa física del modelo OSI, define la forma en la cual tiene que ser conformada la trama de datos provenientes de la capa de enlace de datos además de definir la señalización y cableado de una red de área local LAN.

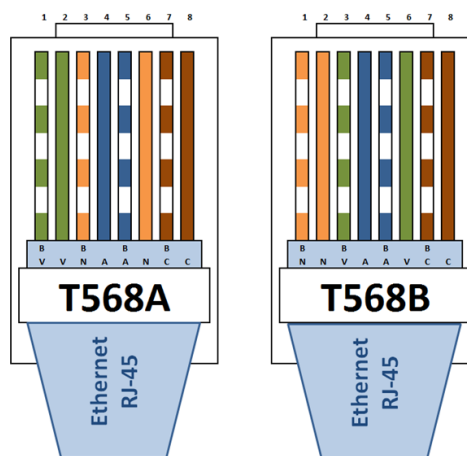
Para el desarrollo de Ethernet se tomó como base el estándar internacional IEEE 802.3; los inicios de esta estándar fueron en el año de 1970 por un estudiante recién graduado en el MIT llamado Robert Metcalfe. En mayo de 1973 terminó de desarrollar este estándar basado en el acceso múltiple con

escucha de portadora y detección de colisiones CSMA/CD, en las oficinas de Xerox en Palo Alto California donde lo bautizó Ethernet en alusión a la teoría física que decía que las ondas electromagnéticas se movían a través del éter.

1.2.1.2. Medios eléctricos

El medio eléctrico más común dentro de una red LAN es el cobre, que se utiliza en forma de 8 cables en pares trenzados para evitar ruido eléctrico, se define en el estándar TIA/EIA T568-A y T568-B como se puede apreciar en la figura 1.

Figura 1. Estándar TIA/EIA T568-A y T568-B



Fuente: Norma TIA/EIA 568 A y B

. <http://tec5quinto.blogspot.com/2017/04/norma-tiaeia-568-y-b>. Consulta: 15 de enero de 2019.

Para esto se utilizan las interfaces físicas RJ-45 categorías 4, 5, 5a, 6, 6a en sus variantes de UTP y STP. Agregando características como inmunidad a inducción electromagnética, altas temperaturas, bajas temperaturas y capacidad de transmisión de datos en cuanto a ancho de banda se refiere.

Por otro lado, es de alta importancia mencionar que los medios eléctricos tienen una limitante en distancia y es que según el estándar Ethernet no se puede sobrepasar una distancia de más de 100 metros para un cable de par trenzado categoría 6 debido a la resistencia que representa con la distancia el cobre.

1.2.1.3. Medios ópticos

Los medios ópticos representan los bits en un haz de luz, generalmente, se utilizan longitudes de onda fuera del rango de la luz visible utilizando dispositivos llamados *Transceiver* el cual cumple con la función de convertir impulsos eléctricos a ópticos.

Posteriormente, a la conversión el haz de luz es transmitido a través de fibra óptica, este es un medio de transmisión en el cual el haz de luz puede viajar libremente valiéndose del fenómeno de reflexión descrito por la ley física del matemático holandés Willebrord Snel mejor conocida simplemente como la ley de Snell. Finalmente, el haz de luz es recibido por otro *Tranceiver* que vuelve los impulsos ópticos en eléctricos.

Fibra óptica las hay de varios tipos, dos en general, monomodo y multimodo. Se diferencian en el tipo de composición, así como el núcleo en su interior, a pesar que ambas tienen un núcleo del orden de los micrómetros la diferencia es bastante notable.

- Multimodo: núcleo de 50 μm
- Monomodo: núcleo de 10 μm

En cuanto a distancia la fibra óptica supera por mucho al cable de par trenzado, alcanzando distancias del orden de los kilómetros lo cual representa una gran ventaja en ese sentido. No obstante, en el aspecto económico, la fibra óptica es extremadamente cara en comparación con el cobre debido a que su fabricación es mucho más difícil y costosa, económicamente hablando su precio es bastante más elevado respecto al cobre.

Referente a conectores los hay de varios tipos LC, SC, FC, ST ilustrados en la figura 2, a su vez, existen variantes de terminaciones de los conectores como APC, UPC y PC los cuales podemos apreciar en la figura tres.

Figura 2. **Conectores de fibra óptica**

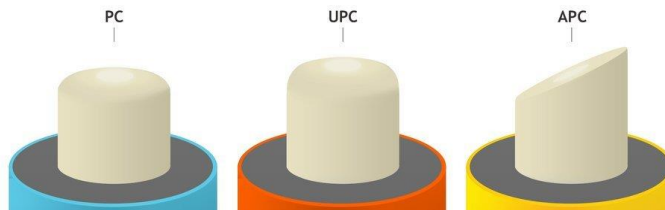


Fuente: *Asis Rodríguez. Conectores para FTTH.*

<https://www.instaladoresdetelecomhoy.com/conectores-para-ftth/>. Consulta: 15 de enero de 2019.

La terminación de los conectores es un factor importante a tomar en cuenta, dado que el uso adecuado de los mismos podría quitar o en su defecto añadir atenuaciones sobre el medio de transporte.

Figura 3. Terminaciones APC/UPC



Fuente: FORREST, Edward J. *Lo que necesitas saber sobre pulidos de en fibra óptica*.
<https://beyondtech.us/blogs/beyondtech-en-espanol/lo-que-necesitas-saber-sobre-conectores-pc-upc-y-apc>. Consulta: 15 de enero de 2019.

1.2.2. Capa 2: análisis de los dispositivos de *switching*, su funcionamiento y direccionamiento físico

La capa de enlace de datos del modelo OSI se basa en conectividad entre los dispositivos que se encuentran dentro de una misma red LAN utilizando direccionamiento físico, es decir, *mac address* para poder cumplir esta tarea.

Específicamente podríamos decir que se basa en el envío y recepción de paquetes desde el nivel físico de la red utilizando direcciones físicas. Una dirección física o *mac address* es aquella dirección única que brinda el fabricante al dispositivo, es un número hexadecimal de 48 *bits* de los cuales los últimos 24 bits los asigna el IEEE y los otros 24 *bits* los asigna el fabricante.

1.2.2.1. Dominios de colisión

Un dominio de colisión es un segmento de red LAN en el cual todos los dispositivos se encuentran conectados entre sí, a medida que crece la red y los dispositivos se disponen a realizar intercambio de paquetes sucede lo que se conoce como una colisión de datos debido a que dos o más dispositivos

intentan comunicarse al mismo tiempo; si no hay ningún dispositivo que lo evite, el desempeño de la red puede ser muy ineficiente.

Es así como surgió la idea de *switching* debido a que en su momento fue necesario crear un dispositivo capaz de enviar, así como recibir datos dentro de una red LAN (un mismo dominio de colisión); disminuye al máximo las colisiones de paquetes.

1.2.2.2. Hub

El *hub* o concentrador era un dispositivo capaz de recibir una trama de datos y reenviarlo a través de todos sus puertos excepto en el puerto del cual recibió el mensaje. Esto es una técnica bastante ineficiente dado que aumentan las colisiones de paquetes en la red, los concentradores no eran capaces de direccionar tráfico simplemente en ellos se concentraba todo el de la red.

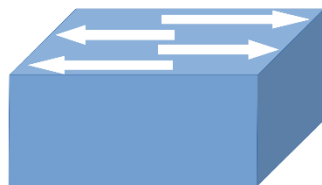
1.2.2.3. Switch

Fue debido a los problemas que presentaba el *hub* que se inventó el *switch*, este es un dispositivo capaz de dirigir tráfico a través de sus interfaces por medio de direccionamiento físico. Es el encargado de brindar conectividad a nivel físico a los dispositivos que se encuentran dentro de una red LAN; también, cabe resaltar que se rige a las especificaciones técnicas Ethernet o IEEE 802.3, generalmente este equipo se utiliza en la capa de distribución de una red en topología de estrella. Es importante mencionar también que realiza la conexión entre dispositivos utilizando técnicas de conmutación para poder llevar así la transferencia de datos.

El *switch* utiliza la técnica de conmutación de reenvío directo, también el método de almacenamiento y reenvío. El reenvío directo es una técnica muy parecida a la del *hub*, en esta técnica de conmutación el *switch* no espera a recibir la trama completa de datos para enviarlo al puerto destino, cuando recibe la dirección MAC lo envía hacia el puerto destino lo cual es bastante eficiente tomando en consideración que los tiempos de respuesta son extremadamente bajos, pero tiene la limitante que solo se puede utilizar si todos los puertos tienen la misma velocidad, adicionalmente otro problema es que debido a la conmutación tan rápido puede que la trama llegue al destino dañada por colisiones o incluso tramas erróneas.

Mientras que aplicando almacenamiento y posteriormente reenvío los datos son almacenados en buffers ubicados en el mismo dispositivo, almacena la trama completa para su posterior reenvío al puerto de destino favoreciendo mucho a que las tramas no se vean afectadas por colisiones o bien lleguen datos erróneos al destino. Simbólicamente, el *switch* se representa como en la figura 4.

Figura 4. **Simbología del *switch***



Fuente: Wikipedia Commons. *Cisco alike*. <https://commons.wikimedia.org/wiki/File:Switch-schematic-image.svg>. Consulta: 16 de enero de 2019.

1.2.3. Capa 3: direccionamiento lógico a través de subredes

La capa denominada red del modelo OSI es aquella en la cual se da direccionamiento lógico a las tramas *Ethernet*; es aquí donde se pueden comunicar dispositivos que se encuentran en diferentes segmentos de red por medio de un enrutador o *router*.

El *router* forma un papel muy importante en este segmento del modelo OSI ya que sin este dispositivo no sería posible encaminar los paquetes que se envían y reciben a través de la red. Este ruteo se da gracias al protocolo IPv4 e IPv6, el protocolo IPv4 es una dirección lógica de 32 *bits* de longitud separada en 4 grupos de 8 *bits* con una representación decimal, que sirven para identificar a un dispositivo dentro de una red; por otro lado, IPv6 es un número de 128 *bits* de longitud separado en 4 grupos de 32 *bits* con una representación hexadecimal, en secciones posteriores se dará una mayor explicación sobre las direcciones IP, su segmentación, entre otros aspectos de importancia.

1.2.3.1. Router

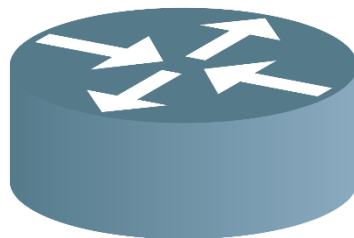
Es un dispositivo capaz de poder dar direccionamiento lógico dentro de varios segmentos de red, utilizando IPv4 o IPv6. Se puede decir básicamente que la función principal de este dispositivo es poder encaminar los paquetes entre subredes, según el segmento IP de destino utilizando la tabla de enrutamiento para enviar el paquete a su destino.

Este dispositivo realiza el enrutamiento más adecuado para los paquetes almacenándolos para evaluar la dirección de origen y destino, posteriormente a esto, evalúa según la tabla de enrutamiento la ruta más adecuada para llegar al destino. Dicha tabla se llena de información en función de los protocolos de

enrutamiento dinámicos configurados en el dispositivo o bien por medio de las rutas estáticas ubicadas en el mismo.

Simbólicamente el *router* se representa como en la figura cinco.

Figura 5. **Simbología del *router***



Fuente: Wikipedia Commons. *Cisco alike*.

<https://de.wikipedia.org/wiki/Router#/media/File:Router.svg>. Consulta: 16 de enero de 2019.

1.2.3.2. Tipos de enrutamiento

Existen básicamente dos tipos de enrutamiento, enrutamiento estático y enrutamiento dinámico. El enrutamiento estático trata en fijar saltos para los diferentes destinos de red; dicho en otras palabras, se asigna un salto específico para un destino de red específico. Esto es una forma de enrutamiento muy básica mas no escalable para redes en puertas de crecimiento o bien redes extremadamente grandes; aplicar enrutamiento estático en redes extremadamente grandes causaría que la red nunca convergiera.

Por su parte el enrutamiento dinámico se basa en diversos algoritmos para poder calcular las distintas rutas hacia diversos segmentos de red. Los hay de vector distancia, estado de enlace y una combinación de ambos en algunos

casos. A continuación, se presentan algunos de protocolos de enrutamiento dinámicos más conocidos.

- RIP
- OSPF
- EIGRP
- BGP

Ya que este no es el enfoque de este estudio no se profundizará más en el funcionamiento de estos protocolos de enrutamiento dinámico.

1.2.4. Capa 4: protocolos de comunicación y transmisión de datos

Llamada también la capa de transporte es una de las más importantes dentro de la estructura del modelo OSI, dado que se encarga de establecer la manera en la cual los datos serán enviados hacia su destino, se encarga de llevar la información en forma precisa y confiable.

Esto lo hace por medio de dos protocolos de transporte sumamente importantes, el protocolo de transmisión de control TCP totalmente orientado a la conexión y el protocolo de datagrama de usuario UDP que está enfocado al envío de datos sin conexión.

1.2.4.1. TCP

El protocolo de transmisión de control está orientado a realizar transmisión de datos por medio de una conexión de manera que cuando esta es creada se genera un flujo de datos a través de la conexión. Este protocolo cuenta con una

gran ventaja debido que se encarga de detectar errores en la transmisión por lo cual la trama de datos puede llegar a su destino sin error alguno.

Este protocolo utiliza un intercambio de 3 vías llamado *3 way handshake*, primero el *host* envía una solicitud de sincronización, seguido de esto el servidor responde con su propia solicitud de sincronización y un acuse de recibido. Finalmente, el *host* que envió la solicitud inicialmente envía un acuse de recibido, creando de esta manera la conexión.

1.2.4.2. UDP

Llamado protocolo de datagrama de usuario, se encarga de realizar transmisión de datos al igual que TCP con la mínima, pero muy notable diferencia que no crea conexiones, por lo cual tampoco tiene control sobre el flujo de datos y detección de errores sobre las tramas de datos enviadas. Esto lo convierte en un protocolo de transmisión no confiable en comparación con TCP, de tal manera que los datos pueden llegar con errores a su destino.

1.2.5. Capa 5: conservación, apertura y cierre de sesiones

La capa de sesión es aquella encargada de crear, mantener y cerrar el diálogo entre aplicaciones de dos sistemas abiertos. En dicho dialogo se desarrolla un intercambio de datos, de forma que la sesión sigue un proceso de dos fases.

- Establecimiento
- Utilización y liberación

1.2.6. Capa 6: presentación de datos

La capa de presentación escoge la forma en la cual los datos serán presentados al usuario final a través de una aplicación sin importar el formato de los caracteres ya sea ASCII, Unicode, etc. Cumple con tres funciones específicas que se encargan de llevar a cabo la comunicación entre aplicaciones de dos sistemas abiertos diferentes.

- Presentación de datos
- Cifrado de datos
- Compresión de datos

1.2.7. Capa 7: aplicaciones y su interacción con el usuario

La capa de aplicación es la última del modelo OSI, la más cercana al usuario y una de las más importantes, sin ella no podría completarse el proceso de la comunicación entre dos sistemas abiertos. En ella se encuentran las aplicaciones de red y los servicios necesarios para que el usuario pueda interactuar con otros dispositivos.

Existen muchos protocolos dentro de esta capa del modelo OSI tales como:

- Protocolo de servicio de nombres DNS
- Protocolo de transferencia de hipertexto HTTP
- Protocolo de transferencia de correo SMTP
- Protocolo de emulación de terminal Telnet
- Protocolo de transferencia de archivos FTP

1.3. TCP/IP

El protocolo de internet IP fue desarrollado en los años setenta por Vinton Cerf y Robert Kahn con el apoyo del Departamento de Defensa de los Estados Unidos; desarrollaron un protocolo capaz de establecer la comunicación entre dos sistemas diferentes. En sus inicios fue pensado como un protocolo que fuera capaz de brindar comunicación al ejercito de dicho país, sin saber que años más adelante se convertiría en lo que hoy en día se llama internet.

TCP/IP se conforma de una serie de protocolos capaces de interconectar redes con muchas computadoras, múltiples nodos conformados por redes de computadoras.

1.3.1. IPv4

Esta es la versión 4 del protocolo de internet, tiene sus orígenes en 1983, se encuentra definido por el RFC 791, una dirección IPv4 es un número conformado por 32 *bits* dividido en 4 octetos de bits separados por puntos, este número puede representar a un dispositivo (*host*) dentro de una red local o pública. Originalmente, las direcciones IPv4 estaban clasificadas en 5 clases: A, B, C, D y E; dichas direcciones estaban ligadas a una máscara de subred fija según la clase de dirección, la distribución era de la siguiente manera.

Tabla I. **Clasificación IPv4**

Clase	Rango IP	Máscara de subred
A	0.0.0.0 – 126.255.255.255	255.0.0.0.
B	128.0.0.0 191.255.255.255	– 255.255.0.0
C	192.0.0.0 223.255.255.255	– 255.255.255.0
D	224.0.0.0 239.255.255.255	– <i>Multicast</i>
E	240.0.0.0 255.255.255.255	– Investigación

Fuente: ODOM, Wendell. *CCENT/CCNA ICND1 100-105*. p. 89.

La clasificación de las direcciones IPv4 fue un problema en cuanto al crecimiento de la red en internet debido a que se desperdiciaban una gran cantidad de direcciones. Posteriormente, se diseñó un método que permitía cambiar la máscara de sub a conveniencia, en función del número de usuarios que se desea tener dentro de la red; este fue llamado como máscaras de subred de tamaño variable o VLSM el cual será explicado en secciones posteriores.

En función del crecimiento de la red se estableció una clasificación adicional en IPv4, naciendo así el concepto de dirección privada y dirección pública. Una dirección privada es aquella dirección que solo puede ser utilizada en ámbitos dentro de una red LAN, este tipo de direccionamiento no puede ser encaminado hacia internet mientras que las direcciones públicas si pueden ser ruteadas fuera de una red privada, es decir, a través de internet.

Es de suma importancia resaltar que las direcciones públicas son únicas a nivel global, lo cual significa que no pueden ser asignadas a diferentes

dispositivos. Cabe mencionar también que son reguladas en cada continente por una entidad encargada de asignarlas según sea requerido.

1.3.2. Segmentación IPv4

Como fue explicado en la sección anterior IPv4 presentó problemas a medida que la red creció, en un intento por volver la red escalable por muchos años más se creó el concepto de IP privado e IP público. Los rangos reservados para IPs privadas se muestran en la tabla a continuación.

Tabla II. Segmentos de IP privados

Segmento 1	10.0.0.0 – 10.X.X.X
Segmento 2	172.16.0.0 – 172.31.X.X
Segmento 3	192.168.0.0 – 192.168.X.X

Fuente: ESCALANTE, Mauro. *Conceptos fundamentales de Mikrotik RouterOS v6.39.2*. p. 106.

Siendo la representación de X cualquier número decimal desde 0 hasta 255, existe un rango reservado para autoconfiguración.

- 169.254.X.X

Un rango destinado a direcciones de *loop back* para pruebas, siendo este.

- 127.X.X.X

Los rangos que no fueron mencionados anteriormente forman parte del segmento de IPs públicas, reguladas por entidades como el ICANN, IANA,

LACNIC, APNIC, RIPE, AFRINIC, etc., exceptuando los rangos asignados para *Multicast* y para investigación.

1.4. Máscara de subred

Es un numero de 32 *bits* dividido en 4 octetos de bits al igual que IPv4, pero con un objetivo totalmente diferente, este tiene la función de identificar la red en la cual se encuentra un dispositivo. Inicialmente, la máscara de subred estaba sujeta al tipo de dirección IPv4 utilizada ya sea clase A, B, C como fue explicado en la sección anterior.

Debido a la problemática de la escalabilidad de la red se realizaron ciertos cambios sobre la máscara de subred; fue de esta manera que surgió la máscara de subred de tamaño variable VLSM volviendo escalable por muchos años la segmentación IP. Consistía en dividir un segmento IP según los requerimientos de la red sin desperdicio de direcciones; a continuación, se presenta en la tabla número tres como puede realizarse la segmentación IP y la cantidad de *host* que pueden estar dentro de una misma red.

Tabla III. Máscaras de subred

Decimal	CIDR	Hosts
255.255.255.255	/32	
255.255.255.254	/31	
255.255.255.252	/30	2
255.255.255.248	/29	6
255.255.255.240	/28	14
255.255.255.224	/27	30
255.255.255.192	/26	62
255.255.255.128	/25	126
255.255.255.0	/24	254
255.255.254.0	/23	510
255.255.252.0	/22	1022
255.255.248.0	/21	2046
255.255.240.0	/20	4094
255.255.224.0	/19	8190
255.255.192.0	/18	16382
255.255.128.0	/17	32766

Continuación de la tabla III.

255.255.0.0	/16	65534
255.254.0.0	/15	131070
255.252.0.0	/14	262142

Fuente: Wikipedia Commons. https://es.wikipedia.org/wiki/M%C3%A1scara_de_red. Consulta:
17 de enero de 2019.

1.5. Diseño preliminar de red de transporte

Con los conocimientos adquiridos de las secciones anteriores ahora se requiere diseñar una red de transporte para tráfico *multicast*, considerando todos los factores necesarios para el transporte, así como poder integrar múltiples señales de origen en un nodo de distribución.

1.5.1. Aspectos importantes

Primero que nada, que debe tomar en cuenta el ancho de banda a utilizar para la red de transporte, capacidad de procesamiento, así como el *Throughput* de los equipos a utilizar.

Técnicamente hablando, también se debe tomar en cuenta el tipo de tráfico que se está transportando, el tamaño de las tramas Ethernet y la forma en la cual serán transportados los datos.

1.5.2. Análisis de tráfico *multicast*

La multidifusión o *multicast* funciona de una manera muy particular, esta clase de tráfico se envía a múltiples host dentro de la red, que se encuentran un determinado grupo y que muestran interés por recibir los paquetes.

Para que un equipo dentro la red reciba el tráfico primero que nada tiene que subscribirse al grupo de multidifusión utilizando un mensaje de IGMP *Snooping*, el protocolo IGMP sirve para que el enrutador pueda realizar mapas de los dispositivos que se encuentran interesados en recibir los paquetes de multidifusión, explícitamente los dispositivos que se encuentran interesados reciben el tráfico.

Existe un rango de direcciones IPv4 reservado para la multidifusión, todas las direcciones de clase D están reservadas para este propósito, generalmente se utiliza con propósitos de video, música, *streaming*, entre otros. La asignación de direcciones IP se realiza por medio de un protocolo llamado SAP, este utiliza un directorio de sesiones para realizar envíos constantemente de sesiones; este intercambio de información se realiza a través de la dirección *multicast* 224.2.127.254 en el puerto 9875.

Según un análisis de tráfico hecho con un *snifer*, la calidad de la señal a transportar influye directamente en el ancho de banda que ocupa cada canal, este reflejo los resultados de la tabla 5.

Tabla IV. **Resultados análisis tráfico *multicast***

Calidad	Ancho de banda
HD	10,5 Mbps
SD	4,5 Mbps

Fuente: elaboración propia.

De forma que el ancho de banda es un factor muy importante a considerar dado que en función de la cantidad de canales a transportar se deben de seleccionar los equipos con el *throughput* adecuado en sus interfaces.

1.5.3. Equipos a utilizar

Como fue mencionado en la sección anterior el ancho de banda es un factor muy importante, dado que generalmente un listado de canales que ofrece el cable-operador es de 180 canales, se necesita que las interfaces del equipo a utilizar tengan un *throughput* de 1Gbps, por lo menos.

Para poder soportar esta demanda se propone utilizar un enrutador de marca Mikrotik CCR 1009-7G-1G-1S+, a excepción del equipo a utilizar en el nodo de distribución dado que necesita una mayor capacidad de procesamiento por que en este se concentrara el tráfico proveniente de múltiples orígenes.

Dada la demanda que necesita satisfacer dicho equipo, se propone un equipo Mikrotik CCR 1036-12G-4S, tiene mayor capacidad de procesamiento, así como las interfaces físicas con el *throughput* adecuado.

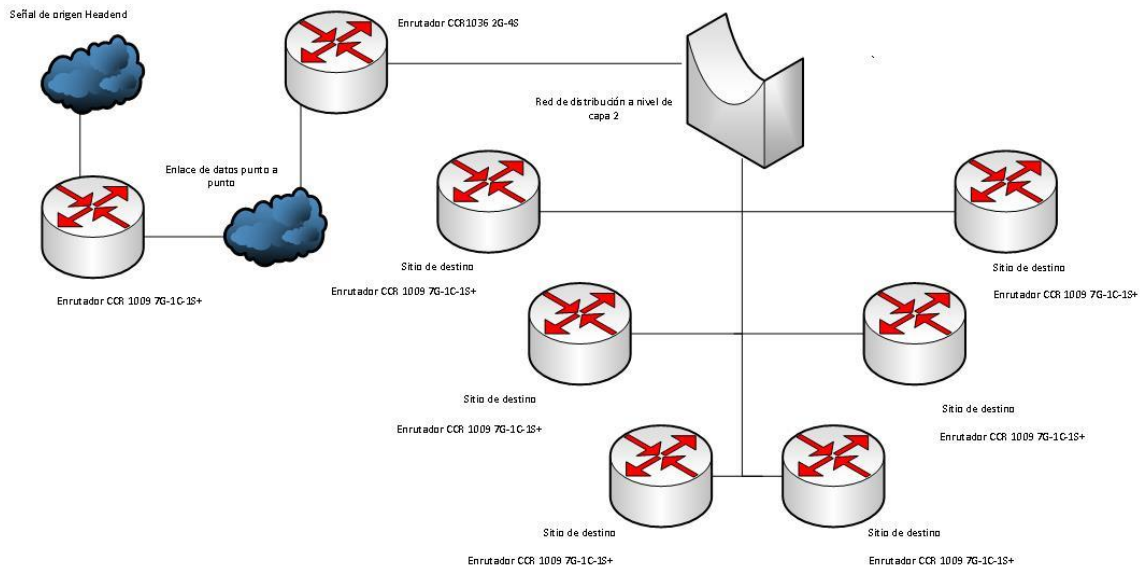
1.5.4. Diagrama de topología lógica

Luego de analizar todos los aspectos necesarios, así como los equipos que se pueden utilizar según las necesidades en cuanto a capacidad, se diseñó la estructura de red en la cual se cuenta con la señal de origen transportada a través de un enlace punto a punto hacia el nodo central de distribución; seguidamente a esto un enrutador se encarga de concentrar todo el tráfico, así como distribuirlo a través de la red de transporte por medio de un puente en sus interfaces.

Finalmente, todo el flujo de tráfico *multicast* llega a los puntos de entrega por medio de enlaces de datos, en esta fase al igual que en el nodo de distribución en cada enrutador hay un puente que se encarga de distribuir el

tráfico en las interfaces deseadas obteniendo como resultado el diseño de red de la figura a continuación.

Figura 6. Red preliminar de distribución



Fuente: elaboración propia, empleando Visio 2016.

2. UNIFICACIÓN DE TRÁFICO MULTICAST EN UN SOLO NODO DE DISTRIBUCIÓN

Dada la necesidad de brindar un servicio estable, la unificación de un nodo de distribución es de carácter sustancial para transportar la señal desde múltiples orígenes y posteriormente distribuir el tráfico.

En secciones posteriores se dará a conocer el transporte de datos a utilizar hasta el nodo de distribución, la justificación de los equipos a utilizar en dicho nodo, así como la integración de múltiples señales de origen.

2.1. Transporte y equipos

A continuación, se presenta el transporte y los equipos.

2.1.1. FTTx

Como bien se sabe los grandes avances tecnológicos de las últimas décadas permitieron que las redes de fibra óptica pudiesen llegar a cualquier ubicación; se puede de esta manera ampliar la capacidad de ancho de banda dentro de una red y la calidad de servicio.

