

**ANÁLISIS COMPARATIVO ENTRE EL TIEMPO DE VIDA DE LOS PAQUETES
Y EL USO DEL ANCHO DE BANDA, EN LAS REDES BASADAS EN SDN Y
LAS REDES BASADAS EN EL MODELO OSI**

**CARLOS MARIO CASTAÑO CASTAÑEDA
JUAN ESTEBAN VALENCIA HENAO**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍAS
INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
PEREIRA
2016**

**ANÁLISIS COMPARATIVO ENTRE EL TIEMPO DE VIDA DE LOS PAQUETES
Y EL USO DEL ANCHO DE BANDA, EN LAS REDES BASADAS EN SDN Y
LAS REDES BASADAS EN EL MODELO OSI**

**CARLOS MARIO CASTAÑO CASTAÑEDA
JUAN ESTEBAN VALENCIA HENAO**

**Asesor del proyecto Juan Carlos Vergara Zapata
Ingeniero de sistemas y computación
Especialista en redes de datos**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍAS
INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
PEREIRA
2016**

Contenido

RESUMEN	8
CAPITULO 1: GENERALIDADES.....	9
1.1 DEFINICIÓN DEL PROBLEMA	9
1.2 JUSTIFICACIÓN.....	10
1.3 OBJETIVOS	11
1.3.1 GENERAL	11
1.3.2 OBJETIVOS ESPECÍFICOS.....	11
1.4 MARCO REFERENCIAL	12
1.5 MARCO CONCEPTUAL	13
1.6 MARCO TECNOLÓGICO	14
1.7 DISEÑO METODOLÓGICO.....	15
CAPITULO 2: REDES DEFINIDAS POR SOFTWARE	17
2.1 HISTORIA.....	17
2.2 CARACTERÍSTICAS	18
2.3 OBJETIVO.....	19
2.4 CAPAS ARQUITECTURA SDN	20
2.5 BENEFICIOS.....	22
2.6 OPENFLOW	23
CAPITULO 3: REDES TRADICIONALES, MODELO OSI.....	24
3.1 El modelo OSI.....	24
3.2 Arquitectura de comunicación por capas.	25
3.3 Protocolos de red.....	26
CAPITULO 4: COMPARATIVO.....	27
4.1 Resultados.....	33
CAPITULO 5: CONCLUSIONES	61
5.1 Conclusiones	61
5.2 Recomendaciones.....	63
5.3 Referencias bibliográficas.....	64

ÍNDICE DE FIGURAS

Figura 1. Arquitectura SDN.....	20
Figura 2. Estructura arquitectura de capas.....	25
Figura 3. Comparación de una red SDN y una red tradicional.....	27
Figura 4. Topología en Packet Tracer.....	31
Figura 5. Topología en Mininet.....	31
Figura 6. Ejecución topología en Mininet.....	32
Figura 7. Ping entre pc1 y pc9 en Packet Tracer.....	33
Figura 8. Ping entre pc1 y pc9 con en Mininet.....	33
Figura 9. Ping extendido entre pc1 y pc9 en Packet Tracer.....	35
Figura 10. Ping extendido entre pc1 y pc9 en Mininet.....	35
Figura 11. Ping entre pc1 y pc5 en Packet Tracer.....	38
Figura 12. Ping entre pc1 y pc5 en Mininet.....	38
Figura 13. Ping extendido entre pc1 y pc5 En Packet Tracer.....	40
Figura 14. Ping extendido entre pc1 y pc5 en Mininet.....	40
Figura 15. Ping entre pc6 y pc4 en Packet Tracer.....	42

Figura 16. Ping entre pc6 y pc4 en Mininet.....	42
Figura 17. Ping extendido entre pc6 y pc4 en Packet Tracer.....	44
Figura 18. Ping extendido entre pc6 y pc4 en Mininet.....	44
Figura 19. Ping entre pc11 y pc12 en Packet Tracer.....	46
Figura 20. Ping entre pc11 y pc12 con en Mininet.....	46
Figura 21. Ping extendido entre pc11 y pc12 en Packet Tracer.....	48
Figura 22. Ping extendido entre pc11 y pc12 en Mininet.....	48
Figura 23. Ping entre pc2 y pc3 en Packet Tracer.....	51
Figura 24. Ping entre pc2 y pc3 en Mininet.....	51
Figura 25. Ping extendido entre pc2 y pc3 en Packet Tracer.....	53
Figura 26. Ping extendido entre pc2 y pc3 en Mininet.....	53
Figura 27. Ping extendido entre pc0 y pc7 en Packet Tracer.....	55
Figura 28. Complementación Ping extendido entre pc0 y pc7 en Packet Tracer...	55
Figura 29. Ping extendido entre pc0 y pc7 en Mininet.....	56
Figura 30. Ping extendido entre pc3 y pc11 en Packet Tracer.....	58
Figura 31. Complementación ping extendido entre pc3 y pc11 en	

Packet Tracer.....	58
Figura 32. Ping extendido entre pc3 y pc11 en Mininet.....	59

