

**DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL
PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y
OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACIÓN DE
MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.**

AUTORES:

CASTRO YANCY ALVARO DE JESUS

PESCA TAFUR CARLOS ALFREDO

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
VALLEDUPAR – CESAR
2022**

**DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL
PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y
OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACIÓN DE
MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.**

ESTUDIANTES:

CASTRO YANCY ALVARO DE JESUS

PESCA TAFUR CARLOS ALFREDO

**Trabajo de Grado presentado como requisito para optar al título de Ingeniero
de Sistemas**

DIRECTOR

DAVID ARTURO MANOTAS FERIAS

INGENIERO DE SISTEMAS

ESPECIALISTA EN ADMINISTRACIÓN DE BASES DE DATOS

ESPECIALISTA EN SISTEMAS DE TELECOMUNICACIONES

MAGÍSTER EN TELEMÁTICA Y TELECOMUNICACIONES

INSTRUCTOR CISCO

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
VALLEDUPAR – CESAR
2022**

Tabla de contenido

SECCIÓN I. Descripción General del Proyecto

- 1.1 Título del Proyecto
- 1.2 Dirección de ejecución
- 1.3 Lapso de ejecución
- 1.4 Organismo o institución responsable
- 1.5 Información de contacto
- 1.6 Línea, Sublínea y grupo de investigación del Proyecto

SECCIÓN II. Descripción Situacional

- 2.1 Estado del arte
- 2.2 Identificación del Problema
 - 2.2.1 Formulación del problema
- 2.3 Impacto del proyecto
- 2.4 Análisis de Participación
- 2.5 Objetivos del Proyecto
- 2.6 Justificación del Proyecto
- 2.7 Cronograma de Actividades del Proyecto

Bibliografía

Anexos

- Anexo A. Modelo Carta del director
- Anexo B. Modelo Carta de los estudiantes
- Anexo C. Modelo Evidencias de asesoría metodológica
- Anexo D. Modelo Carta de aval de entidad responsable
- Anexo E. Modelo Carta declaración antifraude
- Anexo F. Modelo Carta de derechos de autor
- Anexo G. Modelo Carta de compromiso de realizar un artículo científico
- Anexo H. Modelo Carta de declaración de la Universidad
- Anexo I. Imagen del Switch de red definido por software SEL-2740S

1. SECCIÓN I: DESCRIPCIÓN GENERAL DEL PROYECTO

1.1. TÍTULO DEL PROYECTO

DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACIÓN DE MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.

1.2. DIRECCIÓN DE EJECUCIÓN

Laboratorio de Telemática 102 bloque I. Universidad Popular del Cesar Valledupar, Cesar.

1.3. LAPSO DE EJECUCIÓN

El lapso de ejecución es de aproximadamente seis (6) meses

1.4. ORGANISMO O INSTITUCIÓN RESPONSABLE

Universidad Popular del Cesar, Facultad de Ingenierías y tecnologías, Programa de Ingeniería de Sistemas.

1.5. INFORMACION DE CONTACTO DE LOS ESTUDIANTES

NOMBRES	APELLIDOS	CEDULA	TELÉFONO	CORREO
Carlos Alfredo	Pesca Tafur	1065658671	3043632323	Carlpe93@gmail.com
Álvaro De Jesús	Castro Yancy	1079933507	3015107915	alvarocastro@gmail.com

1.6. LINEA Y SUBLINEA DE INVESTIGACION

1.6.1. Línea Sistemas de información.

Este proyecto seguirá la línea de investigación de Tecnologías de la Información y la comunicación, debido a que hace parte de la utilización de sistemas de telecomunicación y representa la aplicación de las redes en la actualidad. Aportando una investigación a través de un diseño en SDN usando protocolos openflow. Los cuales se explicarán durante la ejecución este proyecto.

1.6.2 Sublínea Desarrollo de sistema de información.

Este proyecto se enmarca en la Sublínea de investigación de Telecomunicaciones y teleinformática, el cual es un tema de relevancia en la actualidad de las redes de comunicaciones, aportando investigación, diseños y modelos del sistema en SDN o redes definidas por software. Buscando mostrar las opciones de administración y gestión en las redes de la actualidad, aplicándola en un laboratorio de telemática, teniendo en cuenta dimensiones aproximadas de un campus universitario.

SECCIÓN II. Descripción Situacional

2.1. ESTADO DEL ARTE

Las SDN o redes definidas por software, consisten en una arquitectura virtual. Donde se encuentran desacoplados los planos de control y los planos de datos de una red de comunicaciones. El surgimiento de este tipo de redes ha representado la actualización de las empresas y centro de telecomunicaciones de hoy en día, su concepto y aplicación generan la característica de escalabilidad dentro de las corporaciones e instituciones que ha futuro piensan crecer física y tecnológicamente (**autores**). Dentro del foco de investigaciones que se han realizado sobre esta tendencia a nivel nacional e internacional este proyecto ha encontrado las siguientes:

En el artículo titulado ***SDN y NFV, dos tendencias que revolucionarán las telecomunicaciones -2014***. De la revista digital mundo contacto, se menciona que el Software Definido por redes (SDN) consta de tecnologías que estarán definidas por un controlador, el cual centralizará mediante software la inteligencia de todas las funciones de las capas de la red dentro de un mismo dispositivo, logrando que las funciones de la red sean programables de una manera más ágil y fácil [1]. Dentro del sector empresarial, las tecnologías en SDN responde a la centralización de toda la administración del equipo de redes de comunicaciones y de tener un funcionamiento más lógico de la red, según las necesidades que tengan los usuarios.

De igual manera, la empresa Alcatel-Lucent Enterprise. está claramente ligada con los servicios de SDN, con el uso de las redes autónomas. mostrando los beneficios y las ventajas en la utilización de las redes definidas por software como un sistema de actualización, gestión y de seguridad en la red para las empresas filiales, brindando una infraestructura virtual que ofrece rendimiento, soporte y productividad. Además de,

ofrecer dispositivos tangibles capaces de integrar aplicaciones en SDN como el conmutador LAN apilable OmniSwitch 6860(E), siendo un ejemplo del uso de redes convergentes de alta velocidad y que permiten aplicaciones en SDN para centro de datos [2]

Con lo anterior, se refleja que las tecnologías en SDN son un sistema de tendencia actual que permite generar escalabilidad a nivel empresarial, los centros de datos físicos se interconectan con elementos de infraestructura virtual ofreciendo amplios manejos de trafico de información en la red. Así como, rendimiento y calidad de servicio (QoS). [2]

A NIVEL INTERNACIONAL

El artículo A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks. De la página científica francesa de archivos HAL, menciona que las redes definidas por software se desarrollaron para facilitar la innovación, al permitir un control programático simple de la ruta de datos de red, ya que, la separación del hardware de reenvío con el control lógico, permite el despliegue de nuevos protocolos y aplicaciones. Así como, la consolidación de varias cajas intermedias en el control de software. En lugar de aplicar políticas y ejecutar protocolos en una convolución de dispositivos dispersos, la red se reduce a un reenvío simple, donde el controlador de red es usado para la toma de decisiones [3].

Además, en el mismo artículo se explica que, aunque la comunicación de controlador a controlador es no definida por Openflow, es necesario para cualquier tipo de distribución o redundancia en el plano de control [3]. Un controlador físicamente centralizado representa un punto de falla para toda la red de datos. por lo tanto, Openflow

permite la conexión de múltiples controladores a un solo Switch, lo que permitiría que los controladores de respaldo se hicieran cargo en caso de una avería en el sistema.

En un primer estudio sobre los dispositivos físicos que hacen parte de una SDN, la **empresa SEL**, ofrece el dispositivo SEL-2740S. un switch físico para redes definido por software (SDN) con características de diseño que permiten mejorar el desempeño de Ethernet. Este switch proporciona una ingeniería de tráfico centralizado, puede manejar el controlador de flujo de red definido por software SEL-5056, con el cual generan un control en el flujo dentro de la red. Una de las ventajas más características de este dispositivo es que permite la tolerancia a fallos. Dado que, las topologías de diseño y las rutas de envío se basan en los requisitos de la aplicación en lugar de la optimización de protocolos administrativos de planos de control dinámicos, tales como el protocolo de árbol de expansión rápido (RSTP). Otra de las características es que soporta Openflow 1.3. brindando sincronización de tiempo, por el uso del protocolo PTP. **(Ver anexo I. presentación general SEL-2740S) [4].**

Por otro lado, en el artículo titulado **implementación de un Prototipo de una Red** Definida por Software (SDN) empleando una Solución Basada en Hardware, en el 2014. Para la XXV Jornadas en Ingeniería Eléctrica y Electrónica define la arquitectura SDN como tres grupos de dispositivos: Servidores, Controladores, Dispositivos intermedios, Hosts. Con cuerda este artículo en que “En el plano de control, la comunicación entre el o los servidores controladores y los conmutadores es posible mediante el protocolo OpenFlow, el cual permite establecer la comunicación inicial entre estos dispositivos, para posteriormente enviar la información relacionada a las reglas de flujo que se transmitirán desde la entidad controladora hacia los conmutadores.” [5].