Es así como FTTx hace referencia a diversas redes de fibra óptica como FTTH, FTTN, FTTO, etc. De esta manera es como se reemplazan redes con tecnología antigua que transportaban datos por medios eléctricos, limitadas en distancia, así como de ancho de banda. Este tipo de tecnología se encuentra completamente orientada a contenidos de multimedia como video juegos en

red, audio de alta calidad, video de alta calidad, etc. Es evidente entonces que el medio de transporte ideal para el tráfico *multicast* es la fibra óptica debido a su capacidad de transmisión de datos.

2.1.2. Aspectos a tomar en cuenta sobre los equipos

Para esta fase del proyecto se requiere de un equipo con las siguientes características.

- Alto procesamiento
- Múltiples interfaces físicas ópticas/eléctricas
- *Throughput* de 1 Gbps por interface (por lo menos)
- *Firewall* (para manejo del tráfico)

Se necesita de una alta disponibilidad de procesamiento debido a la cantidad de tráfico que se va a concentrar en este enrutador, múltiples interfaces de distribución, una capacidad de al menos 1 Gbps en cada interface de distribución y manejo de *firewall* a nivel de software para integrar múltiples señales de origen.

2.1.2.1. Justificación de equipos

En la sección 1.5.3 fue propuesto un equipo Mikrotik CCR 1036-12G-4S, según la hoja de datos proporcionada por el fabricante este equipo cuenta con las siguientes características generales.

- CPU: TLR4-03680
- Recuento de núcleos: 36
- Frecuencia CPU: 1,2 Ghz

- RAM: 4 Gb
- Interfaces Ethernet 10/100/1000: 12
- Interfaces ópticas: 4

Figura 7. **Mikrotik CCR 1036-12G-4S**



Fuente: Mikrotik. <https://mikrotik.com/product/CCR1036-12G-4S-149>.

Consulta: 18 de enero de 2019.

El equipo antes descrito cumple con todos los requerimientos para ser utilizado en el nodo central de distribución, cuenta con una gran capacidad de procesamiento, *throughput*, memoria RAM, *firewall*, etc.

2.1.3. Ancho de banda

Este es un aspecto muy importante en el diseño de la red, considerando un listado de 180 canales de televisión se necesitan enlaces de datos de al menos 1Gbps según el estudio de tráfico realizado en la sección 1.5.2; es evidente, entonces, que por cada cliente se necesita un enlace de datos de esta capacidad, así mismo, por cada *headend* donde se origina la señal.

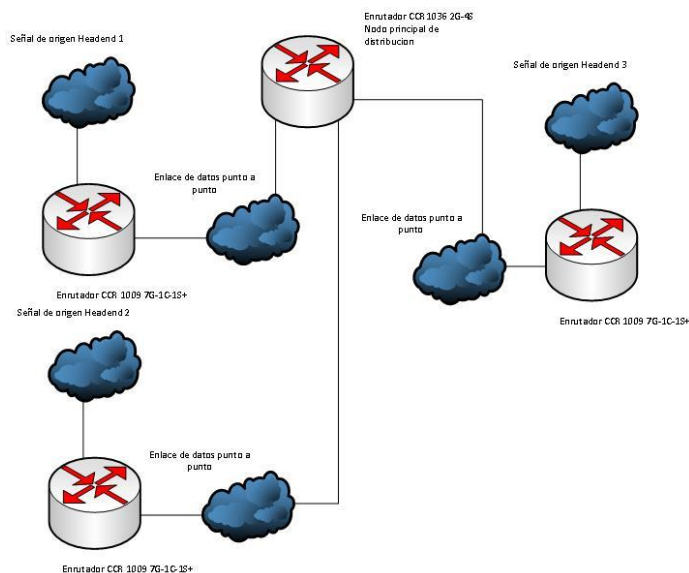
2.2. Diseño de nodo principal de distribución

El nodo de distribución es la piedra angular de este proyecto, es el lugar físico en donde se realizará la mayor implementación de red, en este punto se concentrarán los enlaces necesarios para llevar el tráfico desde su origen hasta los distintos puntos de distribución.

2.2.1. Integración de equipos

Sobre la base de las consideraciones anteriores sabemos que se necesita un equipo de enrutamiento y un equipo de distribución en un nodo principal, a nivel de transporte se va a recibir el tráfico sobre fibra óptica e igualmente este será distribuido por esta misma vía; de manera que al momento el diseño de red propuesto se desglosa de la siguiente manera.

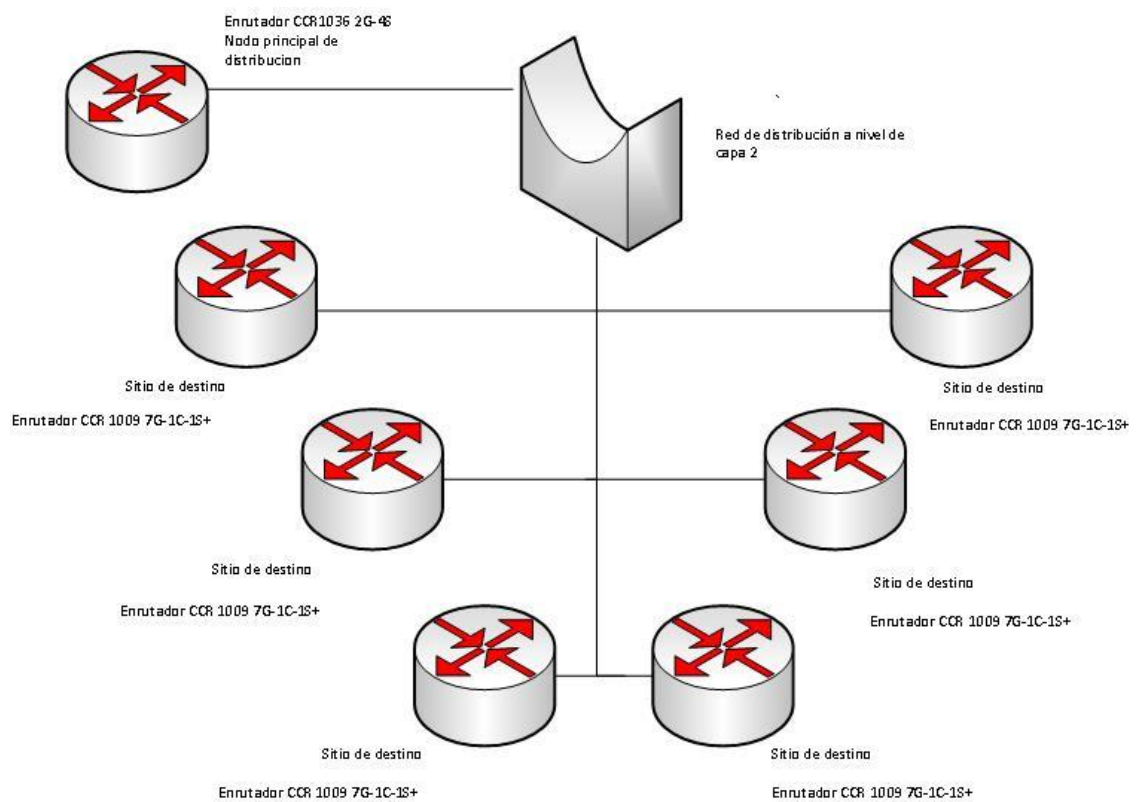
Figura 8. Diseño de red con múltiples señales de origen



Fuente: elaboración propia, empleando Visio 2016.

La integración de múltiples orígenes de suma importancia para el proyecto, luego de esta fase se necesita una red de distribución representada por un puente que sea capaz de entregar el tráfico en los distintos puntos como se ve a continuación en la figura.

Figura 9. Red de distribución



Fuente: elaboración propia, empleando Visio 2016.

Para el diseño de la red de distribución solamente se necesita conectividad a nivel de capa dos; es evidente que solo se necesita enlaces de datos hacia los puntos en donde se entregara el servicio.

2.2.1.1. Enlace punto a punto

También conocido como enlace PTP (*point to point*) es un enlace dedicado que se utiliza únicamente para establecer comunicación entre dos nodos, forman parte principal de la estructura de red presentada en la figura ocho de la sección anterior. Este es el tipo de enlace que se debe de utilizar para transportar el tráfico desde su origen hasta el nodo principal.

2.2.1.2. Enlaces secundarios

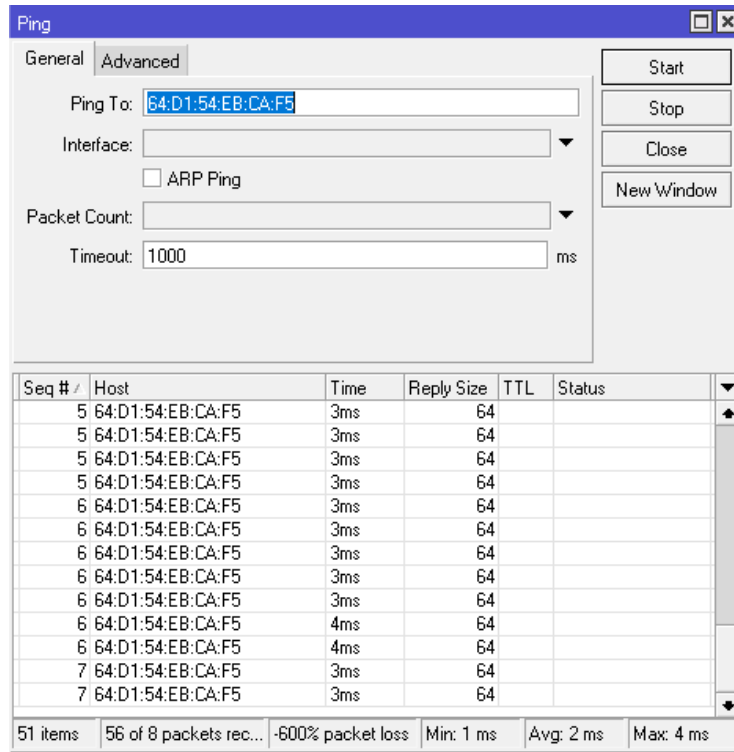
Para la red de distribución como fue mencionado solo se necesita conectividad con los puntos finales; esto puede ser logrado por medio de enlaces dedicados o bien por medio de protocolos de transporte de datos como MPLS.

2.2.2. Pruebas de conectividad

Una vez implementado el enlace punto a punto se pueden realizar pruebas de conectividad entre dispositivos, en este caso el CCR 1036-12G-4S se encuentra directamente conectado mediante un enlace de datos a un CCR 1009-7G-1G-1S+.

Para realizar esta prueba simplemente se configura en el dispositivo una IP de conexión para pruebas o bien por medio de *mac address*, solamente para verificar la conectividad entre dispositivos, realizamos un *ping* extendido para verificar distintos parámetros importantes como la latencia, porcentaje de perdidas, etc. En la figura que a continuación se realizó esta prueba por medio de *mac address* utilizando la herramienta llamada *mac ping* en RouterOS de Mikrotik.

Figura 10. Prueba de conectividad a nivel de capa 2



Fuente: elaboración propia.

Siguiendo la misma línea se pueden realizar pruebas de conectividad sobre la red de transporte con los distintos puntos de destino del tráfico, en dado caso las pruebas fuesen satisfactorias, quiere decir que se tiene conectividad a nivel de capa 2 pudiendo así transportar el tráfico *multicast* a los distintos puntos de entrega del servicio.

2.3. Problemática de integración *headends* secundarios

Sucede que para poder recibir tráfico desde distintos orígenes necesitamos tomar en cuenta que seguramente se presentara una situación muy particular, para transportar los canales en forma digital se utilizan direcciones IP, si en dado caso desde los orígenes de la señal se maneja direcciones semejantes podría llegar a dar un conflicto IP. Esto afectaría directamente el desempeño de la red tal vez incluso abatiendo el tráfico en su totalidad, que satura los enlaces de datos e incluso genera pérdidas de paquetes, lo cual perjudicaría directamente la experiencia del usuario respecto al servicio.

Sumando a esto la necesidad de conmutar el tráfico independientemente del origen se vuelve primordial la necesidad de bloquear de alguna manera el tráfico proveniente de las cabeceras secundarias para que respalden a la principal.

2.3.1. Solución al problema

La solución a dicho problema es la implementación de firewall con la que cuentan los equipos Mikrotik;00 la implementación de esta herramienta cuenta con filtrado de paquetes, dicha función se adecua perfectamente a este requerimiento. Con este tipo de filtros el enrutador es capaz de poder filtrar el flujo de paquetes desde, hacia y a través del enrutador pudiendo incluso cambiar el flujo de tráfico a conveniencia.

Con referencia a lo anterior nos podemos dar cuenta que se pueden realizar filtros a nivel IP bloqueando así el flujo de datos que no se requiere en ciertos momentos, así mismo se podría conmutar el tráfico en casos de falla o

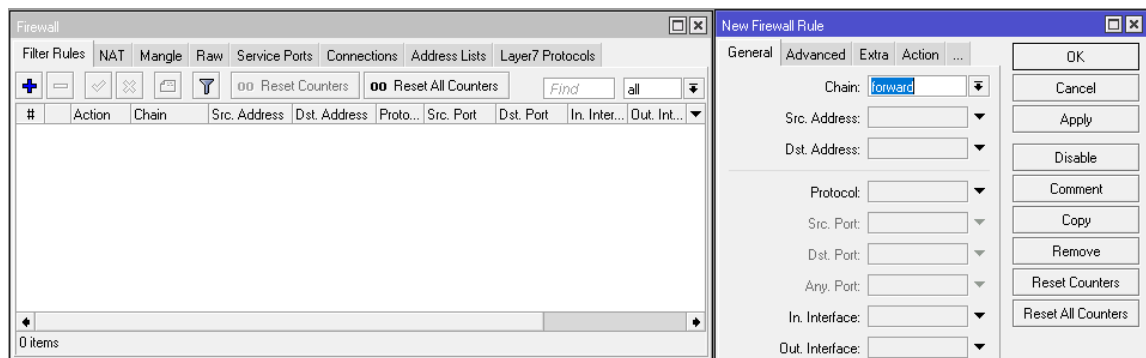
bien hacer una combinación de flujos para poder consolidar un listado de canales completo.

2.3.2. Herramientas a utilizar

El *firewall* implementado en los equipos Mikrotik se puede encontrar en la sección *IP/Firewall*, es aquí donde se localizan diferentes herramientas de bloqueo, marcaje y enmascaramiento de paquetes, entre las que se puede utilizar se encuentran las siguientes.

- Filtros (*filter*)
- NAT
- Marcado de paquetes (*mangle*)
- Filtros de capa 7 (*layer 7 protocols*)

Figura 11. **Firewall mikrotik**



Fuente: elaboración propia.

Para efectos de este informe solo nos centraremos en el estudio de la sección de filtros, en esta sección existen 3 tipos de cadenas (*chain*), *Input*,

output, forward. Estas tienen un significado, así como utilidad en específico, que se describe a continuación.

- *Input*: tráfico que ingresa a la interfaz
- *Output*: tráfico que sale de la interfaz
- *Forward*: tráfico a través de la interfaz

Estos conceptos son muy importantes cuando se trata de realizar reglas sobre el flujo de paquetes, además se pueden utilizar como parámetros para las reglas segmentos IP de destino/origen, listados, puertos físicos, marcas de ruta e incluso filtros de capa 7. Todos estos parámetros de configuración sirven para generar reglas sobre el flujo de paquetes; adicionalmente, existen acciones que determinan que acción realizar cuando las condiciones que establece la regla se cumplen, en este estudio solo se trataran las reglas listadas a continuación.

- *Drop*: deniega el flujo de paquetes
- *Accept*: acepta el flujo de paquetes

La utilización y creación de filtros para la integración de múltiples cabeceras será estudiado en el capítulo 4 de esta investigación; realiza un análisis de los filtros e incluyendo la configuración utilizada.

3. DESARROLLO E IMPLEMENTACIÓN DE CONFIGURACIÓN EN ROUTER OS DE MIKROTIK PARA TRANSPORTE DE TRÁFICO MULTICAST

3.1. Introducción a Mikrotik

Mikrotik es una compañía de origen letones, sus inicios se remontan al año de 1996, inicialmente fue fundada con el objetivo de crear equipos para proveedores de servicios de internet, orientados a conexiones inalámbricas (radio enlaces).

3.1.1. Reseña histórica

Posteriormente a la creación de la compañía en el año de 1997 fue desarrollada una herramienta para brindar control y flexibilidad para cualquier equipo de enrutamiento llamada RouterOS.

Pasados los años en 2002 la compañía tomo un giro radical, comenzaron a fabricar su propio hardware llamándolo RouterBOARD, actualmente Mikrotik es una compañía con presencia a nivel global, cuenta con una gran variedad de equipos para brindar conectividad, desde equipos para montar redes inalámbricas hasta equipos robustos de ruteo y switcheo. Cabe mencionar que se destacan por ser una compañía con precios bastante accesibles en el mercado tan competitivo de hoy.

3.1.2. Introducción a RouterOS

A continuación, se muestran las características del RouterOS.

3.1.2.1. Características

Como fue mencionado anteriormente esta herramienta fue creada en 1997 para brindarle al usuario una mejor experiencia respecto al manejo de los equipos desarrollados por la compañía, cuenta con compatibilidad con la arquitectura i386 por lo que puede ser instalado en la mayoría de sistemas, es compatible con SMP, necesita un mínimo de 32 MB de RAM; adicionalmente, cuenta con soporte para medios de almacenamiento externo IDE, SATA, USB y *flash* bastante útil cuando se utiliza el equipo como un *Sniffer*.

Cabe agregar también que integra herramientas de *Ping*, *Traceroute*, *Bandwidth Test*, *Packet Sniffer*, Telnet, SSH, E-mail, SMS, scripts automatizados (*scheduler*), entre otras más. Cuenta con manejo de *Firewall*, Ruteo, MPLS, VPN, *Wireless*, *HotSpot*, QoS (*queues tree*), *Proxy*, Certificados (SSL), *Queues* (simples) y PPP. Es de suma importancia resaltar que cualquier dispositivo que tenga instalado RouterOS puede utilizar las herramientas anteriormente descritas, así como las características mencionadas, limitados, claramente por sus recursos de hardware.

3.1.2.2. RouterBOARD y arquitecturas

RouterBOARD es un conjunto de tarjetas electrónicas diseñadas por Mikrotik, todas las placas desarrolladas funcionan con el sistema operativo RouterOS, cada placa desarrollada cuenta con actualizaciones de por vida; así mismo, cuentan con una gran relación costo/beneficio debido a que el

rendimiento es excepcional en comparación con su costo tan bajo, en comparación a las de otros proveedores de este tipo de equipos. Dichas tarjetas cuentan con las arquitecturas soportadas en la tabla V que a continuación se presenta.

Tabla V. **Arquitecturas soportadas por RouterBOARD**

Arquitectura	Series
mipsbe	CRS, RB4xx, RB7xx, RB9xx, RB2011, SXT, OmniTik, Groove, METAL, SEXTANT
mipsle	RB1xx, RB5xx, RB Crossroads
ppc	RB3xx, RB600, RB800, RB1xxx
x86	PC / x86, RB230
arm	RB3011
tile	CCR
smips	hAP lite

Fuente: ESCALANTE, Mauro. *Conceptos Fundamentales de Mikrotik RouterOS v6.39.2*. p. 15.

Las actualizaciones para cada arquitectura pueden ser encontradas en el sitio oficial de Mikrotik en el apartado de *software/ downloads*, aquí se encuentra la última versión de actualización disponible; es recomendable actualizar los equipos constantemente para solucionar problemas de versiones anteriores como *bugs*, agujeros de seguridad, etc.

3.1.2.3. Nomenclatura RouterBOARD

Los nombres de cada enrutador se establecen a partir de las características propias del equipo, según parámetros establecidos previamente por la marca específicamente, estos se muestran en la tabla a continuación.

Tabla VI. **Estándar de nomenclatura RouterBOARD**

Nombre de la tarjeta	Características de la tarjeta	-	Tarjeta <i>wireless</i> embebida	Características de la tarjeta <i>wireless</i>	-	Tipo de conector	-	Tipo de <i>enclosure</i>
----------------------	-------------------------------	---	----------------------------------	---	---	------------------	---	--------------------------

Fuente: ESCALANTE, Mauro. Conceptos Fundamentales de Mikrotik RouterOS v6.39.2. p16.

- Nombre de la tarjeta: puede ser un número de tres dígitos que indican el número de serie, interfaces cableadas e interfaces Wireless, este puede ser una palabra clave como OmniTIK, Groove, SXT, METAL.
- Características de la tarjeta
 - U: USB
 - P: *power injection* con controlador
 - i: puerto simple *power injection* sin controlador
 - A: más memoria
 - H: CPU más potente
 - G: *gigabit*
 - L: edición ligera
 - S: puerto SFP
 - e: tarjeta de extensión PCIe
 - x<N>: donde N es el número de núcleos *core*
- Tarjeta *wireless* embebida: si el dispositivo cuenta con tarjeta wireless esta se especifica de la siguiente manera <banda><potencia por chain><protocolo><número de chains>.
 - Banda

- 5 Ghz
 - 2 Ghz
 - Dual 5Ghz y 2Ghz
- Potencia por *chain*
 - H: 23-24dBm a 6 Mbps 802.11a; 24-27dBm a 6Mbps 802.11g.
 - HP: 25-26dBm 6Mbps 802.11a; 28-29dBm a 6Mbps 802.11g.
 - SHP: 27 + dBm a 6Mbps 802.11a; 30 + dBm a 6Mbps 802.11g.
- Protocolo
 - N
 - AC
- Número de *chains*
 - D: dual *chain*
 - T: triple *chain*
- Tipo de conector
 - MMCX
 - u.FL

- Tipo de *enclousure*
 - BU: *board unit* (no *enclousure*)
 - RM: *rack mount*
 - IN: *indoor*
 - EM: *extended memory*
 - LM: *light memory*
 - BE: *black edition*
 - TC: *tower*
 - OUT: *outdoor*

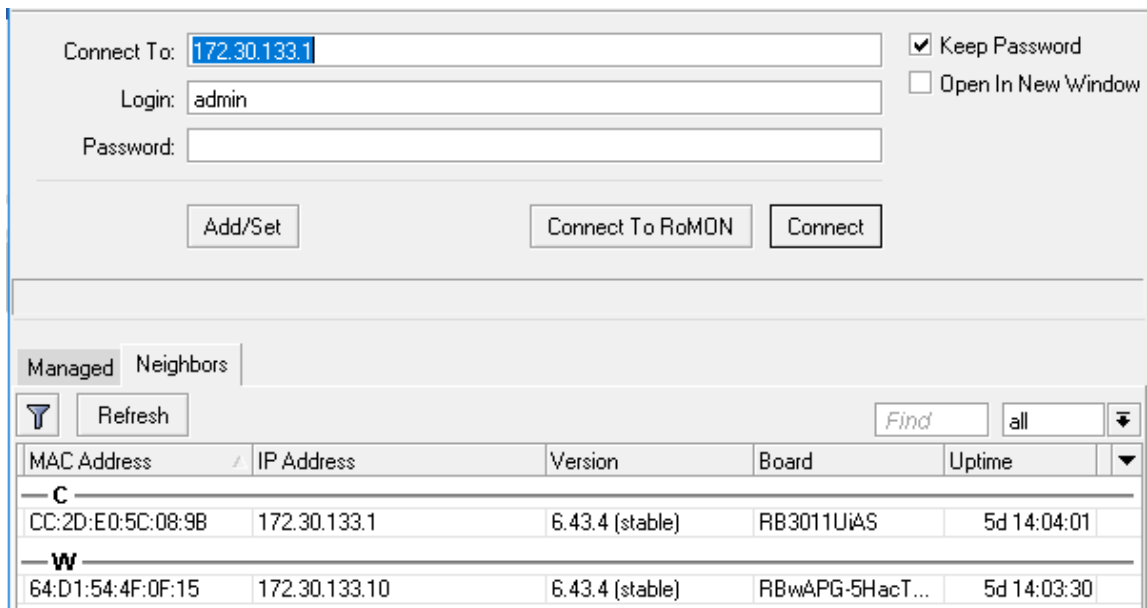
3.1.2.4. Winbox

Winbox es una herramienta desarrollada por Mikrotik para el uso y manejo de sus equipos, posee una simple interface GUI que permite el acceso a los enrutadores que tienen instalado RouterOS. Originalmente, fue desarrollado sobre Win32 pero puede ser ejecutado en Linux como en Mac OSx utilizando Wine, la gran mayoría de funciones de RouterOS se encuentran disponibles en Winbox a excepción de algunas muy críticas como el cambio de dirección física de una interface que solo se encuentra disponible desde la línea de comandos.

Esta herramienta puede ser descargada desde la página oficial de Mikrotik en el apartado de *software*, una vez descargado en nuestro ordenador se puede ingresar al dispositivo por medio de IP (capa 3) o bien *mac address* (capa 2). A primera vista cuando se ejecuta Winbox es posible apreciar todos los dispositivos con RouterOS que se encuentran dentro de la red gracias a que esta herramienta realiza un barrido *broadcast* dentro de la red en busca de dispositivos para poderse conectar. Seguido del apartado de la dirección a la cual deseamos conectarnos encontramos la casilla correspondiente al usuario,

seguidamente la de contraseña; a continuación, se muestra en la figura 12 la primera vista de Winbox ejecutándose.

Figura 12. Primera vista de Winbox



The screenshot displays the Winbox interface. At the top, there are input fields for 'Connect To:' (containing 172.30.133.1), 'Login:' (containing admin), and 'Password:'. To the right, there are checkboxes for 'Keep Password' (checked) and 'Open In New Window' (unchecked). Below these fields are three buttons: 'Add/Set', 'Connect To RoMON', and 'Connect'. The main area is divided into two tabs: 'Managed' and 'Neighbors'. The 'Managed' tab is active, showing a table of devices. The table has columns for MAC Address, IP Address, Version, Board, and Uptime. There are two entries in the table, one starting with 'C' and one with 'W'.

MAC Address	IP Address	Version	Board	Uptime
C				
CC:2D:E0:5C:08:9B	172.30.133.1	6.43.4 (stable)	RB3011UiAS	5d 14:04:01
W				
64:D1:54:4F:0F:15	172.30.133.10	6.43.4 (stable)	RBwAPG-5HacT...	5d 14:03:30

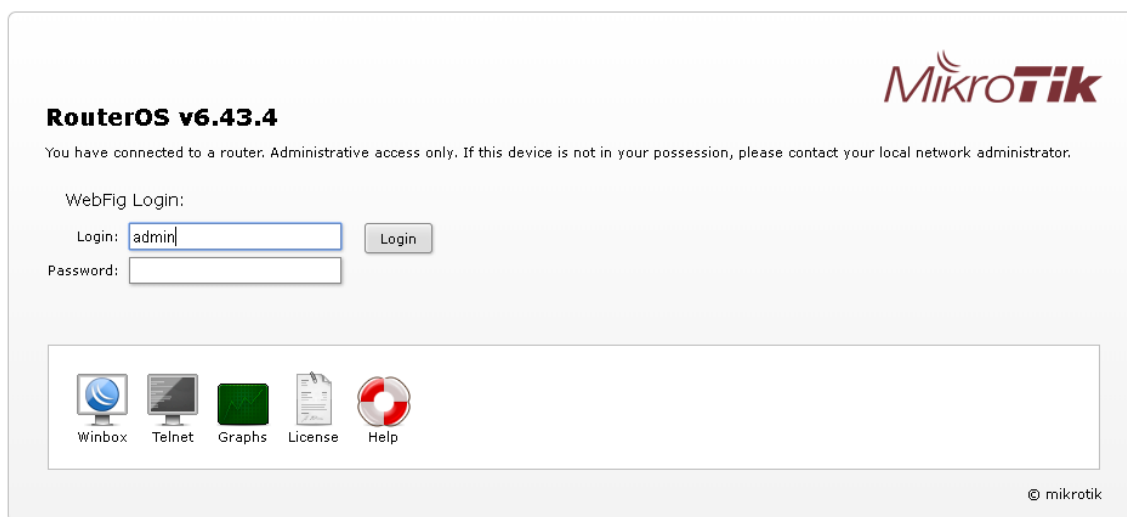
Fuente: elaboración propia.

