



**UNIVERSIDAD TECNICA DE MACHALA
FACULTAD DE INGENIERÍA CIVIL
MAESTRÍA EN SOFTWARE**

**ANÁLISIS COMPARATIVO DE PROTOCOLOS DE COMUNICACIÓN PARA REDES
DEFINIDAS POR SOFTWARE**

ING. GEOVANNY MANUEL MOCHA GUACHO

**“ARTÍCULO PROFESIONAL DE ALTO NIVEL EN OPCIÓN AL TÍTULO DE
MAGISTER EN SOFTWARE”**

**TUTOR: ING. RODRIGO FERNANDO MOROCHO ROMAN
COTUTORA: ING. JENNIFER KATHERIN CÉLLERI PACHECO**

**MACHALA
2021**

DEDICATORIA

A Dios por darme la salud, vida y sabiduría para lograr cumplir cada uno de mis sueños.

A mis padres por ser el apoyo fundamental en el transcurso de esta travesía de estudio, por sus oraciones a la distancia, por sus sabios consejos y muestras de amor incondicional.

A las personas que confiaron en mí y me brindaron apoyo en el transcurso de esta maestría.

AGRADECIMIENTO

Primeramente, agradezco a DIOS, por haberme dado la vida y sabiduría necesaria para lograr una meta más en mi vida.

A mi tutor por ser una guía en el proceso de titulación, por la dedicación y paciencia en el transcurso del desarrollo de la tesis, por el apoyo en cada uno de los procesos dificultosos y por impartir sus conocimientos para cumplir esta meta.

A cada uno de los docentes que formaron parte de esta linda travesía, por la dedicación y sus conocimientos impartidos los mismos que fueron fundamentales para mi formación académica.

RESPONSABILIDAD DE AUTORÍA

Por medio de la presente declaro ante el Comité Académico de la Maestría de Software, de la Universidad Técnica de Machala, que el trabajo de titulación, titulado “ANÁLISIS COMPARATIVO DE PROTOCOLOS DE COMUNICACIÓN PARA REDES DEFINIDAS POR SOFTWARE”, de mi propia autoría, no contiene material escrito por otra persona al no ser referenciado debidamente en el texto; parte de ella o en su totalidad no ha sido aceptada para el otorgamiento de cualquier otro diploma de una institución nacional o extranjera.



Ing. Geovanny Manuel Mocha Guacho
CI: 2300331689

Machala, 12 de marzo de 2021

REPORTE DE SIMILITUD TURNITIN

Informe-Titulación

INFORME DE ORIGINALIDAD

6%	5%	1%	1%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Escuela Politecnica Nacional Trabajo del estudiante	<1 %
2	Submitted to Universidad Internacional de la Rioja Trabajo del estudiante	<1 %
3	www.mysciencework.com Fuente de Internet	<1 %
4	www.slideshare.net Fuente de Internet	<1 %
5	archive.org Fuente de Internet	<1 %
6	www.chrysalis.cl Fuente de Internet	<1 %
7	hdl.handle.net Fuente de Internet	<1 %
8	revistas.uees.edu.ec Fuente de Internet	<1 %

9	www.cta.org.co Fuente de Internet	<1%
10	worldwidescience.org Fuente de Internet	<1%
11	eprints.cs.univie.ac.at Fuente de Internet	<1%
12	gredos.usal.es Fuente de Internet	<1%
13	www.fi.uba.ar Fuente de Internet	<1%
14	es.scribd.com Fuente de Internet	<1%
15	www.coursehero.com Fuente de Internet	<1%
16	www.udc.es Fuente de Internet	<1%
17	www.pedromurillo.com Fuente de Internet	<1%
18	inba.info Fuente de Internet	<1%
19	Jose Aguilar. "", IEEE Latin America Transactions, 12/2007 Publicación	<1%

20	diposit.ub.edu Fuente de Internet	<1%
21	www.iapss.org Fuente de Internet	<1%
22	mgpa.forestaluchile.cl Fuente de Internet	<1%
23	repositorio.uta.edu.ec Fuente de Internet	<1%
24	lalunadetrapo.blogspot.com Fuente de Internet	<1%
25	www.ochina.gov.ve Fuente de Internet	<1%
26	Ana Vidal Pantaleoni. "Técnicas secuenciales y paralelas para la resolución de problemas computacionales en electromagnetismo.", Universitat Politecnica de Valencia, 2013 Publicación	<1%
27	Submitted to Universidad Carlos III de Madrid Trabajo del estudiante	<1%
28	www.scribd.com Fuente de Internet	<1%
29	repositorio.uladech.edu.pe Fuente de Internet	<1%
	www.researchgate.net	

30	Fuente de Internet	<1%
31	R. Goñi-Viguria, E. Yoldi-Arzo, L. Casajús-Sola, T. Aquerreta-Larraya et al. "Fisioterapia respiratoria en la unidad de cuidados intensivos: Revisión bibliográfica", Enfermería Intensiva, 2018 Publicación	<1%
32	futur.upc.edu Fuente de Internet	<1%
33	issuu.com Fuente de Internet	<1%
34	www.urbe.edu Fuente de Internet	<1%
35	repositorio.ucv.edu.pe Fuente de Internet	<1%
36	su-beta.diva-portal.org Fuente de Internet	<1%
37	www.inrena.gob.pe Fuente de Internet	<1%
38	www.science.gov Fuente de Internet	<1%
39	www.netcom.it.uc3m.es Fuente de Internet	<1%

40	uladech.edu.pe Fuente de Internet	<1%
41	docplayer.es Fuente de Internet	<1%
42	repositorio.usfq.edu.ec Fuente de Internet	<1%
43	tesis.pucp.edu.pe Fuente de Internet	<1%
44	www.tdx.cat Fuente de Internet	<1%

Excluir citas Apagado
Excluir bibliografía Apagado

Excluir coincidencias Apagado

CERTIFICACIÓN DEL TUTOR

Por medio de la presente apruebo que el trabajo de titulación titulado “Análisis comparativo de protocolos de comunicación para Redes Definidas por Software” del autor Geovanny Manuel Mocha Guacho, en opción al título de Master en Software, sea presentada al Acto de Defensa.

Firmado digitalmente por RODRIGO
FERNANDO MOROCHO ROMAN
Fecha: 2021.03.12 14:56:01 -05'00'

Rodrigo Morocho Román

C.I: 0703820464

Machala, 12 de marzo de 2021

CESIÓN DE DERECHO DE AUTORÍA

Yo, GEOVANNY MANUEL MOCHA GUACHO, en calidad de autor del presente trabajo titulado “ANÁLISIS COMPARATIVO DE PROTOCOLOS DE COMUNICACIÓN PARA REDES DEFINIDAS POR SOFTWARE”, autorizo a la UNIVERSIDAD TECNICA DE MACHALA, la publicación y distribución en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor declara que el contenido que se publicara es de carácter académico y enmarca en las disposiciones definidas por la Universidad Técnica de Machala.



Ing. Geovanny Manuel Mocha Guacho

CI: 2300331689

Machala, 12 de marzo de 2021

CERTIFICACIÓN DE PUBLICACIÓN



Perú (Jesús María), enero 15 del 2021

Ing. Geovanny Mocha Guacho
Mg. Jennifer Celleri Pacheco
Universidad Técnica de Machala
Ecuador

De nuestra consideración:

Es un honor dirigirme a Usted y a su distinguida Institución para agradecerles la deferencia brindada a nuestra Universidad Alas Peruanas, Vicerrectorado de Investigación, Innovación y Emprendimiento, por las consideraciones otorgadas a través del apoyo brindado al compartir su experiencia y experticia habiendo sido aceptada la publicación en nuestra Revista Científica Hamut'ay el artículo: "Análisis Comparativo de Protocolos de Comunicación para Redes definidas por Software", en el Vol. 7 (3), 2020, el cual se puede visualizar en el siguiente link: <http://revistas.uap.edu.pe/ojs/index.php/HAMUT/article/view/2190/2293>
Como editor en jefe de la revista puedo dar fe de lo relevante del tema, como lo certificaron los pares evaluadores, quienes dieron su visto para la publicación del artículo.

Esperamos que este sea el inicio de una fructífera labor académica y de investigación entre ambas universidades y países hermanos como son Ecuador y Perú.

Sin otro particular hago propicia la oportunidad para testimoniarle los sentimientos de mi más alta estima personal y consideración.

Atentamente,

CLEOFÉ GENOVEVA ALVITES HUAMANI
Editor en Jefe - Revista Hamut'ay
Investigadora RENACYT P0012213



RESUMEN

Las redes definidas por software (en adelante SDN) se las considera como una innovación en la administración de las redes de comunicación basadas en la programación de la red o denominadas también como redes programables. La principal característica de las SDN es el desacoplamiento de los planos de control y datos. El concepto de las SDN ha ido tomando fuerza, sin embargo, no se evidencian trabajos que analicen los protocolos hacia el sur (southbound) que se ubican como un canal de comunicación entre los planos de control y datos. Por su acogida, es necesario realizar estudios que permitan identificar estos protocolos y analizar el factor de optimización en la comunicación entre los planos antes mencionados. El objetivo de este estudio fue analizar los protocolos de comunicación southbound, identificarlos y poner a prueba dos de estos protocolos: OVSDB y OpenFlow. Para dicha prueba, se empleó el enfoque causi experimental lo que permitió conocer el comportamiento de estos protocolos seleccionados por medio de la tasa de transferencia de paquetes ICMP entre los hosts de la red. Para apoyar el método causi experimental, se usó Mininet, una herramienta de emulación que permite la implementación de una red SDN. Se empleó una topología Lineal y Simple. Como resultados de este estudio, se logró evidenciar que el protocolo OpenFlow logro mejorar la comunicación entre los planos de datos y control con respecto a OVSDB. Ante esto, se pudo comprobar que el protocolo SouthBound puede inferir en el rendimiento de la red. Para lograr hacer pruebas con equipos reales, es necesario considerar los controladores adecuados para los protocolos que se vayan a poner a prueba.

Palabras Clave: Mininet, Protocolos de comunicación southbound, Redes definidas por software, Rendimiento de red.

ABSTRACT

Software-defined networks (hereinafter SDN) are considered as an innovation in the management of communication networks based on network programming or also known as programmable networks. The main characteristic is the decoupling of the control and data planes. The concept of SDN has been gaining strength, however, there is no evidence of studies that analyze the protocols to the south (southbound) that are located as a communication channel between the control and data planes. Due to their acceptance, it is necessary to carry out studies that allow these protocols to be identified and to analyze the optimization factor in communication between the aforementioned planes. The objective of this study is to analyze communication protocols to the south, identify them and test two of these protocols: OVSDB and OpenFlow. For this test, the experimental method was used, which allowed us to know the behavior of these selected protocols by means of the ICMP packet transfer rate between the hosts of the network. To support the experimental method, Mininet was used, an emulation tool which allowed the implementation of an SDN network on a Linear and Simple topology. As results of this study, it was possible to show better results using the OpenFlow protocol with respect to OVSDB, which allowed us to verify that the SouthBound protocol can infer the performance of the network. In order to be able to test with real equipment, it is necessary to consider the appropriate drivers for the protocols to be tested.

Keywords: Mininet, Southbound communication protocols, Software-defined networks, Network performance.