Además, un artículo de la universidad de Toronto, titulado HyperFlow. Presentando una herramienta virtual basada en eventos para Openflow. Donde se presenta un diseño del Hyperflow, por medio de switches como elementos de reenvío, controladores de NOX como elementos de decisión controlando toda la red. Con la idea de presentar un sistema lógicamente centralizado, aunque sus estaciones físicas estén

distribuidas. Este artículo concluye que “HyperFlow permite a los operadores de red implementar cualquier número de controladores para ajustar el rendimiento del plano de control en función de sus necesidades. Además, mantiene centralizada la lógica de control de la red y localiza todas las decisiones a cada controlador para minimizar tiempo de respuesta del plano de control. La aplicación HyperFlow, implementada sobre NOX, sincroniza los controladores vistas de toda la red mediante la propagación de eventos que afectan al estado del controlador.” [6]

De la misma manera, se da un análisis parcial Openflow en un artículo entre la Universidad de Houston, Texas e infoblox Inc. de Santa Catalina, donde la gestión de un switch virtual permite mejores herramientas para la gestión de la red. Esta investigación representa el resultado de la implementación y evaluación de un administrador de red llamado NETMAN para gestionar de manera segura y eficiente los recursos de la red de comunicaciones. Teniendo en cuenta que el núcleo de NETMAN es el código HTML / JavaScript. El cual se ejecuta en el usuario final a través de un navegador web como Explorer, Netscape, Chrome, entre otros. Como conclusión “el comando OVSDB implementado en la OFConfig, permite eliminar, imprimir el destino u objetivos del controlador configurado. Además, crear, eliminar, enumerar un puerto(s); agregar un túnel, configurar la QoS, crear una ruta de datos, etc.” [7].

Por último, en el artículo investigativo Cisco ***impulsa un protocolo SDN alternativo a OpenFlow*** en el datacenter dynamics donde afirman que Cisco expuso su propio protocolo abierto para SDN como alternativa a OpenFlow, uniéndose a compañías de tecnología como Citrix, Microsoft, IBM, Canonical, RedHat, Embrane y F5, para colaborar y soportar el estándar, denominado OpFlex. Como una estrategia para brindar infraestructuras virtuales para la gestión de redes. Según el artículo de data center dynamics, la SDN de Cisco que utilizará OpFlex se denomina Application CentricInfrastructure. Demostrándose una vez más, de que las aplicaciones en SDN son

una tendencia positiva para los negocios y empresas que requieren economizar en plantas físicas de intercomunicación, abriéndose campo al manejo de la virtualización [8].

A NIVEL NACIONAL

Continuando con la investigación a nivel nacional se han encontrado los siguientes proyectos que abordan el tema de tendencia SDN con el uso de herramientas virtuales Openflow:

Según el artículo de la Universidad de Manizales titulado como ***OPENFLOW: el protocolo del futuro***. El cual describe que las SDN han despertado la creación de proyectos como Openflow. Refiriéndose a que este nuevo protocolo de comunicaciones se abre paso dentro de los protocolos tradicionales de red y transporte. Dentro de las conclusiones de este artículo se hace énfasis en que las SDN comienzan a tomar fuerza en un momento donde el *software* y el *hardware* están separados. [9].

Así como, el proyecto titulado **DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN DE RED BAJO LA ARQUITECTURA SDN**. De la Universidad JAVERIANA, para la facultad de ingeniería en la maestría en ingeniería electrónica. Donde se representa una red diseñada por software, a la cual se le generan pruebas y comparativas de rendimiento en modo Openflow, sobre los switches virtuales. Con el objeto de representar el comportamiento de la transmisión de datos, ancho de banda y retardos sobre una topología de red. El objetivo del proyecto fue disminuir el número de paquetes en las redes de telecomunicaciones para aplicaciones streaming, modificando los parámetros de enrutamiento y los encabezados de los paquetes transmitidos en la red

de telecomunicaciones. Demostrando que la arquitectura en SDN puede lograr la disminución de paquetes utilizando reglas de **multicast** y reglas de **secuencia** [10]

De igual manera, el proyecto de investigación de la Universidad Distrital de Francisco José de Caldas, titulado **Estudio para la implementación de un sistema de redes definida por software (SDN) para una red de área amplia (WAN)**. El cual se basa en arquitecturas de red híbridas que han visto una creciente popularidad durante años, añaden inteligencia centralizada basada en software que monitoriza, analiza y controla la red, permitiendo a los usuarios finales mezclar y combinar diferentes formas de conectividad (por ejemplo, MPLS, Internet, Ethernet, inalámbrico) y servicios en una red híbrida que puede proporcionar una combinación óptima de costo y rendimiento para cada ubicación y aplicación. [11].

REGIONAL

A nivel regional se encontró el proyecto **titulado “Diseño de Prototipo de Aplicación para Control de Admisión de Flujos en una Red SDN”**, como proyecto de grado de ingeniería en la Universidad del Norte. Donde se referencia que las redes definidas por software o SDN, permiten resolver problemas a los que se enfrentan las redes convencionales, tales como problemas de calidad de servicios (QoS) y seguridad, gestión de movilidad y escalabilidad. El diseño de prototipo es un control de admisión de flujo. El cual determina cuándo admitir o negar flujos entrantes a la red de acuerdo con la disponibilidad de recursos de la red. [12].

Así mismo, se encontró también el proyecto **titulado Algoritmo de enrutamiento multicamino UDIC en redes SDN**, basado en la selección de las rutas dependiendo del

ancho de banda y el retraso de los enlaces. El resultado final del proyecto es un sistema programado, que permite elegir la ruta más óptima en base a los costos de cada uno de los enlaces usando paquetes UDP. Aplicando el sistema de enrutamiento con base a la prioridad de paquetes, por medio de un controlador Ryu [13].

2.2 IDENTIFICACIÓN DEL PROBLEMA

Actualmente, la mayoría de las empresas desconocen las redes definidas por software o SDN. Tropezando con problemas de inversión a nivel tecnológico. Atribuyendo esto a distintos aspectos tales como gastos de mantenimiento, gastos en ampliación de estructuras físicas, gastos de personal. Otro de los problemas, es la existencia de una gran variedad de plataformas en SDN que podrían ayudar a solventar los problemas antes mencionados. Sin embargo, la selección de la más adecuada depende de su rendimiento.

Por otro lado, el laboratorio de Telemática del bloque i, no posee una herramienta virtual de simulación para evaluar el desempeño de las redes definidas por software (SDN) basadas en Openflow. De acuerdo con Marín Muro [14]. Es definido como una arquitectura de red muy dinámica, operativamente gestionable, y adaptable con otras tecnologías de red, de costo eficiente. Lo cual la hace ideal para las altas demandas de ancho de banda y la naturaleza dinámica de las aplicaciones actuales. Esta arquitectura desacopla el control de la red y la funcionalidad de reenvío de información permitiendo que el control de la red pueda ser completamente programable logrando que las aplicaciones y servicios de red se abstraigan de la infraestructura de red subyacente. Aprovechando que su planta física posee elementos tangibles como dispositivos finales, y dispositivos intermedios. Así, como una buena conectividad a Internet, se pretende la generación de este proyecto.

Además de lo anterior, en la comunidad académica asociada al laboratorio de Telemática Bloque I, no hay la intermediación requerida que permita aprovechar la capacidad de almacenamiento gestión y seguridad de la nube; hoy más necesaria si se tiene en cuenta la facilidad de comunicación guardando distancias físicas, por efecto de la

covid-19 o ahorro en infraestructura e inversión, lo cual permitiría brindar grandes beneficios para las empresas. Los contactos físicos no serían del todo necesarios y la presencialidad puede darse gradualmente o alternada; pero se mantienen intercambios para una comunicación eficaz.