El usuario por defecto para los equipos Mikrotik es *admin* sin contraseña alguna; una vez se ingresa al equipo se puede visualizar cada una de las funcionalidades disponibles en RouterOS; el puerto por defecto para realizar la conexión vía Winbox es el 8291, este puede ser cambiado en la sección de *IP/Services* que evita así problemas de seguridad y mantiene el acceso restringido al equipo.

3.1.2.5. WebFig

Siguiendo la misma línea de Winbox, se trata de otra forma de acceso a los equipos por medio de un navegador web; para ingresar al equipo simplemente se coloca la IP en la barra de direcciones e inmediatamente se desplegará una página web solicitando usuario y contraseña para ingresar al equipo. Además, de poder administrar el dispositivo, se puede ver el tráfico histórico sobre las interfaces, estas graficas pueden ser visualizadas en el enlace “*Graphs*”, en este apartado se encuentra un histórico de todas las interfaces del enrutador.

Figura 13. Primera vista WebFig



Fuente: elaboración propia.

Por defecto al igual que Winbox se utiliza el usuario *admin* sin contraseña, una vez adentro se mostrará la información más relevante sobre el dispositivo como las direcciones IP configuradas, parámetros de la red inalámbrica (si la

tuviese configurada), dirección física, así como todas las funcionalidades de RouterOS.

3.1.2.6. Otras formas de acceso y servicios

Además de ingresar a los equipos por medio de Winbox o bien vía WebFig, también, se puede ingresar por medios como SSH y Telnet, simplemente se necesita que se encuentren habilitados dentro del equipo en el apartado de *IP/Services*.

En este apartado se especifica si se desea habilitar servicios como SSH, Telnet, Winbox, HTTP, FTP, entre otros. Adicionalmente a esto se pueden especificar los puertos en los cuales se puede realizar la conexión, los puertos por defecto se muestran en la tabla VII.

Tabla VII. **Puertos por defecto servicios Mikrotik**

Servicio	Puerto
Telnet	23
FTP	21
www	80
SSH	22
www-ssl	443
api	8 728
Winbox	8 291

Fuente: elaboración propia.

Estos puertos pueden ser editados según lo requiera el administrador del dispositivo, existen otros medios de conexión como *Mac Telnet* y cable de consola los cuales no serán citados en este estudio.

3.1.3. Conceptos básicos

Ya que fue propuesto un diseño de red preliminar es necesario comenzar a pensar la configuración que se realizará sobre los equipos para transportar tráfico *multicast* desde su origen hasta los puntos de distribución. Antes que nada, se necesita explicar algunas de las funciones básicas sobre el manejo de los equipos, pero bastante ineludibles para el funcionamiento del proyecto.

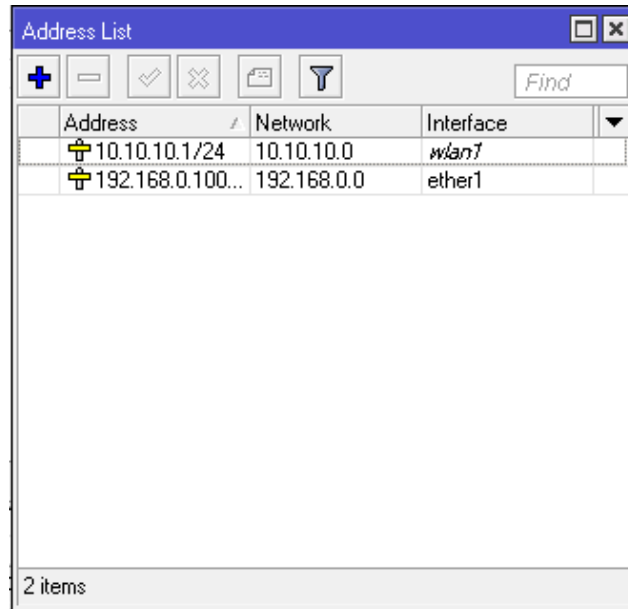
Como fue mencionado en secciones anteriores, siguiendo en la misma línea, RouterOS cuenta con una gran cantidad de herramientas, funciones y servicios orientados al desempeño de redes con un alto nivel de exigencia. Para enfoque de este estudio se tratarán a fondo los temas que a continuación se presentan.

3.1.3.1. Direccionamiento

En el capítulo 1 fue citado el concepto de una dirección IP llegando a la conclusión que es un número que identifica a un dispositivo dentro de un segmento de red; de manera que para el diseño de la red es necesario asignar direcciones a los dispositivos, teniendo así gestión remota de ellos.

Para dar direccionamiento a una interfaz física o virtual de un equipo, debemos ubicarnos en la sección de *IP/Addresses* dentro de RouterOS, debería de mostrarse una ventana como la que se ve a continuación en la figura.

Figura 14. Manejo de direcciones RouterOS



The screenshot shows the 'Address List' window in RouterOS. It features a toolbar with icons for adding (+), deleting (-), checking (✓), unchecking (✗), refreshing (refresh), and filtering (funnel), along with a 'Find' search box. Below the toolbar is a table with the following data:

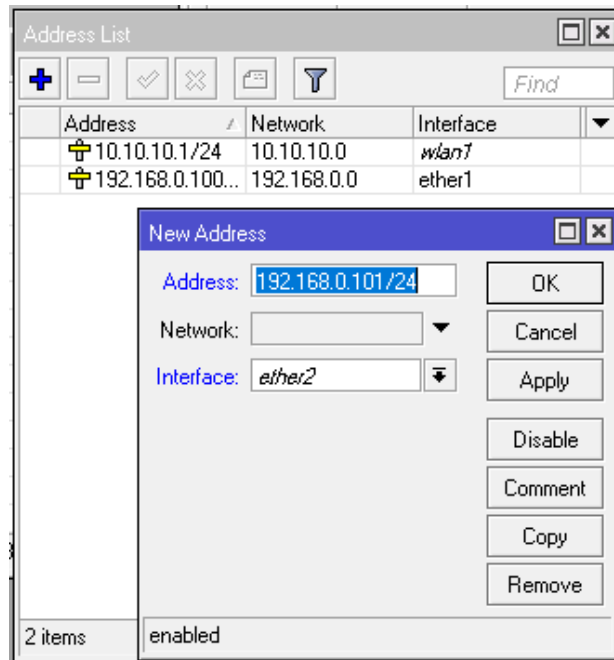
Address	Network	Interface
10.10.10.1/24	10.10.10.0	wlan1
192.168.0.100...	192.168.0.0	ether1

At the bottom left of the window, it indicates '2 items'.

Fuente: elaboración propia.

Seguido de esto para agregar una dirección IP se coloca sobre el botón con el signo “+”, se mostrará una nueva ventana en donde se podrá agregar la dirección IP; es de suma importancia resaltar que junto a la dirección IP se debe especificar la máscara de subred, como el ejemplo a continuación.

Figura 15. Agregando una dirección IP



Fuente: elaboración propia.

Finalmente, para confirma el cambio presionamos *Apply* y por ultimo *Ok* para cerrar la ventana; luego este cambio se reflejada en la tabla de dirección del enrutador.

3.1.3.2. Enrutamiento

Con los dispositivos Mikrotik se puede realizar enrutamiento dinámico y estático; se enfocará en el enrutamiento estático ya que la red en cuestión no es tan robusta; es topología simple que no requiere de la aplicación de enrutamiento dinámico.

Para agregar rutas estáticas se tiene que posicionar en el apartado de *IP/Routes* en RouterOS; una vez dentro aparecerá una ventana en donde se podrá encontrar todas las rutas que se encuentran en la tabla de enrutamiento del dispositivo.

Figura 16. **Tabla de rutas**

The screenshot shows the 'Route List' window in RouterOS. It features a toolbar with icons for adding (+), removing (-), enabling (checkmark), disabling (X), and refreshing (circular arrow), along with a search field labeled 'Find' and a dropdown menu set to 'all'. The main area contains a table with the following data:

	Dst. Address	Gateway	Distance	Routing Mark	Pref. Source
AS	▶ 0.0.0.0/0	192.168.0.1 reachable ether1	1		
DC	▶ 10.10.10.0/24	wlan1 unreachable	255		10.10.10.1
DAC	▶ 192.168.0.0/24	ether1 reachable	0		192.168.0.100

At the bottom left of the window, it indicates '3 items'.

Fuente: elaboración propia.

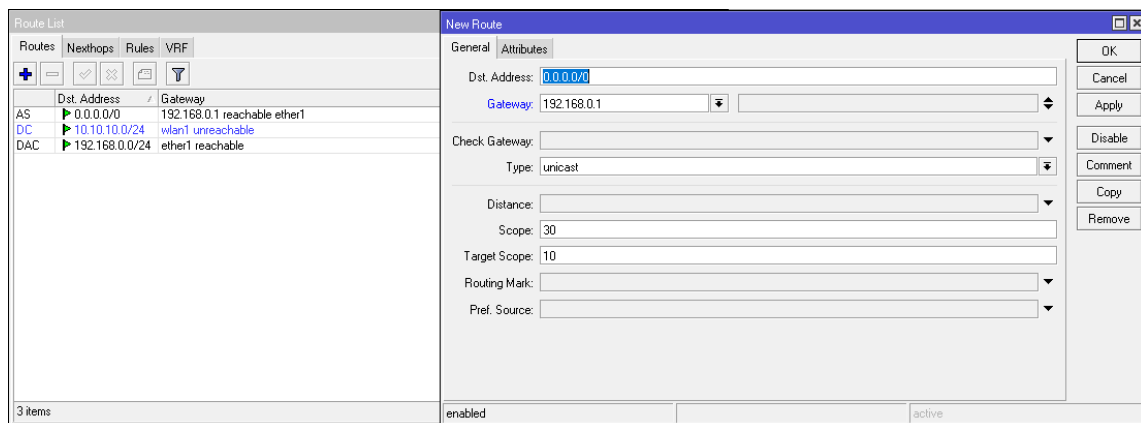
En este listado se puede encontrar todas las rutas del dispositivo, al lado de cada ruta se encuentran parámetros muy importantes como la distancia administrativa, marca de ruta, puerta de enlace predeterminada, así como las etiquetas ubicadas al lado izquierdo que indican el estado de la ruta en ese momento, a continuación, se muestra un listado con las etiquetas y su significado.

- X: deshabilitada

- A: activa
- D: dinámica
- C: conectada
- S: estática

Al igual que en la sección de direcciones para agregar una ruta se ubica en el botón con el signo +, luego de esto se colocan los parámetros de la ruta, puerta de enlace predeterminada, red de destino, marca de ruta, distancia administrativa, entre otros.

Figura 17. **Agregando una ruta**



Fuente: elaboración propia.

Por último, se presiona *Apply*, posteriormente *Ok* para terminar el proceso agregando así una nueva ruta estática al dispositivo.

3.1.3.3. Puente

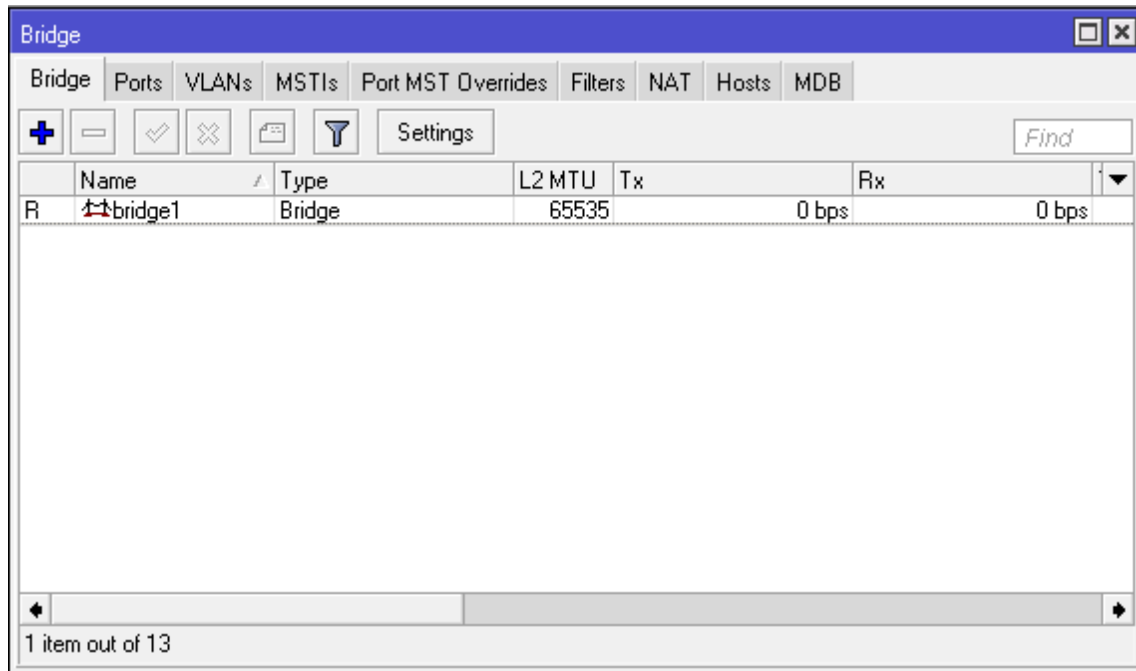
El puente se utiliza para la interconexión entre segmentos de red, trabaja sobre la capa número 2 del modelo OSI realizando la transmisión de datos desde un segmento de red hacia otro. Trabaja con el protocolo CSMA/CD al igual que Ethernet lo cual le permite realizar un censo sobre la red antes de transmitir los datos, la transferencia de datos se realiza en base a la dirección física de destino en cada paquete; cabe mencionar también que el término puente se utiliza para dispositivos que operan de acuerdo al estándar IEEE 802.1D.

Algo muy interesante de esto es que permite la comunicación entre diferentes segmentos de red unificándolos en una sola subred, todo esto sin necesidad de utilizar un enrutador. Por otro lado, el direccionamiento de paquetes se realiza de acuerdo a la tabla de direcciones físicas que se encuentran dentro de los segmentos de red conectados al puente; cuando detecta que un segmento de red está tratando de transmitir algún dato toma la trama de datos para posteriormente enviarla a la dirección física especificada.

Utiliza una dinámica de autoaprendizaje, de forma que el encaminamiento de las tramas no requiere de alguna configuración adicional manual, se vale de una tabla denominada de reenvío en la que se almacenan las direcciones físicas asociadas a la interfaz física donde se encuentran conectadas; si un puente no encuentra dentro de su tabla la dirección física de destino envía la trama por todas las interfaces excepto por la interfaz por lo que vino el paquete. Este tipo de interfaces pueden ser creadas por medio de RouterOS, algunos dispositivos Mikrotik cuentan con la ventaja de hacer conmutación por medio de chip de *switch* y no en *software* lo cual representa una gran ventaja respecto al tiempo de respuesta de la red, procesamiento, entre otros.

Para crear uno en RouterOS se posiciona sobre el apartado de Bridge, seguido de esto lo agregamos en el botón con el signo + al igual que en otras secciones, finalmente aplicamos los cambios.

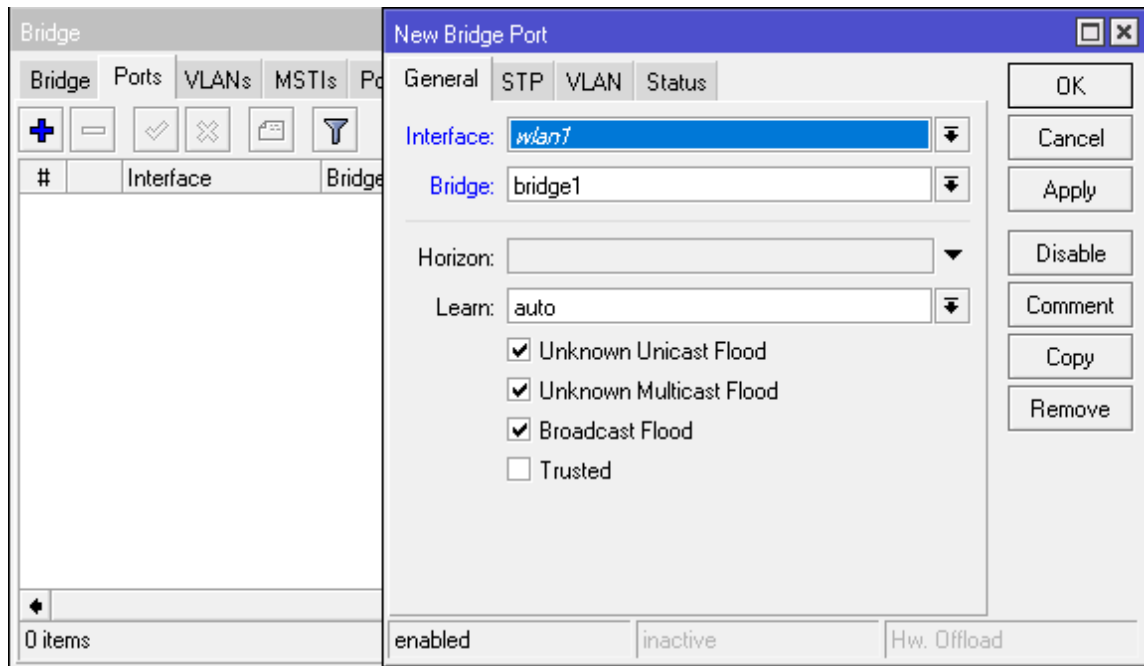
Figura 18. **Puente**



Fuente: elaboración propia.

Para agregar los puertos físicos que se desean estén dentro del puente, es necesario ir a la pestaña llamada *Ports*, aquí se seleccionan los puertos agregándolos uno a la vez presionando el botón + al igual que cuando se creó el puente.

Figura 19. **Agregando puertos a un puente**



Fuente: elaboración propia.

3.1.3.4. **Firewall**

También conocido como corta fuegos es la parte de una red encargada de establecer políticas sobre el manejo del tráfico desde, hacia y a través del dispositivo. Básicamente, maneja el tráfico entre dos o más redes, hoy en día es un dispositivo indispensable dado que protege los datos que se manejan a través de nuestra red volviéndola segura.

RouterOS por defecto tiene implementado un paquete de *firewall*, este generalmente se utiliza para que usuarios no autorizados desde internet tengan acceso a la red privada, contenidos web peligrosos, análisis de puertos, filtrado

de paquetes, marcado de paquetes, enmascaramiento de la red (NAT), entre otros.

La versión actual de RouterOS en la que se encuentra implementado el *firewall* está basada en el *kernel* versión 3.3.5 de Linux, funciona como un *stateful firewall*, esto quiere decir que desarrolla una inspección del estado de los paquetes, también chequea el estado de las conexiones que viajan a través del enrutador. Utiliza dos reglas denominadas *the matcher* y *the action*.

El *matcher* analiza los siguientes parámetros:

- *Source mac address*
- Direcciones IP (Red o lista)
- Tipos de direcciones (*broadcast, multicast, unicast, local*)
- Puerto
- Protocolo
- Interface por donde entra o sale el paquete
- *DSCP byte*

Finalmente determinando una acción en dado caso se cumplen los parámetros, posteriormente a la revisión se procede con una determinada acción. A primera vista el repertorio de la sección de *firewall* se ve de la siguiente manera.

Figura 20. Primera vista *firewall*

#	Action	Chain	Src. Address	Dst. Address	Proto...	Src. Port	Dst. Port	In. Inter...	Out. Int...	Bytes	Packets
0 D	jump	forward								0 B	0
1 D	jump	forward								0 B	0
2 D	jump	input								0 B	0
3 D	drop	input			6 (tcp)		64872-64875			0 B	0
4 D	jump	hs-input								0 B	0
5 D	acc...	hs-input			17 (u...		64872	hs-input		0 B	0
6 D	acc...	hs-input			6 (tcp)		64872-64875	hs-input		0 B	0
7 D	jump	hs-input								0 B	0
8 D	reject	hs-unauth			6 (tcp)					0 B	0
9 D	reject	hs-unauth								0 B	0
10 D	reject	hs-unauth-to								0 B	0
::: place hotspot rules here											
11 X	pas...	unused-hs...								0 B	0

Fuente: elaboración propia.

Para objetos de este estudio solo profundizara sobre sección de *filter rules*, dado que las otras secciones no son de utilidad para este caso.

3.1.3.5. Copia de respaldo

Realizar un respaldo de la configuración de cada equipo es un paso muy importante dentro de las buenas prácticas del mantenimiento de una red, sobre todo cuando se realizan cambios o bien actualizaciones en los equipos. Por su parte Mikrotik con la implementación RouterOS brinda una herramienta para realizar estos respaldos; se pueden realizar de dos tipos: de configuración binaria o bien en formato de texto plano.

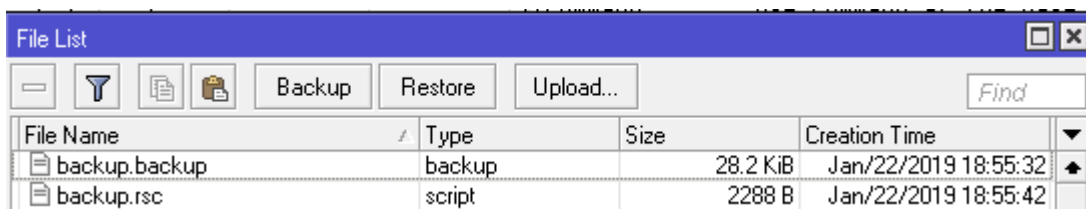
Para esto se debe ir a la sección llamada *New Terminal* seguido de esto se deben utilizar los comandos que a continuación se presentan.

- Binario: system backup save name=backup

- Texto legible: export file=backup

Finalmente, los archivos generados podrán ser encontrados en la sección *Files* dentro de RouterOS como se ve en la figura.

Figura 21. **Copia de respaldo**



The screenshot shows a 'File List' window with a blue title bar and standard window controls. Below the title bar is a toolbar with icons for back, filter, copy, and paste, followed by buttons for 'Backup', 'Restore', and 'Upload...'. A search box labeled 'Find' is on the right. The main area contains a table with the following data:

File Name	Type	Size	Creation Time
backup.backup	backup	28.2 KiB	Jan/22/2019 18:55:32
backup.rsc	script	2288 B	Jan/22/2019 18:55:42

Fuente: elaboración propia.

3.2. Requerimientos de la configuración

Como fue mencionado en la sección 1.5.2, los paquetes *multicast* llegan a quien muestra interés por recibirlo, habiéndose suscrito al grupo de multidifusión previamente para recibir el tráfico, por lo cual se requiere lo siguiente.

- Conectividad a nivel de capa 2
- Enrutamiento a nivel de capa 2
- Gestión vía remota de equipos a nivel de capa 3
- Flexibilidad de conmutación por implementación de filtros

3.2.1. Planteamiento de la configuración

Para realizar la configuración se necesitan delimitar ciertos requerimientos, en el origen del tráfico *multicast*, así como en el punto de entrega el tráfico se recibirá en una interface y será distribuido en múltiples interfaces de acceso, es lógico entonces que en estos puntos solamente se necesitaría enrutamiento físico para los paquetes.

Al contrario, la configuración del equipo ubicado en el nodo de distribución presenta una complicación con respecto a los otros, sucede que todo el tráfico proveniente de diversos puntos será concentrado en este equipo. Se debe considerar que se tiene que restringir el tráfico que se deja pasar por este enrutador hacia la red de distribución, fijar una línea entre el tráfico necesario e innecesario.

Finalmente, pero no menos importante, se debe considerar que el ancho de banda transportado no puede sobrepasar un límite establecido, dado que de suceder esto, afectaría drásticamente la calidad de la señal que se está transportando, de tal forma que la señal se degradaría totalmente.

3.2.2. Máxima unidad de transferencia (MTU)

Es un dato que limita la longitud máxima en *bytes* de un datagrama que puede ser transferido a través de un enlace utilizando un protocolo de comunicaciones. Por defecto un datagrama Ethernet tiene un MTU de 1 500 *bytes*, aunque puede llegar teóricamente hasta 65 535 dependiendo de los enlaces de transmisión.

El manejo de dicho parámetro es crucial para el funcionamiento de la red de transporte, dado que un MTU muy bajo puede derivar en un resultado bastante contraproducente para la red, mientras que si se utiliza un valor alto se asegura que en los datagramas se utilizará la máxima capacidad de carga útil; pero se debe ser cuidadoso en cuanto a posibles cuellos de botella en la red. Un valor bajo genera lo que se denomina como fragmentación de paquetes, cuando un paquete sobrepasa la unidad máxima de transferencia el enrutador divide los datos en diferentes datagramas.

Cada datagrama creado por el enrutador tiene distintos valores en sus cabeceras, ocupando de esta manera una mayor cantidad de ancho de banda en la red, lo cual afecta el desempeño de la misma. En el caso de *multicast* afecta la calidad de la señal resultando en una calidad baja de imagen, audio e incluso la transmisión del canal; paraliza por momentos la transmisión de datos; derivado de esto es que se hace necesario realizar un estudio más a fondo sobre el tráfico que se desea transportar, se evita así una afectación sobre el servicio.

3.2.3. Implementación de puente

Como fue mencionado en el planteamiento de la configuración, solamente se necesita direccionamiento físico dentro de la red para la fase de distribución del tráfico, es así como se hace evidente que se puede implementar un puente en las interfaces del enrutador para poder distribuir el tráfico *multicast*.

A pesar de que la fase de distribución también puede ser realizada por un *switch* común, la implementación de un puente dentro de la configuración del mismo enrutador que recibe el tráfico, se presenta como la opción más viable debido a que no se agregan gastos en otros equipos e incluso es posible

gestionar el equipo remotamente. La implementación dentro de la fase de distribución es más simple de lo que parece, se agrega un puente como fue indicado en la sección 3.1.3.3; posteriormente se agregan al puente los puertos en los que se desea distribuir el flujo *multicast*; es muy importante agregar el puerto donde se recibe el tráfico. Finalmente, en el puente deberían de estar involucrados los puertos de distribución y el puerto de entrada de la señal, implementando de esta forma la fase de distribución.

3.2.4. Posibles dificultades

A pesar de que la fase de distribución se limita simplemente a la conectividad a nivel de capa 2 por medio de un Bridge, es importante resaltar que la posible integración de múltiples *headends* puede representar una complicación con respecto a la configuración del nodo de distribución.

Dado que un mismo canal por ejemplo puede provenir desde dos fuentes diferentes e incluso con el mismo direccionamiento IP por lo cual es de suma importancia encontrar una solución en la configuración que ofrezca la integración de varias fuentes; este tema será explicando a fondo en el próximo capítulo de esta investigación.

Además de la dificultad que representa lo expuesto anteriormente, se debe tomar en cuenta que se deben conmutar las diferentes fuentes de origen del tráfico, enlaces redundantes capaces de cubrir las necesidades en casos de fallas.

3.3. Desarrollo de configuración

Con base en todas las consideraciones, así como los requerimientos anteriores se desarrolló la configuración que a continuación se presenta, cabe agregar que serán explicadas detalladamente las configuraciones desarrolladas para los enrutadores.

3.3.1. Configuración *headend*

Primero se identifica el equipo para diferenciarlo dentro de la red de transporte; seguido, se crea un puente para recoger el tráfico desde el origen y distribuirlo en las interfaces pertinentes, por comodidad se presenta la configuración aplicada desde la línea de comandos; también, puede ser realizada en Winbox a través del modo gráfico como fue presentado en secciones anteriores.

Figura 22. Código *headend*

```
/system identity
set name=Headend
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast"
interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.1/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.0
```

Fuente: elaboración propia.

La interface 1 en este caso es la de entrada, el resto de interfaces que se encuentran en el puente son las de distribución; es decir, las interfaces donde se puede recoger el tráfico; también, fue agregada una IP de gestión al equipo sobre el puente con el objetivo de tener gestión remota del equipo sobre el enlace punto a punto.

3.3.2. Configuración nodo de distribución

La conexión punto a punto entre el nodo de distribución y la cabecera hace que la configuración de este equipo sea muy parecida a la presentada anteriormente, el equipo es identificado con un nombre, se le coloca una dirección IP de gestión, se incluye un puente que sirve de distribución, así como de gestión para la red.