ÍNDICE GENERAL

DEDICATORIA.....	1
AGRADECIMIENTO	3
RESPONSABILIDAD DE AUTORÍA	4
REPORTE DE SIMILITUD TURNITIN	5
CERTIFICACIÓN DEL TUTOR.....	10
CESIÓN DE DERECHO DE AUTORÍA.....	11
CERTIFICACIÓN DE PUBLICACIÓN	12
RESUMEN.....	13
ABSTRACT	14
ÍNDICE GENERAL	15
INTRODUCCIÓN.....	18
CAPÍTULO 1: MARCO REFERENCIAL	23
1.1. Antecedentes históricos de la Investigación	23
1.1.1. Objetivos del estudio.....	23
1.2. Antecedentes conceptuales	29
CAPÍTULO 2 MATERIALES Y MÉTODOS	40
2.1. Tipo de Estudio o Investigación.....	40
2.2. Paradigma o enfoque	40
2.3. Cálculo de la población y muestra	41
2.4. Métodos teóricos con los materiales utilizados.....	43
2.5. Métodos empíricos con los materiales utilizados.....	43
2.6. Técnicas estadísticas para el procesamiento de datos obtenidos....	46
CAPÍTULO 3: RESULTADOS	48
CAPÍTULO 4: DISCUSIÓN DE LOS RESULTADOS	52
CONCLUSIONES.....	53
RECOMENDACIONES.....	54
BIBLIOGRAFÍA	55

LISTA DE ILUSTRACIONES

Ilustración 1: Protocolo para realizar un para realizar un RSL	24
Ilustración 2: Filtro aplicado a los resultados de estudios primarios.....	26
Ilustración 3: Propuesta para separación del plano de control y datos	27
Ilustración 4: Categorías de las variables.....	29
Ilustración 5: Diferencia entre Red tradicional y SDN.....	31
Ilustración 6: Arquitectura SDN.....	34
Ilustración 7: Contexto del Estudio	38
Ilustración 8: Tipo y diseño de investigación	40
Ilustración 9: Causa – efecto de las variables de investigación	41
Ilustración 10: Población y muestra del estudio.....	41
Ilustración 11: Arquitectura de red SDN	42
Ilustración 12: Proceso sistemático del estudio.....	43
Ilustración 13: Diseño de la topología (Lineal).....	45
Ilustración 14: Diseño de la topología alternativa (Simple)	46
Ilustración 15: Resultado de OVSDb con topología lineal	48
Ilustración 16: Resultado de OpenFlow con topología lineal	48
Ilustración 17: Resultados con la topología Lineal con OpenFlow.....	49
Ilustración 18: Resultados con diferentes tamaños de paquetes ICMP	50
Ilustración 19: Resultado de OVSDb con topología single o simple.....	51
Ilustración 20: Resultado de OpenFlow con topología single o simple	51

LISTA DE TABLAS

Tabla 1: Criterios de inclusión y exclusión para RSL	25
Tabla 2: Resultados de las cadenas de búsqueda	26
Tabla 3: Redes SDN vs Redes tradicionales	32
Tabla 4: Protocolos de comunicación - SDN.....	37
Tabla 5: Resumen de Protocolos Southbound	44
Tabla 6: Protocolos Southbound soportado por Mininet	44

LISTA DE ABREVIATURA Y SÍMBOLO

API: Application Programming Interface, en español Interfaz de programación de aplicaciones

DOI: Digital Object Identifier, en español, Identificador De Objeto Digital

RSL: Research Literature Review, en español Revisión de la literatura de investigación

SDN: Software-Defined Networking, en español, Redes Definidas por Software

TI: Tecnología de la información

INTRODUCCIÓN

El presente trabajo consiste en realizar una comparación de protocolos de comunicación que se consideran para la implementación de una red basada en software. El análisis se realiza bajo la evaluación de una topología, la misma que será ejecutada en una herramienta de emulación de redes. Entre algunos de los ambientes de emulación se encuentra Mininet, GNS3, etc. Estas herramientas son parte de un análisis para seleccionar el ambiente adecuado que permita la ejecución de redes SDN. Cabe destacar que este trabajo no tiene como finalidad determinar cuál es el mejor protocolo, sino, ser un punto de partida para la selección de un adecuado protocolo.

Para el cumplimiento del presente trabajo se parte de un análisis bibliográfico de los protocolos de comunicación. Esto va permitir la construcción de un cuadro comparativo de los protocolos existentes que soportan SDN. Finalmente se escogerá dos de los protocolos de comunicación y serán puestos a prueba en un ambiente de emulación bajo una misma topología, lo cual permite analizar el comportamiento de los protocolos seleccionados. El resultado del presente trabajo es de gran valor para la comunidad estudiantil e investigadores que estén inmersos en el área de redes de comunicación, puesto que representan una línea base para la selección de un protocolo adecuado en una SDN.

La nueva era digital ha llevado a muchas empresas públicas y privadas a tener presencia en internet, esto como estrategia para el crecimiento y la sobrevivencia en esta nueva era que avanza de forma rápida y que exige mucha competitividad [1]. Como se menciona en el trabajo de Virrueta y Millán, la era digital juega un “papel importante en el desarrollo económico de una sociedad o de un país” [2]. Lo cual se conoce, como la Transformación digital. Ante esto, nace la Cloud Computing, que es la representación de la infraestructura TI a través de la red. Cabe destacar que en la actualidad la

infraestructura TI es considerada como un activo estratégico en los negocios u organizaciones [3], [4] y como se menciona en [3] [5], mejoran el modelo de negocio, generan beneficios y ahorro de los costos sobre las soluciones tecnológicas.

El Ministerio de Telecomunicaciones y de la Sociedad de la Información, en su documento del Plan de Telecomunicaciones y TI resalta el incremento de consumo de datos (computadores y móviles). Ante esto, el ministerio impulsa la “necesidad de actualizar continuamente sus redes fijas y móviles a las últimas tecnologías, las cuales proveen mayores velocidades de subida y bajada de datos a los usuarios” [6]. El ministerio promueve entre uno de sus principales objetivos (denominados también como macro-objetivos) lo siguiente: “Completar y fomentar el despliegue de infraestructura de Telecomunicaciones”, por lo cual, es importante la implementación y el uso de nuevas tecnologías para un mejor despliegue de la infraestructura de red.

Hoy en día, el uso de las redes tradicionales o arquitecturas de red tradicionales, son evidentes en la mayoría de organizaciones. Las redes tradicionales son una “estructura cerrada y estática que no se puede adaptar en tiempo real a la demanda de las aplicaciones” [7]. Este tipo de arquitectura se componen de tres planos que son: control, datos y gestión. Estos tres planos, se encuentran embebidos dentro de los dispositivos de red, lo que conlleva hacer muy compleja su administración [8]. Además, no permiten satisfacer las necesidades o nuevos requerimientos de los modelos de negocios y limitan la reestructuración de los equipos o diseño de la red. La implementación y mantenimiento de este tipo de red es costosa y limitada. Esto ha motivado, desde años anteriores, a la comunidad que hace investigación sobre redes a llevar un paso adelante en busca de nuevas propuestas considerando el avance de la tecnología y el éxito del internet [9].

Por su parte, Benzekki en su artículo [10], menciona que es necesario dinamizar la administración de las redes, provocando la innovación mediante la programación de la red. En la actualidad los administradores de redes realizan un control y administración de las redes de forma manual, la cual es una desventaja en esta era digital, puesto que no pueden responder de forma óptima a patrones de tráfico impredecibles.

Ante esto, nacen las redes definidas por software (SDN de aquí en adelante), o también conocidas como redes programables [11]. Las redes SDN son una alternativa para la innovación de las redes [12] [7]. Se presentan como un nuevo paradigma en redes bajo un modelo programable, lo cual permite generar redes más eficientes [13] [14]. Las SDN son una propuesta para mejorar los aspectos de velocidad, infraestructura y costos en plataformas orientadas a cloud [15].

La principal característica de las redes SDN es que generan un control centralizado, lo que permite la administración o automatización de la red desde un punto único [15] [16]. Es decir, las SDN dan la oportunidad de “desplegar nuevas aplicaciones y servicios en cuestión de horas o días, en lugar de las semanas o meses necesarios en la tecnología de redes anteriores” [7]. Cabe destacar que las SDN, a diferencia de las redes tradicionales, en su arquitectura desacoplan el plano de control y datos del equipo de red, lo que da la posibilidad de “programar los dispositivos de red y en general la automatización de las redes de datos” [16]. De este modo, el funcionamiento de forma general, inicia desde el plano de aplicaciones en donde se definen las configuraciones de la red, las cuales son comunicadas al controlador SDN que está en el plano de control. El plano de control recibe las instrucciones y toma decisiones adecuadas. Desde el plano de control se comunican las decisiones a los dispositivos de red (switches y routers) que conforman el plano de datos o infraestructura.

Para la comunicación entre el plano de aplicaciones y el plano de control se emplean NorthBound APIs, cuya función es comunicar las instrucciones de las aplicaciones al controlador SDN. Para la comunicación entre el plano de control y de datos, se emplea un protocolo o interfaz de comunicación, al cual se lo conoce también como SouthBound APIs [16]. Este último es el cual se analizará en el presente trabajo. Los SouthBound o protocolos de comunicación, permiten el fácil acceso a dispositivos de red. Uno de los protocolos de comunicación más conocido es OpenFlow, sin embargo existen otros como ForCES y OpFlex [17].

En varios trabajos se puede observar que las SDN han sido aprovechadas para proyectos de IOT [18], [19], transmisión de videos [20] y Cloud gaming [21], aprovechando las ventajas que presenta su arquitectura. Sin embargo, existe poca información acerca de la comparación de los protocolos de comunicación para SDN. Ante esto, como menciona [12] [22], surge la necesidad de analizar los protocolos de comunicación que se emplean para la implementación de una red SDN. Actualmente, se están desarrollando investigaciones sobre las redes SDN [16], por lo cual es necesario inducir nuevas líneas de investigación. En el trabajo de Valencia [16] se menciona que, las SDN se han convertido en un foco de interés para investigaciones, debido a las ventajas que ofrece este nuevo paradigma frente a las redes tradicionales.

Sin embargo, las SDN no son una idea nueva, puesto que desde el siglo pasado se ha tratado de introducir la ejecución de comandos sobre la red [23]. Con la llegada de la SDN, la administración de la red se la realiza desde un software, por lo cual surgen amenazas, esto como consecuencia de problemas en el diseño de software y/o fallas de implementación [24]. Ante lo mencionado anteriormente, se considera a las SDN como un eje para posibles ataques. Por lo cual, es importante realizar un análisis de la arquitectura, controladores y/o

protocolos, y demás componentes de las redes SDN, con el fin de minimizar estos riesgos.

La selección adecuada de un protocolo para redes basadas en software permite optimizar la comunicación entre el plano de control y el plano de datos. Además, realizar un análisis comparativo de los protocolos de comunicación, va permitir alternativas a la hora de la implementación de una SDN. Conocer las ventajas y desventajas que brindan los protocolos permite a los desarrolladores y administradores de red optimizar tiempos en el desarrollo o administración de una SDN. Tener un protocolo adecuado para una determinada red, genera beneficios no solo al desarrollador o administrador de una red, sino a la sociedad que va emplear o navegar sobre esa infraestructura.

Este documento está estructurado de los siguientes capítulos, en el capítulo 1, se encuentran antecedentes históricos y conceptuales que sustentan el desarrollo del trabajo. En el capítulo 2, se revisan los protocolos de comunicación y herramientas que se van a emplear para la comparación. En el capítulo 3 se describen los resultados del análisis de los protocolos seleccionados y puestos a pruebas. Finalmente, en el capítulo 4 se establecen las conclusiones de la investigación.

CAPÍTULO 1: MARCO REFERENCIAL

El desarrollo de este capítulo permite conocer el estado del arte sobre las redes SDN, y en especial sobre los protocolos de comunicación empleados en la conexión entre el plano de control y datos. Para ello, se realizó un Mapeo Sistemático de Literatura, con el objetivo de conocer de manera general sobre las redes SDN y de forma particular sobre los protocolos de comunicación empleados entre los planos de control y datos

1.1. Antecedentes históricos de la Investigación

1.1.1. Objetivos del estudio

Objetivo general

- Analizar los protocolos de comunicación southbound en una arquitectura SDN mediante el uso de herramientas de emulación Mininet para optimizar la comunicación entre los planos de control y datos.

Objetivos específicos

- Implementar una topología de red SDN para el análisis de los protocolos de comunicación southbound.
- Realizar un cuadro comparativo de los protocolos de comunicación southbound empleados en redes SDN.
- Validar el cuadro comparativo mediante la emulación de los protocolos de comunicación en una herramienta de emulación.

1.1.2. Mapeo Sistemático de Literatura

En el desarrollo del presente mapeo sistemático de literatura se considera como guía el siguiente protocolo propuesto por Eduardo Benavides [25], el cual se describe en la siguiente ilustración 1:

Ilustración 1: Protocolo para realizar un para realizar un RSL



Fuente: Elaborado por el autor

Definir las preguntas de investigación

RQ1: ¿Los protocolos SouthBound pueden mejorar la comunicación en una arquitectura SDN?

RQ2: ¿Cuáles son los protocolos que permiten la comunicación entre el plano de control y de datos?

RQ3: ¿Cuáles son las métricas que permiten evaluar los protocolos de comunicación en redes SDN?