2.2.1 Formulación del problema.

¿De qué manera el diseño de un modelo en infraestructura SDN usando el protocolo Openflow en las plataformas abiertas Onos y Opendaylight a través de la herramienta de configuración de mininet para laboratorios de redes y telecomunicaciones representará eficiencia, escalabilidad y QoS para cualquier tipo de red?

2.3. Impacto del proyecto

Social:

Este proyecto tiene un impacto social, ya que con la aplicación de plataformas virtuales se puede ofrecer mayores servicios al público en general. Además de, ampliar los servicios de atención a usuarios dentro y fuera de la región. Control de flujo informacional de relevancia como noticias, avisos, foros, capacitaciones.

Económico:

En el aspecto económico, este proyecto muestra un grado de economía empresarial, basado en disminución de gastos de adquisición, implementación, mantenimiento y configuración de dispositivos físicos necesarios para administrar el lote de información de la empresa. En vez de invertir en dispositivos o bienes tangibles, se puede redirigir los rubros de utilidad en otros recursos de gran valor para la entidad.

Tecnológico:

En relación con este tema, se usan aplicaciones de tendencia como plataformas Onos y Open daylight, redes de nueva generación presentándose como una solución que permite ampliar los recursos de ancho de banda, eficacia y gestión de red.

2.4 Análisis de Participación

Beneficiarios

Los beneficiarios directos son las empresas que desean aplicar el sistema de redes definidas por software, para manejar un tráfico de información más rápido y grandes volúmenes de información requeridos por la empresa.

Aliados

Los aliados directos para que se implemente este diseño, ingenieros de sistemas y electrónicos que operan en virtuales dentro de entornos en SDN, NGN o SDWAN. Empresas que ofrece el servicio: Aruba, Hewlett Packard, Cisco.

Opositores

Empresas que desconocen las ventajas del uso de las SDN y prefieren seguir usando infraestructuras tradicionales y poco eficientes.

2.5 Objetivos del proyecto

Objetivo General

Diseñar un modelo en infraestructura en SDN usando el protocolo Openflow en las plataformas abiertas Onos y Opendaylight a través de la herramienta de configuración de Mininet para medir eficiencia, escalabilidad y calidad de servicio en diferentes tipos de red de comunicación.

Objetivos específicos

- Identificar los requerimientos para la generación del diseño del modelo en infraestructura en SDN.
- Seleccionar el controlador más adecuado para la generación del modelo de infraestructura en las redes definidas por software.
- Establecer el tipo de red de comunicaciones a diseñar para analizar el modelo según el controlador seleccionado.
- Diseñar la arquitectura del sistema usando la herramienta de configuración de mininet sobre Centos.
- Desarrollar el modelo de infraestructura teniendo en cuenta niveles de escalabilidad, seguridad y QoS.

2.6 Justificación del proyecto

2.6.1 Justificación teórica.

El tema que plantea el proyecto de investigación se basa en investigaciones y conceptos generados de las redes definidas por software, sus alcances, beneficios y aplicaciones.

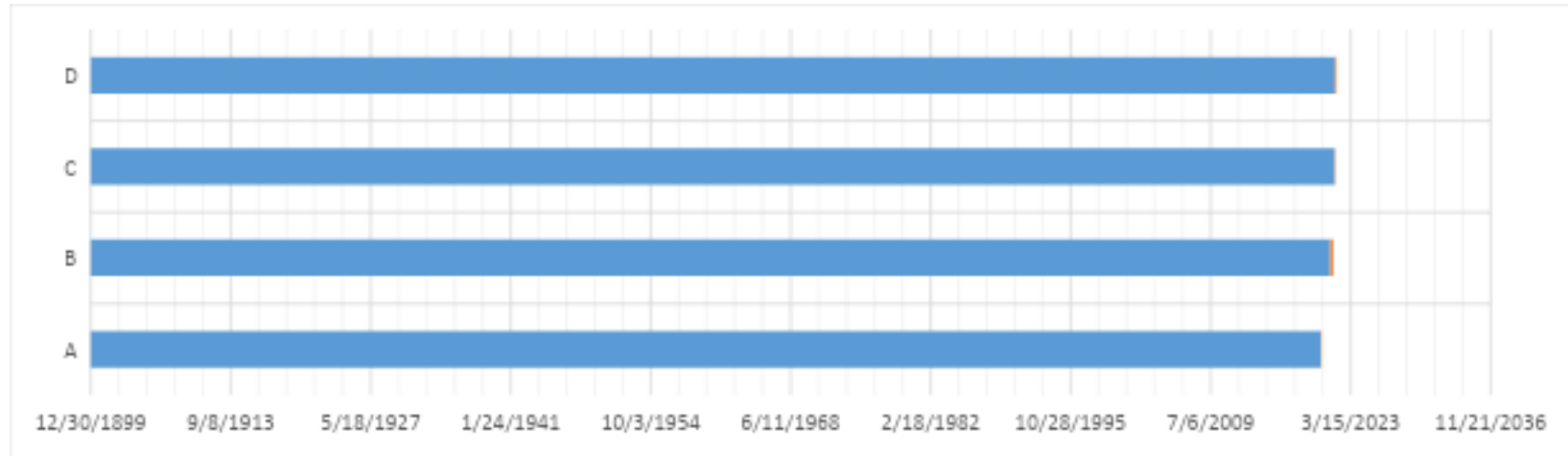
2.6.2. Justificación Práctica:

Este proyecto se pretende realizar porque existe la necesidad de mejorar el nivel de desempeño empresarial, y la generación de recursos que permiten ampliar las redes de comunicación dentro de las entidades de la región.

2.6.3. Justificación metodológica:

Esta investigación servirá como base para próximas investigaciones sobre SDN, donde se reflejará el uso de un controlador seleccionado por los autores para estimar el mejoramiento en el tráfico de información en una red de datos. Con la elaboración del modelo de infraestructura virtual, usando métodos científicos, ya que se presentarán situaciones que pueden ser investigadas por la ciencia. Aplicando la observación, hipótesis y experimentación.

2.7. CRONOGRAMA



	ACTIVIDAD
A	Identificar los requerimientos para la generación del diseño del modelo en infraestructura en SDN.
B	Seleccionar el controlador más adecuado para la generación del modelo de infraestructura en las redes definidas por software.
C	Diseñar la arquitectura del sistema basados en sistemas de simulación como Mininet sobre sistemas operativos abiertos.
D	Desarrollar el modelo de infraestructura teniendo en cuenta niveles de escalabilidad, seguridad y QoS.

3.0. MARCO TEORICO

En el presente ítem se describe los conceptos que fundamentan la realización del presente proyecto.

3.1. La SDN

Los sistemas de SDN permiten ofrecer ventajas a nivel de red para las empresas que deseen expandir su infraestructura usando bajos costos. Este modelo para las telecomunicaciones de las empresas permite administrar y controlar los recursos de la red, ofreciendo un alto rendimiento, utilizando algún tipo de controlador para la expansión de la red. Algunos ejemplos: ONOS, Opendaylight, NOX, ODL, POX.

3.1.1 Onos.

Onos es un sistema de código abierto, que permite crear, inicializar, controlar, cambiar y gestionar el comportamiento de reenvío del tráfico de una red mediante APIs abiertas. Una de las características principales de este sistema, es permitir la escalabilidad en la red interna de la empresa. Además, puede ejecutarse como un sistema distribuido en múltiples servidores, lo que le permite utilizar la unidad central de proceso y los recursos de memoria de los servidores.

Requerimientos

- RAM: 8 GB
- Disco duro: espacio libre mínimo (20 GB).
- Sistema operativo: Windows, Mac OS X o Linux.