Figura 23. Código nodo de distribución

```
/system identity
set name="Nodo de distribucion"
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast"
interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de gestion" interface=ether4
/ip address
add address=10.10.10.2/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.0
add address=10.10.10.5/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.0
```

Fuente: elaboración propia.

En este caso la interfaz 1 es utilizada para recibir el tráfico, la interfaz 2 y 3 para distribución; la interfaz 4 es para gestionar los equipos.

3.3.3. Configuración cliente final

En el caso del cliente final se utilizará una configuración similar a la utilizada en la señal de origen; recoge el tráfico en la interface 1, tomando como interfaces de acceso de la 2 a la 5.

Figura 24. Código cliente final

```
/system identity
set name="Cliente final"
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast"
interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.6/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.0
```

Fuente: elaboración propia.

También, fue agregada una IP de conexión punto a punto para gestionar el equipo vía remota desde el nodo de distribución.

4. IMPLEMENTACIÓN DE RED Y APLICACIÓN DE FILTROS PARA ENLACES REDUNDANTES

Luego de considerar todos los factores presentados en secciones anteriores, desarrollada la configuración necesaria, es posible realizar la implementación final de la red. No obstante, hay un factor de suma importancia a tomar en cuenta, la implementación final de la red requiere de múltiples señales de origen dada a necesidad de que la señal sea ininterrumpida.

4.1. Problemática de integración *headend* secundario

La integración de un segundo e incluso un tercer *headend* es una pieza fundamental de este proyecto, sin esto no es posible garantizar que el servicio brindado responda en cualquier circunstancia. Para esto primero que nada debemos de considerar que el tráfico proveniente de los diversos orígenes debería tener el mismo direccionamiento IP *multicast*, dado que en caso de falla en alguno de los orígenes de la señal simplemente se debe de tener la facilidad de conmutar el tráfico hacia la red de distribución, a eso se debe sumar la limitante del ancho de banda dado que fueron considerados enlaces de datos de 1 Gbps; derivado de esto, en función de la capacidad del enlace es vital limitar la cantidad de canales que se distribuyen por que no se puede saturar la red de distribución.

Por lo cual se vuelve evidente que para integrar múltiples orígenes de la señal se necesita de alguna manera filtrar el contenido que se desea dejar pasar hacia la red de distribución a través del enrutador ubicado en el nodo de central. También, es necesario encontrar la manera en que se pueda tomar el

tráfico e incorporarlo a la programación de canales existente, integrando así una señal unificada.

4.1.1. Implementación de *firewall*

Como fue mencionado en el capítulo anterior, el *firewall* implementado en RouterOS maneja el tráfico que fluye desde, hacia y a través del enrutador. Dadas las condiciones es claro entonces que se adapta perfectamente a las necesidades descritas con anterioridad debido que se necesita controlar el tráfico que fluye hacia la red de distribución que será manejado por el enrutador en el nodo central.

Recuérdese entonces que las cadenas (*chains*) manejadas por el *firewall* son 3, *input*, *output* y *forward*, dado que el tráfico está pasando a través del enrutador se utilizará la cadena *forward*. En el mismo orden de las ideas anteriores se utilizarán las acciones *drop* y *accept* dado que simplemente deseamos darle paso o no al flujo de datos a través del enrutador. A manera de resumen final, se aplicarán filtros sobre direcciones IP *multicast* utilizando la cadena *forward*, así mismo utilizando la acción *Drop* generalmente para bloquear o en su defecto la acción *accept* para permitir el paso a través del enrutador principal.

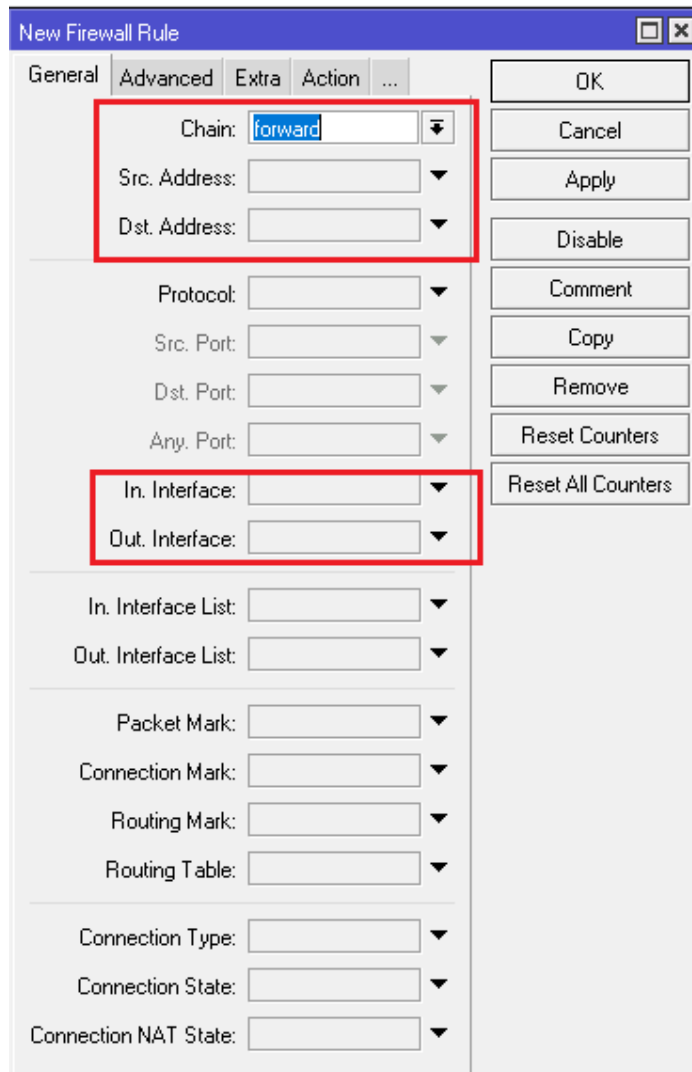
4.1.1.1. Filtros a nivel de capa 3 del modelo OSI

Previamente al diseño e implementación de los filtros es necesario comprender perfectamente el funcionamiento de los filtros a utilizar, dichas reglas serán cimentadas sobre la capa 3 del modelo OSI, como bien fue mencionado en el capítulo 1 es en esta capa donde se maneja el concepto de

dirección IP e interviene el enrutador como dispositivo base para dar direccionamiento lógico dentro de una red de área local.

Para crear un filtro se ubica. sobre la sección de *firewall/filter rules*, agregar una nueva regla, se selecciona la dirección ya sea de origen (*Src address*) o destino (*Dst address*). También, se puede seleccionar la interface por la cual está entrando o saliendo el tráfico; para el caso de los filtros a diseñar, se utilizarán direcciones de origen y destino, así como interfaces de entrada o salida.

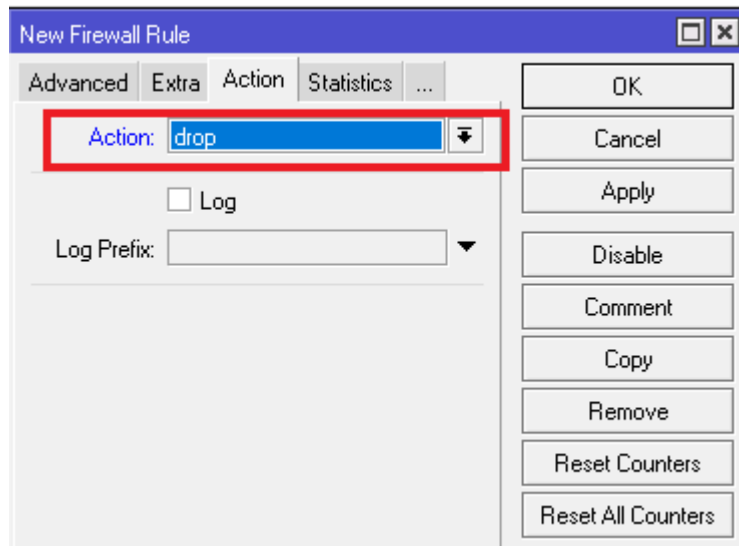
Figura 25. **Filtro *firewall***



Fuente: elaboración propia.

Finalmente, se selecciona la acción a realizar, en dado caso se cumplen las condiciones previamente establecidas para la regla.

Figura 26. **Parámetros finales filtro *firewall***



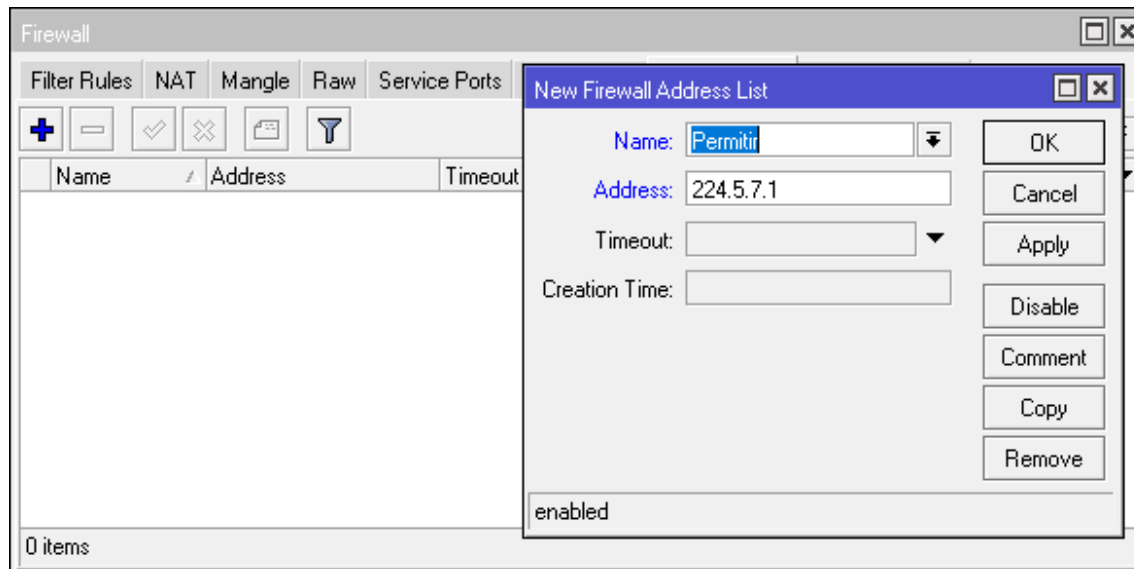
Fuente: elaboración propia.

4.1.2. **Address list**

Dado que cada canal es transportado en la red utilizando una dirección IP *Multicast* en el desarrollo del filtro no es posible restringir un segmento de direcciones completo; se necesita permitir o bloquear el flujo de determinadas direcciones. En RouterOS existe una herramienta llamada *address list* (lista de direcciones); básicamente, se puede crear un directorio completo de direcciones específicas sin necesidad de agregar un segmento entero.

Dicha funcionalidad se encuentra en la sección de *firewall/address list*, una vez dentro se debe agregar una dirección, seguido de esto pedirá los siguientes parámetros.

Figura 27. **Address list**



Fuente: elaboración propia.

Finalmente, se tendrá una lista seleccionada de las direcciones que se desean gestionar para utilizarlas en los filtros que serán explicados en la sección posterior.

4.1.3. Diseño e implementación de filtros

El diseño del filtro es realmente simple, para objeto de ejemplo se desarrolló un filtro capaz de bloquear el tráfico contenido en las IP listadas a continuación.

- 224.5.7.1:3080
- 224.6.6.7:3080
- 224.5.8.2:1030
- 224.7.1.1:1030

- 224.8.8.5:1030

Las direcciones que no fueron enlistadas anteriormente tendrán paso libre a través del enrutador; a continuación, se muestra la configuración para la implementación de un filtro de este tipo; la configuración será mostrada por comodidad desde la línea de comando.

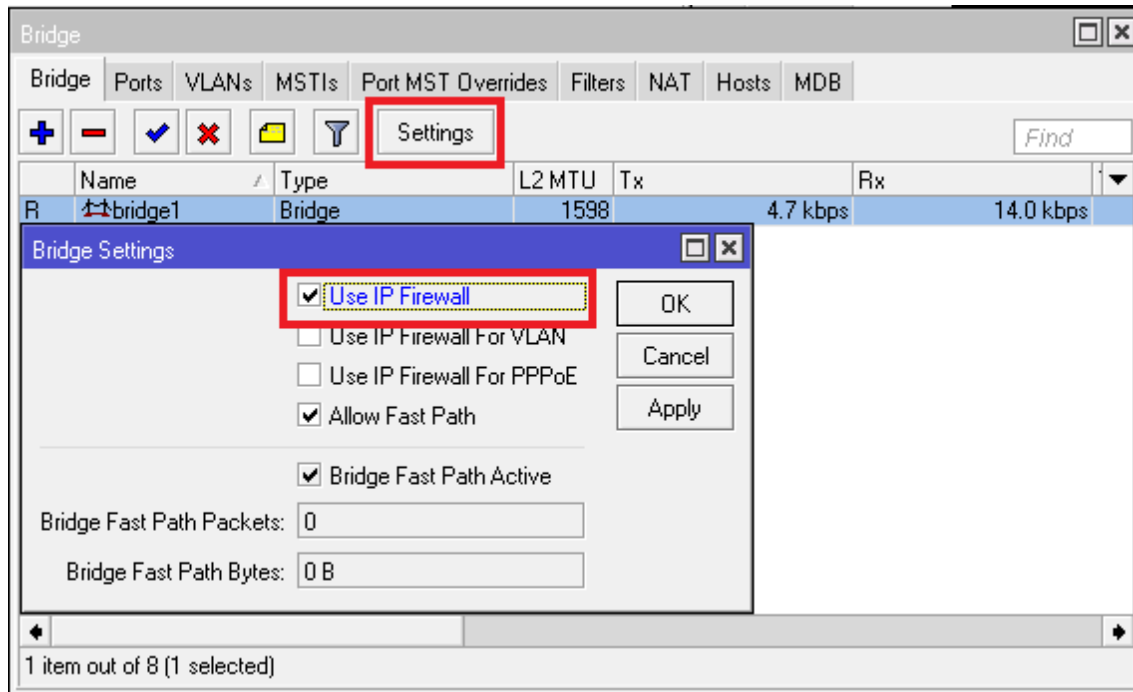
Figura 28. **Filtro simple**

```
/ip firewall address-list
add address=224.5.7.1 list=Bloquear
add address=224.6.6.7 list=Bloquear
add address=224.5.8.2 list=Bloquear
add address=224.7.1.1 list=Bloquear
add address=224.8.8.5 list=Bloquear
/ip firewall filter
add action=drop chain=forward dst-address-list=Bloquear out-interface=ether4
src-address-list=Bloquear
```

Fuente: elaboración propia.

En este caso se crea un *address list* llamado bloquear, esta contiene las direcciones IP de destino que se desea bloquear en el enrutador; finalmente, se crea un filtro en el *firewall* que especifica la interface de salida, lista de direcciones y la correspondiente acción en dado caso se cumplan las condiciones establecidas. Cabe resaltar que si una interface se encuentra en modo *slave* (esclavo), es decir, dentro de un puente, no se pueden aplicar políticas de *firewall* sobre esta interface; las políticas de Firewall podrán ser aplicadas si se habilita la función llamada *use ip firewall* que se puede usar solamente entrando a la sección de *bridge* (puente) en el botón llamado *settings*, como se muestra en la figura a continuación.

Figura 29. IP Firewall Bridge



Fuente: elaboración propia.

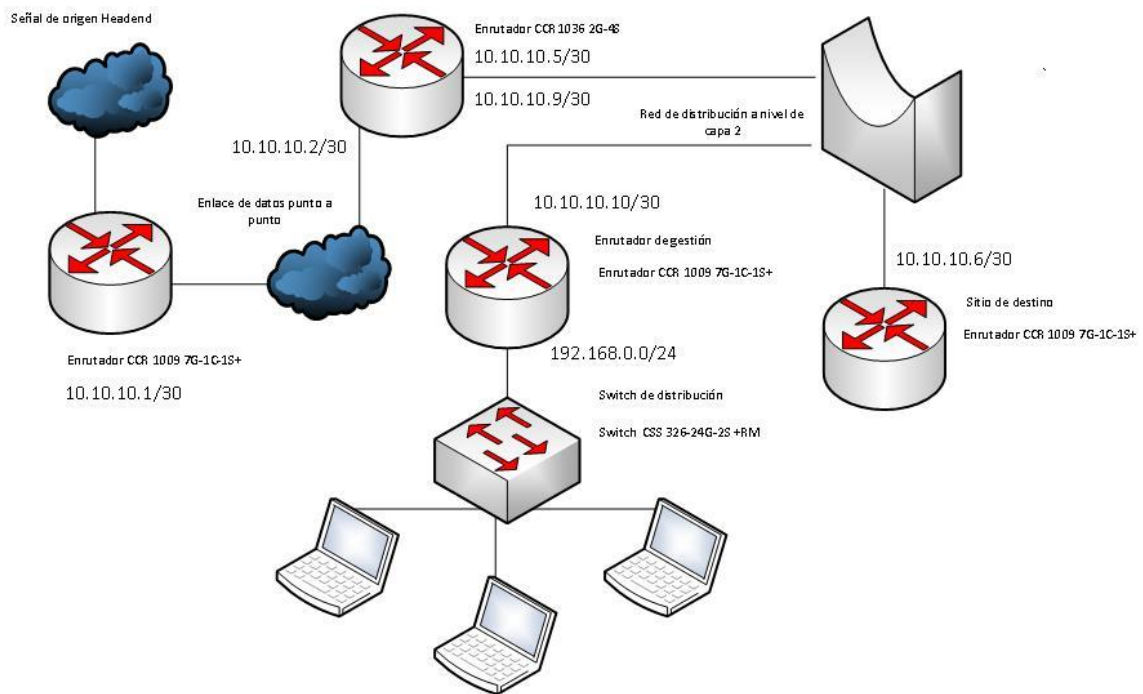
Como fue explicado en secciones anteriores, el uso de un puente es obligatorio dentro de la red de distribución, por lo cual derivado de esto la habilitación de esta función es de carácter obligatorio.

4.2. Implementación de red de gestión

En el capítulo anterior dentro de la configuración fue asignada una segmentación de direcciones IP privadas para gestionar los equipos remotamente; para mantener el monitoreo sobre los canales digitales que se transportan a través de la red. Para integrar la red de gestión, es necesario agregar un enrutador y un equipo de distribución que podría ser un *switch*, agregando la última pieza para que el diseño de la red este completo; la red de

gestión y monitoreo es de suma importancia dado que sin el debido monitoreo de la red podría llegar a colapsar con el pasar del tiempo; también, es necesario realizar operaciones preventivas y correctivas a la red. El cambio en el diseño de la red es mínimo, por su lado la configuración no sufre cambios abruptos, simplemente se deben de agregar rutas por defecto dirigidas hacia el nodo de distribución. La nueva estructura de la red se presenta a continuación en la figura.

Figura 30. Red de gestión



Fuente: elaboración propia, empleando Visio 2016.

Como se puede ver la red de gestión se encuentra dentro de otro segmento de red, el encargado de realizar el enrutamiento hacia los diferentes puntos es el enrutador ubicado en el nodo de distribución. Adicionalmente al monitoreo de los equipos desde la red de gestión, se pueden visualizar los

canales transportados dado que se encuentra directamente conectada a la red de distribución.

4.3. Implementación de seguridad en la red

Dada la incorporación de la red de gestión, pero sobre todo por la seguridad de la red en general es importante implementar medidas de seguridad para evitar tener una penetración con fines maliciosos en la red. Lógicamente, la información transportada tiene cierto valor, es muy importante cuidar la integridad de los datos implementando sistemas de detección de intrusos (IDS), así como reglas de seguridad a nivel de *firewall* para que el acceso a los equipos sea exclusivamente desde la red de gestión; también, sería de suma importancia homologar los puertos y servicios habilitados en la red e incluso una revisión periódica de la red a nivel de seguridad en busca de alguna debilidad.

4.3.1. Políticas de *firewall*

La implementación de políticas de seguridad en el *firewall* es tan solo uno de los pasos propuestos para restringir el acceso a la red, se propone implementar reglas de control en el *router* principal de distribución debido a que toda la red converge en este dispositivo. Dichas reglas se encargarían de prohibir la comunicación entre los equipos de los clientes, evitando de esta manera alguna infiltración en la red desde algún equipo cliente; a continuación, se muestra una regla en la cual no se permite la comunicación entre clientes a través del *router* principal, con la excepción de la red de gestión.

Figura 31. **Deshabilitando comunicación entre clientes**

```
/ip firewall address-list
add address=192.168.0.0/24 list="Excepcion gestion"
/ip firewall filter
add action=drop chain=forward dst-address=10.0.0.0/8 src-
address=10.0.0.0/8 src-address-list="!Excepcion gestion"
```

Fuente: elaboración propia.

En este ejemplo se implementa el uso de *address list* presentadas en capítulos anteriores, se utiliza una lista por que da la flexibilidad de poder agregar más segmentos de red en dado caso fuera necesario, pero no permite la comunicación entre clientes a través del *router* por eso se utilizó cadena *forward*. Por lo cual se evita que los clientes se puedan comunicar con otros clientes, acotando la vista de la red desde la perspectiva del cliente, solamente podría ver el equipo ubicado en el nodo central (si se lo permite el administrador) mas no el resto de la red.

4.3.2. Descubrimiento de vecinos

Neighbor Discovery es un protocolo IPv6 que utiliza mensajes ICMPv6 para descubrir a los dispositivos que puede ver a través de sus interfaces, esto podría representar un riesgo para la seguridad de la red en cuestión, por lo cual se recomienda el uso adecuado del protocolo en los dispositivos. Mikrotik brinda la manera de poder restringir las interfaces que pueden realizar la búsqueda de vecinos, por defecto todas las interfaces pueden realizar la búsqueda por lo que se recomienda tomar en consideración cambiar esta configuración.

Para restringir la búsqueda de dispositivos, se utiliza la herramienta *interface list*, por defecto en el equipo vienen configuradas tres listas con las siguientes funciones.

- *All*: contiene todas las interfaces
- *Dynamic*: contiene las interfaces dinámicas
- *None*: no contiene interfaces

En nuestro caso será agregada una nueva lista en la que se especificarán las interfaces en las cuales estará habilitada la búsqueda de vecinos, lógicamente la única interfaz que podría tener habilitada la búsqueda sería las interfaces de distribución en el caso del nodo central y la interface WAN en el caso del cliente final. Esto se realiza por medio de los comandos que a continuación se presentan; para este ejemplo se agregó una nueva lista con la única interfaz en la que estará habilitado el protocolo *neighbor discovery*, posteriormente en la tabla de vecinos del dispositivo se especifica la lista que contiene las interfaces que pueden correr este protocolo.

Figura 32. **Restricción de Neighbor Discovery vía CLI**

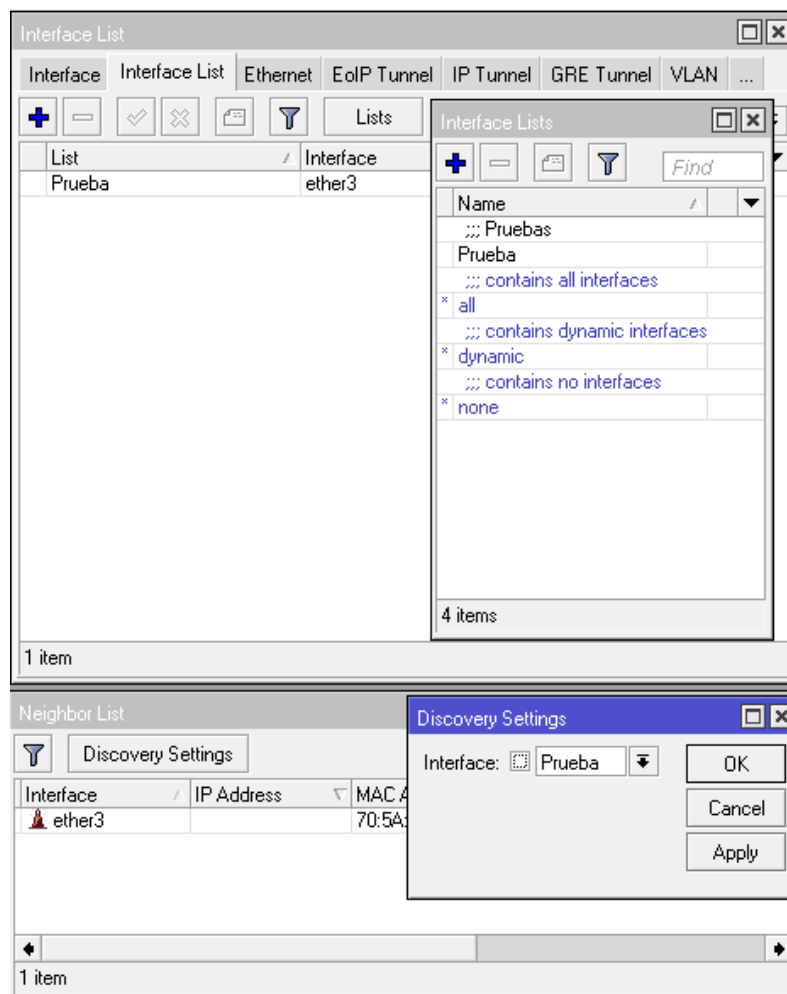
```
/interface list
add comment=Pruebas name=Prueba
/interface list member
add interface=ether3 list=Prueba
/ip neighbor discovery-settings
set discover-interface-list=Prueba
```

Fuente: elaboración propia.

De la misma manera se pueden agregar más interfaces a juicio del administrador de la red, todo esto con el fin de evitar que los equipos que están

conectados al CPE del cliente final puedan tener visualización de la red de transporte internamente. Si se desea realizar el cambio desde la interfaz gráfica, se utilizan las opciones *interface list* en conjunto con *neighbor list*, como se muestra en la figura.

Figura 33. Restricción de Neighbor Discovery vía GUI

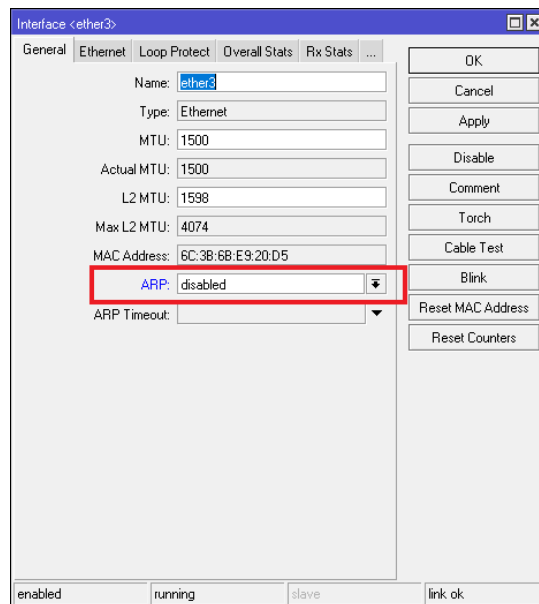


Fuente: elaboración propia.

4.3.3. Restricción de ARP

ARP es un protocolo bastante parecido a *neighbor discovery*, se encarga de encontrar la dirección física que pertenece a determinada IP, Mikrotik también incluye la opción de poder deshabilitar la utilización de este protocolo a través de cualquiera de sus interfaces evitando así que los equipos sean visibles en la red. Para este trabajo fines sería una buena práctica evitar que desde la sede de los clientes finales puedan ver los demás equipos de la red por lo que se recomienda hacer uso adecuado de este protocolo; la deshabilitación del protocolo es muy sencilla, simplemente se ingresa a la interfaz en la cual se desea deshabilitarlo y en la opción llamada ARP se cambia el valor a *Disabled* como se muestra a continuación, por defecto viene habilitado en todas las interfaces.

Figura 34. Deshabilitación ARP vía GUI



Fuente: elaboración propia.