Definir criterios de inclusión y exclusión para la RSL

Para la selección de los estudios se consideraron los títulos y resúmenes correspondientes en los que se denotará los siguientes criterios de inclusión y exclusión:

Tabla 1: Criterios de inclusión y exclusión para RSL

CRITERIOS DE INCLUSIÓN	CRITERIOS DE EXCLUSIÓN
<ul style="list-style-type: none"> - Estudios en el campo de las redes y telecomunicaciones - Estudios que abarquen información sobre los protocolos de comunicación en redes SDN - Estudios sobre nuevas tecnologías - Artículos publicados desde 2015 - Artículos científicos y conferencias. 	<ul style="list-style-type: none"> - Estudios no incluidos en las bases de datos seleccionadas - Estudios duplicados <p>Otros criterios:</p> <ul style="list-style-type: none"> - Para estudios duplicados en congresos y revistas, se dará prioridad a los de revistas.

Fuente: Elaborado por el autor

Identificar las bases de datos y motores de búsqueda

Para obtener información adecuada y verídica, se considera estudios publicados en revistas y congresos de alto impacto, bases digitales de editores de ciencias de la computación, tales como:

- IEEE Computer Society (<http://www.computer.org/>)
- ACM (<http://dl.acm.org/>)
- Springer (<http://www.springerlink.com/>)
- Science Direct (<https://www.sciencedirect.com>).

Definir los términos de búsqueda

La definición de la cadena de búsqueda se realizó con la identificación de los términos claves, comunes y sus sinónimos. Para lo cual, se buscaron palabras relacionadas con SDN, protocolos de comunicación, protocolos Southbound en los motores de búsquedas anteriormente definidas. Para un mejor control de los resultados, a los términos de búsqueda se agregaron operadores lógicos: OR para agregar sinónimos de términos y AND para agregar atributos. La cadena definida para la búsqueda es la siguiente:

(comparison and (Communication protocol OR Southbound APIS) AND SDN)

Buscar en bases de datos científicas

Tabla 2: Resultados de las cadenas de búsqueda

Cadena de búsqueda sugerida	Resultados	
(comparison and (Communication protocol OR Southbound APIS) AND SDN)	Google Scholar	1640
	Science Direct	18
	IEEE Explore	98

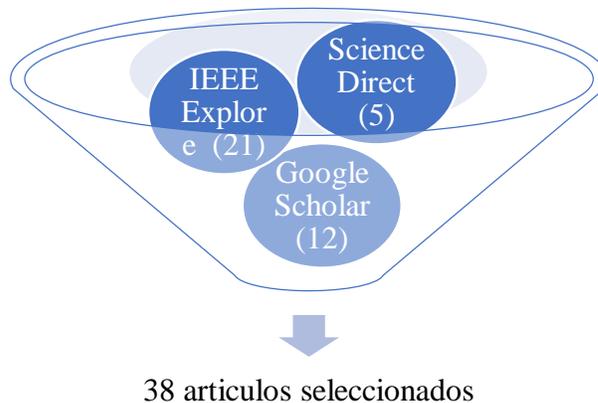
Fuente: Elaborado por el autor

Como se puede observar en la tabla anterior, los resultados obtenidos mediante la cadena de búsqueda definida anteriormente arrojan una cantidad de estudios adecuados para considerarlos para la fase de revisión.

Fases de Revisión

En la fase de revisión, se realizó la selección de los estudios que cumplan con los criterios de inclusión y exclusión, además considerando los filtros que fueron anteriormente definidas (ver Ilustración 2). Finalmente, mediante la lectura del artículo completo se obtuvo como resultado un listado de 38 estudios primarios.

Ilustración 2: Filtro aplicado a los resultados de estudios primarios



Fuente: Elaborado por el autor

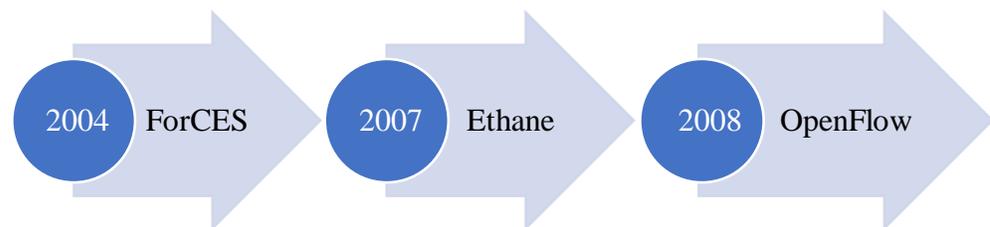
Extracción de Datos

La extracción de los datos, se realizó llenando una matriz con los datos más relevantes de los estudios primarios. Considerando los siguientes atributos: título, abstract, revista, DOI, año, autores.

Presentación de Resultado

Las infraestructuras TI han crecido en tamaño, complejidad y heterogeneidad [26], por lo cual es necesario administrar sin afectar el modelo del negocio. Es evidente que desde años atrás se ha considerado que “administrar una gran red es demasiado costoso” [9], mucho más con el excesivo uso de la red y cambios de patrones. En los últimos años ha sido notorio el incremento de varios servicios como, el comercio electrónico, redes sociales, aplicaciones móviles, cloud computing, entre otros [27]. Esto ha motivado la reestructuración de las redes tradicionales para mejorar su control, siendo una gran alternativa, hacer programable la red [12] por medio de su centralización en un controlador.

Ilustración 3: Propuesta para separación del plano de control y datos



Fuente: Elaborado por el autor

Como consecuencia de esto, empiezan a surgir varias propuestas para separar el plano de control y datos (Ilustración 3). A continuación, se describe tres de los más conocidos:

En el año 2004, surge Forwarding and Control Element Separation con la intención de separar el control y reenvío de datos [28]. FORCES es parte de un trabajo presentado por el IETF (Internet Engineering Task Force).

Otra de las alternativas que se presenta es Ethane, desarrollado por la Universidad de Stanford en 2006 [22]. Se considera como una arquitectura de red para empresas [29]. Ethane se basa en la misma idea de tener un controlador centralizado que permita la admisión y enrutamiento de los flujos de datos.

Finalmente, surge OpenFlow con su primera versión, la cual fue publicada en el 2008. OpenFlow se basa en el mismo paradigma de ForCes que es la separación del plano de control y datos. Ante esto, surgen las redes definidas por software o denominadas SDN por sus siglas, que tiene como objetivo la separación del plano de control y el plano de datos [15] [16]. Las redes SDN se componen sobre una arquitectura de 3 planos, las cuales son: plano de aplicación, plano de datos y plano de control. Las redes SDN no son una idea nueva, puesto que desde el siglo pasado se han venido concibiendo la idea de generar comandos para administrar la red [23].

Para el 2011, nace ONF (Open Networking Foundation) [22], una comunidad fundada por: Deutsche Telekom, Facebook, Google, Microsoft, Verizon, y Yahoo.

Las redes basadas en software, se han convertido en una arquitectura innovadora en redes [12]. Esto ha llevado a ser un “importante impulso en el desarrollo de nuevos protocolos y proyectos de investigación e innovación en la infraestructura de redes de información” [30].

Las redes SDN han ido creciendo en varios aspectos como “escala, complejidad e importancia, por lo que resulta necesario estudiar y evaluar el comportamiento de las aplicaciones, protocolos y servicios que se implementen para estas redes” [31].

En la mayoría de trabajos que emplean la arquitectura SDN [21], [27], [32]–[34], se habla del uso de los protocolos OpenFlow, dejando de considerar otros protocolos como ForCes. Para la actualidad, existen varios protocolos

que pueden ser empleados para la comunicación entre el plano de control y de datos, entre algunos están: ForCes, OVSDDB, NETCONF Y BGP.

En el transcurso de los años, y considerando el avance que ha tenido las redes SDN surge la necesidad de analizar los protocolos de comunicación [12] [22], lo cual ha motivado a generar nuevas líneas de investigación. Las SDN son un nuevo foco de investigación para la academia, como menciona Valencia [16] son un foco de interés para conocer las ventajas que ofrece este nuevo paradigma frente a las redes tradicionales. Es importante analizar cada uno de los componentes de la red, en especial los protocolos y API que son empleados para la comunicación entre los diferentes planos.

1.2. Antecedentes conceptuales

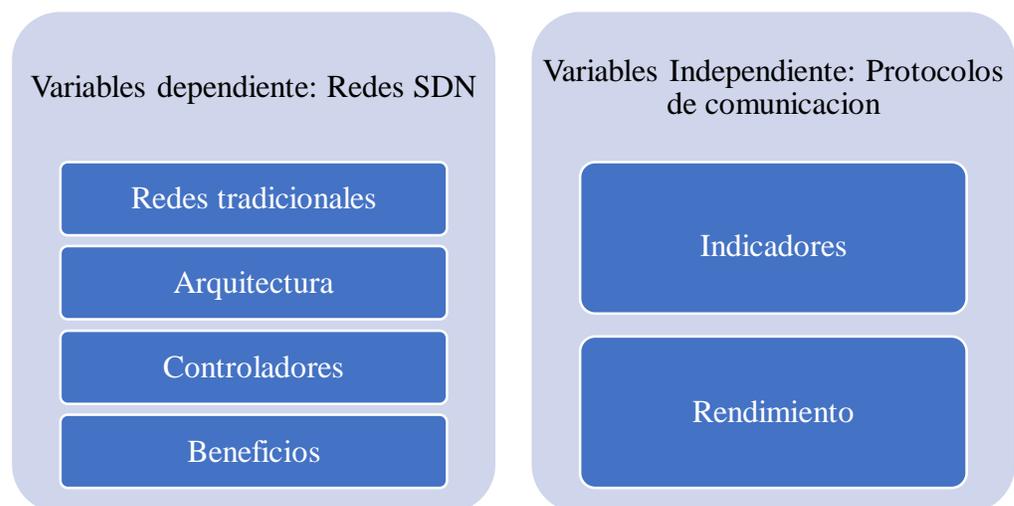
Hipótesis de Investigación

Se ha definido la hipótesis de la investigación considerando la variable dependiente e independiente:

SI se selecciona un adecuado protocolo de comunicación para una red SDN ENTONCES se puede lograr optimizar la comunicación entre el plano de control y datos.

Red de Categorías de las variables

Ilustración 4: Categorías de las variables



Fuente: Elaborado por el autor

Redes tradicionales

A las redes tradicionales se las considera como un conjunto de elementos independientes (dispositivos, host, etc) y a su vez relacionados entre sí, que transfieren datos entre ellos bajo estándares y protocolos [23].

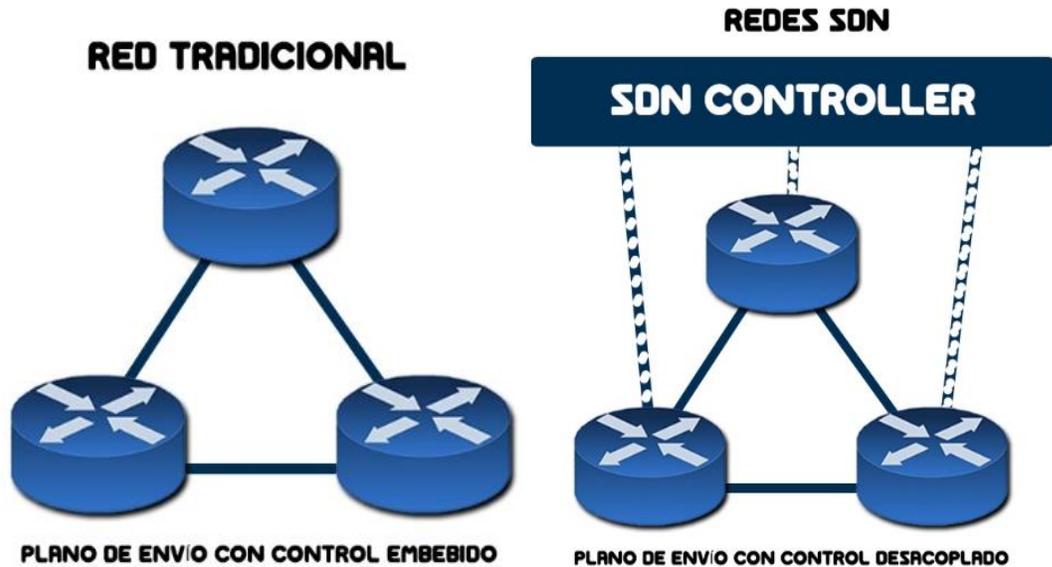
Una de las desventajas de las redes tradicionales es la compleja administración (Ver Tabla 1), condicionando su reconfiguración a probables fallas y cambios [35]. Como menciona [36], las redes tradicionales tienen limitada adaptación a las nuevas tecnologías, escalabilidad e ineficiente uso de políticas de control de acceso a la red.