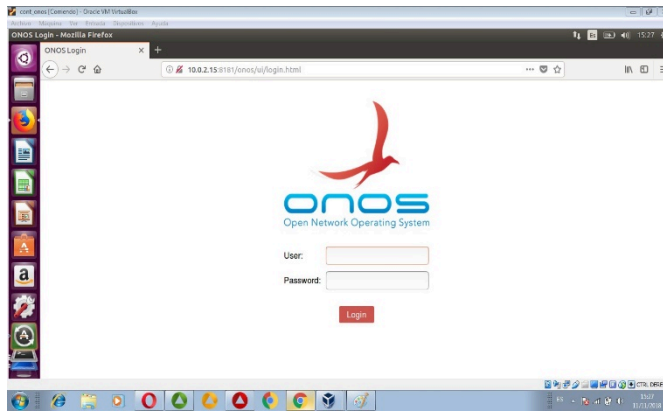


Figura 1, visualización de la entrada de Onos Gui, name: onos , passw: rocks., Fuente:
<https://wiki.onosproject.org/display/ONOS/Basic+ONOS+Tutorial>

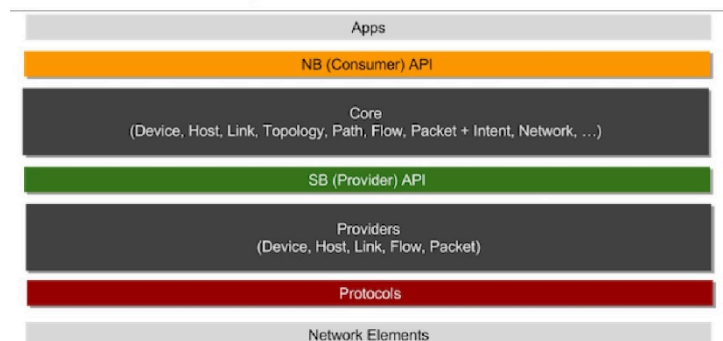


figura 1.1, Arquitectura de Onos 1.14.

De la figura 1.1, se entiende que el núcleo del sistema abierto (Onos) es responsable de rastrear la información sobre el entorno de red, el estado de selección y la sincronía entre los pares del clúster. Además, de distribuirla a las aplicaciones de forma sincrónica a través de solicitud de consultas o también de llamada entre los terminales.

Dentro de las versiones oficiales de Onos se encuentra la versión Búho 1.14, con la que se estará desarrollando la aplicación de pruebas para este proyecto. Una de las características del sistema de Onos es que se centra en un nivel básico como se muestra en la figura 1.1., dejando los protocolos al nivel de los proveedores.

En conclusión, **el funcionamiento de ONOS se efectúa a través del Core, el cual buscará la información para distribuirla a los sistemas de aplicación.** Esta información refiere al entorno en el cual está operando el sistema Onos, siendo el nivel de núcleo el responsable de la **sincronización y administración de red**, a través del envío de consultas. Con lo cual, la función de los proveedores es de interactuar con los elementos interconectados a la red.

3.2 Opendaylight

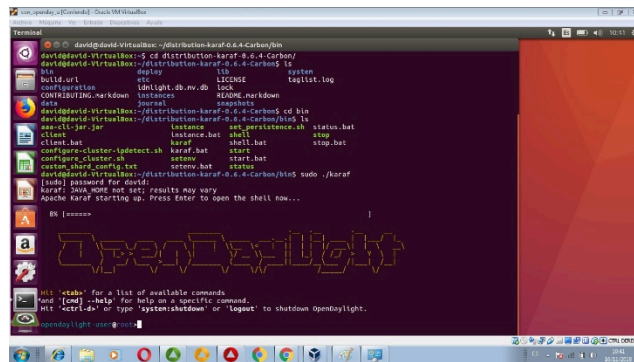


Figura 2, visualización Opendaylight en funcionamiento.

El controlador Opendaylight, sigue un procedimiento de instalación a través de las ODL, las cuales permitirán que se escuche a través del puerto 6653, (figura 2) el cual será el puerto de servicios de comunicación para openflow. habilitándose el flujo de transferencia TCP.

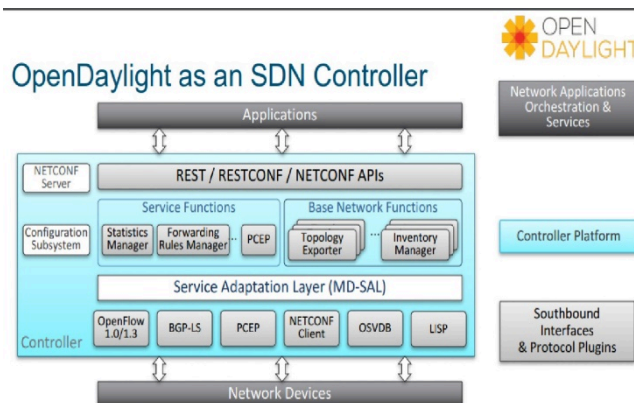


Figura 2.1 , estructura de Opendaylight, como una SDN.

Como se puede ver en la figura 2.1 el enlace entre la red y las aplicaciones es el controlador el cual genera el flujo de información, servicios y control en el sistema. Para Opendaylight es necesaria la instalación de la odl md-sal la cual permite brindar servicios de adaptación en un nivel interno entre los plugin y las funciones de red.

Una vez instalado las ODL el sistema puede ingresar activamente por medio del puerto 8181 el cual brindara los servicios de protocolo de transporte (TCP) usando la IP del controlador principal como medio de enlace para toda la red.

3.3. Mininet

Es un emulador de redes definidas por software, que utiliza el kernel de Linux y otros recursos para emular elementos de la SDN (figura 3) como el controlador, los switch OpenFlow y los hosts. ya sea en modo grafico o en modo comando. Es un sistema simple para simular la gestión de la red, que ofrece un entorno grafico para establecer la configuración de la red, a través de la opción miniedit.

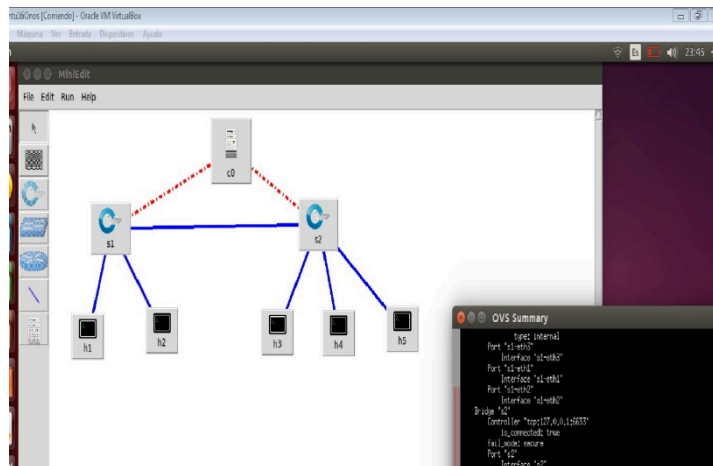
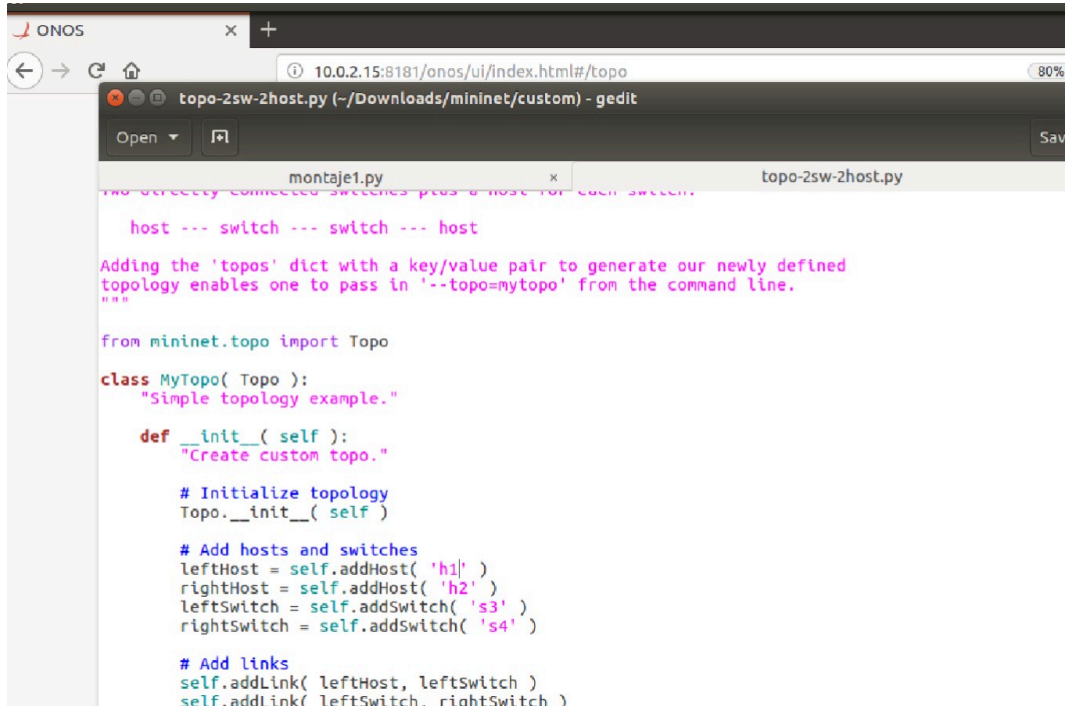


Figura 3, visualización de la aplicación miniedit, de mininet. demostración funcionamiento de mininet,

controlador: Onos activado

switch=1 host=16



```

topo-2sw-2host.py (~/Downloads/mininet/custom) - gedit
montaje1.py
topo-2sw-2host.py

host --- switch --- switch --- host

Adding the 'topos' dict with a key/value pair to generate our newly defined
topology enables one to pass in '--topo=mytopo' from the command line.
"""

from mininet.topo import Topo

class MyTopo( Topo ):
    "Simple topology example."

    def __init__( self ):
        "Create custom topo."

        # Initialize topology
        Topo.__init__( self )

        # Add hosts and switches
        leftHost = self.addHost( 'h1' )
        rightHost = self.addHost( 'h2' )
        leftSwitch = self.addSwitch( 's3' )
        rightSwitch = self.addSwitch( 's4' )

        # Add links
        self.addLink( leftHost, leftSwitch )
        self.addLink( leftSwitch, rightSwitch )

```

Figura 7, visualización del api con de la topología lineal. figura scripts de la topología simple o lineal.