Esto también puede ser realizado desde la línea de comando como se muestra en la siguiente figura, para este caso se está deshabilitando el protocolo en la interfaz Ethernet 3 del dispositivo.

Figura 35. **Deshabilitación ARP vía CLI**

```
/interface ethernet  
edit 2 value-name=arp
```

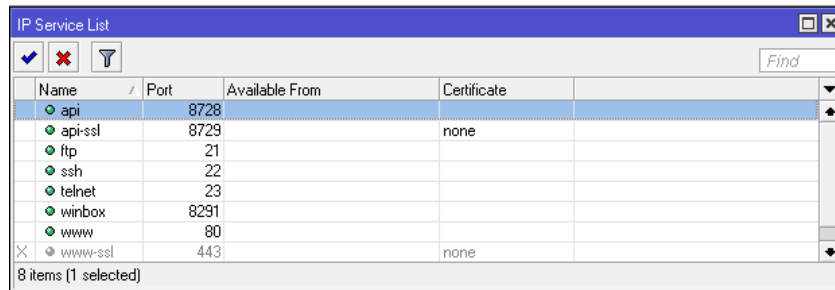
Fuente: elaboración propia.

4.3.4. Homologación de servicios y puertos

Establecer los puertos y también los servicios que serán utilizados en la red de distribución es un paso importante para controlar el acceso a la misma, los servicios disponibles dentro RouterOS pueden ser deshabilitados; también, se puede establecer el puerto específico para un servicio e incluso se puede hacer un arreglo para que solo pueda ser desde un segmento de red en específico.

Como fue mostrado en la sección 3.1.2.6 de esta investigación, existen varias formas de gestionar los equipos, para deshabilitar/habilitar dichos servicios, esto puede ser hecho desde la línea de comando o bien desde la interfaz gráfica de RouterOS. Para hacer esto desde la interfaz gráfica, se va al menú de IP, luego se busca *Services*, aparecerá una ventana como la que a continuación se presenta en la figura.

Figura 36. Listado de servicios vía GUI



The screenshot shows a window titled "IP Service List" with a table of services. The table has columns for Name, Port, Available From, and Certificate. The "api" service is selected, indicated by a blue highlight and a small "X" icon in the left margin. Other services listed include api-ssl, ftp, ssh, telnet, winbox, www, and www-ssl.

Name	Port	Available From	Certificate
api	8728		
api-ssl	8729		none
ftp	21		
ssh	22		
telnet	23		
winbox	8291		
www	80		
www-ssl	443		none

Fuente: elaboración propia.

Para deshabilitar un servicio, nos ubicamos sobre el botón marcado con una X, luego de presionar automáticamente el servicio se encuentra deshabilitado; si se desea habilitar nuevamente simplemente se presiona el botón que se encuentra al lado izquierdo del que se utiliza para deshabilitar. El manejo de los servicios también se puede realizar desde la línea de comando a partir de los comandos presentados a continuación.

Figura 37. Habilitación y des habilitación de servicios vía CLI

```
/ip service disable X  
/ip service enable X
```

Fuente: elaboración propia.

Donde X representa el número de servicio según el listado del equipo; para ver el listado completo se puede utilizar el comando *print* desde la línea de comandos, lo cual mostraría una vista muy parecida a la siguiente.

Figura 38. Listado de servicios vía CLI

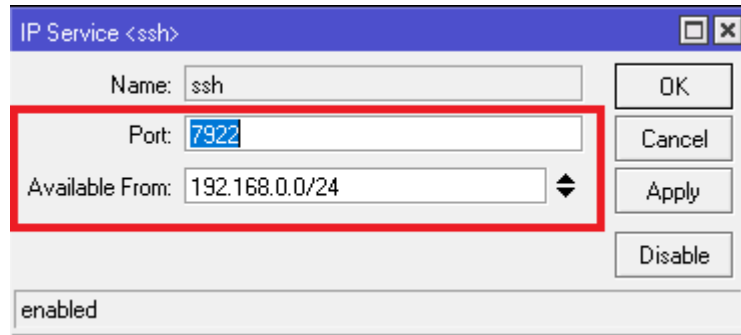
```
[admin@MikroTik] > /ip service print
Flags: X - disabled, I - invalid
#  NAME      PORT ADDRESS      CERTIFICATE
0  XI telnet   23
1  ftp       21
2  www       80
3  ssh       22
4  XI www-ssl  443           none
5  api       8728
6  winbox    8291
7  api-ssl   8729           none
```

Fuente: elaboración propia.

De aquí puede ser tomado el número de servicio que se desea editar, se recomienda que si los servicios no son utilizados estos sean deshabilitados, limitando de esta manera formas de acceso a los equipos, acotando las posibilidades. Otra buena práctica sería cambiar los puertos por defecto de los servicios y cambiar el rango de direcciones IP desde el cual el servicio se encuentra disponible, en dado caso se encuentren habilitados; el cambio de estos valores también puede ser realizado desde la línea de comando como desde la interfaz gráfica.

Para realizar el cambio desde la interfaz gráfica, se necesita seleccionar el servicio en cuestión dando doble clic sobre el mismo; a continuación, se mostrarán los valores que pueden ser editados, para nuestro caso serían los campos *port* y *available from*; en este ejemplo se especifica un puerto diferente para el servicio SSH; adicionalmente, se especifica el único segmento de red desde el cual podrá ser accesible este servicio, en este caso la red de gestión.

Figura 39. **Editando servicios vía GUI**



Fuente: elaboración propia.

Utilizando la línea de comando podría ser de la siguiente manera, especificando los mismos parámetros para ese servicio, donde X representa el número de servicio.

Figura 40. **Editando servicios vía CLI**

```
/ip service  
edit X value-name=port value  
/ip service  
edit X value-name=address
```

Fuente: elaboración propia.

Cuando se ejecuta cualquiera de estos comandos se apertura una ventana en la cual se debe de especificar el valor para esta característica del servicio. Se recomienda homologar esta configuración para todos los equipos dentro de la red para evitar ataques a través de los puertos por defecto de los servicios, deshabilitar los que no se utilizan y delimitar el segmento de red que puede consultarlos.

4.3.5. Sistemas de protección

Incorporar sistemas de protección ante ataques externos sería de gran ayuda para garantizar la integridad de la información que se transporta en la red, por lo cual se presentan a continuación alternativas de implementación de sistemas de detección, así como sistemas de protección de intrusos.

4.3.5.1. *Intrusion detection system (IDS)*

Un IDS es un sistema de detección de intrusos (*intrusion detection system*) se trata de un sistema diseñado para captar información proveniente de la red, analizar los datos en busca de coincidencias con ataques conocidos y advertir cuando la red se encuentra ante una posible amenaza. Es de suma importancia para la implementación de seguridad en una red, utiliza herramientas como por ejemplo *sniffers* para que los datos puedan ser analizados, cabe resaltar que un IDS por sí solo no es capaz de detener las amenazas detectadas.

La única manera por la que puede actuar para repelar la amenaza es trabajando en conjunto con un *firewall*, a partir de esto surgen dos tipos de IDS: los pasivos que no realizan ninguna acción y los reactivos que además de advertir dan instrucciones al *firewall* para que actúen contra la posible amenaza.

4.3.5.2. *Intrusion protection system (IPS)*

Muy similar a un IDS se trata de un sistema que además de la detección de intrusos es capaz de poder realizar alguna acción sin la dependencia de un *firewall*. Los IPS (*intrusion protection system*) son una extensión de los IDS, a su vez se consideran mejores que los *firewall* convencionales debido a su

capacidad de actuar ante posibles amenazas en función del análisis sobre el tráfico en la red.

4.3.6. Implementación de IDS

A continuación, se muestra la implementación de IDS.

4.3.6.1. Snort

Hoy en día existe una gran variedad de software que pueden ser utilizados como IDS, los hay gratuitos como pagados, Snort es un software *open source* que realiza esta función; básicamente, se encarga del análisis de tráfico en busca de alguna anomalía en el mismo; también, se realiza búsquedas basadas en firmas y políticas que puedan representar alguna amenaza para la red.

Snort realiza la detección de intrusos en tiempo real en línea con un sistema de detección de ataques con escaneo de puertos, analiza todo el tráfico con base a reglas específicas, a su vez cuenta con almacenamiento de bitácoras en texto plano e incluso en bases de datos de código abierto como MySQL.

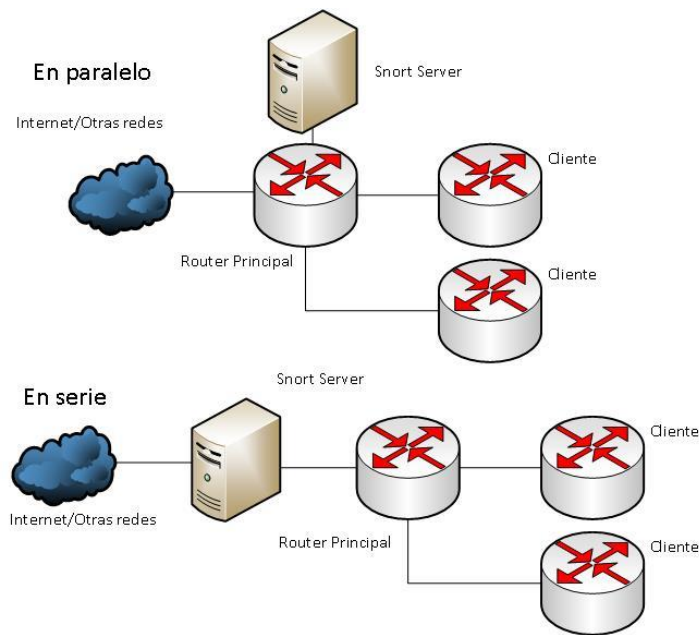
La descarga de este *software* es de forma gratuita; también, cuenta con actualizaciones constantes basadas en la seguridad gracias a la comunidad abierta de desarrolladores; la página oficial se encuentra a continuación en donde se pueden encontrar más especificaciones técnicas, guías específicas de instalación y descargas gratuitas.

- <https://www.snort.org>

4.3.6.2. Integración de snort con mikrotik

La integración de snort con un equipo mikrotik es relativamente sencilla; dependiendo de la configuración que se desea utilizar; este se puede utilizar de dos formas: en paralelo a la topología o en línea como se muestra a continuación.

Figura 41. Integración de snort en la red

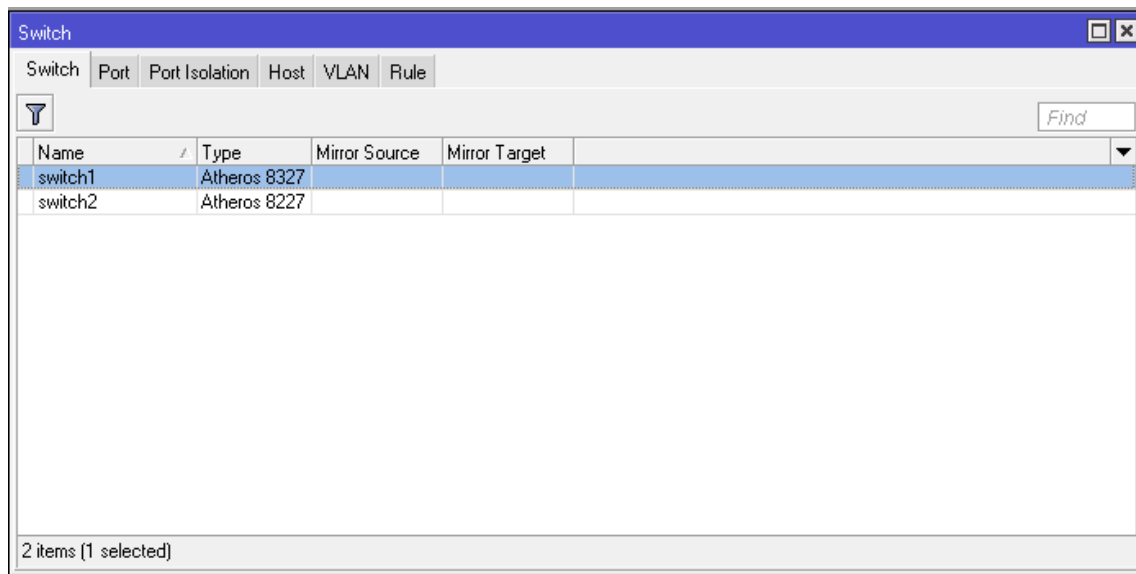


Fuente: elaboración propia, empleando Visio 2016.

Para los fines de esta investigación se integrará en paralelo a la red, dado que no se cuenta con una salida directa hacia internet u otro tipo de red, el servidor con el *software* IDS debería de estar conectado al *router* principal en el nodo de distribución.

La configuración en el *router* es extremadamente sencilla, simplemente se tiene que hacer un espejo de las interfaces que se desean monitorear hacia una en común (la que tendrá conectado el servidor IDS), para realizar este, se debe utilizar la sección *switch* del equipo como se muestra a continuación.

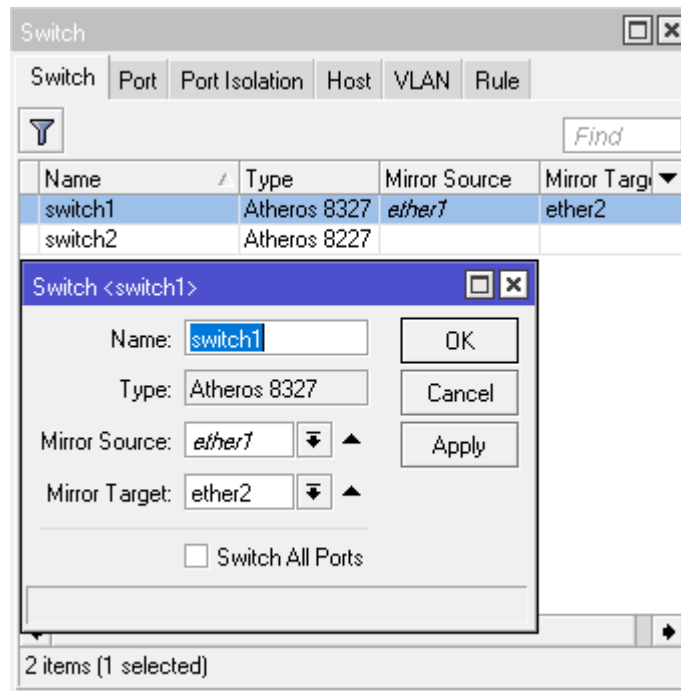
Figura 42. **Configuración Switch Mikrotik**



Fuente: elaboración propia.

El equipo utilizado en este ejemplo es un rb2011 UiAS, este cuenta con dos chips de *switch* Atheros 8327; cuando se ingresa a la configuración del chip este permite realizar un espejo entre puertos como se muestra en la siguiente ilustración.

Figura 43. Espejo entre puertos



Fuente: elaboración propia.

Dicha configuración se denomina como *port mirroring*, también conocida como duplicación de puertos, permite enviar una copia de los paquetes entrantes, así como los salientes de una interfaz (*mirror source*) y enviarlos desde otro puerto (*mirror target*).

El objetivo en este caso es que los paquetes recibidos a través de la interfaz de distribución en el nodo central sean duplicados hacia la interfaz en la cual se encuentra instalado el servidor IDS; es muy importante resaltar que para utilizar esta funcionalidad se recomienda que el equipo cuente con un chip de *switch*, con el objetivo de evitar sobre cargar el procesador del equipo, así mismo, considerar el chip de *switch* del equipo a utilizar, a pesar de que todos

los chips utilizados por Mikrotik cuentan con *port mirroring* algunos tienen mejor desempeño en comparación a otros, toda esta información puede ser consultada en la documentación oficial.

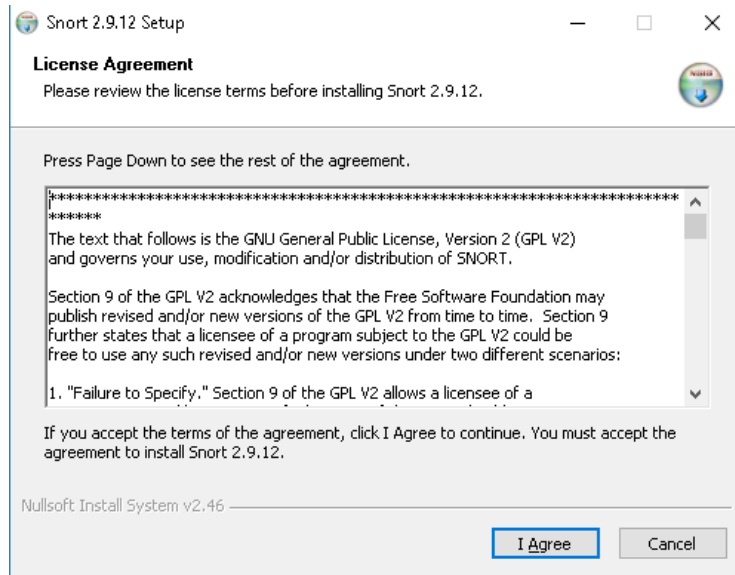
4.3.7. Instalación de Snort

La instalación de este *software* IDS puede ser realizada tanto en el sistema operativo Windows como en distribuciones de Linux; a continuación, se muestra la instalación en ambos.

4.3.7.1. Windows

Para realizar la instalación en este sistema operativo primero que nada se debe contar con el programa WinPcap instalado en el ordenador que se utilizará como servidor; en este caso se instaló la versión 4.1.3, seguidamente a esto es necesario descargar Snort desde la página oficial para instalarlo posteriormente. Es importante resaltar que todas las pruebas que a continuación se presentan fueron realizadas sobre la versión de Windows 10 Pro, actualizada en abril de 2019.

Figura 44. Instalación Snort en Windows



Fuente: elaboración propia.

Si la instalación se llevó a cabo correctamente se puede obtener un resultado como el que se muestra a continuación, donde se muestran las interfaces disponibles para realizar el escaneo, versión del *software*, entre otras características del mismo.

Figura 45. Primera vista Snort en Windows

```
c:\>cd c:\Snort\bin
c:\Snort\bin>snort.exe -W

  _*_
  o" )~
  ....

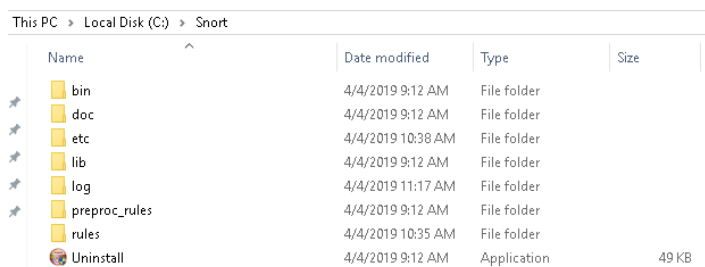
-*> Snort! <*-
Version 2.9.12-WIN32 GRE (Build 325)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2018 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using PCRE version: 8.38 2015-11-23
Using ZLIB version: 1.2.8

Index  Physical Address      IP Address      Device Name      Description
-----
1      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:1063:e769  \Device\NPF_{CAF72ED2-8487-4430-90A9-A7ABEA0F1B4
C}
Microsoft
2      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:8424:4e6e  \Device\NPF_{74ED7E27-91EB-44C8-9CB4-B1E6B1D82FF
8}
Microsoft
3      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:9860:ac17  \Device\NPF_{EBFFAE96-679F-4EE0-AD95-8A86F073BEA
8}
Realtek Ethernet Controller
4      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:31a8:08b9  \Device\NPF_{E224F6D9-C531-43C3-9B6D-01776F4EBAA
7}
VMware Virtual Ethernet Adapter
5      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:2918:c518  \Device\NPF_{8196CE45-531F-430D-A70D-15476178284
9}
VMware Virtual Ethernet Adapter
```

Fuente: elaboración propia.

Luego de terminar la instalación inicial, en nuestro directorio raíz aparecerá la carpeta que contiene todos los archivos, reglas y demás para el funcionamiento del *software*.

Figura 46. Directorio de Snort en Windows

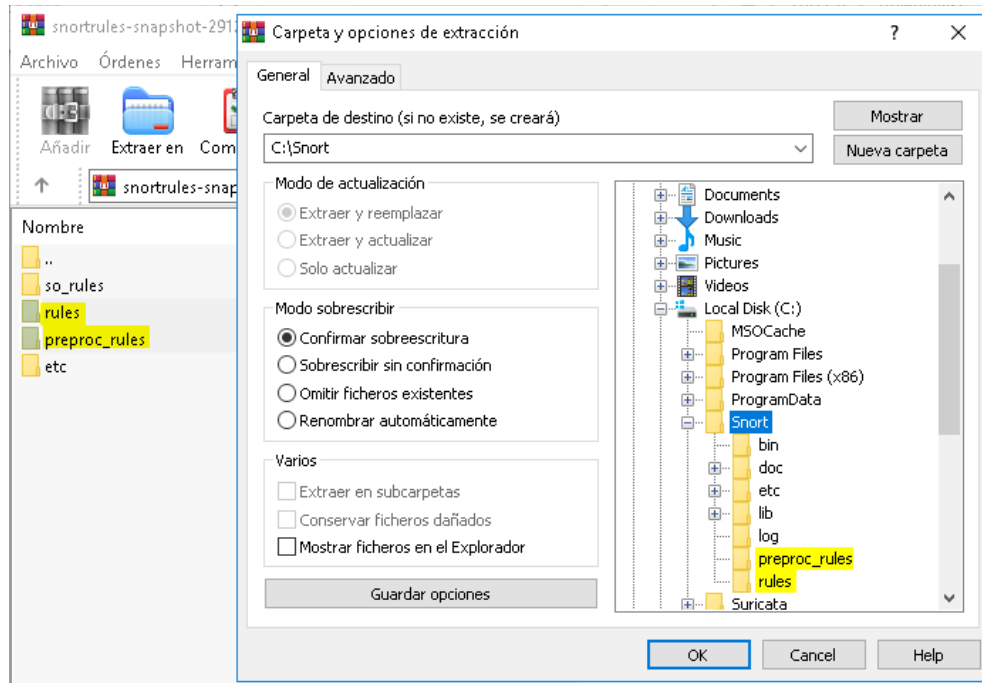


Fuente: elaboración propia.

Siguiendo con el proceso de instalación, es necesario registrarse en la página oficial de Snort para obtener las reglas necesarias en el funcionamiento

del IDS. Es importante tomar en cuenta que las reglas descargadas desde la página deben coincidir con la versión que se instaló del software, para este ejemplo se utilizará la versión 2.9.12. Luego de descargar las reglas, están deben ser descomprimidas en el directorio que fue creado en la instalación inicial como se muestra en la figura.

Figura 47. **Descomprimiendo las reglas en Windows**



Fuente: elaboración propia.

Una vez fue terminada la extracción de las reglas en el directorio correspondiente, es necesario editar el archivo de configuración de snort ubicado en la siguiente dirección.

- C:\Snort\etc\snort.conf

Con un editor de texto, primero hay que ubicarse en la línea 45 para editar la variable HOME_NET que contiene la red en la que se desea realizar el escaneo de paquetes; para este ejemplo se utilizó la red privada 192.168.0.0/24 quedando de la siguiente manera.

- ipvar HOME_NET 192.168.0.0/24

Luego unas líneas más abajo se edita la variable EXTERNAL_NET simplemente negando la variable anterior.

- ipvar EXTERNAL_NET !\$HOME_NET

Así mismo, es necesario reemplazar de la línea 104 a la 106 con los siguientes comandos respectivamente, especificando los directorios desde los cuales serán tomadas las reglas para el funcionamiento del software; es decir, las reglas que fueron descomprimidas en el directorio raíz de Snort.

- var RULE_PATH ../rules
- #var SO_RULE_PATH ../so_rules
- var PREPROC_RULE_PATH ../preproc_rules

También, es necesario definir el directorio en el que se encuentran los archivos *white* y *black list*, esto en la línea 113. Es importante mencionar que por defecto estos archivos no se encuentran creados, es parte del proceso de instalación crearlos para ubicarlos en el directorio correspondiente.

- `var WHITE_LIST_PATH C:\Snort\rules`
- `var BLACK_LIST_PATH C:\Snort\rules`

Es necesario especificar el directorio en el cual serán almacenados los *logs* resultado del análisis de los paquetes, en la línea 186.

- `config logdir: C:\Snort\log`

En los pasos finales se define el directorio en el cual se encuentra el motor que se utiliza para el análisis de paquetes en la línea 247 y 250; la línea 253 se comenta.

- `dynamicpreprocessor directory C:\Snort\lib\snort_dynamicpreprocessor`
- `dynamicengine C:\Snort\lib\snort_dynamicengine\sf_engine.dll`
- `# dynamicdetection directory /usr/local/lib/snort_dynamicrules`

Finalmente, se especifican las rutas absolutas de las reglas cambiando la diagonal por diagonal invertida, desde la línea 546 hasta la línea 651.

- `include $RULE_PATH\local.rules`

Para probar el resultado de la instalación se debe de utilizar el siguiente comando, el cual especifica la interfaz que será analizada y el directorio en el cual se encuentra ubicado el archivo de configuración que a su vez especifica las reglas, direcciones IP, motor de búsqueda, etc.

Figura 48. Comando de prueba en Windows

```
snort.exe -i #interface -c c:\Snort\etc\snort.conf -T
```

Fuente: *IDS snort detección de intrusos (Windows)*.

<https://www.youtube.com/watch?v=oBq8VrcHDms&t=1784s>. Consulta: 15 de abril de 2019.

Si el resultado de la instalación es satisfactorio, se obtiene un resultado parecido a este.

Figura 49. Ejecutando Snort sobre una interfaz en Windows

```
---= Initialization Complete =---
o''~
  .'.
  ....

-*> Snort! <*-
Version 2.9.12-WIN32 GRE (Build 325)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2018 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using PCRE version: 8.38 2015-11-23
Using ZLIB version: 1.2.8

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 3.0 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>

Snort successfully validated the configuration!
Snort exiting
```

Fuente: elaboración propia.

De no ser así, el mismo software especificará si es que existe algún error en el archivo de configuración de Snort. El resultado del análisis de paquetes, puede ser visto en la propia consola, a través del comando especificado a continuación.

Figura 50. **Comando de prueba con log en consola en Windows**

```
snort.exe -i #interface -c c:\Snort\etc\snort.conf -A console
```

Fuente: *IDS Snort detección de intrusos (Windows)*.

<https://www.youtube.com/watch?v=oBq8VrcHDms&t=1784s>. Consulta: 16 de abril de 2019.

4.3.7.2. Linux

La instalación en dicho sistema operativo es muy parecida a la anterior, en este caso se utilizó la distribución de Linux Lubuntu en su versión 19.04, antes que nada, se debe de llenar algunos pre requisitos para el funcionamiento de Snort, primero ejecutar los siguientes comandos desde la terminal del sistema operativo.

Figura 51. **Upgrade**

```
sudo apt-get update  
sudo apt-get dist-upgrade
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.

<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 16 de abril de 2019.

Reiniciar el equipo, posteriormente al reinicio realizar la ejecución de los comandos.

Figura 52. **Instalaciones previas en Linux**

```
sudo apt-get install build-essential
sudo apt-get install -y libpcap-dev libpcrc3-dev libdumbnet-dev
sudo apt-get install -y zlib1g-dev liblzma-dev openssl libssl-dev
sudo apt-get install bison flex
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 15 de abril de 2019.

Adicionalmente, es necesario instalar Daq para que Snort funcione correctamente es necesario instalarlo por medio de los siguientes comandos.

Figura 53. **Instalación Daq en Linux**

```
cd ~/snort
wget https://www.snort.org/downloads/snort/daq-2.0.6.tar.gz
tar -xvzf daq-2.0.6.tar.gz
cd daq-2.0.6
./configure
make
sudo make install
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 16 de abril de 2019.

Al finalizar la instalación de Daq, el ordenador se encuentra listo para la instalación de Snort, esto por medio de la ejecución de los comandos que a continuación se presentan.

Figura 54. **Instalación Snort en Linux**

```
cd ~/snort
wget https://www.snort.org/downloads/snort/snort-2.9.8.3.tar.gz
tar -xvzf snort-2.9.8.3.tar.gz
cd snort-2.9.8.3
./configure
make
sudo make install
sudo ldconfig
sudo ln -s /usr/local/bin/snort /usr/sbin/snort
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 16 de abril de 2019.