Las redes tradicionales están configuradas en base a políticas predefinidas, las cuales no pueden responder eficazmente a cambios [30]. En la actualidad, las redes tradicionales crecen de forma rápida, siendo muy complicado administrarlas [37] debido a la integración de los planos de datos y control en el mismo equipo. Esta integración lleva a realizar las configuraciones directas en cada uno de los dispositivos que forman parte de la red, lo cual dificulta la adaptación eficiente de nuevos requerimientos.

Redes basadas en software

Las redes basadas en software o denominadas SDN por sus siglas, se han convertido en una arquitectura innovadora en redes. Esto ha llevado a ser un “importante impulso en el desarrollo de nuevos protocolos y proyectos de investigación e innovación en la infraestructura de redes de información” [30]. La comunidad ONF (Open Networking Foundation) es la encargada de generar recomendaciones, documentación y estandarización de las redes SDN [22].

Ilustración 5: Diferencia entre Red tradicional y SDN



Fuente: Elaborado por el autor

Como se observa en la ilustración 5, la principal diferencia entre una red tradicional y SDN es el desacoplamiento del plano de control y datos. En las redes tradicionales el plano de control y datos vienen encapsuladas en los dispositivos de red. A diferencia del paradigma de las SDN que es el desacoplamiento de estos planos, lo que permite convertirla en una red programable y administrable. Mendoza-Rubio destaca en su trabajo [38], que las SDN son una base para el control de las redes móviles de próxima generación. Sin embargo, se debe considerar políticas [4] para llevar un control adecuado.

Una de las grandes ventajas de las SDN, es permitir la programación de las redes (Ver Tabla 3) con los requerimientos exactos del usuario mediante el control de la configuración. Una red tradicional o no centralizada limita el control de la red, por lo cual no permiten adaptarse a patrones de tráfico cambiantes [39]. Otro aspecto a analizar en cuanto a la acogida de las SDN por parte de los administradores u operadores de red, es el nacimiento de este nuevo paradigma como consecuencia de la gran cantidad de dispositivos

conectados en la red, y además del aumento de los servicios en la nube [40] destacando el éxito del internet.

Tabla 3: Redes SDN vs Redes tradicionales

CARACTERÍSTICAS	REDES SDN	REDES TRADICIONALES
Programación de la red	✓	X
Control centralizado	✓	X
Fácil implementación	✓	X
Mejor gestión de la red	✓	X
Flexibilidad de la red	✓	X
Control complejo de la red	X	✓

Fuente: [10]

Las redes SDN permiten el control de la red debido a desacople de las funciones de reenvío de los dispositivos de red lo cual hace que la red sea programable.

Esta nueva tendencia permite a los administradores de red una fácil configuración, administración y optimización de los recursos de red mediante software.

Las redes tradicionales son complejas de administrar y de elevado costo [40] lo que ha llevado a muchas organizaciones a implementar SDN, como una solución óptima a estos problemas. Como se menciona en el trabajo de [19], estos problemas son incluso derivados por la propia arquitectura que se emplean para el desarrollo de soluciones debido a la funcionalidades del equipo.

Las SDN han venido incursionando en muchas áreas de desarrollo de nuevas soluciones tecnológicas. Un claro ejemplo es el trabajo de [18], el cual emplea

los principios de las redes SDN para mejorar la seguridad en una granja inteligente, considerando la virtualización de los equipos que forman parte de su diseño.

En resumen, las redes SDN crean un cerebro centralizado (controlador), para permitir una mejor comunicación y administración de toda la red. De este modo, se puede responder de forma óptima a los cambios que se generen, ya sea por nuevas políticas del negocio o por necesidad de satisfacer nuevas demandas.

Las SDN han venido a revolucionar el mundo de las redes de datos, y como se menciona en el trabajo de [41], algunos casos de éxitos son:

- Empresa de Transporte en Metro - East Japan Railway Company, presente en el área este de Japón hasta el área metropolitana de Tokyo.
- Hospital Universitario - Nagoya City University Hospital, en Nagoya Japón.
- Datacenter Interno de NEC.
- Empresa Nippon Express.
- Empresa Genesis Hosting Solutions.
- Hospital Universitario - Kanazawa University Hospital, en Kanazawa Japón.

Arquitectura y elementos de una SDN

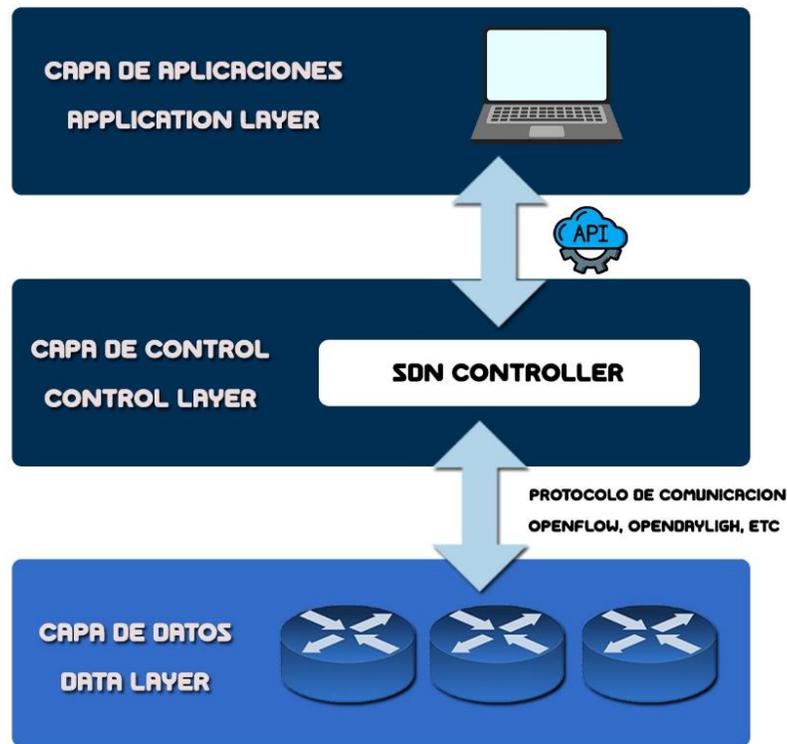
La arquitectura SDN desacopla el plano de control y reenvío de datos en la red [38] [42] [24], creando un controlador centralizado. Esto permite, como se menciona en los trabajos de [13] [39], que la red sea programable y fácil de administrar.

Como se ha expuesto anteriormente, el paradigma SDN permite administrar o programar el comportamiento de la red en base a las propias decisiones que determine el administrador de la red [41]. Esto permite que se pueda redireccionar (conmutar o enrutar) paquetes a un puerto determinado. Las

redes SDN se constituyen en una arquitectura por planos, lo cual hace que se convierta en una red programable.

Como se observa en la ilustración 6, una arquitectura SDN se despliega en tres planos: aplicación, control y datos.

Ilustración 6: Arquitectura SDN



Fuente: Elaborado por el autor

Plano de aplicaciones: Está compuesta por aplicaciones diseñadas por usuarios para personalizar y administrar las redes [16]. Cabe destacar que para la comunicación entre el plano de aplicaciones y el de control se emplean APIs conocidas como NorthBound APIs,

Plano de control: Es el plano intermedio que permite la toma de decisiones y programación de la tabla de flujo para el control de los dispositivos de red [16]. Para la comunicación con el plano de datos, se emplean protocolos o interfaces de comunicación denominadas SouthBound APIs, una de las empleadas es OpenFlow.

Plano de datos: Está compuesta por los dispositivos de red que componen una topología. Estos van a ser controlados por medio del controlador [16].

En resumen, el funcionamiento de esta arquitectura se da de la siguiente forma: Desde el plano de aplicaciones se definen las configuraciones de la red, las cuales son comunicadas al controlador SDN (plano de control), el cual se encarga de tomar decisiones adecuadas. Estas decisiones son comunicadas a los dispositivos de red que se encuentran en el plano de datos o infraestructura, empleando las SouthBound APIs o interfaces de comunicación.

La arquitectura SDN se basa en 3 principales principios detallados en el trabajo de [43], los cuales son:

1. Desacoplamiento del plano de control y plano de datos.
2. Control centralizado.
3. Programación de los servicios de red.

Controladores SDN

Los controladores SDN son parte esencial de una arquitectura SDN. Estos controladores son un nexo entre el cliente y los dispositivos de red [43] para proveer servicios. Estos controladores permiten interactuar por medio de las interfaces Northbound (NBI). Tanta es la importancia de los controladores, que en el trabajo de Centeno [7] se lo considera como el “centro de la arquitectura de las SDN”, debido a que es el controlador quien gestiona el flujo de comunicaciones entre las aplicaciones y los dispositivos físicos. En el mismo trabajo de Centeno, destaca que por medio de estos controladores se puede obtener datos estadísticos o eventos.

En el trabajo del estado de arte de [22], se considera el análisis de los protocolos de comunicación como una línea base para generar investigación sobre las SDN. En el mismo trabajo menciona [22], que en los últimos años se ha empezado a realizar investigación de la SDN con respecto a la arquitectura, controladores o protocolos, aplicaciones y/o reglamentos.

Algunos de los controladores de código abierto son: Beacon, Floodlight, NOX, POX, Ryu, Trema y OpenDayLight (ODL).

Beneficios de las redes SDN

Como se ha venido analizando a lo largo de este capítulo, las SDN son un nuevo paradigma que se presenta como una innovadora arquitectura en redes. Por lo cual, genera beneficios para los negocios y administradores de redes. A continuación, se describe algunos de los beneficios de las redes SDN:

- **Redes programables:** este beneficio es gracias al desacoplamiento del plano de control y reenvío de datos.
- **Control centralizado de la red:** Este control centralizado es proporcionado al controlador SDN, el cual es el componente esencial de una red SDN, puesto que es el centro de inteligencia de la red.
- **Reducción de Gastos de capital y operativos:** Las redes SDN permite la reducción de gastos en hardware, gracias a los algoritmos que permiten un mejor control de los dispositivos de la red. Lo cual permite escalar y automatizar la red.
- **Agilidad y flexibilidad para implementar nuevas aplicaciones:** este nuevo paradigma permite un fácil y rápido despliegue de nuevas aplicaciones y/o servicios en la red.

Protocolos de comunicación hacia el sur

Conocidos también como SouthBound API y en otros casos como API de comunicación ascendente y descendente. Estos protocolos permiten la comunicación (Intercambio de mensajes) entre los elementos del plano de control y datos [17], en la tabla 4 se definen cada una de ellas.

En el trabajo de [44], se menciona la necesidad de tener un protocolo de comunicación que permita acceder a los elementos de la red. Ante esto surgen varios protocolos y uno de los más empleados es OpenFlow [45]. Este protocolo es destacado por ser guiado por la ONF (Open Networking

Foundation). Sin embargo, existen otros como ForCES, SoftRouter, LIST, PCEP, NETCONF/YANG y OpFlex [41].

OpenFlow, a pesar de ser el protocolo de comunicación más empleado en SDN, partió de la misma idea de ForCES (Forwarding and Control Element Separation), que tenía como objetivo el permitir que los elementos de control se comuniquen con los elementos de la red [44]. Pese a ser propuesto para ser un protocolo estándar no logró consolidarse a nivel de los proveedores.

Tabla 4: Protocolos de comunicación - SDN

ITEM	PROTOCOLO	DESCRIPCIÓN
1	ForCES [46]	Su funcionamiento se basa en una arquitectura maestro-esclavo donde los elementos de reenvío (FE) son esclavos y permiten que el elemento de control maestro (CE) los controle. Esto quiere decir, que FE está a cargo de procesar y manejar los paquetes, mientras que CE se encarga de la administración y ejecución del enrutamiento de los paquetes.
2	BGP [47]	Se lo clasifica como protocolo de ruta vectorial o un protocolo de vector de distancia. Se basa en su tabla de enrutamiento con las direcciones que se puede alcanzar, y se las van asociando a una métrica de costo (valor que representa la conexión a cada router).
3	OpenFlow [48]	Es un protocolo a través del cual un controlador lógicamente centralizado puede controlar un interruptor OpenFlow. El enrutamiento de los paquetes se basa en las tablas de flujo (uno o más) de cada conmutador OpenFlow.
4	OVSDB	El protocolo de administración OVSDB usa JSON. Permite que las aplicaciones se conecten a la base de datos Open vSwitch, en donde se encuentra la configuración.
5	NetConf [49]	Proporciona mecanismos para instalar, manipular y eliminar la configuración de los dispositivos de red. Emplean el lenguaje XML (Extensible Markup Language) para los datos de configuración y los mensajes de protocolo. Tienen como objetivo la reducción de la complejidad y mejora del rendimiento de la red.