ONOS lineal	6.61Mbps	6.61Mbps	6.61Mbps	6.61Mbps	6.61Mbps	6.61Mbps	6.61Mbps
	1610 Paquetes	1500 Paquetes	1550 Paquetes	1700 Paquetes	800 Paquetes	400 Paquetes	2000 Paquetes
	mss.	ms	ms	ms	ms	ms	ms
rtt min	40,98	40,6	42,3	55	20,24	9,73	78,96
rtt Max:	1923,74	1900,11	960,3	820	960,9	456,21	3125,03
rtt mdev:	65,49	64,3	61,02	44	31,08	15,7	70,06

Tabla. Resultado rtt prueba onos lineal fuente: autores

ONOS tree	9,16 Mbps 1610 Paquetes	9,16 Mbps 1500 Paquetes	9,16 Mbps 1550 Paquetes	9,16 Mbps 1700 Paquetes	9,16 Mbps 800 Paquetes	9,16 Mbps 400 Paquetes	9,16 Mbps 2000 Paquetes
	mss.	ms	ms	ms	ms	ms	ms
rtt min	82,13	81	81,65	90,6	41,065	20,53	100,78
rtt Max:	295,8	250	270,5	350	147,9	72,95	504,8
rtt mdev:	11,76	11,0	11,8	16,55	5,88	3,20	18,85

Tabla. Resultado rtt prueba ampliando ancho de banda Onos tree

En las tablas, se muestran los resultados obtenidos de la aplicación de las topologías lineal y tree cuando se envía un paquete entre los host o terminales. Se referencia el tiempo de ida y vuelta que se toman los paquetes enviados entre h1 y h16.

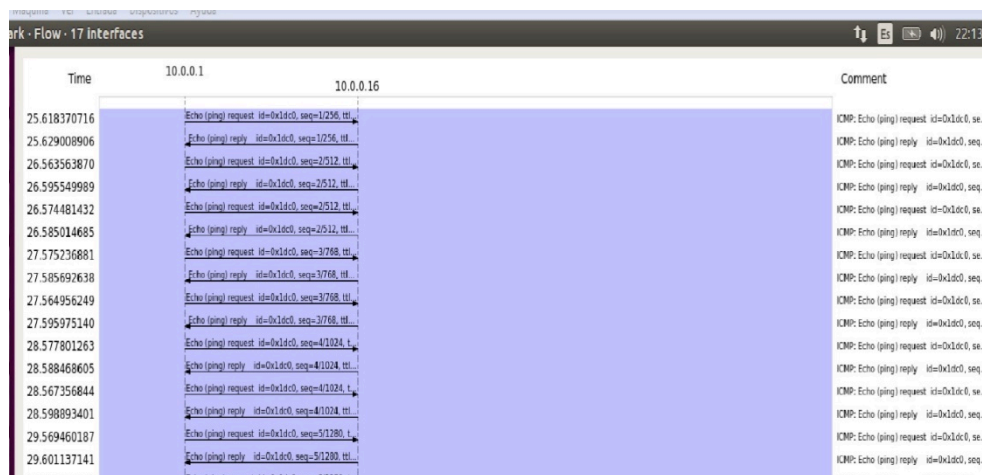


Figura 8, visualización de flow grap de ping entre h1 y h16.

En la Figura 8, se aprecia el flujo de información entre los hosts de cada switch, representado gráficamente como una captura de flujo tomada por la aplicación de wireshark.

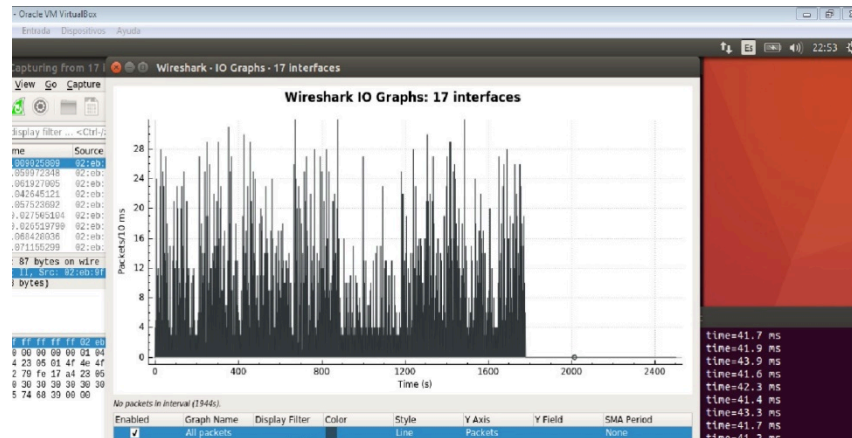


Figura 9, visualización grafica de entrada y salida de paquetes en s1 por transmisión de paquetes. Estado de ping entre h1-h16.

o.	Time	Source	Destination	Protocol	Length	Info
4767	254.334416330	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6702817 Win=889344 Len=0 TSval=15
4768	254.335925126	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6714401 Ack=1 Win=29696 Len=2896 TSval=
4769	254.336914151	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6705713 Win=889344 Len=0 TSval=15
4770	254.338436772	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6717297 Ack=1 Win=29696 Len=2896 TSval=
4771	254.338922294	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6706609 Win=889344 Len=0 TSval=15
4772	254.341003523	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6720193 Ack=1 Win=29696 Len=2896 TSval=
4773	254.341973701	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6711505 Win=889344 Len=0 TSval=15
4774	254.343413088	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6723089 Ack=1 Win=29696 Len=2896 TSval=
4775	254.343950201	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6714401 Win=889344 Len=0 TSval=15
4776	254.345515041	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6725905 Ack=1 Win=29696 Len=2896 TSval=
4777	254.346440221	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6717297 Win=889344 Len=0 TSval=15
4778	254.348013001	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6728881 Ack=1 Win=29696 Len=2896 TSval=
4779	254.349137646	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6720193 Win=889344 Len=0 TSval=15
4780	254.350514120	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6731777 Ack=1 Win=29696 Len=2896 TSval=
4781	254.351422844	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6723089 Win=889344 Len=0 TSval=15
4782	254.352928334	10.0.0.1	10.0.0.2	TCP	2962	36062 → 5001 [ACK] Seq=6734673 Ack=1 Win=29696 Len=2896 TSval=
4783	254.353971872	10.0.0.2	10.0.0.1	TCP	66	5001 → 36062 [ACK] Seq=1 Ack=6725905 Win=889344 Len=0 TSval=15

Figura 10, visualización de ping entre h1 y h2. captura del protocolo TCP.

Como resultado de la aplicación de transmisión de datos entre los hosts, se obtuvo unos picos gráficos en la salida del sistema; las pruebas mostraron el diagrama resultante en la figura 7 donde se hicieron pasar 1610 paquetes, estableciéndose un tiempo variado entre los mismo. Es decir, un retardo distinto entre los paquetes enviados y recibidos, a través de los puertos de cada dispositivo.