Para realizar una prueba, en el ejercicio de verificar el funcionamiento correcto de Snort se puede utilizar el comando Snort de la siguiente manera.

Figura 55. **Comando de prueba en Linux**

```
sudo snort -V
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 15 de abril de 2019.

Si el resultado de la instalación es correcto, se obtiene un resultado como el de la figura.

Figura 56. Prueba Snort en Linux

```
victor@victor-pc:~$ snort -V
o"  )~
' ' ' '

-*> Snort! <*-
Version 2.9.12 GRE (Build 325)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2018 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.8.1
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11
```

Fuente: elaboración propia.

En este punto el software se encuentra instalado, mas no configurado, para proceder con la configuración del mismo es necesario crear algunos directorios en donde estarán ubicados los *log*, reglas, errores, entre otros.

Figura 57. Creación de directorios en Linux

```
sudo groupadd snort
sudo useradd snort -r -s /sbin/nologin -c SNORT_IDS -g snort

sudo mkdir /etc/snort
sudo mkdir /etc/snort/rules
sudo mkdir /etc/snort/rules/iplists
sudo mkdir /etc/snort/preproc_rules
sudo mkdir /usr/local/lib/snort_dynamicrules
sudo mkdir /etc/snort/so_rules

sudo touch /etc/snort/rules/iplists/black_list.rules
sudo touch /etc/snort/rules/iplists/white_list.rules
sudo touch /etc/snort/rules/local.rules
sudo touch /etc/snort/sid-msg.map

sudo mkdir /var/log/snort
```

Continuación de la figura 57.

```
sudo mkdir /var/log/snort/archived_logs

sudo chmod -R 5775 /etc/snort
sudo chmod -R 5775 /var/log/snort
sudo chmod -R 5775 /var/log/snort/archived_logs
sudo chmod -R 5775 /etc/snort/so_rules
sudo chmod -R 5775 /usr/local/lib/snort_dynamicrules

sudo chown -R snort:snort /etc/snort
sudo chown -R snort:snort /var/log/snort
sudo chown -R snort:snort /usr/local/lib/snort_dynamicrules

cd ~/snort/snort-2.9.8.3/etc/
sudo cp *.conf* /etc/snort
sudo cp *.map /etc/snort
sudo cp *.dtd /etc/snort
cd ~/snort/snort-2.9.8.3/src/dynamic-
preprocessors/build/usr/local/lib/snort_dynamicpreprocessor/
sudo cp * /usr/local/lib/snort_dynamicpreprocessor/
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 16 de abril de 2019.

Luego de crear los directorios, muy parecido a la instalación en Windows se procede a editar algunas líneas del archivo snort.conf por medio de algunos comandos; primero se van a comentar todas las reglas del archivo, por medio de la siguiente instrucción en la cual se especifica comentar todas las reglas incluidas en el archivo de configuración.

Figura 58. **Comentando reglas del archivo snort.conf en Linux**

```
sudo sed -i "s/include \$RULE_PATH/#include \$RULE_PATH/" /etc/snort/snort.conf
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.
<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 17 de abril de 2019.

Luego por medio del editor de texto nano, se editan algunas líneas del archivo en cuestión; primero, al igual que en la instalación en Windows se editará la línea que especifica el segmento de red que será escaneado y que la variable IP_EXTERNAL sea la negación de la variable HOME_NET.

- ipvar HOME_NET 192.168.0.0/24
- ipvar EXTERNAL_NET !\$HOME_NET

Una vez configurado esto, a partir de la línea 104, verificar que los directorios que especifican las reglas, así como las listas, se encuentren de la siguiente manera.

- var RULE_PATH /etc/snort/rules
- var SO_RULE_PATH /etc/snort/so_rules
- var PREPROC_RULE_PATH /etc/snort/preproc_rules
- var WHITE_LIST_PATH /etc/snort/rules/iplists
- var BLACK_LIST_PATH /etc/snort/rules/iplists

Finalmente, colocar la ruta absoluta de las reglas en el archivo de configuración, haciéndolas ver de la siguiente manera.

- \$RULE_PATH/local.rules

Para probar la configuración realizada, se procede a utilizar el siguiente comando, lógicamente especificando el identificador de la interfaz que se desea monitorear.

Figura 59. **Comando de prueba sobre una interface en Linux**

```
sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -I #interface
```

Fuente: USMAN, Nasir. *How to install snort intrusion detection system on Ubuntu*.

<https://cyberpersons.com/2016/07/18/install-snort-ubuntu/>. Consulta: 17 de abril de 2019.

Obteniendo un resultado como el que se puede ver a continuación, en este caso monitoreando la interfaz física del servidor configurado con Snort.

Figura 60. **Ejecutando Snort sobre una interfaz en Linux**

```
victor@victor-pc:~$ sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf -i wlp1s0
04/06-00:36:32.056301  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::8e61:a3ff:fe74:6d04 -> ff02::1
04/06-00:36:32.056636  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::8e61:a3ff:fe74:6d04 -> ff02::1
04/06-00:36:32.056892  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::8e61:a3ff:fe74:6d04 -> ff02::1
04/06-00:36:32.057114  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::8e61:a3ff:fe74:6d04 -> ff02::1
04/06-00:36:32.072436  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::3bd:f882:1335:5cf2 -> ff02::16
04/06-00:36:32.372431  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {TCP} 2803:d100:8800:1aa2:995b:449:d6f6:2568:36432 -> 2607:f8b0:4008:810::2003:80
04/06-00:36:32.415296  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {TCP} 2607:f8b0:4008:810::2003:80 -> 2803:d100:8800:1aa2:995b:449:d6f6:2568:36432
04/06-00:36:32.884465  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {IPV6-ICMP} fe80::3bd:f882:1335:5cf2 -> ff02::16
04/06-00:36:33.140442  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {TCP} 2803:d100:8800:1aa2:995b:449:d6f6:2568:57442 -> 2606:2800:220:de:468:2285:c1:4a3:443
04/06-00:36:33.175759  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
  {TCP} 2606:2800:220:de:468:2285:c1:4a3:443 -> 2803:d100:8800:1aa2:995b:449:d6f6:2568:57442
04/06-00:36:33.396457  [**] [1:354234:0] ATTACK RESPONSES id check returned root [**] [Priority: 0]
```

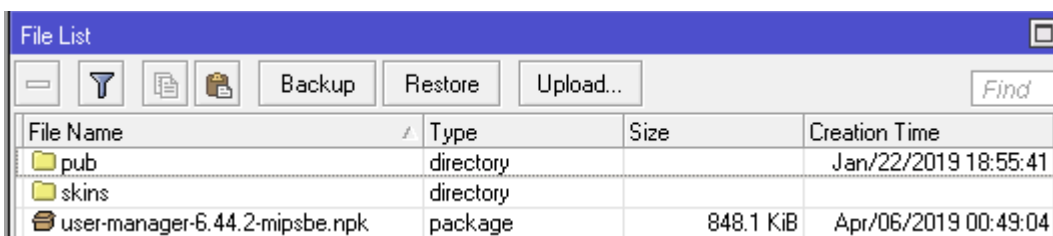
Fuente: elaboración propia.

4.3.8. Radius

La implementación de un servidor *radius* también podría ayudar en cuanto a seguridad se refiere, este se encarga de administrar los usuarios que tienen permitido el ingreso a los equipos, usuarios *hotspot*, usuarios PPP, entre otros. Para el enfoque de este estudio se utilizará como servidor para el control de ingreso de usuarios en el equipo, el uso de esta funcionalidad requiere de una serie de instalaciones adicionales en el equipo que será utilizado como servidor; primero es necesario descargar el archivo denominado *all packages* desde la página oficial de Mikrotik, es importante tomar en cuenta que estos paquetes no vienen incluidos dentro de las actualizaciones periódicas del equipo.

Una vez descargado dentro del archivo comprimido se debe de ubicar el paquete *user-manager*; luego, se debe de cargar al equipo y posteriormente reiniciarlo.

Figura 61. Instalación User Manager Mikrotik



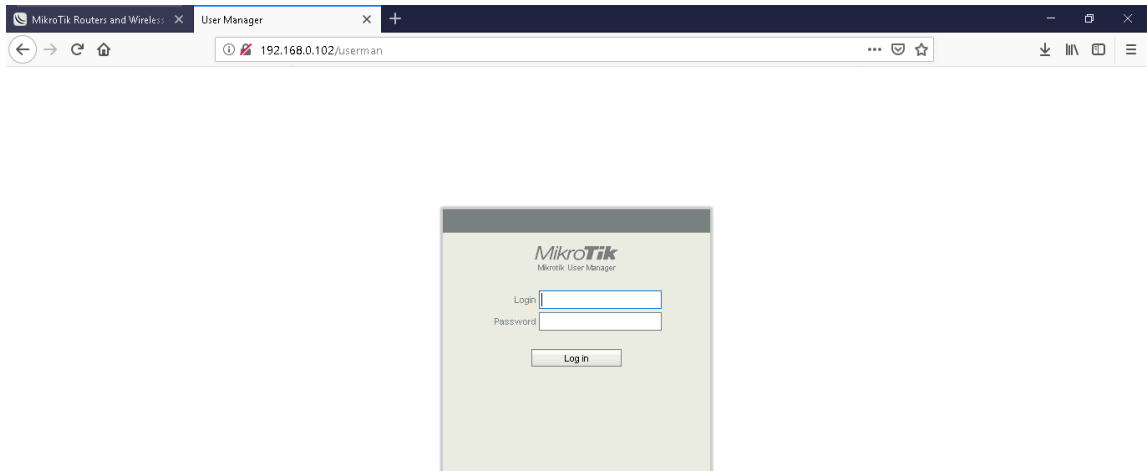
The screenshot shows a 'File List' window with a blue title bar. Below the title bar is a toolbar with icons for back, forward, and search, along with buttons for 'Backup', 'Restore', and 'Upload...'. A 'Find' text box is on the right. The main area is a table with columns for 'File Name', 'Type', 'Size', and 'Creation Time'. The table contains three entries: a 'pub' directory created on Jan/22/2019 at 18:55:41, a 'skins' directory, and a 'user-manager-6.44.2-mipsbe.npk' package of 848.1 KiB created on Apr/06/2019 at 00:49:04.

File Name	Type	Size	Creation Time
pub	directory		Jan/22/2019 18:55:41
skins	directory		
user-manager-6.44.2-mipsbe.npk	package	848.1 KiB	Apr/06/2019 00:49:04

Fuente: elaboración propia.

Luego del reinicio, deberían de aparecer las carpetas correspondientes al paquete recién instalado en la sección *Files*, para prueba de esto se debe entrar a gestionar los usuarios a través de la interface web del equipo.

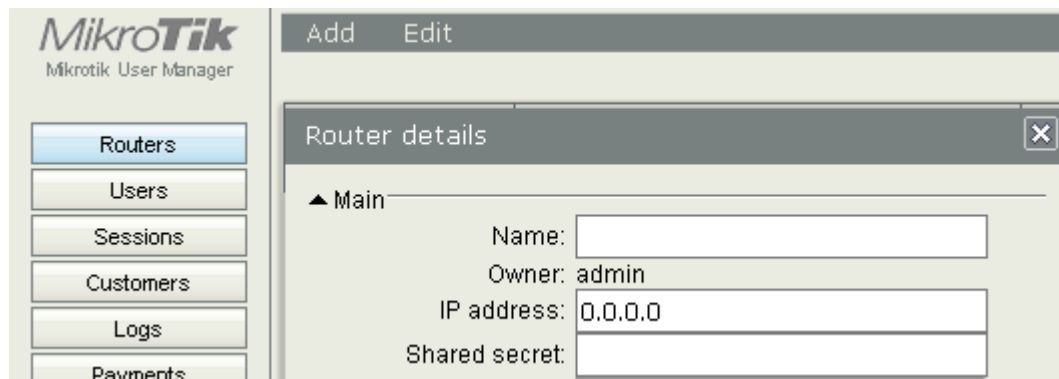
Figura 62. **Gestión de usuarios interfaz web**



Fuente: elaboración propia.

El usuario por defecto para ingresar a la plataforma es *admin*, sin ninguna contraseña; antes que nada, se debe agregar en la sección *routers* la dirección IP de conexión y la contraseña correspondiente para establecer la conexión con los otros *routers* en la red.

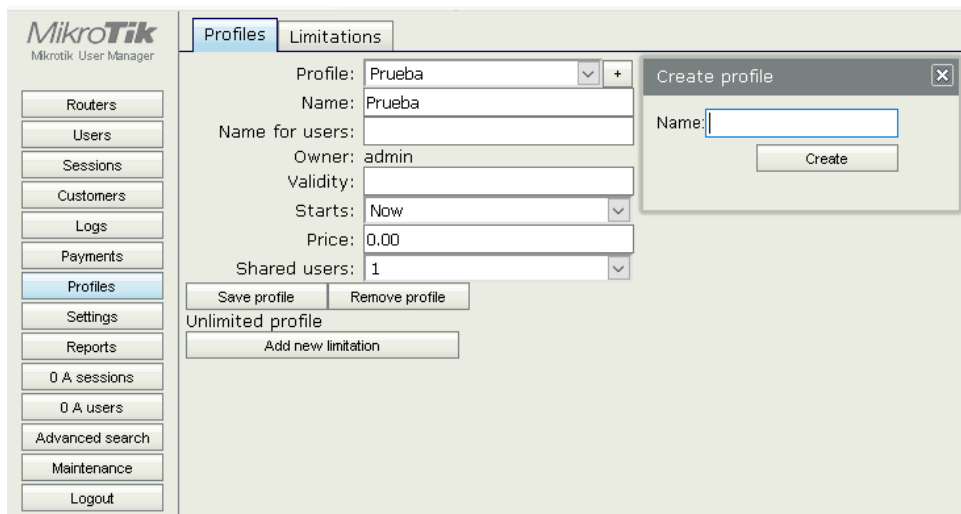
Figura 63. **Agregando IP y contraseña de conexión**



Fuente: elaboración propia.

Para manejar los usuarios se debe crear un perfil especificando los usuarios permitidos por cuenta.

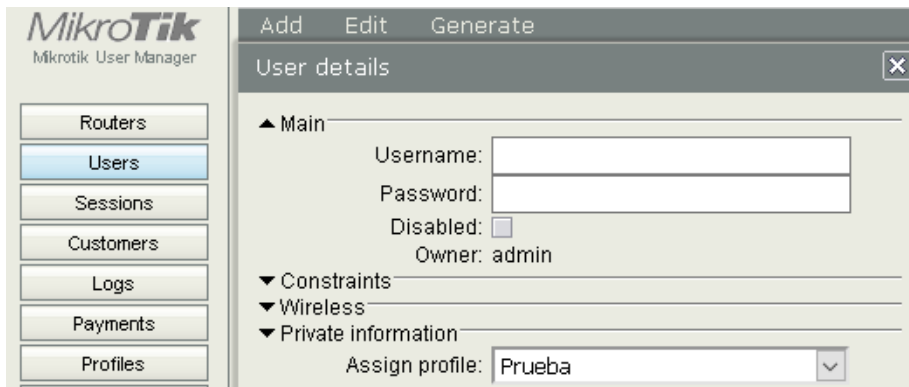
Figura 64. **Perfiles de usuarios**



Fuente: elaboración propia.

Una vez fue creado un perfil, se procede a crear los usuarios, como se muestra a continuación.

Figura 65. **Agregando usuarios**



The screenshot shows the Mikrotik User Manager web interface. On the left is a navigation menu with buttons for 'Routers', 'Users', 'Sessions', 'Customers', 'Logs', 'Payments', and 'Profiles'. The 'Users' button is highlighted. The main area is titled 'User details' and contains a form with the following fields: 'Username' (text input), 'Password' (text input), 'Disabled' (checkbox), 'Owner' (text input with value 'admin'), 'Constraints' (dropdown), 'Wireless' (dropdown), 'Private information' (dropdown), and 'Assign profile' (dropdown with value 'Prueba'). At the top of the form are buttons for 'Add', 'Edit', and 'Generate'.

Fuente: elaboración propia.

Una vez fueron creados los usuarios necesarios, en todos los equipos de la red se debe de agregar un nuevo servidor *radius*, especificando que se quiere gestionar solamente la opción *login* del equipo, para completar la configuración, se debe activar la opción *use radius* en el menú *system/users*; a continuación, la configuración para un equipo que apunta a un servidor con la dirección IP 192.168.0.102/24.

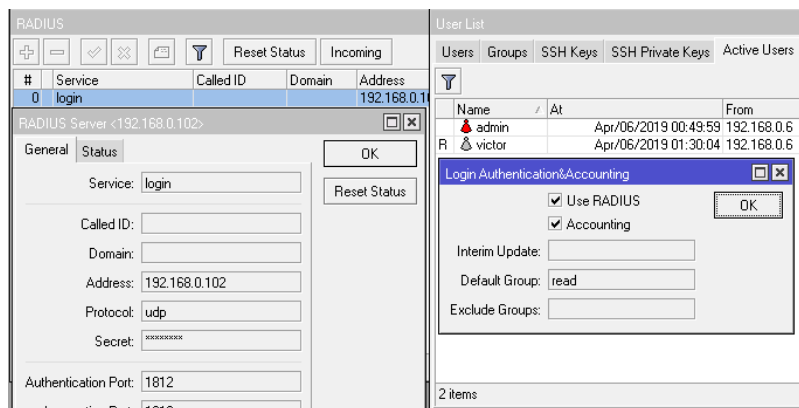
Figura 66. **Configuración CPE**

```
/radius
add address=192.168.0.102 secret=12345678 service=login
/user
add comment="system default user" group=full name=admin
/user aaa
set use-radius=yes
```

Fuente: elaboración propia.

Si la configuración es exitosa, todos los usuarios que fueron creados a través de la interface web del *user manager* deben ingresar a todos los *routers* que apuntan al servidor radius.

Figura 67. Prueba radius

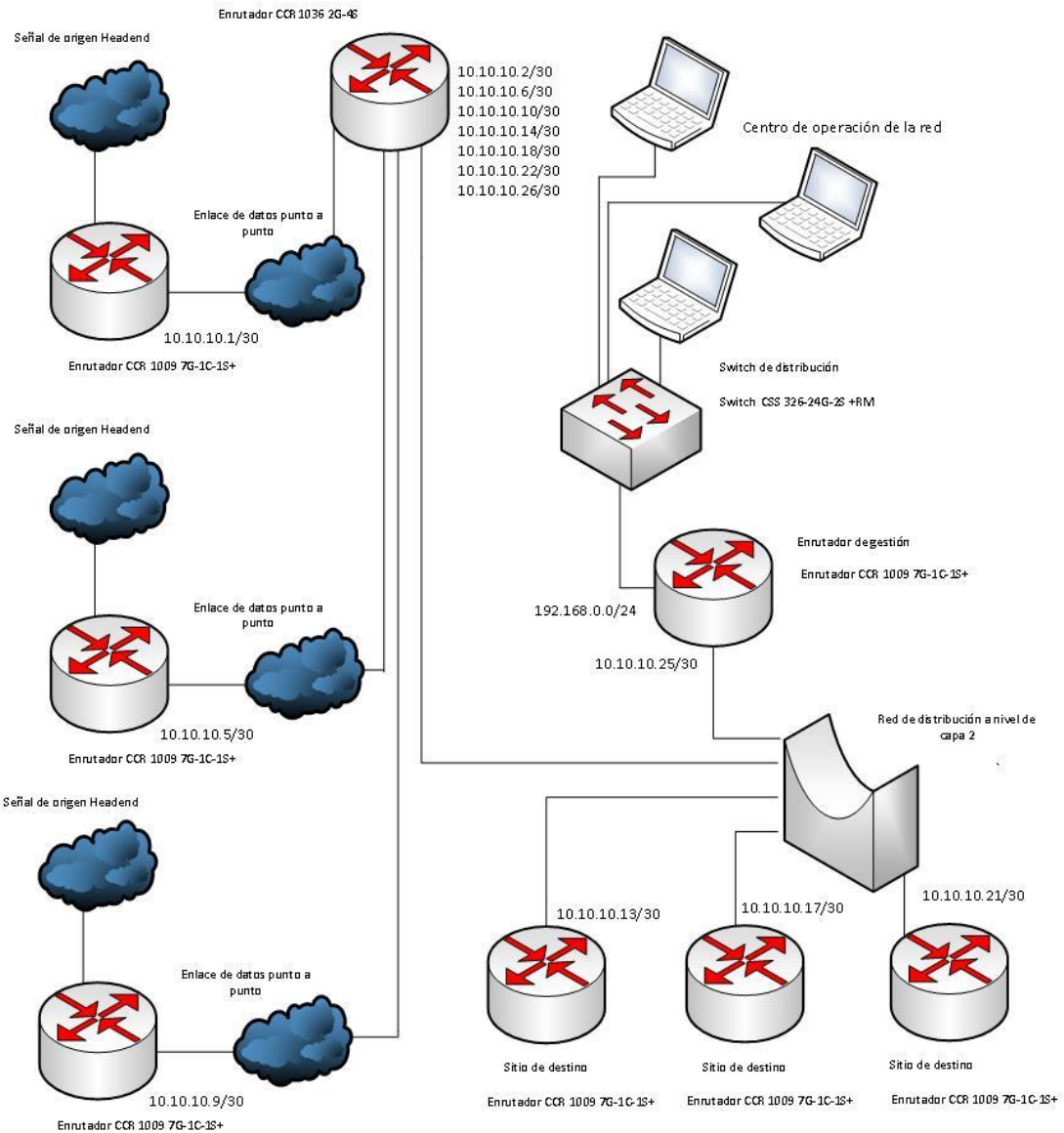


Fuente: elaboración propia.

4.4. Diseño final de la red

A continuación, se presenta en la figura 68, incluyendo el direccionamiento, así como especificaciones de equipos de la red.

Figura 68. Diseño final de red



Fuente: elaboración propia, empleando Visio 2016.

Dentro del diseño de red ya se encuentra incorporada una red de gestión sobre la de distribución, también se pueden incorporar múltiples señales de origen luego de haber sido explicados los filtros aplicables en el *firewall* de

RouterOS, lo cual brinda como resultado final un diagrama red como el de la figura anterior.

4.5. Aspectos y configuraciones finales

Debido a los cambios incorporados dentro de la red, la configuración de los equipos debe de ser levemente mejorada, es importante mencionar que los cambios están enfocados a la integración de la red de gestión y los *headends* adicionales para ser utilizados de respaldo.

La configuración de los *headends* no varía significativamente con respecto a la que fue presentada en el capítulo 3, como se observa a continuación.

Figura 69. **Configuración final *headend 1***

```
/system identity
set name=Headend1
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast" interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.1/30 comment="IP de gestion" interface=bridge1 network=\10.10.10.0
/ip route
add comment="Ruta por defecto hacia nodo de distribucion" distance=1 gateway=10.10.10.2
```

Fuente: elaboración propia.

Figura 70. **Configuración final *headend 2***

```
/system identity
set name=Headend2
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast"
interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.5/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.4
/ip route
add comment="Ruta por defecto hacia nodo de distribucion" distance=1
gateway=10.10.10.6
```

Fuente: elaboración propia.

Figura 71. **Configuración final *headend 3***

```
/system identity
set name=Headend3
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast" interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.9/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.8
/ip route
add comment="Ruta por defecto hacia nodo de distribucion" distance=1
gateway=10.10.10.10
```

Fuente: elaboración propia.

La configuración para cada *headend* es muy parecida, la única variación entre cada punto es la dirección de gestión y la ruta por defecto, en cada punto pueden ser agregados filtros para el manejo del tráfico, pero en este caso dichos filtros serán ocupados en la fase del nodo de distribución, dado que se desea centralizar el control del flujo de datos en una sola ubicación.

Figura 72. **Configuración final nodo de distribución**

```

/system identity
set name="Nodo de distribucion router"
/interface bridge
add comment="Puente de distribucion y gestion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Entrada Multicast " interface=ether1
add bridge=bridge1 comment="Entrada Multicast " interface=ether2
add bridge=bridge1 comment="Entrada Multicast " interface=ether3
add bridge=bridge1 comment="Interface de distribution" interface=ether4
add bridge=bridge1 comment="Interface de gestion" interface=ether5
add bridge=bridge1 comment="Interface de servidor IDS" interface=ether6
/interface ethernet switch
set 0 mirror-source=ether4 mirror-target=ether6
/ip address
add address=10.10.10.2/30 comment="Conexion punto a punto HE1" interface=bridge1
network=\10.10.10.0
add address=10.10.10.6/30 comment=" Conexion punto a punto HE2" interface=bridge1
network=\10.10.10.4
add address=10.10.10.10/30 comment=" Conexion punto a punto HE3" interface=bridge1
network=\10.10.10.8
add address=10.10.10.14/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.12
add address=10.10.10.18/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.16
add address=10.10.10.22/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.20
add address=10.10.10.26/30 comment="Conexion punto a punto red de gestion"
interface=bridge1 network=\10.10.10.0
/ip firewall filter
add action=drop chain=forward dst-address-list=Bloquear in-interface=ether2 src-address-
list=Bloquear
/ip firewall filter
add action=drop chain=forward dst-address-list=Bloquear in-interface=ether3 src-address-
list=Bloquear

```

Fuente: elaboración propia.

Para este proyecto se tomaron en cuenta tres fuentes de datos digitales: una red de gestión y tres clientes, como es descrito en la figura 68; todo esto es integrado en el enrutador de distribución, se creó un puente tomando en cuenta los enlaces punto a punto de las cabeceras, la interface de gestión, así como la de distribución. Sobre esta misma interfaz (bridge) se agregaron las direcciones de gestión de los equipos remotos, clientes y origen; para cada conexión se creó una red punto a punto utilizando segmentos de red 10.10.10.X/30. Finalmente, se aplicaron reglas en el *firewall* sobre las interfaces eléctricas dos y tres, esto con el objetivo de utilizar estos como redundancias pasivas, activándolas solo en casos emergentes.

Figura 73. **Configuración final cliente**

```
/system identity
set name="Cliente final"
/interface bridge
add comment="Puente de distribucion" name=bridge1
/interface bridge port
add bridge=bridge1 comment="Interface de entrada Multicast"
interface=ether1
add bridge=bridge1 comment="Interface de distribucion" interface=ether2
add bridge=bridge1 comment="Interface de distribucion" interface=ether3
add bridge=bridge1 comment="Interface de distribucion" interface=ether4
add bridge=bridge1 comment="Interface de distribucion" interface=ether5
/ip address
add address=10.10.10.13/30 comment="IP de gestion" interface=bridge1
network=\10.10.10.12
```

Fuente: elaboración propia.

Por su lado, la configuración del cliente final es sumamente sencilla, solamente se creó un puente entre la interface de recepción y distribución del flujo *multicast*; lógicamente, se creó una dirección de gestión para el equipo, en caso de fallas puntuales.

Figura 74. **Red de gestión**