Fuente: Elaborado por el autor

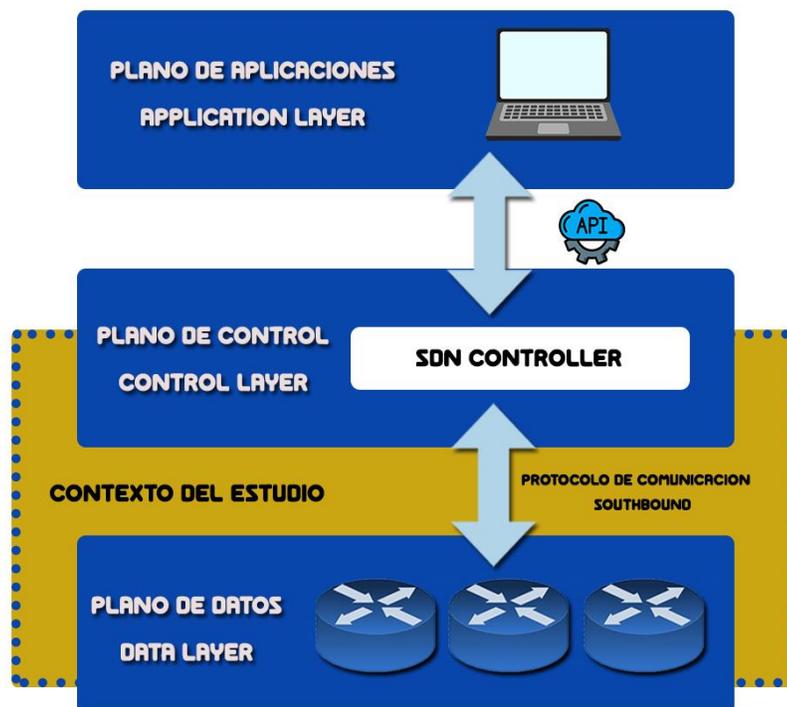
Métricas de protocolos de comunicación

Para analizar un protocolo de comunicación se emplean métricas, con las cuales pueden obtener resultados cuantitativos y cualitativos. En el trabajo de [31], para analizar el protocolo de comunicación OpenFlow se emplean como métricas cuantitativas las siguientes: tiempo de respuesta, tiempo de activación del protocolo y duración de una entrada en la tabla de flujo. Además, se emplean métricas cualitativas como disponibilidad, capacidad, rendimiento, eficiencia y escalabilidad.

1.3. Antecedentes contextuales de la investigación

Delimitación del contexto del Estudio

Ilustración 7: Contexto del Estudio



Fuente: Elaborado por el autor

Considerando que el estudio de las Redes SDN es muy amplio, el presente trabajo se enfoca en describir los protocolos de comunicación Southbound como se destaca en la Ilustración 7 y realizar un análisis mediante un entorno

controlado para la emulación de una red SDN, considerando los siguientes lineamientos:

- Seleccionar una herramienta de emulación que soporte una arquitectura SDN.
- Describir los protocolos de comunicación Southbound que se emplean en una arquitectura SDN.
- Seleccionar los protocolos de comunicación Southbound que soporte la herramienta de emulación seleccionada.

Propuesta de solución y contribuciones

Para cumplir con el objetivo de este estudio se ha definido el siguiente proceso de solución:

- **Recopilación de información:** Obtener información sobre los protocolos de comunicación Southbound y sus características. Además, conocer el funcionamiento de la arquitectura de una red SDN.
- **Diseño de escenario para laboratorio:** Seleccionar una herramienta de emulación que permita la ejecución de una arquitectura SDN. Seleccionar adecuadamente los protocolos que cumplan los criterios de la herramienta de emulación seleccionada. Finalmente, diseñar la topología en la herramienta.
- **Pruebas y análisis de resultados:** Se realiza la ejecución de la topología y la obtención de los datos para el análisis de los resultados.

La principal contribución de este estudio es generar un artículo académico que evidencie los resultados obtenidos en el emulador Mininet. De tal forma, que se detalle el procedimiento para la implementación de una red SDN.

CAPÍTULO 2 MATERIALES Y MÉTODOS

En este capítulo, se describe la metodología y todos los materiales utilizados para el desarrollo de este estudio. A continuación, se detallan el tipo y enfoque de investigación (Ilustración 8) empleados para la consecución de los objetivos planteados.

Ilustración 8: Tipo y diseño de investigación



Fuente: Elaborado por el autor

2.1. Tipo de Estudio o Investigación

Para el presente estudio acerca de los protocolos de comunicación Southbound empleados en una arquitectura de red definida por software, se alineó al tipo de investigación exploratoria y descriptiva. El enfoque exploratorio [50] se aplica, para examinar un tema o problema de investigación poco estudiado. Ante esto, se obtuvo información de los protocolos de comunicación existentes para redes SDN.

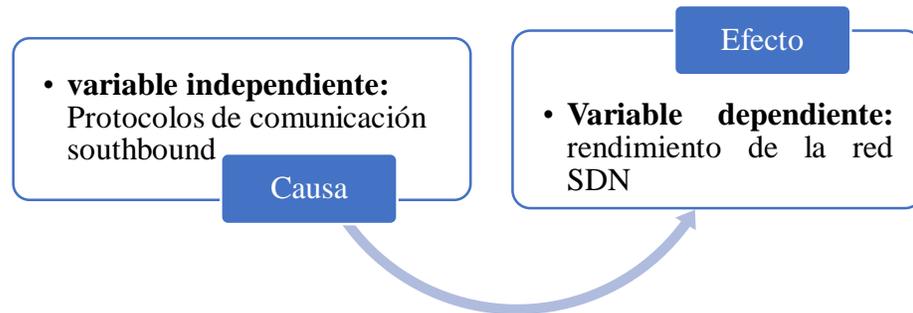
El enfoque descriptivo [50], permite describir personas, proceso, objeto o cualquier otro fenómeno. Por lo tanto, en este estudio se emplea el enfoque descriptivo para especificar las características de los protocolos de comunicación para redes SDN.

2.2. Paradigma o enfoque

Cuantitativo: por medio de este enfoque se pretende recolectar datos sobre la cual se probó la hipótesis. Para seguir ese enfoque se partió por definir una idea general, del cual se han derivado los objetivos y preguntas de investigación. Y posteriormente se ha revisado la literatura sobre la cual se generó el marco teórico.

Causi experimental: el estudio se basa en el análisis de los protocolos de comunicación southbound por lo cual se requiere manipular la variable independiente para conocer los efectos en el rendimiento de la red en una arquitectura SDN como se observa en la Ilustración 9.

Ilustración 9: Causa – efecto de las variables de investigación

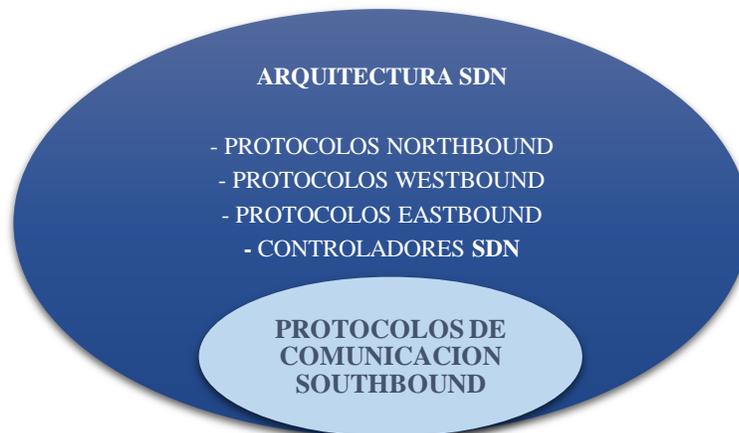


Fuente: Elaborado por el autor

2.3. Cálculo de la población y muestra

En una arquitectura SDN los planos de aplicación, control y datos, están enlazados por medio de protocolos de comunicación o APIs: SouthBound y NorthBound entre las más conocidas. Sin embargo, también se incluyen protocolos de comunicación Westbound y Eastbound como en la arquitectura mostrada en el trabajo de Valencia [16]. Por lo cual, como se muestra en la ilustración 10, la muestra de este estudio fue los protocolos de comunicación SouthBound considerando que la población fueron los componentes de la arquitectura SDN.

Ilustración 10: Población y muestra del estudio

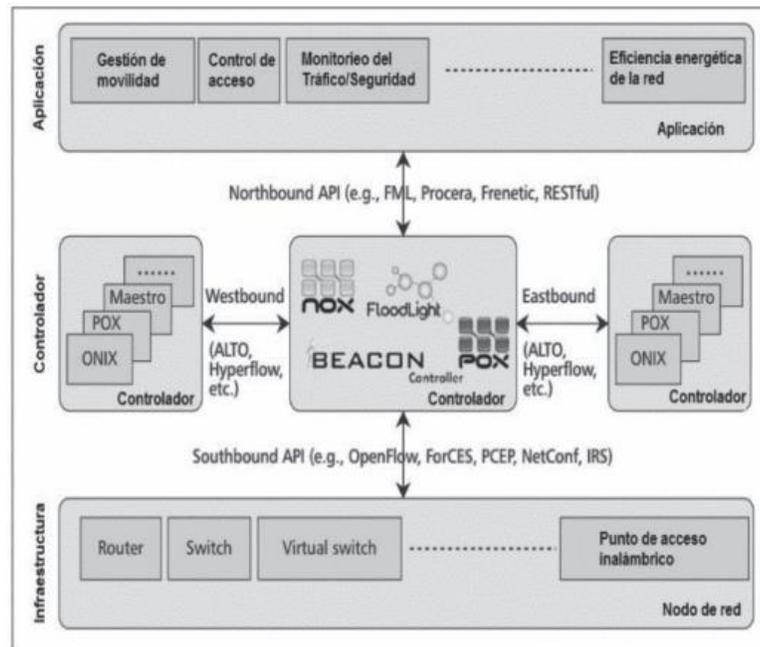


Fuente: Elaborado por el autor

Se ha considerado seleccionar esta parte de la arquitectura debido a:

- Escaso estudio centrado en los protocolos Southbound los cuales son empleados para la comunicación entre el plano de control y datos.
- Importancia de la comunicación para el envío de paquetes entre los planos de control y datos.

Ilustración 11: Arquitectura de red SDN



Fuente: Tomado del trabajo de Valencia [16]

Como se puede observar en la Ilustración 11, los protocolos de comunicación southbound permiten el envío de información entre el plano de control (controlador) y el plano de datos (elementos de red). Son considerados como una interfaz que permite que el controlador establezca el comportamiento de los equipos de red [51]. Para lo cual, se realizó una exploración sobre bases teóricas que permitió conocer los protocolos de comunicación southbound existentes. No obstante, hay que mencionar que no solo existen protocolos de comunicación (ver Tabla 5), sino APIs que permiten esta comunicación entre los planos antes mencionados.

2.4. Métodos teóricos con los materiales utilizados

Comparativo: se ha empleado el método comparativo para analizar el rendimiento de la red considerando la tasa de transferencia en el envío de paquetes ICMP entre hosts aplicando los dos protocolos de comunicación seleccionados (OPENFLOW y OVSDB).

Experimental: para poner a prueba la hipótesis del estudio, se ha empleado el método experimental. Por medio de este método se pretende observar las variables de investigación, es decir, si los protocolos de comunicación pueden incidir en el rendimiento de una SDN.

2.5. Métodos empíricos con los materiales utilizados

Como se observa en la Ilustración 12, se ha planteado un proceso sistemático que consta de tres fases para el cumplimiento del desarrollo de la emulación de una SDN aplicando los protocolos de comunicación Southbound. A continuación, en el gráfico se resume cada fase:

Ilustración 12: Proceso sistemático del estudio



Fuente: Diseñado por el autor

Recopilación de Información

En esta fase se detalla el resumen del análisis bibliográfico sobre los protocolos de comunicación Southbound. Los protocolos que se detallan en la Tabla 4, han sido recopilados por medio de la revisión de la literatura. En

dicha tabla se describen varios de los protocolos de comunicación empleados en las redes definidas por software.

Además, se ha diseñado una tabla (Tabla 5) con los diferentes protocolos Southbound en la que se indican si es un protocolo, API o framework. Además, se detalla bajo qué controlador puede ser implementado.