TOPOLOGIA TREE CARACTERISTICAS

controlador: Onos activado switch=6 host=25

```
s1 s2 s3 s4 s5 s6
*** Adding links:
(10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s1, s2) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s1, s3) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s1, s4) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s1, s5) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s1, s6) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s2, h1) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s2, h2) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s2, h3) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s2, h4) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s2, h5) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s3, h6) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s3, h7) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s3, h8) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s3, h9) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s3, h10) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s4, h11) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s4, h12) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s4, h13) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s4, h14) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s4, h15) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s5, h16) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s5, h17) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s5, h18) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s5, h19) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s5, h20) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s6, h21) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s6, h22) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s6, h23) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s6, h24) (10.00Mbit 10ms delay) (10.00Mbit 10ms delay) (s6, h25)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16 h17 h18 h19 h20 h21 h22 h23 h24 h25
*** Creating controller
```

Figura 11, visualización de enlaces entre switch y sw y sw-host.

> iperf h1-h25 comando CLI usado para ping

> h1 ping h2 comando de ejecución de ping entre dispositivos terminales.

Estado de resultados transferencias TCP

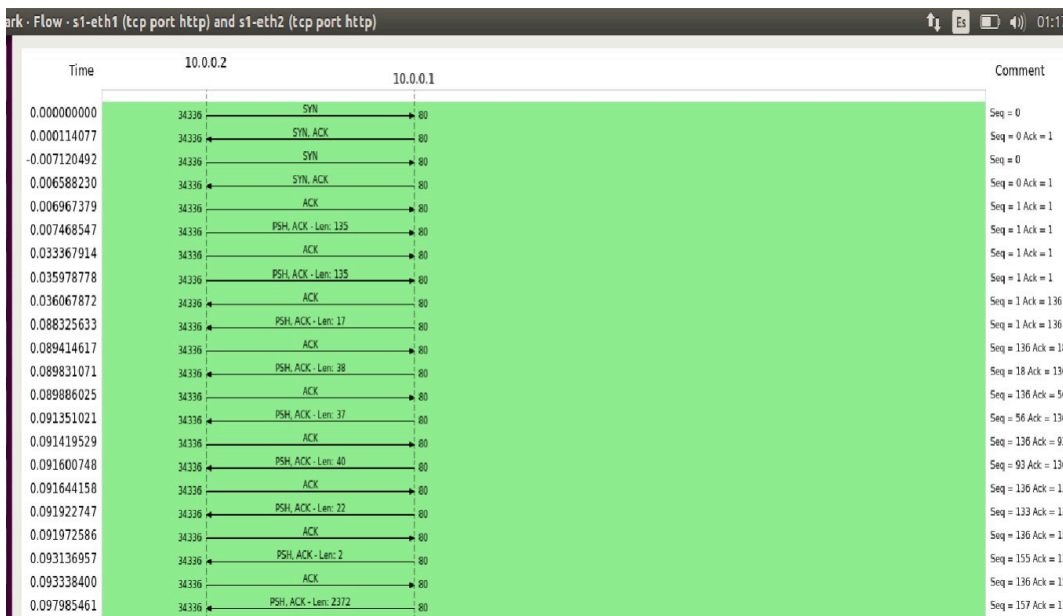


Gráfico 1, flowgraph visualización de protocolo de transferencia TCP h2 llama a un servidor simple HTTP conectado al h1.

en el grafico anterior se hace un requerimiento GET a un server desde el host 2 obteniendo como resultado la captura de paquetes TCP.

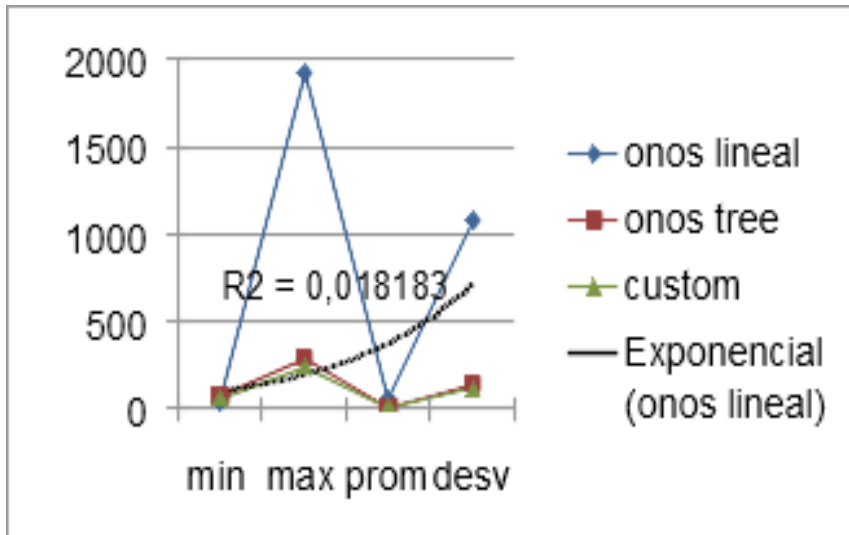


Grafico 2, paquetes enviados Vs tiempo(ms). comparación entre topologías por defecto en el sistema. representación crecimiento exponencial de la topología lineal valor Y.

Análisis gráfico:

según el resultado de enviar 50 paquetes a través de los dispositivos terminales, se define que la topología lineal es más eficiente dado que puede enviar más paquetes usando el mismo tiempo que en la topología tree.

En cuanto al resultado de correlación en el grafico entre las topologías se halló como resultado que fue de 0,018183.

Como resultado el crecimiento exponencial de la eficiencia de la topología lineal, que demuestra mayor número de paquetes que la topología tree.

comando en modo CLI:

```
$sudo --topo=topo --link tc,bw=10,delay=10ms.
```

ANALISIS DE LA TABLA

caso topología linear:

Como el retraso de cada enlace es de 10 ms, el tiempo de ida y vuelta (RTT) debe ser de aproximadamente 1923,7 ms, ya que la solicitud ICMP atraviesa un enlace (uno al conmutador, uno al destino) y la respuesta ICMP atraviesa un enlace que regresa rápidamente.

Es de concluir que según los resultados obtenidos en RTT, es más rápido y eficiente la topología linear usando el controlador Onos.

Calcular el Throughput

para generar este experimento se ha creado un archivo llamado “pruebadiseg.png” el cual tendrá un tamaño de 10Mbps. con el fin de ejecutar una transferencia en FTP “FTP son las siglas de “File Transfer Protocol” o en español “Protocolo de transferencia de archivos”, que es un protocolo para la transferencia de archivos entre sistemas conectados a una red. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo. Este protocolo para la subida y bajada de archivos no es del todo seguro, ya que la información que se manda por medio del protocolo, no va cifrada, sino en texto plano.

```
$ falocate -l 10MiB pruebadiseg.png
```

se instalará un servidor FTP en mininet para generar el experimento.

```
$ sudo apt-get install ftpd
```

Desde el host2 (h2) se hará la transferencia al host 16 usando las topologías. a través del puerto 21.

```
$ tcpdump -w ftp-transfer host2 10.0.0.2 and host12 10.0.0.16  
and port 21
```

Para generar esta prueba fue necesario tener en cuenta que el protocolo de transporte requiere de un servidor y de un cliente por lo cual se programó un host como server y el otro como cliente. los resultados obtenidos fueron los siguientes: ancho de banda 100mbps, retardo a 5 ms.

Onos	sec	transf Mbps
lineal (h1S1 a h16S2)	17	4,12
tree (h1 a h16)	101,3	12,2
custom	9,2	8,2

Tabla 2, de datos de TCP puerto 5566

4. *diseño con el controlador Opendaylight*

Este controlador hace parte de la gama de modelos actuales de TI, de la SDN, el cual permite generar administración, control, y gestión en la red de topologías de red.