```
/system identity
set name="Red de gestion"
/ip address
add address=10.10.10.25/30 comment="IP de gestion" interface=ether1
network=\10.10.10.24
add address=192.168.0.1/24 comment="Red de gestion" interface=ether2
network=\192.168.0.0
/ip route
add comment="Ruta por defecto hacia nodo de distribucion" distance=1
gateway=10.10.10.26
/ip pool add name=dhcp-pool ranges=192.168.0.10-192.168.0.254
/ip dhcp-server network add address=192.168.0.0/12 gateway=192.168.0.1
/ip dhcp-server add interface=ether2 address-pool=dhcp-pool
```

Fuente: elaboración propia.

La red de gestión cuenta con un segmento de red distinto a los anteriores, este segmento de red cuenta con un servidor DHCP, esto para dar direccionamiento a los equipos que van a monitorear la red. También, cuenta con una IP de conexión punto a punto y una ruta estática dirigida hacia el nodo de distribución; la implementación de este equipo es debido a la necesidad de un centro de operación de la red (NOC).

4.6. Ventajas y desventajas de la implementación final de red

Finalmente, realizando un análisis técnico sobre la implementación de red se concluye que cuenta con un manejo de gestión centralizado, implementando un NOC para monitorear la red; así mismo, se puede habilitar una cabecera de

respaldo e incluso una segunda cabecera de respaldo lo cual representa un punto muy importante a favor del diseño de la red.

Desde el centro de operaciones de la red se puede monitorear el tráfico en las interfaces, monitorear los equipos remotos a través de un sistema dedicado. En el mismo orden de las ideas es posible transportar el tráfico *multicast* a través de la red de gestión; de esta manera, se pueden monitorear los canales digitales que se están distribuyendo en la red utilizando *software* cliente para visualizarlos y descartar problemas de origen.

También, es posible realizar una combinación de los canales digitales, tomando canales de diferentes orígenes para unificarlos en la señal digital que se transporta hacia los clientes, siendo esto posible por medio de la aplicación de filtros de capa 3 en el *firewall* de los equipos terminales de los clientes o bien en el nodo principal. Siendo en conclusión una red sumamente flexible, sencilla, sin mayores complejidades en cuanto a diseño o configuración; por otro lado, cuenta con la desventaja principal que la conmutación de las cabeceras debe de ser de forma manual habilitando y deshabilitando las reglas en el *firewall* que manejan el flujo del tráfico.

5. ANÁLISIS TÉCNICO FINANCIERA DE IMPLEMENTACIÓN

Posteriormente a todos los argumentos técnicos presentados en capítulos anteriores, se realizará un análisis financiero sobre dicho proyecto, que resalta los factores por los cuales podría mejorar la rentabilidad del cable operador guatemalteco.

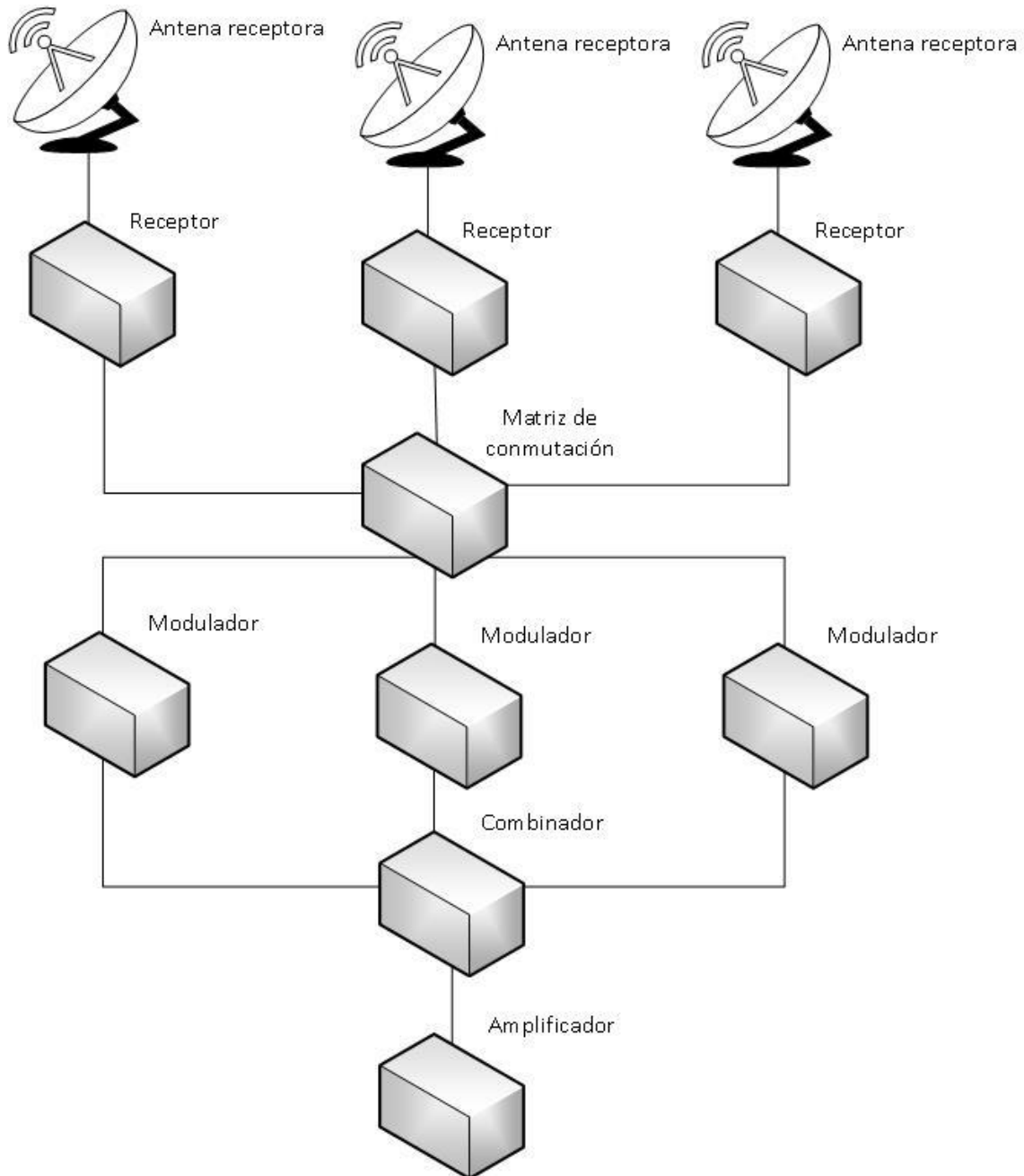
5.1. *Headend*

Las cabeceras realizan una función importante, si no es que la más fundamental sobre la estructura de red, este se encuentra destino a aterrizar la señal satelital y posteriormente redistribuirla hacia los diferentes destinos. Dado que se utiliza una gran cantidad de equipos dedicados, los costos de operación son bastante elevados, dentro de los dispositivos a utilizar se incluyen.

- Antenas
- Receptores
- Codificadores
- Moduladores
- Combinadores
- Amplificadores

Sin una red de transporte se tendría que tener un *headend* por cada punto de distribución de la señal de cable, lo cual vuelve dicho proyecto demasiado costoso. En el orden de las mismas ideas, se presenta a continuación una ilustración sobre la estructuración básica, así como los procesos por los cuales pasa la señal para una posterior distribución.

Figura 75. **Estructura headend**



Fuente: elaboración propia, empleando Visio 2016.

Los costos aproximados de cada dispositivo, se presentan en la tabla a continuación; dichos datos fueron tomados con base en las ventas en línea, específicamente del popular sitio de compras en internet Ebay, utilizando una tasa de cambio de 7.69 según el banco de Guatemala en la fecha 15 de marzo de 2019 por otro lado en los anexos es posible encontrar el precio de los equipos en dólares.

Tabla VIII. **Costos equipos *headend***

Dispositivo	Costo
Receptor satelital Cisco d9858	Q.7 520,00
Matriz de conmutación Roland XS-84H	Q.62 000,00
Modulador PCM55SAW	Q.1 000,00
Combinador PHC12G	Q.1 000,00
Transmisor óptico EDFA	Q.20 000,00
Encoder Blonder Tongue	Q.34 500,00
Enrutador CCR 1009 7G-1C-1S+	Q.4 000,00
Total	Q.130 020,00

Fuente: elaboración propia.

Derivado de esto es evidente que aterrizar la señal satelital en todos los puntos de distribución de los canales no es viable; si a esto se suman los costos de operación, mantenimiento de dispositivos y personal se vuelve un tanto costoso poder sustentar múltiples *headends*. Es por esto que se propone la creación de una red de transporte, reduciendo así los costos de operación dado que la cantidad de *headends* se reduce.

Es importante resaltar que el debido mantenimiento a esta parte del proyecto es de suma importancia, por lo cual es necesario que la operación y mantenimiento del *headend* deben ser tomados en cuenta dentro del

presupuesto de implementación para brindar de esta manera un servicio de calidad.

5.2. Nodo central

La implementación del nodo central de distribución requiere básicamente de equipos de red como *switch*, *router*, entre otros. Como fue presentado en la figura 68 del capítulo anterior, se necesita de dos enrutadores y un *switch* para la topología; a continuación, se muestra la tabla con los costos de aproximados de los equipos tomados del popular sitio de ventas en línea Ebay utilizando una tasa de cambio de 7,69 según el banco de Guatemala en la fecha 15 de marzo de 2019 por otro lado en los anexos es posible encontrar el precio de los equipos en dólares.

Tabla IX. **Costos equipos nodo de distribución**

Equipo	Costo
Enrutador CCR1036 2G-4S	Q.7 700,00
Enrutador CCR 1009 7G-1C-1S+	Q.4 000,00
Switch CSS 326-24G-2S +RM	Q.1 100,00
Total	Q.12 800,00

Fuente: elaboración propia.

Adicionalmente a esto habría que considerar equipos como computadoras, monitores, así como el personal a cargo de dar soporte y mantenimiento a la red, dado que el objetivo de este proyecto es evaluar la implementación de la red de transporte esto no será tomado en cuenta dentro del presupuesto. También, es muy importante mencionar que la estructura de red está pensada para tener el control absoluto de toda la red de transporte a través del centro de

operación de la red NOC; por otro lado, toda la red está pensada para trabajar con equipos Mikrotik dado que vuelven el proyecto mucho más rentable.

En el mismo orden de las ideas anteriores, es importante mencionar la flexibilidad de la red, a pesar de ser diseñada con equipos Mikrotik en ella pueden ser introducidos diferentes equipos que fácilmente podrían cumplir con la misma función. Los equipos recomendados también son un factor importante, dado que la configuración de los mismos no requiere de mayores complicaciones, es aquí donde la flexibilidad de configuración suma un valor agregado al proyecto.

Por otro lado, la red de gestión puede tener acceso a toda la red como fue mencionado anteriormente, de manera que se recomienda encarecidamente aplicar políticas de restricción dentro de la red dado que el valor de los datos transportados puede ser muy alto, derivado de esto cada uno de los usuarios debería de contar con los permisos respectivos en función de su cargo para poder evitar inconvenientes dentro de la red.

5.3. Transporte de datos

El transporte es muy importante para la gestión del proyecto, podría considerarse alquilar el transporte hacia el sitio de destino o bien utilizar la infraestructura propia, si es que se tuviese alguna. Se recomienda imprescindiblemente que el medio de transporte sea por medio de fibra óptica, dado que es un medio en el cual se presenta una menor cantidad de pérdidas y latencia, de ser posible tener enlaces redundantes para un servicio sin interrupciones.

De la mano del tema de transporte se encuentra íntimamente ligado el factor ancho de banda, para esta investigación se tomará en cuenta que se alquilará el transporte hasta el sitio final. El ancho de banda que se considerara como mínimo para la implementación del proyecto es de 1 Gbps tanto en los enlaces desde el nodo central hasta los clientes finales, así como los enlaces punto a punto desde el *headend* hacia el nodo de distribución dado que se está considerando transportar al menos 180 canales de televisión. El precio por el transporte puede ser variable dependiendo del proveedor de servicios, para objetivo de simplicidad se tomó un estándar del costo de 1 Mbps en el mercado actual, según artículos escritos por el del Diario de Centro América y la BBC en diciembre de 2018, afirman que el precio por una conexión de banda ancha (es decir, al menos 1 Mbps) ronda los \$ 10.

Tabla X. **Costo de transporte**

Ancho de banda	Precio
1 024 Mbps (dedicados)	Q.78 000,00

Fuente: elaboración propia.

5.4. Cliente final

La entrega del servicio con en el punto terminal también tiene un costo, basándonos en la topología de la figura 68, desglosados en la tabla que a continuación se presenta.

Tabla XI. **Costo entrega de servicio**

Equipo	Costo
Enrutador CCR 1009 7G-1C-1S+	Q.4 000,00

Fuente: elaboración propia.

Dado que la configuración del cliente es la sencilla de todas, solo se quiere de un enrutador para poder aterrizar la señal para su posterior distribución.

5.5. Análisis financiero

Ahora bien, ya que fueron presentados los costos de equipos necesarios para la implementación y el transporte de datos se presenta a continuación un análisis financiero del proyecto, con el objetivo de dar a conocer si unificar un listado de canales es de provecho para una empresa, más allá del punto de vista técnico.

5.5.1. Rentabilidad

La rentabilidad de un producto o empresa se define como el provecho promedio respecto a la inversión monetaria realizada, es decir un indicador de la ganancia inmediata de la entidad en cuestión respecto a todos los gastos asegurados. El cálculo de este valor nos puede dar un indicio, mas no la seguridad de que un negocio puede prosperar, dicho valor se representa en porcentaje y puede variar desde 0 % hasta un 100 %; sin embargo, también puede ser un número negativo lo cual es un fuerte indicio de pérdidas monetarias o dicho en otras palabras un negocio no rentable, si esto pasara es

necesario encontrar la manera de poder subir el precio de venta de nuestro producto o bien reducir los costos para que nuestra rentabilidad suba.

Para el cálculo de este valor se utiliza la fórmula presentada a continuación, también se explican el significado de los valores y su interpretación.

Figura 76. **Fórmula rentabilidad**

$$R = \frac{P - C}{P} * 100$$

Fuente: *Fórmula para calcular la rentabilidad de un producto.*

<http://www.elgrannegocio.com/formula-para-calculer-la-rentabilidad-de-un-producto/>. Consulta:

18 de marzo de 2019.

Obtener una rentabilidad de 100 % no es posible, al menos no en el mundo de los negocios convencionales, se puede observar a partir de la fórmula que la única manera de obtener un dato así es que el costo de un producto sea cero, lo cual es imposible ya que cualquier producto tiene un costo. Ahora bien, siguiendo la fórmula presentada en la figura anterior y considerando los datos de la tabla VIII, tenemos un costo total aproximado por cada *headend* de Q.130 020,00 representando con esta cifra únicamente los equipos.

Sin una red de transporte sería necesario colocar uno en cada punto de distribución, para caso de ejemplo se asumirá que se tienen 6 puntos, por lo cual el costo total sería de Q.780 120,00, una cifra bastante elevada; si a eso le suma gastos de operación y mantenimiento, es decir, el sueldo de algunos

empleados, gastos de energía eléctrica, reparaciones, distribución e instalaciones los números se elevan aún más pudiendo llegar incluso hasta los Q.900 000,00.

Para determinar una venta mínima del producto despejamos de la fórmula presentada en la figura anterior, la cual queda de la siguiente manera.

Figura 77. **Fórmula precio de venta**

$$P = \frac{100 C}{100 - R}$$

Fuente: elaboración propia.

Derivado de lo anterior es evidente que para tener una rentabilidad de por lo menos 1 %, la venta total del producto tendría que ser de al menos Q.909 090,90; a continuación, se presenta el desglose de los datos.

Tabla XII. **Rentabilidades múltiples *headends***

Descripción	Cantidad
Costo por <i>headend</i>	Q.130 020,00
Costo total <i>headends</i> (6)	Q.780 120,00
Costo de operación y mantenimiento	Q.119 880,00
Costos totales	Q.900 000,00
Rentabilidad mínima (1%)	Q.909 090,90

Fuente: elaboración propia.

Por otro lado, si se realiza el mismo análisis con una menor cantidad de *headends*, tomando de ejemplo la topología final presentada en la figura 68 del

capítulo anterior, con tres *headends* los costos se reducen drásticamente, adicionalmente a esto se toma en cuenta el costo de los enlaces de datos hacia los clientes finales, así como los costos del nodo central de distribución y el costo de operación y mantenimiento de las cabeceras, nodo central e incluso el centro de operaciones de la red; el desglose de dichos datos se presenta a continuación en la tabla, asumiendo que los costos de operación se mantienen.

Tabla XIII. **Rentabilidad topología final**

Descripción	Cantidad
Costo por Headend	Q.130 020,00
Costo de operación y mantenimiento	Q.119 880,00
Costo nodo central	Q.12 800,00
Costo enlaces dedicados (6)	Q.78 000,00
Costo equipos cliente final (5)	Q.4 000,00
Costos totales	Q.750 700,00
Rentabilidad mínima (1 %)	Q.758 282,82

Fuente: elaboración propia.

Realizando una comparativa sobre los datos contenidos en las tablas anteriores, es indiscutible que es más barato transportar el tráfico desde centros de datos dedicados y también es más sencillo obtener una rentabilidad mayor debido a que se puede dar cobertura a la misma cantidad de puntos (6) con un costo menor. Esto representa una mayor posibilidad de crecimiento para el empresario guatemalteco, así como una posición firme ante la competencia de los proveedores de servicios multinacionales que se encuentran en Guatemala y la posibilidad de poder generar más oportunidades de empleo.

5.5.2. Solvencia

Como bien se sabe este término representa básicamente la capacidad económica de una entidad o empresa, está dada a partir de la relación entre el activo y el pasivo de la entidad en cuestión.

Para comprender un poco más a fondo es necesario explicar los conceptos de activos y pasivos, estos son presentados a continuación.

5.5.2.1. Activo

Se conoce como un activo todos los bienes que pueden llegar a ser convertidos en efectivo en algún momento; es decir, productos, acciones, mobiliarios y cualquier cosa de la cual se dueña la institución, como los ejemplos enlistados.

- Vehículos
- Maquinaria
- Computadoras
- Derechos de patentes
- Equipos

5.5.2.2. Pasivo

Este valor se representa por todos los gastos y obligaciones de una empresa, quiere decir que, en él se ven reflejados los pagos a proveedores, pagos de préstamos e incluso deudas con empleados. A continuación, algunos ejemplos más específicos de lo que representa un pasivo.

- Remuneraciones pendientes
- Proveedores
- Acreedores

Ahora bien, ya que fueron explicados ambos conceptos, en la siguiente figura podemos apreciar la fórmula para poder obtener la solvencia de un negocio.

Figura 78. **Ecuación de solvencia**

$$S = \frac{A}{P}$$

Fuente: *Ratio de solvencia: definición, fórmula y ejemplos*. <https://anatrenza.com/ratio-de-solvencia/>. Consulta: 18 de marzo de 2019.

Por otro lado, la interpretación de este valor es muy importante, primero que nada, para tener un índice de solvencia aceptable los activos deben de ser mayores a los pasivos; si se tuviese un índice de solvencia de 3,70, es decir, que la entidad cuenta con 3,70 quetzales por cada quetzal de deudas. De manera que, si no se cuenta con una solvencia mayor a uno, no se cuenta con el suficiente capital para afrontar las deudas de la empresa, derivado de esto y considerando los datos de las tablas en la sección anterior se presenta a continuación la cifra mínima de activos corrientes.

Tabla XIV. **Activos mínimos múltiples *headends***

Descripción	Monto
Activos corrientes	Q.909 090,90

Fuente: elaboración propia.

Tabla XV. **Activos corrientes mínimos topología final**

Descripción	Monto
Activos corrientes	Q.758 282,82

Fuente: elaboración propia.

5.5.3. **Liquidez**

Este concepto se confunde comúnmente con el de solvencia, son muy parecidos mas no iguales; por su parte, la liquidez representa la virtud de nuestros activos para volverse efectivo. Esta medida se da a partir de la razón entre los activos corrientes y pasivos corrientes de la empresa, es decir todos los activos que inmediatamente se vuelven dinero; dicha fórmula se presenta a continuación.

Figura 79. **Fórmula, liquidez**

$$L = \frac{A_c}{P_c}$$

Fuente: *Índice de liquidez: conozca los indicadores y mejore sus finanzas.*

<https://www.myabcm.com/es/blog-post/indice-de-liquidez-conozca-los-indicadores-y-mejore-sus-finanzas/>. Consulta: 19 de marzo de 2019.

Por lo cual los activos corrientes lógicamente deben ser mayores a los pasivos corrientes, al igual que en la sección anterior. De manera que se puede concluir que ambos conceptos son semejantes y se complementan, básicamente uno expresa la capacidad de pago a largo plazo (solvencia), siendo el otro a corto plazo (liquidez), factores importantes a tomar en cuenta para nuestra empresa dado que nos da un indicio del comportamiento financiera de la misma.

5.5.4. TIR VAN

Cuando un proyecto se encuentra en vías de desarrollo es importante poder estimar en cuanto tiempo podría recuperarse la inversión inicial, derivado de esto se encuentran las herramientas TIR (tasa interna de retorno) y VAN (valor actual neto). Con base en una estimación de ingresos monetarios que tenga la empresa menos los gastos netos, se puede realizar una proyección de tiempo para que el capital inicial sea recuperado. Para calcular el VAN se utiliza una sumatoria de la razón del flujo neto por año y la tasa de interés más uno elevado a una potencia que representa el año, menos la inversión inicial; dicha fórmula se presenta a continuación.

Figura 80. **Formula VAN**

$$VAN = \sum_{n=1}^N \frac{Q_n}{(1+r)^n} - I$$

Fuente: *Tasa interna de retorno (TIR): definición, cálculo y ejemplos.*

<https://www.rankia.cl/blog/mejores-opiniones-chile/3391122-tasa-interna-retorno-tir-definicion-calculo-ejemplos>. Consulta: 20 de marzo de 2019.

Por otro lado, el TIR representa el valor del interés, cuando el VAN se hace cero, la interpretación de estos números es muy importante debido que si se obtiene un VAN positivo; es decir, se recuperará la inversión hecha inicialmente y podría tener un valor agregado para la entidad en cuestión; en caso contrario, significa que no se cubre la deuda e incluso no se obtienen ganancias sobre el proyecto.

Por su parte, el valor del TIR puede ser un factor muy importante al tomar una decisión empresarial; si este valor fuese alto este es un proyecto rentable que dará un retorno del capital invertido, de no ser así, la inversión no retornaría. Para casos de ejemplo se tomarán los datos de la tabla XII, si se realiza una proyección de seis años, el cálculo del VAN quedaría de la siguiente manera.

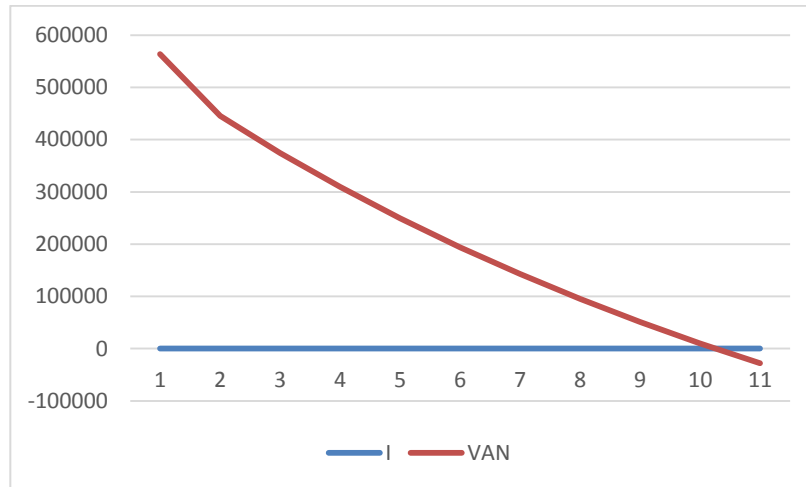
Tabla XVI. **Cálculo VAN/TIR múltiples Headends**

Descripción	Monto
Inversión inicial	Q.900 000,00
Flujo neto año 1	Q.300 000,00
Flujo neto año 2	Q.320 000,00
Flujo neto año 3	Q.340 000,00
Flujo neto año 4	Q.360 000,00
Flujo neto año 5	Q.380 000,00
Tasa de interés	10 %
VAN	Q.374 372,06
TIR	25 % (a 5 años)

Fuente: elaboración propia.

Con el gráfico como se ve a continuación.

Figura 81. **Gráfica VAN múltiples *headends***



Fuente: elaboración propia.

Siguiendo el caso de estudio, ahora se tomarán en cuenta los datos de la tabla XIII; con el mismo flujo de efectivo neto, se obtuvieron los siguientes resultados.

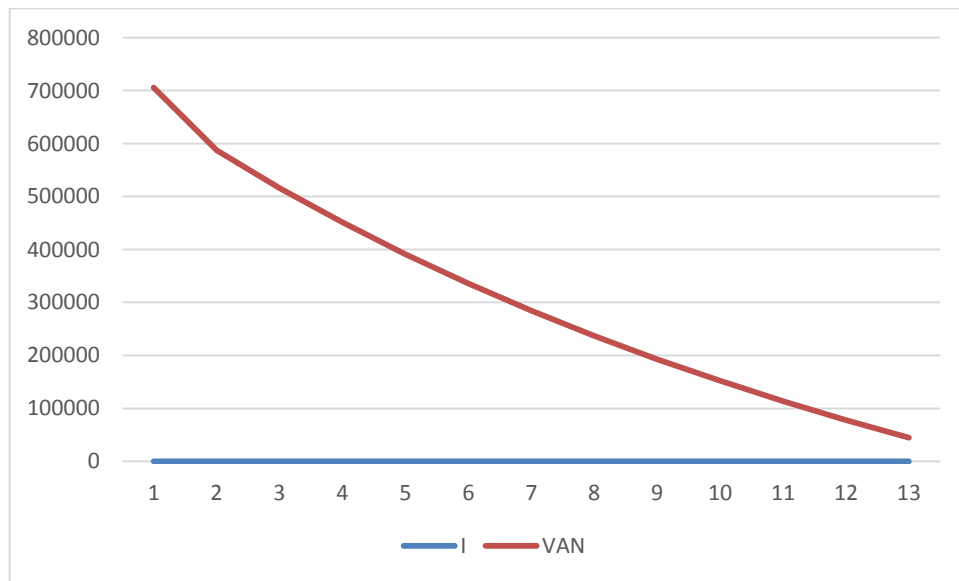
Tabla XVII. **Cálculo VAN/TIR, topología final**

Descripción	Monto
Inversión inicial	Q.758 282,82
Flujo neto año 1	Q.300 000,00
Flujo neto año 2	Q.320 000,00
Flujo neto año 3	Q.340 000,00
Flujo neto año 4	Q.360 000,00
Flujo neto año 5	Q.380 000,00
Tasa de interés	10 %
VAN	Q.516 189,24
TIR	32 % (a 5 años)

Fuente: elaboración propia.

Utilizando los datos de la tabla anterior, se presenta el gráfico a continuación.

Figura 82. Gráfica VAN, topología final



Fuente: elaboración propia.

De manera que en ambos casos el VAN es positivo; sin embargo, algo muy importante a resaltar es que el VAN obtenido en el caso de la topología presentada en este proyecto es 1,38 veces más que utilizando múltiples *headends*; por su parte, el TIR varía con el diseño final de la red aumentando a un 33 % indicando que el proyecto es mucho más rentable según la proyección realizada a 5 años.

Es indiscutible entonces la mejora sustancial a un proyecto de este tipo, con la aplicación del diseño presentado en este proyecto, tanto técnicamente como económicamente los resultados fueron contundentes.

CONCLUSIONES

1. Es posible consolidar una programación de canales digitales utilizando filtros de nivel de IP, manejando el flujo del tráfico en la red de distribución en función del desempeño adecuado de la red.
2. La topología de red fue diseñada para conmutar el tráfico e incluso adicionarlo desde otras vías; da de esta manera respaldo y flexibilidad a la red en caso de fallas.
3. El código desarrollado para los equipos es capaz de realizar la adición y conmutación de tráfico; mitiga el tiempo de falla de las incidencias en la red.
4. Según los argumentos técnicos presentados a lo largo del capítulo 3 de esta investigación, este proyecto es capaz de satisfacer las necesidades para la implementación de red, considerando su escalabilidad.

RECOMENDACIONES

1. Considerar que la utilización de filtros IP consume bastantes recursos de *hardware* en los equipos, por lo cual se recomienda la utilización de equipos de gama alta.
2. Integrar enlaces redundantes dentro de la red que funcionen como *failover*, para que en caso de corte funcionen inmediatamente, dándole mayor fiabilidad al proyecto.
3. El manejo del tráfico repercute directamente en la capacidad del enlace de datos; sumar muchos canales a la programación podría saturar el enlace y afectar el desempeño del servicio.
4. A medida que la red crezca se recomienda monitorear constantemente el desempeño de los equipos para prever futuros crecimientos, manteniendo así su escalabilidad. También, dar el mantenimiento adecuado a los equipos, así como las actualizaciones pertinentes de *firmware* y *software*.

BIBLIOGRAFÍA

1. ESCALANTE, Mauro. Conceptos fundamentales de Mikrotik Router OS. [En línea]. <<https://academyxperts.com/index.php/conceptos-fundamentales-de-mikrotik-routeros>>. [Consulta: 16 de mayo de 2019].