Tabla 5: Resumen de Protocolos Southbound

	Protocolo	API	Framework	Controlador Compatible
ForCES	X		X	Vendedor específico
OpenFlow	X			Openaylight, RYU, ONOS, Openvirtex, POX
OVSDB	X			Openaylight, RYU, ONOS, Opencontrail
Netconf		X		Openaylight, RYU, ONOS, Opencontrail
Xmpp		X		Opencontrail
BGP LS	X			Openaylight
BGP	X			Opencontrail

Fuente: Diseñado por el autor

Diseño de escenario para laboratorio

En esta fase, se realizó una revisión teórica de Mininet para conocer los controladores que puede soportar en la implementación de una arquitectura SDN.

Tabla 6: Protocolos Southbound soportado por Mininet

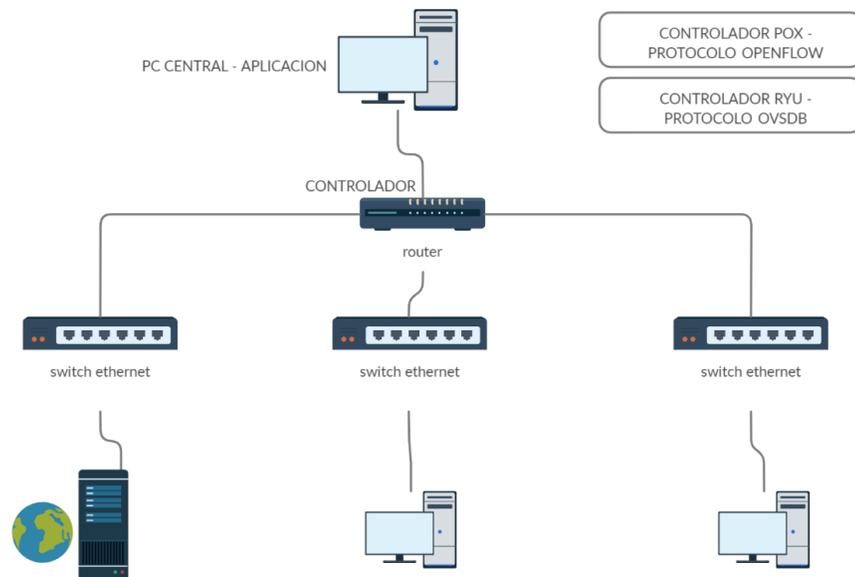
Protocolos	Controlador Compatible	Soportado por MININET
ForCES	Vendedor específico	
OpenFlow	Openaylight, RYU, ONOS, Openvirtex, POX	X
OVSDB	Openaylight, RYU, ONOS, Opencontrail	X

Fuente: Diseñado por el autor

Como resultado de la revisión se obtuvo que OpenFlow y OVSDB están soportados por la herramienta de emulación Mininet (Tabla 6).

Además, en esta fase se diseñaron dos topologías para realizar el análisis de los protocolos. Para el primer caso, se ha diseñado una topología lineal, que consiste conectar un switch a cada host como se observa en la ilustración 13. La topología propuesta se adapta a la arquitectura SDN y sobre la cual se realizan las pruebas de los protocolos de comunicación southbound.

Ilustración 13: Diseño de la topología (Lineal)

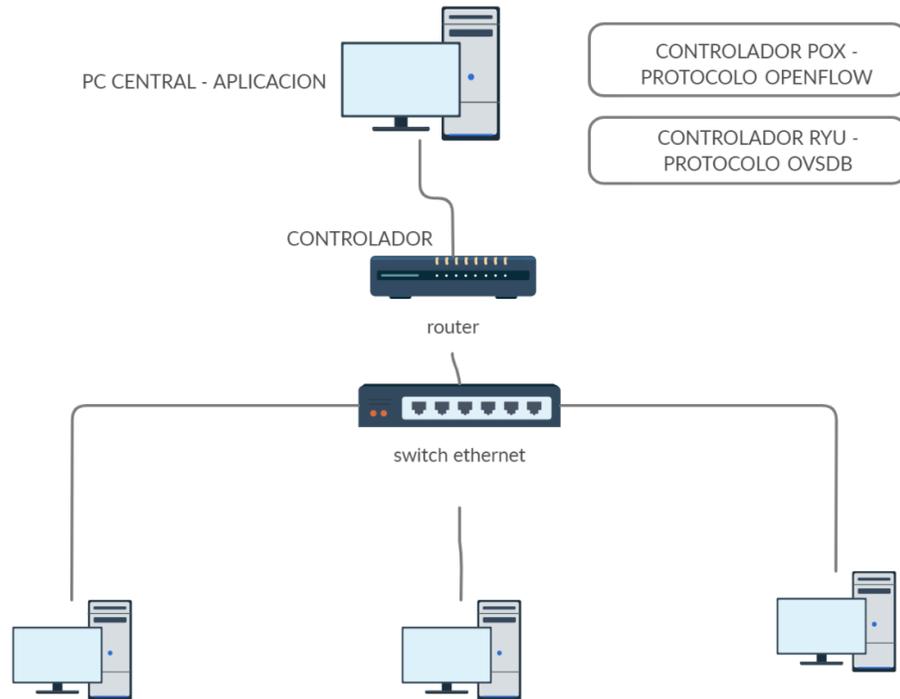


Fuente: Diseñado por el autor

La topología lineal se empleó para experimentar con los protocolos seleccionados en los controladores adecuados.

Para comprobar los datos que se obtuvieron con los protocolos de comunicación puestos a prueba en la topología antes mencionada, se ha diseñado una topología alternativa, la cual es un diseño simple, que consiste en crear una red con varios hosts conectados a un switch como se observa en la ilustración 14.

Ilustración 14: Diseño de la topología alternativa (Simple)



Fuente: Diseñado por el autor

Pruebas y análisis de resultados

En esta fase final se implementó la red SDN y se procedió a realizar las pruebas. La información de este apartado se describe en el capítulo 3 de este documento.

2.6. Técnicas estadísticas para el procesamiento de datos obtenidos

Para el procesamiento de los datos obtenidos por medio de la herramienta de emulación, se ha considerado emplear las medidas de tendencia central, el cual es la media aritmética. Esta técnica permitió agrupar en valores promedios los valores en milisegundos que se obtuvieron en los envíos de paquetes ICMP.

Por lo cual, se aplicó la técnica de observación, que permite explorar y describir atentamente el fenómeno, hecho o caso, por lo tanto, es una técnica adecuada para determinar el comportamiento de los protocolos de comunicación.

Al realizar el estudio en dos escenarios distintos con el propósito de validar los resultados, se considera la técnica de casos de estudio. Con esta técnica se pretende obtener más datos que permitan realizar una comparación adecuada de los protocolos que se estudiaron.

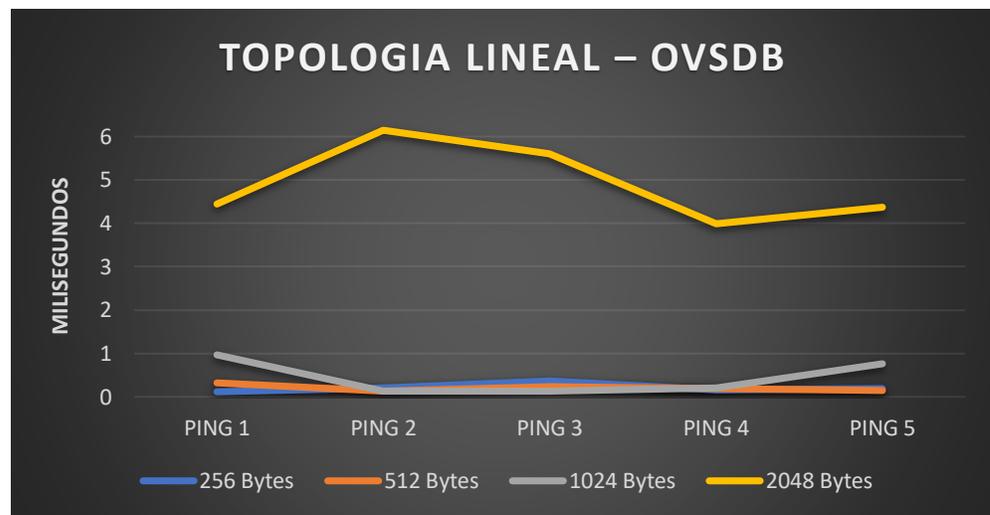
CAPÍTULO 3: RESULTADOS

Siguiendo el proceso del estudio, en este capítulo se describen los resultados que se han obtenido mediante la emulación de la red SDN. En esta sección se presentan los resultados obtenidos de la implementación de los dos protocolos de comunicación SouthBound (OVSDB y OpenFlow), estos protocolos fueron puestos a prueba en dos topologías diferentes.

3.1. Resultados

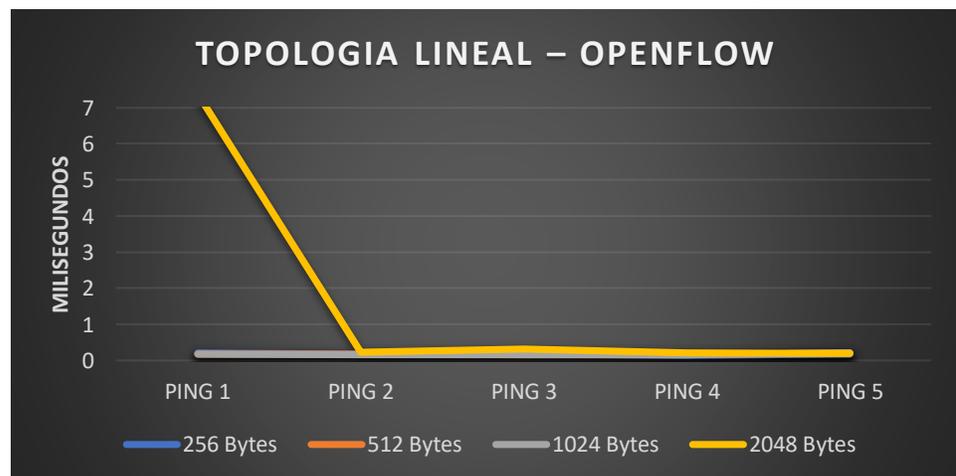
Una vez realizadas las pruebas en la topología lineal, aplicando el proceso que se describió en el capítulo anterior, se han obtenido los siguientes resultados:

Ilustración 15: Resultado de OVSDB con topología lineal



Fuente: Elaboración propia (2020)

Ilustración 16: Resultado de OpenFlow con topología lineal

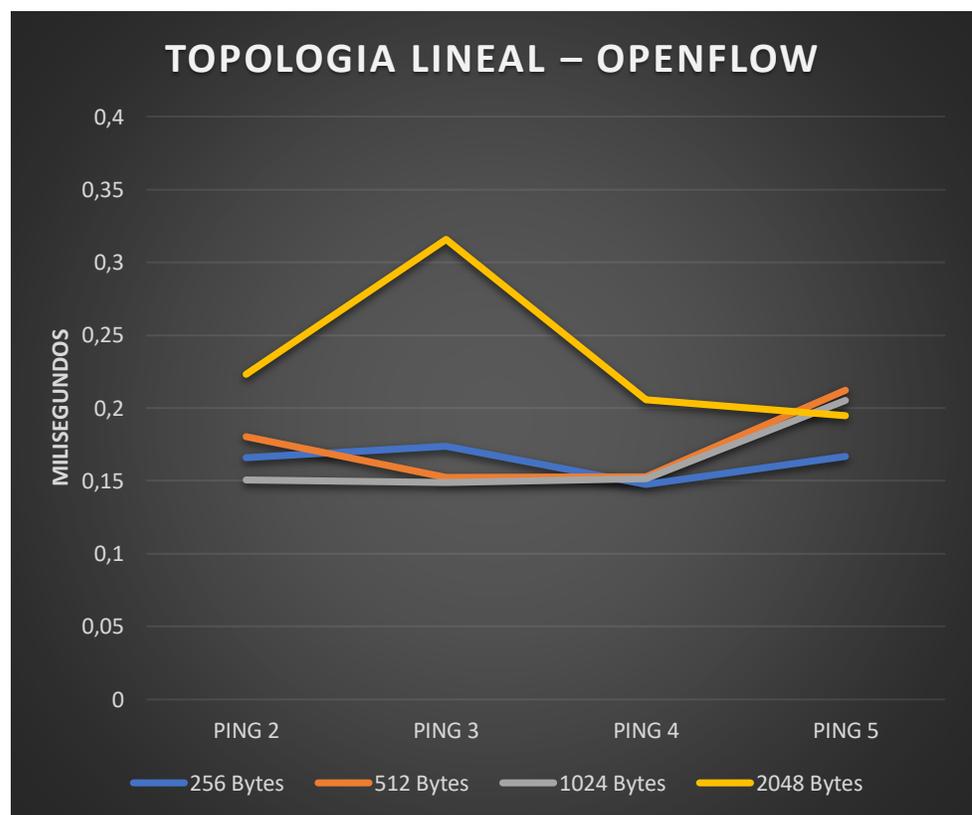


Fuente: Elaboración propia (2020)

Como se puede observar en las ilustraciones 15 y 16, los resultados de la implementación de los protocolos de comunicación hacia el sur OpenFlow y OVSDB, el primero presenta mejores resultados con respecto a la tasa de transferencia en el envío de paquetes con diferentes tamaños (256, 512, 1024 y 2018 bytes) entre hosts que conforman la red.