Una de las principales ventajas del controlador Opendaylight es que elimina las barreras de adopción, ya que algunas organizaciones no quieren comprometerse con un fabricante. Al ser una plataforma abierta, las empresas disponen de la opción de optar por tecnologías de fabricantes diferentes que serían interoperables, estos son: Big Switch Networks, Brocade, Cisco, Citrix, Ericsson, IBM, Juniper Networks,

Round-trip delay time

Opendaylight (1610 Paquetes)	lineal	tree	`custom
ancho de banda h1 a h16	9.38Mbps	9,28Mbps	6.97Mbps
	mss.		
rtt min	61,382ms	82,516ms	61,23ms
rtt Max:	168,34ms	640,58ms	163,45ms
rtt mdev:	6,57ms	13,92ms	6,33ms

tabla 3, visualización de datos captados ping h1-h16 opend. cont

Para generar esta tabla se transmitió una ráfaga de paquetes 1610 con los que se obtuvo los anteriores resultados. Teniendo en cuenta que el ancho de banda se consideró en 10 Mbps y delay de 10 ms

4. Grafica de resultados generados por la aplicación de la simulación.

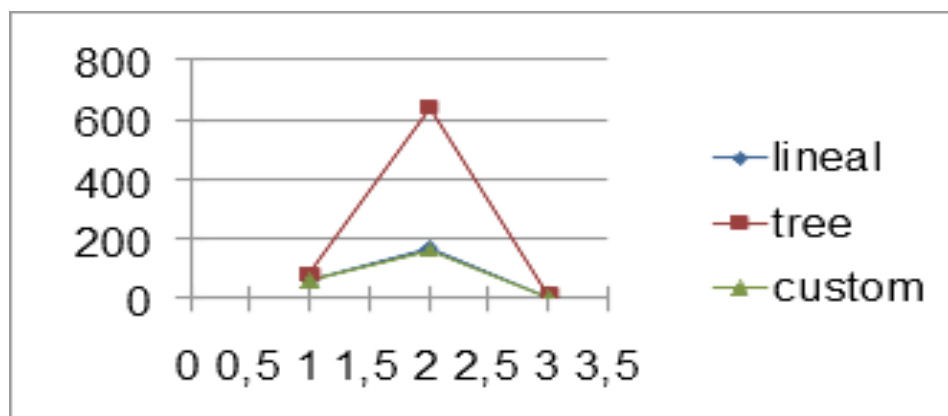


grafico 3, grafica del estado comparativo entre las topologías de red para el controlador Opendaylight.

Integrando las gráficas resultantes en grafico 1 y 2. como comparativa de los controladores Opendaylight y Onosproject.

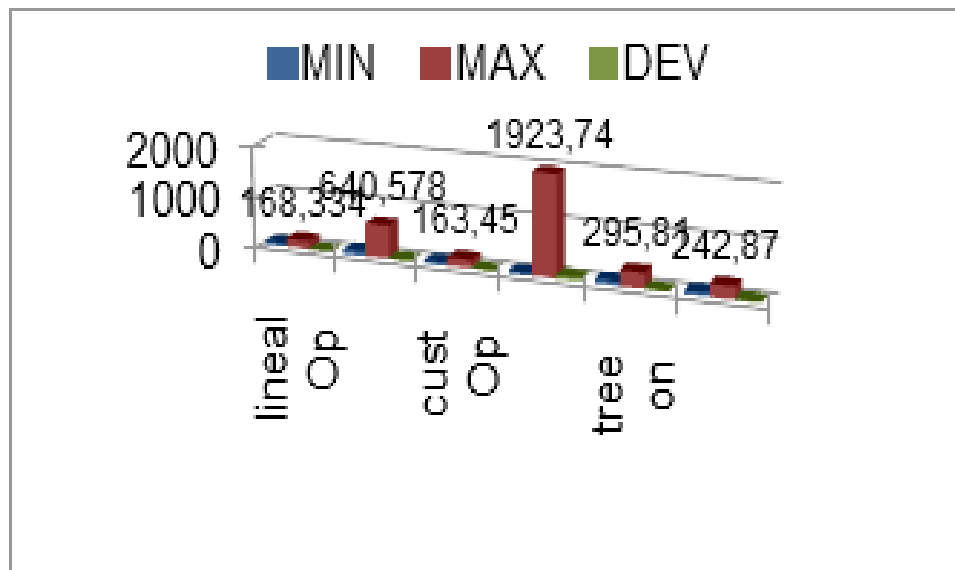


gráfico 4, Diagrama de barras resultante, cruce de datos entre las topologías de ambos controladores.

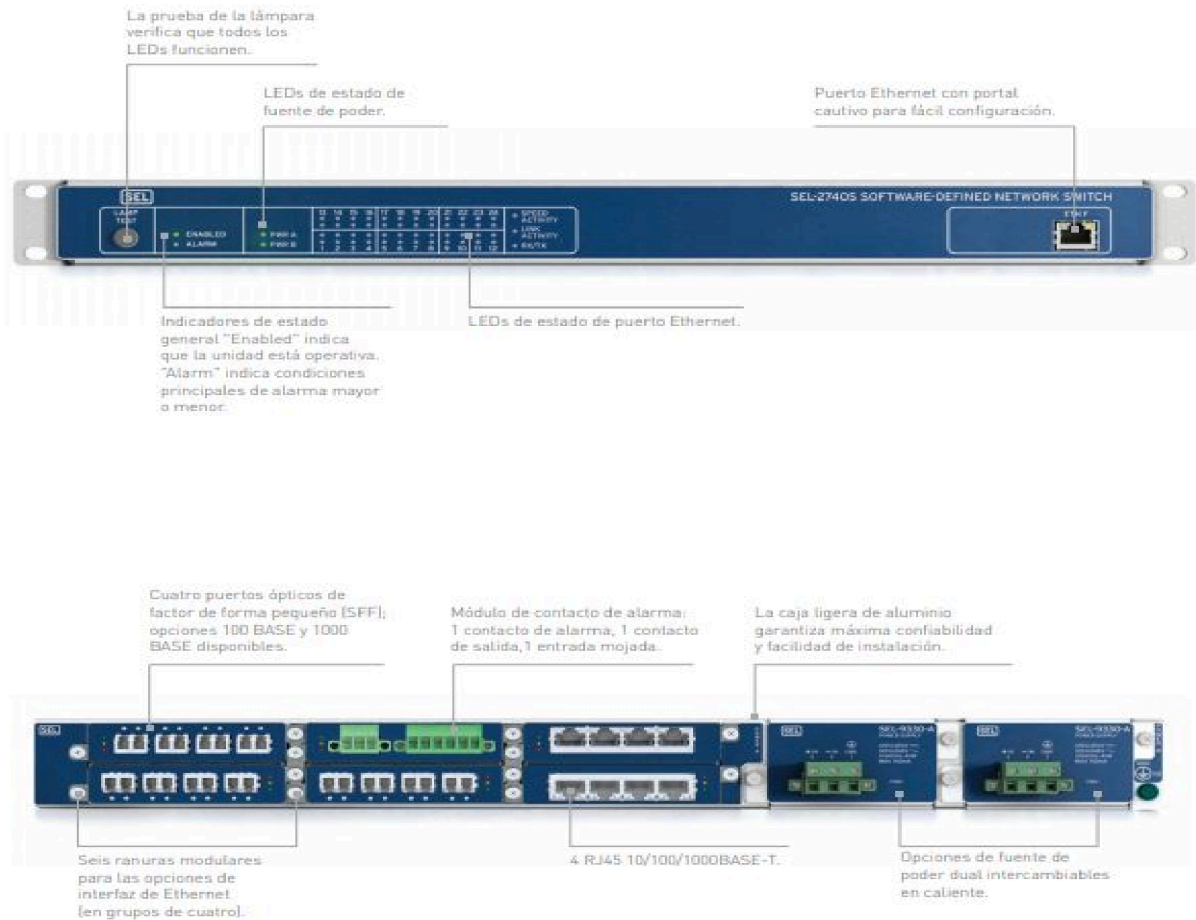
Según el gráfico 4, se puede notar que la topología lineal en el controlador Opendaylight, es la mayor. Demostrando que el controlador Opendaylight es una herramienta más eficiente que el controlador Onos.

Bibliografía

- [1] E. Pineda, «SDN y NFV, dos tendencias que revolucionarán las telecomunicaciones,» *Mundo Contact*, 2014.
- [2] Alcatel-Lucent, «<https://www.al-enterprise.com/>,» 2013. [En línea].
- [3] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka y T. Turletti, «A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks,» *IEEE Communications Surveys & Tutorials*, vol. 16, pp. 1617 - 1634, 2014.
- [4] Sel, «<https://selinc.com/>,» 2020. [En línea].
- [5] J. C. Chico, D. Mejía y I. Bernal, «Implementación de un Prototipo de una Red Definida por Software (SDN),» <https://bibdigital.epn.edu.e>, Quito.
- [6] A. Tootoonchian y Y. Ganjali, «HyperFlow A Distributed Control Plane for OpenFlow,» Toronto, 2010.
- [7] D. G. L. D. ., N. ., S. N. ., B. A. Malishevskiy, «Gestión de red basada en OpenFlow con visualización de elementos gestionados,» santa catalina, 2014.
- [8] V. Toledo, «Cisco impulsa un protocolo SDN alternativo a OpenFlow,» Toledo, Virginia, 04 04 2014. [En línea]. Available: <https://www.datacenterdynamics.com/es/noticias/cisco-impulsa-un-protocolo-sdn-alternativo-a-openflow/>.
- [9] D. Blandon, «EL PROTOCOLO DEL FUTURO* Openflow: The future protocol,» researchgate.net, marzo 2013. [En línea]. Available: https://www.researchgate.net/publication/338753687_OPENFLOW_EL_PROTOCOLO_DEL_FUTURO_Openflow_The_future_protocol.
- [10] M. H. Diego, «Diseño e implementación de una aplicación de red bajo la arquitectura SDN,» bogota, 2014.]
- [11] M. C. Bryan y S. S. José Luis, «Estudio para la implementación de un sistema de redes definida por software para una red de área amplia (WAN).,» bogota, 2019.
- [12] A. C. Cueto Altahona y D. A. Polo Pérez, «Diseño de Prototipo de Aplicación para Control de Admisión de Flujos en una Red SDN,» norma, valledupar, 2019.
- [13] J. A. Vela Samudio y J. J. Santis Venegas, «Algoritmo de enrutamiento multicamino UDIC en redes SDN,» 2019.]
- [14] Y. A. Marín Muro, «PLATAFORMA DE PRUEBAS PARA EVALUAR EL DESEMPEÑO DE LAS REDES DEFINIDAS,» Santa Clara, Cuba, 2016.]