2. _____ . *Control de tráfico, Firewall y QoS con Mikrotik Router OS*. [En línea]. <<http://www.abcxperts.com/index.php/control-de-traffic-firewall-y-qos-con-mikrotik-routeros>>. [Consulta: 16 de mayo de 2019].
3. _____ . *Ruteo avanzado y alta disponibilidad con Mikrotik Router OS*. [En línea]. <<http://www.abcxperts.com/index.php/ruteo-avanzado-y-alta-disponibilidad-con-mikrotik-routeros>>. [Consulta: 16 de mayo de 2019].
4. ODOM, Wendell. *Routing & Switching. Learn, prepare, and practice for exam success*. [En línea]. <<http://www.ciscopress.com/store/ccent-cna-icnd1-100-101-official-cert-guide-9781587143854>>. [Consulta: 16 de mayo de 2019].

APÉNDICE

Apéndice 1. **Modelo OSI**

Básicamente hablamos de un modelo pensado y creado para poder establecer, así como hacer posible la comunicación de dos sistemas abiertos, establece en sus normativas una serie de pasos a seguir para cumplir dicho objetivo. Fue creado para resolver principalmente la problemática de la comunicación entre dispositivos de diferentes fabricantes, por esta razón es que en nuestros días sin importar la marca del dispositivo puede establecer sesiones e incluso enviar datagramas ethernet a un destinatario específico sin ninguna condicionante.

Aunque sus inicios se remontan a 1977 este fue oficialmente creado en el año de 1980 por la Organización Internacional de Normalización (ISO), desde 1984 la Unión Internacional de Telecomunicaciones lo presentó al mundo como un estándar, lo cual se mantiene vigente hasta hoy en día. Por otra parte, este se desglosa en una arquitectura jerárquica de 7 diferentes capas en las que se especifica los diferentes protocolos a tomar en cuenta para establecer la comunicación, también es llamado el modelo de referencia OSI dado que se utiliza para educar respecto al proceso de comunicación entre dispositivos. La idea de implementar una arquitectura de esta forma es descomponer la problemática del intercambio de datagramas en varias capas sin que una dependa de la otra, en otras palabras, poder establecer límites para la resolución de problemas (*troubleshooting*) gracias a la separación lógica de los 7 niveles.

Continuación del apéndice 1.

También es importante comentar que el modelo OSI originalmente se vio sumergido en una especie de competencia con respecto al modelo antecesor, es decir el modelo TCP/IP, el cual fue creado por el departamento de defensa de los Estados Unidos de América. Debido a la gran cantidad de protocolos y diferencias con respecto al anterior fue visto como algo bastante difícil de implementar e innecesario dada la simplicidad del modelo TCP/IP, a pesar de esto finalmente fue aceptado por todo el ámbito de telecomunicaciones principalmente por la segmentación de los posibles problemas, con el tiempo demostró ser un modelo sólido siendo prueba de ello su utilización hasta hoy en día.

La segmentación del modelo OSI se presenta a continuación.

- Capa 1: física
- Capa 2: enlace
- Capa 3: red
- Capa 4: transporte
- Capa 5: sesión
- Capa 6: presentación
- Capa 7: aplicación

Siendo la más cercana al usuario la capa de aplicación (capa 7), en contraste con la capa física (capa 1) siendo esta la más cercana a la máquina. Lógicamente cada nivel cuenta con características, protocolos y funcionalidades diferentes, por lo cual a continuación se realiza una explicación más a profundidad sobre el tema.

Continuación del apéndice 1.

- Capa 1, Física

Como fue mencionado anteriormente, esta es la más cercana a la máquina dado que en este punto los datos se manejan en forma de bits, lo cual quiere decir que solo pueden ser comprendidos por la máquina. Aquí mismo los datos son preparados para ser enviados hacia otro dispositivo en forma de impulsos eléctricos u ópticos, según sea definido por el usuario o en función de las capacidades del equipo.

Es aquí donde se define en que forma, así como el medio en que serán transmitidos los datos, aquí mismo los equipos son capaces de manejar la codificación de los bits provenientes de la capa de enlace de datos siendo la única limitante en este punto la capacidad de los equipos e incluso el medio de transmisión. Dichos equipos deciden la representación física de los valores 1 o 0, la duración de cada uno de los bits, también entra en consideración los aspectos eléctricos (niveles de tensión), mecánicos (componentes/conectores) y medios de transmisión ya sea eléctricos u ópticos lo cual quiere decir que se define si el medio de comunicación será un par de cable trenzado, fibra óptica, aire, guías de onda o bien cable coaxial, por ejemplo.

Resulta oportuno mencionar que también se encarga de definir las características propias de cada interfaz física, así como el establecimiento, mantenimiento y liberación del enlace, además de garantizar la conexión entre interfaces. A continuación, un listado con los dispositivos que podrían ser ubicados en esta capa del modelo OSI.

Continuación del apéndice 1.

- Repetidores
- Amplificadores
- Multiplexores
- Codecs
- Transductores
- Cables
- Conectores
- Tarjetas

Dentro de los protocolos más utilizados podríamos encontrar, por ejemplo.

- RS232
- X.21
- DSL
- RDSI
- USB

Todo lo anterior nos lleva a la conclusión de que esta capa hace referencia únicamente a dispositivos físicos, manejo de datos a nivel de *bits* y establecimiento de la conexión entre interfaces.

Continuación del apéndice 1.

- Capa 2, Enlace

Cuando un conjunto de dispositivos se encuentra dentro del mismo dominio de colisión, básicamente todos los dispositivos tienen conectividad entre sí, lo cual es la base sobre la cual se fundamenta la capa 2 del modelo OSI. Por defecto cada dispositivo que genera un fabricante cuenta con una dirección física, esta dirección es única e irrepetible, es un número hexadecimal de 48 *bits* de los cuales los últimos 24 *bits* los asigna el IEEE y los otros 24 *bits* los asigna el fabricante.

Dicha conectividad se logra a través del direccionamiento a nivel físico por medio de la dirección en cada dispositivo mejor conocida como *mac address*. Otro aspecto importante en esta capa es la detección de errores, revisa la trama en busca de algún error, de ser así obliga a realizar una retransmisión de los datos, todo esto por medio de software lo cual les da la seguridad a las capas superiores que la transmisión de datos por medio del enlace físico se llevó a cabo de manera correcta.

En ese mismo sentido se encarga del control de flujo, acceso al medio, distribución de las tramas, entre otros. Se pueden mencionar principalmente dos dispositivos representativos de esta capa, enlistados a continuación.

- Hub
- Switch

Continuación del apéndice 1.

El *Hub* fue uno de los primeros dispositivos, este hacia real la capacidad de comunicar dispositivos dentro de un mismo dominio de colisión, tomaba la trama a transmitir desde el puerto emisor, posteriormente lo retransmitía a todos los puertos excepto el puerto emisor lo cual provocaba una gran cantidad de colisión de paquetes en el dispositivo volviéndolo poco funcional. Por su lado el *switch* aplica una dinámica de funcionamiento diferente, utiliza la técnica reenvió directo que consiste en reenviar la trama hacia el puerto destino inmediatamente después de su recepción (muy parecido al *hub*) lo cual brinda una comunicación muy rápida, pero poco eficiente porque se generan muchas colisiones. También, los hay con la técnica de almacenamiento y reenvió (la más utilizada), consiste en almacenar la trama completa en buffers internos para ser enviada finalmente al puerto destino.

Dentro de los protocolos que se manejan se pueden mencionar los siguientes.

- HDLC
- LLC

Ambos protocolos se encargan de la detección de errores a nivel de enlace datos, del entramado, direccionamiento de *mac address*, entre otros.

Continuación del apéndice 1.

- Capa 3, red

En esta fase se da el direccionamiento lógico a los paquetes, se determina el camino más adecuado para que puedan llegar a su destino, dicho direccionamiento se puede realizar a través de direcciones IP, rutas estáticas o en su defecto protocolos de enrutamiento dinámicos.

El protocolo de internet IP cuenta con dos versiones IPv4 (versión 4) e IPv6 (versión 6), a pesar de que el concepto es parecido, existen algunas variaciones entre ambas versiones. Por su lado IPv4 es una dirección lógica de 32 *bits* de longitud separada en 4 grupos de 8 *bits* con una representación decimal, que sirven para identificar a un dispositivo dentro de una red, IPv6 es un número de 128 *bits* de longitud separado en 4 grupos de 32 *bits* con una representación hexadecimal, ambas versiones del protocolo son manejadas por un dispositivo llamado enrutador (*router*), encargado de enviar los paquetes desde su origen hasta su destino.

Esta capa se puede sub dividir en tres más, sub capa de acceso que trabaja sobre la interfaz directamente, sub capa dependiente de la convergencia y sub capa independiente de convergencia que se encargan de la operación entre sub redes, enrutamiento de paquetes, control de concurrencia en la red, entre otros. Para realizar el enrutamiento adecuado de los paquetes se utilizan diferentes algoritmos para calcular rutas, los hay de dos tipos, vector distancia y estado de enlace, en cuanto a enrutamiento dinámico se refiere, adicionalmente, las rutas estáticas complementan la tabla de enrutamiento de un *router*, a continuación, un listado de protocolos más utilizados.

Continuación del apéndice 1.

- RIP
- OSPF
- EIGRP
- BGP

Dichos protocolos de enrutamiento cuentan con similitudes, simplemente difieren en ciertos aspectos respecto a su funcionamiento, pero cualquier protocolo de enrutamiento dinámico hace que un *router* aprenda la información de subredes IP de sus *routers* vecinos, anuncie las rutas hacia las subredes aprendidas a todos sus vecinos, si existe más de una ruta hacia el destino decide cual es la mejor con base en la métrica de la ruta, reacciona ante los cambios en la red inmediatamente.

- Rutas estáticas

Dichas rutas son definidas por el administrador de la red, tal como su nombre lo dice implícitamente estas no nunca cambian, están compuestas por una red de destino y el siguiente salto (*next hop*) para alcanzar el destino. Es muy importante mencionar que, a pesar de la simplicidad de este método, no es escalable en redes grandes o en puertos de un crecimiento, dado que las rutas deberían de ser agregadas a todos los *routers* en la red, volviendo demasiado lenta la convergencia de la misma.

Generalmente, solo se recomienda utilizar rutas estáticas cuando se desconoce la red que se encuentra más allá de nuestro siguiente salto, por ejemplo, en nuestra conexión hacia un ISP, también cuando se quiere forzar que el tráfico salga por una vía específicamente.

Continuación del apéndice 1.

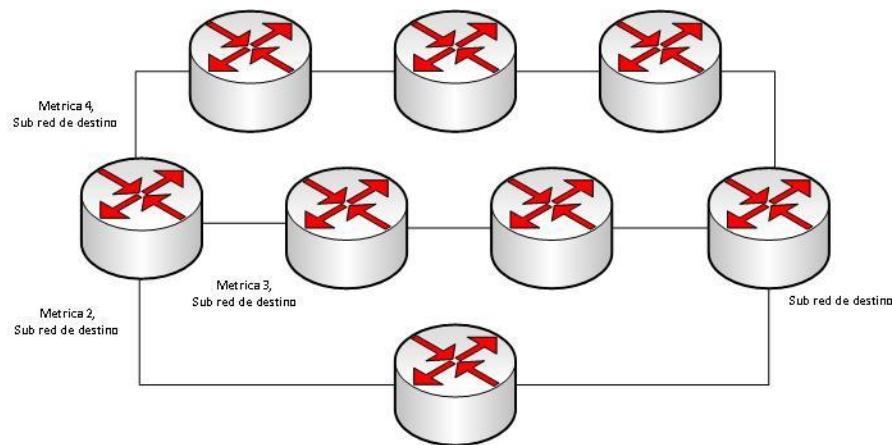
- RIP

De los primeros protocolos en su categoría, el protocolo de información de encaminamiento tiene sus orígenes en la década de 1980 dada la necesidad del crecimiento de las redes fue necesario formular un protocolo capaz encaminar el tráfico dinámicamente, dicho protocolo cuenta con dos versiones, la primera versión se volvió el protocolo de enrutamiento dinámico más popular de su época, caracterizado principalmente por tener un comportamiento *classful* y transmitir sus actualizaciones por *broadcast*. La segunda entrega de este protocolo vino con la segunda oleada de protocolos de enrutamiento, en el año de 1990, muy similar a su antecesor con la diferente de que sus actualizaciones eran por medio de una dirección *multicast*, agregando la característica muy importante de soportar *classless*, ya no haciendo distinciones entre clases de direcciones IP.

Considerado como un IGP (*interior gateway protocol*), se trata de un protocolo de enrutamiento dinámico de vector distancia, calcula la métrica de una ruta en función de los saltos que tiene que realizar para llegar al destino. Elige la mejor ruta en base de la métrica más baja, es decir la ruta que cuenta con la menor cantidad de saltos para poder llegar a su destino.

Continuación del apéndice 1.

Figura A. IGP



Fuente: elaboración propia, empleando Visio 2016.

- OSPF

También considerado un IGP, es un protocolo de enrutamiento dinámico que calcula la métrica de las rutas en función del estado de enlace por medio del algoritmo SPF (*shortest path first*), identificado con el número de protocolo 89, este trabaja en la capa 3 del modelo OSI.

Soporta Classless, autenticación y envía sus actualizaciones por medio de la dirección *multicast* 224.0.0.5, además OSPF envía la información de sus vecinos a todos los demás Routers dentro de la topología. La métrica es basada en el ancho de banda, utilizando la siguiente fórmula para calcularla.

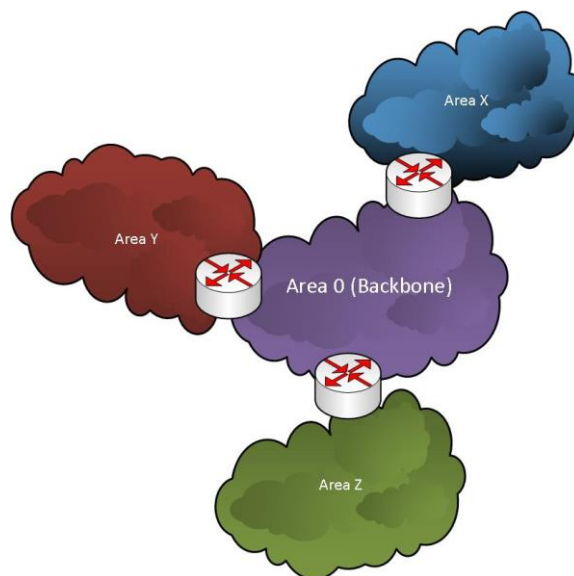
$$\text{Costo} = \frac{\text{ancho de banda de referencia}}{\text{ancho de banda de la interface}}$$

Continuación del apéndice 1.

Por otro lado, OSPF mantiene 3 diferentes tablas para su funcionamiento, la tabla de rutas que contiene la diferentes rutas hacia diversos destinos, vecinos que almacena la información de todos los dispositivos que el propio *router* ve a través de alguna de sus interfaces y topología la cual almacena la información de todos los dispositivos que se encuentran dentro de la misma red ejecutando OSPF.

Otro aspecto importante es que OSPF basa su funcionamiento en áreas con el objetivo principal reducir la dimensión de las tablas mencionadas anteriormente, reduciendo así las actualizaciones entre dispositivos, el área número cero es considera como *backbone* (la columna vertebral) fuera de esta área todas son consideradas como estándar.

Figura B. **OSPF**



Fuente: elaboración propia, empleando Visio 2016.

Continuación del apéndice 1.

- EIGRP

EIGRP es un protocolo sustituto de su predecesor IGRP, lanzado en 1993 por la compañía fabricante de equipos Cisco, por muchos años fue propietario Cisco, liberado como un estándar abierto en 2013.

Es un protocolo de enrutamiento vector distancia que incorpora en su funcionamiento ciertas características de estado de enlace, soporta *classless*, se identifica con el número de protocolo 88 y envía sus actualizaciones a sus vecinos por la dirección de multidifusión 224.0.0.10.

Al contrario de OSPF, este no cuenta con una visión completa de la topología de la red por lo cual no garantiza la utilización de la mejor ruta hacia el destino. Para su funcionamiento se vale del algoritmo llamado DUAL (*diffusing update algorithm*), las actualizaciones hacia los vecinos solamente se envían cuando la topología cambia, lo cual favorece mucho a la convergencia rápida de la red.

Adicionalmente a lo anterior, el cálculo de la métrica de EIGRP es extremadamente complicado, dado que toma en cuenta la carga de tráfico, el ancho de banda, errores y retraso inducido por las interfaces.

$$\text{Métrica} = \frac{10^7 * 256}{\text{ancho de banda (interface más lenta)}} + \sum \text{Retrasos} * 256$$

Continuación del apéndice 1.

Al igual que OSPF, EIGRP guarda 3 tablas, la de enrutamiento, vecinos y topología, con la pequeña diferencia que las mejores rutas aprendidas a través de la tabla de vecinos serán copiadas a la tabla de enrutamiento como respaldo.

- BGP

Considerado como EGP (*exterior gateway protocol*), es el único protocolo de enrutamiento utilizado en el encaminar tráfico hacia internet.

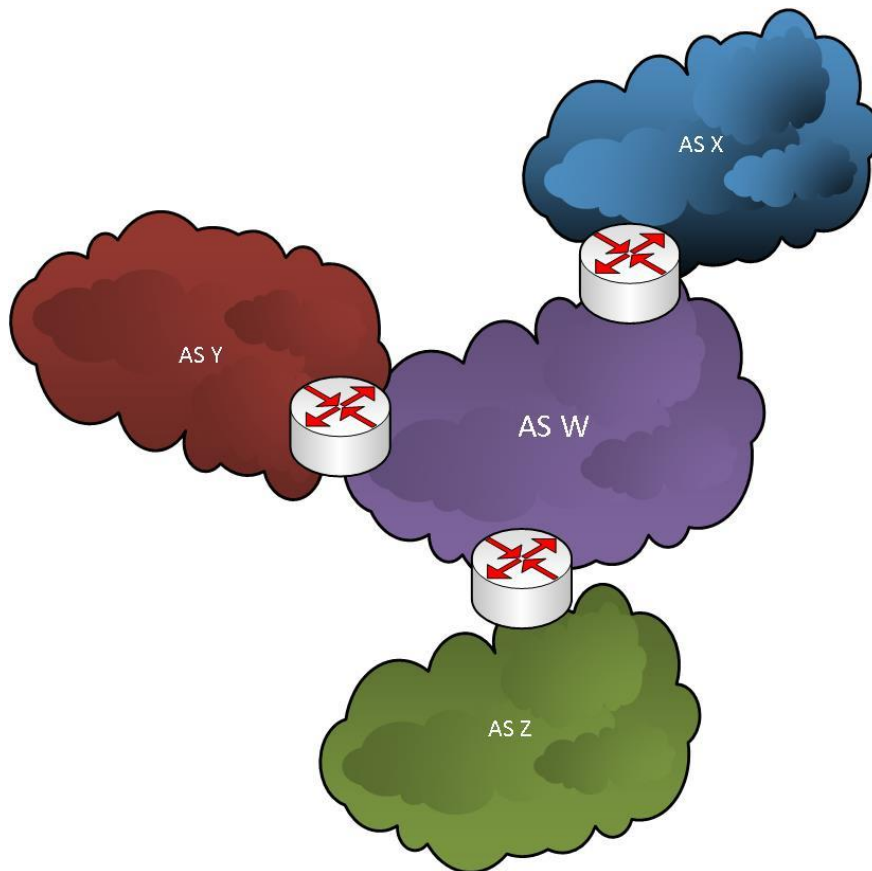
Continuación del anexo 1.

Para establecer una sesión BGP es necesario utilizar un número de sistema autónomo (AS), dicho número identifica a nivel global a un conjunto de equipos que forman una red.

Funciona bajo el algoritmo de vector distancia, muy parecido a RIP, con la diferencia que no cuenta los saltos entre *routers*, lo hace contando los saltos entre sistemas autónomos. A través de esto encamina el tráfico hacia la mejor ruta disponible a través de la red global, para su funcionamiento utiliza el protocolo TCP en el puerto 179. En el caso de BGP los vecinos son llamados *Peers*, este protocolo también puede ser utilizado en escenarios privados, lo cual marca la diferencia entre iBGP (Interno) y eBGP (Externo).

Continuación del apéndice 1.

Figura C. **BGP**



Fuente: elaboración propia, empleando Visio 2016.

- Capa 4, transporte

La capa 4 del modelo OSI establece la forma en la cual los datos serán enviados a su destino, se encarga de llevar la información en forma precisa y confiable.

Continuación del apéndice 1.

Esto lo hace por medio de dos protocolos de transporte sumamente importantes, el protocolo de transmisión de control TCP totalmente orientado a la conexión y el protocolo de datagrama de usuario UDP que está enfocado al envío de datos sin conexión.

Técnicamente se encarga de recibir los datos de la capa superior, verificar el estado de los datos en busca de algún error en la transmisión y finalmente pasarlo a la capa inferior para que se envíen al destino final. Por otro parte, se encarga del control de flujo, establecimiento, mantenimiento, así como el cierre de conexión entre dos sistemas. Este nivel del modelo OSI es de suma importancia dado que marca la diferencia entre los niveles superiores e inferiores, el protocolo usado por esta capa marca mucho la diferencia en cuanto a la calidad de servicio se refiere, por su parte TCP es mucho más confiable que UDP, pero en contraste UDP es más rápido, existen 5 clases de protocolos orientados a la conexión, dicha clasificación es en función de las capacidades de cada protocolo.

- Clase 0
 - Sin mecanismo de detección de errores
 - Sin mecanismo de corrección de errores
- Clase 1
 - Recuperación básica de errores
- Clase 2

- Continuación del apéndice 1.
 - Implementa mecanismos de control de flujo
 - Permite conexiones de transporte multiplexadas
 - Clase 3
 - Recuperación de errores
 - Permite conexiones de transporte multiplexadas
 - Clase 4
 - Detección
 - Recuperación de errores
 - Permite conexiones de transporte multiplexadas
- TCP

El protocolo de transmisión de control está orientado a realizar transmisión de datos por medio de una conexión de manera que cuando esta es creada se genera un flujo de datos a través de la conexión. Este protocolo cuenta con una gran ventaja debido que se encarga de poder detectar errores en la transmisión por lo cual la trama de datos puede llegar a su destino sin error alguno.

Este protocolo utiliza un intercambio de 3 vías llamado *3 way handshake*, primero el host envía una solicitud de sincronización, seguido de esto el servidor responde con su propia solicitud de sincronización y un acuse de recibido. Finalmente, el host que envió la solicitud inicialmente envía un acuse de recibido, creando de esta manera la conexión.

Continuación del apéndice 1.

- UDP

Llamado protocolo de datagrama de usuario, se encarga de realizar transmisión de datos al igual que TCP con la mínima, pero muy notable diferencia que no crea conexiones, por lo cual tampoco tiene control sobre el flujo de datos y detección de errores sobre las tramas de datos enviadas. Esto lo convierte en un protocolo de transmisión no confiable en comparación con TCP, de tal manera que los datos pueden llegar con errores a su destino.

- Capa 5, sesión

La capa de sesión es aquella encargada de crear, mantener y cerrar el diálogo entre aplicaciones de dos sistemas abiertos. En dicho diálogo se desarrolla un intercambio de datos, de forma que la sesión sigue un proceso de dos fases.

- Establecimiento
- Utilización y liberación

Cuenta con los siguientes servicios.

- Control del diálogo: este puede ser *full-duplex* (ambos sentidos) o *half-duplex* (un sentido).
- Agrupamiento: los datos pueden ser etiquetados para formar grupos de datos.

Continuación del apéndice 1.

- Recuperación: si en dado caso se pierde la conexión, cuenta con la capacidad de poder recuperarla en el punto donde estaba anteriormente y no desde el principio.

Sobre todo, se ocupa de la sincronización entre *host* mientras se realiza la transferencia de datos de cualquier tipo, para que la sesión se mantenga activa durante la transferencia. Dentro de los protocolos más conocidos podemos encontrar los siguientes.

- SMTP
 - FTP
 - SAP
 - ZIP
 - RCP
 - SCP
- Capa 6, presentación

La capa de presentación escoge la forma en la cual los datos serán presentados al usuario final a través de una aplicación sin importar el formato de los caracteres ya sea ASCII, Unicode, entre otros. Cumple con tres funciones específicas que se encargan de poder llevar a cabo la comunicación entre aplicaciones de dos sistemas abiertos diferentes.

- Presentación de datos
- Cifrado de datos
- Compresión de datos

- Continuación del apéndice 1.

Se encarga de que la información se envíe de forma que el receptor pueda comprenderla, el cifrado de datos también se da en esta capa del modelo OSI, trata temas de semántica y sintaxis, de manera que protocolos que hablan diferentes idiomas pueden entenderse actuando como un traductor.

- Capa 7, aplicación

La capa de aplicación es la última del modelo OSI, la más cercana al usuario y una de las más importantes, sin ella no podría completarse el proceso de la comunicación entre dos sistemas abiertos. En ella se encuentran las aplicaciones de red y los servicios necesarios para que el usuario pueda interactuar con otros dispositivos.

Dentro de los protocolos más conocidos podemos encontrar.

- HTTP
- HTTPS
- POP3
- IMAP
- NFS
- AFP
- DNS
- Telnet
- SSH

Continuación del apéndice 1.

▪ Costos de equipos a utilizar

	Mikrotik CSS326-24G-2S+RM 24 port Gigabit Ethernet switch with two SFP+ Nuevo	Cant. <input type="text" value="1"/>	US \$152.89 Envío gratis
		USPS Priority Mail De 2 a 5 días	
	Mikrotik CCR1009-7G-1C-1S+ - Cloud Core Router CCR1009-7G-1C-1S+ w/ Dual Power Nuevo	Cant. <input type="text" value="1"/>	US \$578.99 Envío gratis
		Standard Shipping from outside US De 5 a 21 días	
	MikroTik Cloud Core Router CCR1036-12G-4S 12x Gigabit ports, 36 CPU Nuevo	Cant. <input type="text" value="1"/>	US \$904.58 Envío gratis
		USPS Priority Mail De 2 a 5 días	
	Bo-Erbium-doped Fiber Amplifier EDFA Nuevo	Cant. <input type="text" value="1"/>	US \$2 599.00 Envío gratis
		DHL Global Mail Parcel Direct De 26 a 43 días	
	Blonder Tongue HDE-2H/2S-QAM MPEG-2 HD Encoder (2 HDMI&HD-SDI/4 Usado ÚLTIMO	Cant. 1	US \$4 495.00 Envío gratis
		UPS Ground De 4 a 9 días	
	New Pico Macom PHC-12G 1GHz Broadband Passive Headend Combiner Open box ÚLTIMO	Cant. 1	US \$95.00 Más US \$16.95
		FedEx Ground or FedEx Home Delivery De 2 a 7 días	
	Roland XS-84H 8-in x 4-out Multi-Format AV Matrix Switcher Nuevo	Cant. <input type="text" value="1"/>	US \$7 995.00 Envío gratis
		FedEx Ground or FedEx Home Delivery De 3 a 8 días	

Continuación del apéndice 1.



**Pico Digital PCM55SAW 550 MHz
Channelized-Agile PLL SAW-Filtered A/V**
Nuevo

Cant.

US \$118.00



**Cisco D9858-1 Advanced Receiver
Transcoder with CLC: ATP P/N**
Usado
ÚLTIMO

Cant. 1

US \$995.00

Fuente: elaboración propia.