En el caso del protocolo OVSDB (ilustración 15), se han obtenido valores de hasta 6 milisegundos, a diferencia de OpenFlow que presenta mejores valores, como los de 1 milisegundo.

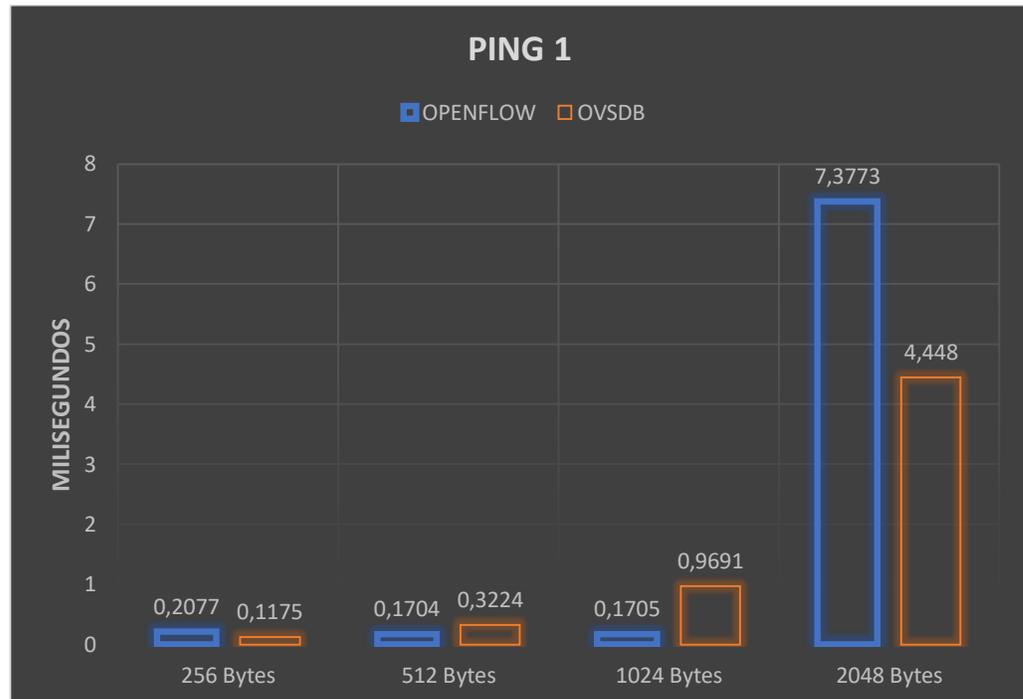
Ilustración 17: Resultados con la topología Lineal con OpenFlow



Fuente: Elaboración propia (2020)

Para mejorar la visualización de los resultados del protocolo OpenFlow, se han omitido los valores del primer envío de paquetes ICMP, y como se observa en la Ilustración 17, los resultados obtenidos en la topología propuesta no superan los 0.35 milisegundos.

Ilustración 18: Resultados con diferentes tamaños de paquetes ICMP



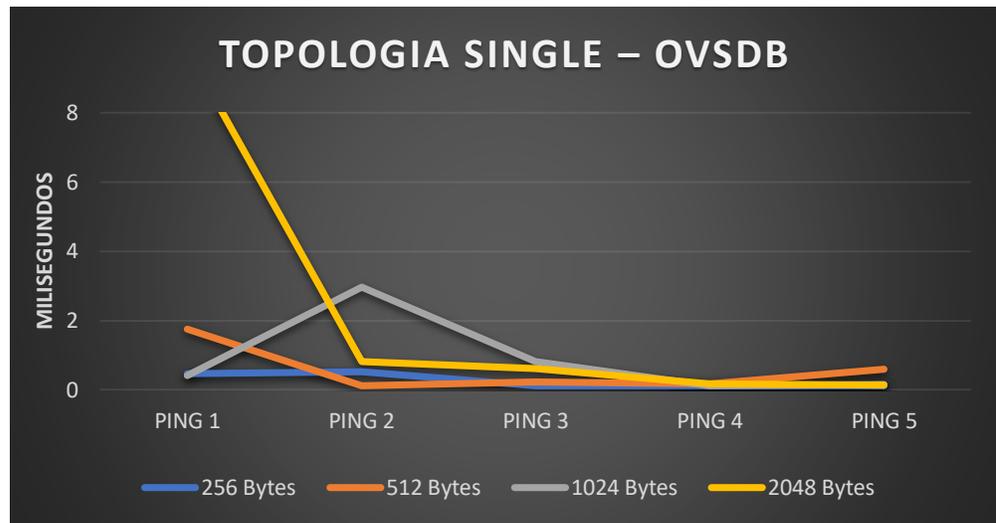
Fuente: Elaboración propia (2020)

Otro aspecto importante a destacar, es el comportamiento de estos protocolos, en el envío de paquetes con carga de 2048 Bytes (Ilustración 18) en su primer envío, toma valores altos, es decir, mayor latencia en relación a los siguientes paquetes que se envían, y esto como consecuencia del tiempo de convergencia de la red SDN. A esto, hay que considerar que en la topología lineal en su diseño se agrega un Switch para cada Host o grupo de hosts lo que provocaría el aumento de la latencia.

Sin embargo, el protocolo OpenFlow a pesar de presentar un retardo inicial alto, en el transcurso de los envíos de paquetes se estabiliza hasta su convergencia, en comparación con del protocolo OVSDB que en envíos de paquetes de tamaño 2048 bytes se mantiene con valores iguales a los del primer envío.

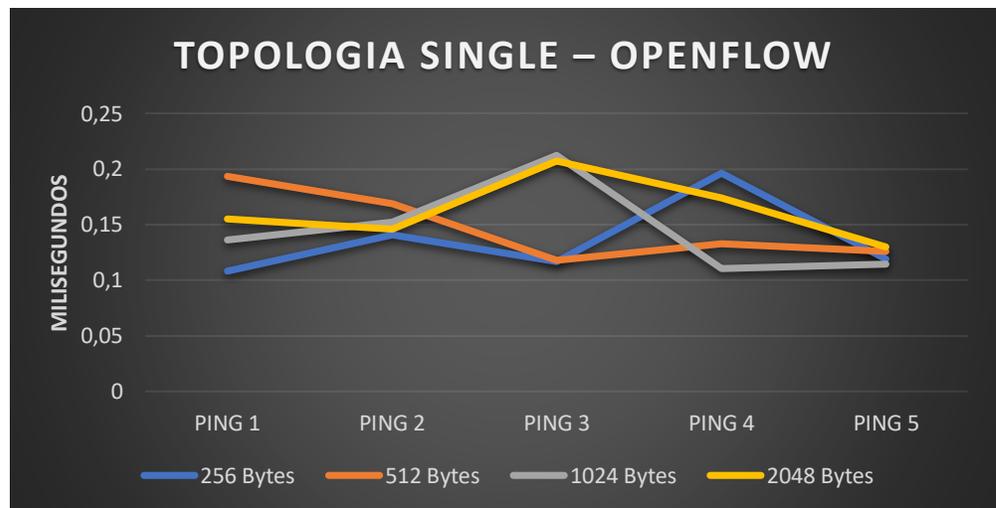
Para validar este primer resultado, se consideró aplicar una topología diferente, para lo cual se selecciona la topología Simple (Single) que proporciona la herramienta Mininet. Con este diseño se obtuvieron los siguientes resultados:

Ilustración 19: Resultado de OVSDB con topología single o simple



Fuente: Elaboración propia (2020)

Ilustración 20: Resultado de OpenFlow con topología single o simple



Fuente: Elaboración propia (2020)

Para este caso, con la topología Simple, se obtuvieron resultados similares a los obtenidos en la topología lineal. OpenFlow se presenta como un protocolo que se adapta a ese diseño. Como se puede observar en la Ilustración 19, el protocolo hacia el sur OVSDB alcanza un valor por encima de los 8 milisegundos (9,815ms) para el primer envío de paquetes ICMP de tamaño 2048 bytes. Por otra parte, el protocolo OpenFlow (ver Ilustración 20) presenta valores debajo de los 0.25 milisegundos en todos los envíos realizados con los distintos tamaños de paquetes.

CAPÍTULO 4: DISCUSIÓN DE LOS RESULTADOS

Las pruebas en diferentes topologías de los protocolos de comunicación SouthBound permiten tener alternativas para la implementación de una red SDN y conseguir la comunicación entre el plano de control y datos.

Como se puede observar en los resultados obtenidos, estos protocolos pueden inferir en un mejor rendimiento en el envío de paquetes entre hosts en una red SDN, sin embargo, este puede variar dependiendo de la topología sobre la que se implemente. Tener una topología simple con un switch puede proporcionar un mejor rendimiento en el envío de los paquetes, esto como consecuencia de que los hosts se encuentran conectados al mismo Switch y la tabla de conmutación es única haciendo el proceso de envío de paquete óptimo. A diferencia de una topología lineal, en la que se le agrega un switch para la conexión de cada host o grupo de hosts de la red, y que por tanto agrega latencia para procesar el paquete desde el switch origen hacia el controlador, y posteriormente al switch del destino para que reenvíe al host receptor. Con este análisis, se deduce que el diseño de la topología es importante para la aplicación de un determinado protocolo de comunicación y la implementación de la SDN.

Con esta validación, por medio de los datos obtenidos en la emulación de la red SDN, se comprobó la hipótesis planteada para este estudio, los protocolos de comunicación southbound empleada entre el plano de control y datos pueden optimizar la comunicación y/o envío de paquetes entre los dispositivos de red.

Otro hallazgo que se pudo evidenciar en este estudio son los resultados obtenidos con el protocolo OpenFlow para la comunicación entre los planos de control y datos. En OpenFlow, el tiempo para procesar las cabeceras de los paquetes de datos es mejor a diferencia de OSVDB, por lo que, al observar los resultados, la tasa de transferencia mejora con el uso del protocolo OpenFlow en las dos topologías diseñadas para la prueba.

Cabe destacar que este estudio abarca una parte de la arquitectura SDN: los protocolos de comunicación southbound. Los componentes de la arquitectura SDN, como se ha mencionado en el presente documento, son varias, por lo cual este estudio motiva a realizar análisis y prueba de los demás componentes de la arquitectura SDN.

CONCLUSIONES

El resultado de este análisis permitió evidenciar la buena respuesta del protocolo de comunicación OpenFlow, considerando que pruebas de conexión se han realizado entre los dispositivos de red y el controlador empleando como parámetro los envíos de paquetes ICMP entre los hosts. Esto ha llevado a considerar que los protocolos pueden optimizar la comunicación entre los planos de datos y control.

De este estudio se ha generado un listado de protocolos de comunicación que son empleados en una red SDN, el mismo que permitió diferenciar los protocolos de comunicación y APIs, además de conocer sobre qué controladores son soportados.

El ambiente de prueba, la herramienta Mininet, fue la adecuada para realizar un análisis de la red SDN por todas las prestaciones que dispone haciendo una herramienta útil para emular redes SDN, sin embargo, se puede mejorar las pruebas con equipos reales (dispositivos y controladores adecuados).

Los diseños de las topologías utilizadas han sido las apropiadas para poner a prueba una red SDN considerando aspectos como el número de hosts, servicios web y controlador.

Mininet tiene gran capacidad para la implementación de una arquitectura SDN, otorga varios comandos necesarios y fáciles de usar para la implementación y configuración de una SDN. Además, permite la ejecución de diversas instrucciones desde las consolas de los hosts creados

RECOMENDACIONES

Realizar análisis y comprobar otros protocolos hacia al sur considerando el mismo controlador o varios controladores para los protocolos que se han puesto a prueba.

Para la implementación de una red con dispositivos físicos, se debe tener en cuenta que los controladores soportan los protocolos seleccionados.

Usar los recursos que proporciona el sitio web de Mininet, el cual para este estudio ha funcionado correctamente.

Combinar las dos topologías empleadas en este estudio para analizar el comportamiento de los protocolos de comunicación incluso de los mismos controladores que se emplean, se pueden también incluir topologías personalizadas.