Anexo I, Imagen SWITCH DE RED DEFINIDO POR SOFTWARE SEL – 2740S

SWITCH DE RED DEFINIDO POR SOFTWARE



Anexo A. Carta del director del proyecto

Valledupar, 21 marzo del 2022.

Señores:
COMITÉ DE INVESTIGACION.
Facultad de Ingenierías y Tecnológicas
Programa ingeniería de sistemas
Universidad Popular Del Cesar

Cordial saludo

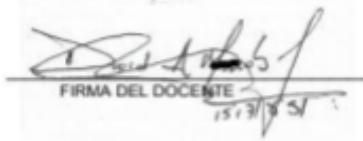
Yo, David Arturo Manotas Ferias identificado con la cédula de ciudadanía No. 15171851, certifico que he revisado el documento correspondiente al proyecto que lleva por título "DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACION DE MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.", presentada por los estudiantes CARLOS ALFREDO PESCA TAFUR y ALVARO DE JESUS CASTRO YANCY, y, después de haberle realizado las respectivas correcciones, cuenta con mi aprobación para ser presentada ante el comité. Sugiero la aprobación por parte de ustedes.

Línea de investigación: Tecnologías de la Información y la comunicación

Sublínea: Telecomunicaciones y teleinformática

Agradezco la atención prestada

Atentamente,



FIRMA DEL DOCENTE

DAVID MANOTAS FERIAS
CC 15171851 de Valledupar, Cesar
Director de Proyecto

Anexo B. Carta de los estudiantes

Valledupar, 08 de septiembre 2021

Señores:

COMITÉ DE INVESTIGACION
Facultad de Ingenierías y Tecnológicas
Programa ingeniería de sistemas
Universidad Popular Del Cesar

Cordial saludo

Nosotros los abajo firmantes, estudiantes del programa de Ingeniería de sistemas, presentamos a ustedes el documento correspondiente al proyecto de grado denominado **DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACIÓN DE MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.**

Quedamos a la espera del concepto emitido por el comité respecto de la viabilidad y aceptación de dicha propuesta.

Agradecemos la atención prestada

Atentamente,

Alfredo Pesca Tafur

CC. 1 . 065. 658. 671 de Valledupar

Álvaro De Jesús Castro Yancy

CC.1.079.933.507de Pivijay

Carlos

Anexo D. Carta aval



Valledupar, Cesar 26/05/2020

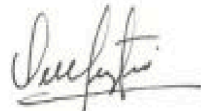
Señores:
COMITÉ DE PROYECTOS DE GRADO
Facultad de Ingenierías y Tecnológicas
Programa de Ingeniería de Sistemas
Universidad Popular Del Cesar

Cordial Saludo respetados Ingenieros,

Me permito informarle que los estudiantes: Carlos Alfredo Pesca Tafur y Álvaro Castro Yancy, se encuentran autorizados por la Universidad Popular Del Cesar, ubicada en la Dig. 21 #2992, teléfono 5842472, para realizar su proyecto de grado, titulado **DISEÑO DE UN MODELO EN INFRAESTRUCTURA DE SDN USANDO EL PROTOCOLO OPENFLOW ATRAVES DE MININET PARA EL LABORATORIO 102 DE TELEMATICA BLOQUE I UNIVERSIDAD POPULAR DEL CESAR.**

Aclaro de antemano que el desarrollo del proyecto no genera ningún vínculo laboral con la entidad.

Atentamente,



OMAIRA LUZ TAPIAS DÍAZ
Jefe Departamento Programa Ingeniería Electronica

Anexo E. Carta antifraude.

**PRESENTACIÓN DE PROPUESTA DE PROYECTO O TESIS DE GRADO HOJA DE
DECLARACION ANTI FRAUDE**

SEMESTR E	
FECHA (aaaa/mm/dd)	08/09/2021

PROYECTO DE GRADO PARA OPTAR EL TÍTULO DE: INGENIEROS DE SISTEMAS

NOMBRES Y APELLIDOS DEL ESTUDIANTE:

Carlos Alfredo Pesca Tafur CC. 1.065. 658. 671 de Valledupar

Álvaro De Jesús Castro Yancy CC.1.079.933.507 de Pivijay

TÍTULO DE LA TESIS O PROYECTO:

DISEÑO DE UN MODELO EN INFRAESTRUCTURA SDN USANDO EL PROTOCOLO OPENFLOW EN LAS PLATAFORMAS ABIERTAS ONOS Y OPENDAYLIGHT A TRAVÉS DE LA HERRAMIENTA DE CONFIGURACIÓN DE MININET PARA LABORATORIOS DE REDES Y TELECOMUNICACIONES.

DECLARACIÓN:

1 - Soy consciente que cualquier tipo de fraude en este proyecto es considerado como una falta grave en la Universidad. Al firmar, entregar y presentar esta propuesta de Proyecto de Grado, doy expreso testimonio de que esta propuesta fue desarrollada de acuerdo con las normas establecidas por la Universidad. Del mismo modo, aseguro que no participé en ningún tipo de fraude y que en el trabajo se expresan debidamente los conceptos o ideas que son tomadas de otras fuentes.

2- Soy consciente de que el trabajo que realizaré incluirá ideas y conceptos del autor y del Director y/o Asesor y podrá incluir material de cursos o trabajos anteriores realizados en la Universidad y por lo tanto, daré el crédito correspondiente y utilizaré este material de acuerdo con las normas de derechos de autor. Así mismo, no haré publicaciones, informes, artículos o presentaciones en congresos, seminarios o conferencias sin la revisión o autorización expresa del Asesor, quien representará en este caso a la Universidad.

<i>NOMBRE</i>	<i>Firma</i>
Carlos Alfredo Pesca Tafur	
CC: 1.065. 658. 671	
Álvaro De Jesús Castro Yancy	
CC: 1.079.933.507	

Anexo F. Derechos de Autor

**PRESENTACIÓN DE PROPUESTA DE PROYECTO O TESIS DE
GRADO HOJA DE DERECHOS DE AUTOR
(A DILIGENCIAR POR EL ESTUDIANTE)**

SEMESTRE	2021
FECHA (aaaa/mm/dd)	08/09/2021

PROYECTO DE GRADO PARA OPTAR EL TÍTULO DE: INGENIEROS DE SISTEMAS

NOMBRES Y APELLIDOS DEL ESTUDIANTE:

Carlos Alfredo Pesca Tafur CC. 1.065. 658. 671 de Valledupar
 Álvaro De Jesús Castro Yancy CC.1.079.933.507 de Pivijay

AUTORIZACIÓN DE SU USO A FAVOR DE LA UNIVERSIDAD:

Autorizo a LA UNIVERSIDAD POPULAR DEL CESAR, para que en los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, Decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia, utilice y use en todas sus formas, los derechos patrimoniales de reproducción, comunicación pública, transformación y distribución (alquiler, préstamo público e importación) que me corresponden como creador de la obra objeto del presente documento. PARÁGRAFO: La presente autorización se hace extensiva no sólo a las facultades y derechos de uso sobre la obra en formato o soporte material, sino también para formato virtual, electrónico, digital, óptico, usos en red, internet, extranet, intranet, etc., y en general para cualquier formato conocido o por conocer.

NOMBRE	Firma
Carlos Alfredo Pesca Tafur	
CC: 1.065. 658. 671	
Álvaro De Jesús Castro Yancy	
CC: 1.079.933.507	