Realizar análisis y pruebas de los componentes de la arquitectura SDN como son controladores, protocolos hacia el norte (NorthBound) y demás protocolos que se vayan a emplear, además considerar pruebas de seguridad entre los planos de control y aplicaciones.

BIBLIOGRAFÍA

- [1] Y. Fernández Torres, M. Gutiérrez Fernández, y R. Palomo Zurdo, «¿Cómo percibe la banca cooperativa el impacto de la transformación digital?», *CIRIEC-Esp. Rev. Econ. Pública Soc. Coop.*, n.º 95, p. 11, abr. 2019, doi: 10.7203/CIRIEC-E.95.12724.
- [2] E. Virrueta y M. Millán, «Brecha digital en los Negocios», *Ecorfan J.*, vol. 3, n.º 7, pp. 851-860, 2012.
- [3] M. Peña Casanova, C. Anías Calderón, M. Peña Casanova, y C. Anías Calderón, «Sistema para ejecutar políticas sobre infraestructuras de Tecnologías de la Información», *Ingeniare Rev. Chil. Ing.*, vol. 27, n.º 3, pp. 479-494, sep. 2019, doi: 10.4067/S0718-33052019000300479.
- [4] M. Peña y C. Anías Calderón, «Empleo de modelos de información en arquitectura modificada para gestión de redes y servicios basada en políticas», vol. 39, pp. 77-88, oct. 2018.
- [5] J. Céleri-Pacheco, J. Andrade-Garda, y S. Rodríguez-Yáñez, *Cloud Computing para PYMEs*. Machala: Universidad Técnica de Machala, 2018.
- [6] «Ministerio de Telecomunicaciones y de la Sociedad de la Información – Ecuador». <https://www.telecomunicaciones.gob.ec/> (accedido jun. 27, 2020).
- [7] A. G. Centeno, C. M. R. Vergel, C. A. Calderón, y F. C. C. Bondarenko, «Controladores SDN, elementos para su selección y evaluación.», *Telemática Mag.*, vol. 13, n.º 3, Art. n.º 3, nov. 2014.
- [8] A. A. L. Carbonell, «APLICACIÓN SDN PARA EL CONTROL DE ACCESO A LA RED».
- [9] S. Latifis, «Análisis al futuro de las arquitecturas de internet», *Rev. Ing. USBMed*, vol. 2, n.º 1 (Enero-Junio), p. 18, 2011.
- [10] K. Benzekki, A. El Fergougui, y A. Elbelrhiti Elalaoui, «Software-defined networking (SDN): a survey», *Secur. Commun. Netw.*, vol. 9, n.º 18, pp. 5803-5833, dic. 2016, doi: 10.1002/sec.1737.
- [11] C. E. M. Gilces y R. P. Villamar, «Aplicación de Balanceo De Carga Dinámico Para Servidores, Basada En Redes Definidas Por Software.», *RISTI Rev. Ibérica Sist. E Tecnol. Informação*, n.º 32, pp. 67-82, 2019.
- [12] Y. A. Muro, F. Alvarez Paliza, y A. Carbonell, «COLABORACION DE IMS Y SDN-OPENFLOW. UNA ARQUITECTURA PARA MITIGAR PROBLEMAS DE SEGURIDAD EN REDES FUTURAS», *Rev. TONO*, vol. 13, pp. 34-39, jun. 2016.
- [13] G. C. Pérez y M. F. Marín, «Redes definidas por software: Solución para servicios portadores del Ecuador», *INVESTIGATIO*, n.º 6, pp. 41-63, 2015.
- [14] S. P. Pejathaya, «A SOFTWARE DEFINED NETWORK APPROACH ON NETWORK ROUTING ALGORITHM», vol. 2, n.º 6, p. 9, 2020.
- [15] D. F. B. Gómez, «OPENFLOW: EL PROTOCOLO DEL FUTURO», *Rev. Académica E Inst.*, n.º 93, pp. 61-72, 2013.

- [16] B. Valencia, S. Santacruz, L. Y. Becerra, y J. J. Padilla, «Mininet: una herramienta versátil para emulación y prototipado de Redes Definidas por Software», *Entre Cienc. E Ing.*, vol. 9, n.º 17, pp. 62–70, 2015.
- [17] A. Kumari y A. S. Sairam, «A Survey of Controller Placement Problem in Software Defined Networks», *ArXiv190504649 Cs*, may 2019, Accedido: jul. 11, 2020. [En línea]. Disponible en: <http://arxiv.org/abs/1905.04649>.
- [18] E. Barrientos-Avendaño, D. Rico-Bautista, L. A. Coronel-Rojas, y F. R. Cuesta-Quintero, «Granja inteligente: Definición de infraestructura basada en internet de las cosas, IPv6 y redes definidas por software», *Rev. Ibérica Sist. E Tecnol. Informação*, n.º E17, pp. 183–197, 2019.
- [19] L. R. Amondaray, F. J. A. Fuentes, y C. A. Calderón, «Redes de Sensores Inalámbricos Definidas por Software», *Rev. Ing. Electrónica Automática Comun. ISSN 1815-5928*, vol. 41, n.º 2, pp. 39–50, 2020.
- [20] F. W. S. Navarro, J. G. Bustos, y W. E. C. Hernández, «Adaptive video transmission over software defined networks», *Visión Electrónica*, vol. 13, n.º 1, Art. n.º 1, feb. 2019, doi: 10.14483/22484728.14398.
- [21] F. D. Trejos y N. Alzate, «Cloud gaming: a survey», *Entre Cienc. E Ing.*, vol. 10, n.º 20, pp. 82-91, dic. 2016.
- [22] M. Á. Barrera Pérez, N. Y. Serrato Losada, E. Rojas Sánchez, y G. Mancilla Gaona, «State of the art in software defined networking (SDN)», *Visión Electrónica*, vol. 13, n.º 1, pp. 178-194, ene. 2019, doi: 10.14483/22484728.14424.
- [23] P. Alcívar y M. Navia, «Comparativa entre red tradicional y red definida por», p. 13, 2020.
- [24] T. Alharbi y M. Portmann, «The (In)Security of Virtualization in Software Defined Networks», *IEEE Access*, vol. 7, pp. 66584-66594, 2019, doi: 10.1109/ACCESS.2019.2918101.
- [25] E. Benavides, «Revisión Sistemática de la Literatura (RSL). De la Teoría a la Práctica», jul. 06, 2020.
- [26] M. Peña y C. Anías Calderón, «Modelo para la gestión de infraestructuras de tecnologías de la información», *TecnoLógicas*, vol. 23, pp. 31-53, may 2020, doi: 10.22430/22565337.1449.
- [27] J. Carlos, D. Mejia, y I. Bernal, «Implementación de un Prototipo de una Red Definida por Software (SDN) Empleando una Solución Basada en Hardware», feb. 2014.
- [28] L. Yang, T. Anderson, R. Dantu, y R. Gopal, «Forwarding and Control Element Separation (ForCES) Framework», may 2004.
- [29] M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, y S. Shenker, «Ethane: Taking Control of the Enterprise», p. 12.
- [30] S. Manzano, J. Pallo, P. González, y A. Escobar, «Gestión de flujo de datos en una red definida por software en relación a variables externas», *UTCiencia Cienc. Tecnol. Al Serv. Pueblo*, vol. 3, n.º 2, pp. 73–84, 2017.
- [31] L. G. Picó, C. A. Calderón, S. B. Macías, y C. M. R. Vergel, «Evaluación de desempeño y configuraciones de las SDN mediante la simulación», *Rev. Téc.*, p. 29, 2016.

- [32] A. R. Ampuño Avilés y M. M. Chávez Cristóbal, «Diseño y simulación de una red de Datacenters basada en topología FAT-TREE en un ambiente de redes definidas por software (SDN).», B.S. thesis, 2015.
- [33] F. Ibáñez Moruno, J. A. Lévano Lévano, y E. S. Nieto Maldonado, «Diseño e implementación de una herramienta de visualización para análisis en tiempo real de redes SDN/OpenFlow», 2016.
- [34] J. C. Chico, D. Mejía, y I. Bernal, «Implementación de un Prototipo de una Red Definida por Software (SDN) Empleando una Solución Basada en Hardware», vol. 25, p. 10, 2014.
- [35] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, y S. Uhlig, «Software-Defined Networking: A Comprehensive Survey», *Proc. IEEE*, vol. 103, n.º 1, pp. 14-76, ene. 2015, doi: 10.1109/JPROC.2014.2371999.
- [36] I. Bernal y D. Mejía, «Las redes definidas por software y los desarrollos sobre esta temática en la Escuela Politécnica Nacional», *Rev. Politécnica*, vol. 37, n.º 1, pp. 43–43, 2016.
- [37] J. Machado, A. F. Ramos, y J. C. Cuéllar, «Implementación de OpenFlow sobre NetFPGA», *Ing. Región*, vol. 11, pp. 57–71, 2014.
- [38] J.-M. Mendoza-Rubio, J. CARMONA-MURILLO, J.-L. González-Sánchez, J. Calle-Cancho, y D. Cortés-Polo, «SR-DMM: A SDN-Based DMM Solution For Future Mobile Networks», *IEEE Lat. Am. Trans.*, vol. 17, n.º 05, pp. 734-741, may 2019, doi: 10.1109/TLA.2019.8891941.
- [39] C. Trois, M. D. Del Fabro, L. C. E. de Bona, y M. Martinello, «A Survey on SDN Programming Languages: Toward a Taxonomy», *IEEE Commun. Surv. Tutor.*, vol. 18, n.º 4, pp. 2687-2712, Fourthquarter 2016, doi: 10.1109/COMST.2016.2553778.
- [40] J. H. Cox *et al.*, «Advancing Software-Defined Networks: A Survey», *IEEE Access*, vol. 5, pp. 25487-25526, 2017, doi: 10.1109/ACCESS.2017.2762291.
- [41] M. R. Giraldo y A. M. L. Echeverry, «Redes de datos definidas por software - SDN, arquitectura, componentes y funcionamiento», p. 7, 2018.
- [42] F. Hu, Q. Hao, y K. Bao, «A Survey on Software-Defined Network and OpenFlow: From Concept to Implementation», *IEEE Commun. Surv. Tutor.*, vol. 16, n.º 4, pp. 2181-2206, Fourthquarter 2014, doi: 10.1109/COMST.2014.2326417.
- [43] S. Schaller y D. Hood, «Software Defined Networking Architecture Standardization», *Comput. Stand. Interfaces*, vol. 54, ene. 2017, doi: 10.1016/j.csi.2017.01.005.
- [44] A. Lara, A. Kolasani, y B. Ramamurthy, «Network Innovation using OpenFlow: A Survey», *IEEE Commun. Surv. Tutor.*, vol. 16, n.º 1, pp. 493-512, First 2014, doi: 10.1109/SURV.2013.081313.00105.
- [45] A. Bianco, V. Krishnamoorthi, N. Li, y L. Giraudo, «OpenFlow driven ethernet traffic analysis», en *2014 IEEE International Conference on Communications (ICC)*, jun. 2014, pp. 3001-3006, doi: 10.1109/ICC.2014.6883781.

- [46] E. Haleplidis *et al.*, «Network Programmability With ForCES», *IEEE Commun. Surv. Tutor.*, vol. 17, n.º 3, pp. 1423-1440, thirdquarter 2015, doi: 10.1109/COMST.2015.2439033.
- [47] «Estudios de caso BGP - Cisco», 2008. https://www.cisco.com/c/es_mx/support/docs/ip/border-gateway-protocol-bgp/26634-bgp-toc.html (accedido nov. 07, 2020).
- [48] S. Denazis, E. Haleplidis, J. H. Salim, O. Koufopavlou, D. Meyer, y K. Pentikousis, «Software-Defined Networking (SDN): Layers and Architecture Terminology», 2015. <https://tools.ietf.org/html/rfc7426#section-4.3> (accedido nov. 07, 2020).
- [49] R. Enns, M. Bjorklund, y J. Schoenwaelder, «Network Configuration Protocol (NETCONF)». <https://tools.ietf.org/html/rfc6241> (accedido nov. 07, 2020).
- [50] R. Hernández Sampieri, C. Fernández Collado, y P. Baptista Lucio, *Metodología de la investigación*. 2014.
- [51] G. Pereira, J. Silva, y P. Sousa, «Comparative Study of Software-Defined Networking (SDN) Traffic Controllers», en *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, jun. 2019, pp. 1-6, doi: 10.23919/CISTI.2019.8760997.