ÍNDICE DE TABLAS

Tabla 1. datos ping entre pc1 y pc9.....	34
Tabla 2. datos ping entre pc1 y pc5.....	36
Tabla 3. Datos ping entre pc6 y pc4.....	39
Tabla 4. Datos ping pc11 a pc12.....	41
Tabla 5. Datos entre pc2 y pc3.....	43
Tabla 6. Datos ping extendido entre pc1 y pc9.....	45
Tabla 7. Datos ping extendido entre pc1 y pc5.....	47
Tabla 8. Datos ping extendido entre pc6 y pc4.....	49
Tabla 9. Datos ping extendido pc11 a pc12.....	52
Tabla 10. Datos ping extendido entre pc2 y pc3.....	54
Tabla 11. Datos ping extendido entre pc0 y pc7.....	56
Tabla 12. Datos ping extendido entre pc3 y pc11.....	59

ÍNDICE DE GRÁFICAS

Gráfica 1. Packet Tracer vs Mininet ping entre pc1 y pc9.....	34
Gráfica 2. Packet Tracer vs Mininet ping entre pc1 y pc5.....	37
Gráfica 3. Packet Tracer vs Mininet ping entre pc6 y pc4.....	39

Gráfica 4. Packet Tracer vs Mininet ping entre pc11 y pc12.....	41
Gráfica 5. Packet Tracer vs Mininet ping entre pc2 y pc3.....	43
Gráfica 6. Packet Tracer vs Mininet ping extendido entre pc1 y pc9.....	45
Gráfica 7. Packet Tracer vs Mininet ping extendido entre pc1 y pc5.....	47
Gráfica 8. Packet Tracer vs Mininet ping extendido entre pc6 y pc4.....	50
Gráfica 9. Packet Tracer vs Mininet ping extendido entre pc11 y pc12.....	52
Gráfica 10. Packet Tracer vs Mininet ping extendido entre pc2 y pc3.....	54
Gráfica 11. Packet Tracer vs Mininet ping extendido entre pc0 y pc7.....	57
Gráfica 12. Packet Tracer vs Mininet ping extendido entre pc3 y pc11.....	60

RESUMEN

El propósito de las redes definidas por software (SDN), es facilitar la implementación de servicios de red, a través de una configuración más eficiente de mejor rendimiento y mayor flexibilidad para dar la apertura a nuevos diseños innovadores de la redes. Esta tecnología proporciona velocidad y agilidad al implementar nuevas aplicaciones.

En este documento se examina las SDN, ofreciendo una perspectiva de las redes programables. Luego presentamos la arquitectura SDN y el protocolo Openflow, se discuten las ventajas de las redes definidas por software frente a los modelos tradicionales, comparando dos factores particulares el uso de ancho de banda y tiempo de vida de los paquetes.

Se realiza una emulación y una simulación de dos redes, permitiendo comparar los aspectos mencionados de forma técnica, los cuales facilitaran las argumentaciones acerca de las variables entre los dos escenarios.

CAPITULO 1: GENERALIDADES

1.1 DEFINICIÓN DEL PROBLEMA

La arquitectura de capas del modelo OSI utilizada hoy en día en la manipulación de datos, no es flexible y su complejidad es alta. Su enfoque se basa en prestar un servicio de transmisión de datos, minimizando agilidad, flexibilidad y escalabilidad de las mismas. Su gestión es poco automatizada, y para realizar políticas de configuración se debe hacer manualmente a cada dispositivo, incrementando los riesgos de afectación del servicio por error de quien realiza la configuración. De igual manera los servicios han ido creciendo considerablemente como también su diversidad, pero la arquitectura que soporta dichas aplicaciones no ha evolucionado a la misma velocidad, convirtiéndose en una limitante.

Las redes de datos tradicionales se han ido adaptando a las necesidades tecnológicas que el medio les ha planteado, en la cual el constante crecimiento de las mismas resulta tedioso y costoso para los diseñadores, el uso de dispositivos como los racks, routers, switch, hubs, entre otros, generan la necesidad de ampliar espacios físicos y aumentar recursos financieros, convirtiéndose en una mala práctica empresarial.

Como alternativa a la nueva tecnología emergente surgen las redes definidas por software (SDN), que pretenden llevar a cabo un análisis sobre las características y beneficios que podría brindar el uso de las mismas sobre las redes Ethernet 802.3.

Las redes definidas por software (SDN) son una forma de crear soluciones tecnológicas orientadas a la infraestructura de red, en la cual se tiene en cuenta dos aspectos fundamentales: la aplicación y el controlador. El término SDN (Software Defined Network o RED DEFINIDA POR SOFTWARE) ha estado como tendencia desde hace un par de años atrás y hace referencia a una arquitectura de red que permite separar el plano de control del plano de datos, para conseguir redes más programables, automatizables y flexibles.

Teniendo en cuenta las características del funcionamiento entre las redes SDN gestionadas por openflow y las redes basadas en el modelo OSI, ¿se puede afirmar que las redes definidas por software SDN presentan más ventajas que el modelo OSI respecto al uso del ancho de banda y tiempo de vida de los paquetes?

1.2 JUSTIFICACIÓN

Las redes de datos requieren una evolución que les permita crecer a la par con sus servicios, pero que a su vez no signifique un aumento en el gasto de recursos. Esto no debería quedarse solo en palabras o textos que digan que es necesario cambiar el modelo de implementación o de diseño, y que las redes SDN son la solución, esto debe ir acompañado de criterios fundamentados que permitan justificar su impacto y aporte a las nuevas tendencias de redes.

Se pretende realizar un análisis comparativo del funcionamiento entre ambos tipos de redes, que permita identificar las ventajas costo beneficio de cada una de ellas en algunos aspectos puntuales establecidos dentro del desarrollo de la investigación.

Como medio de verificación y fundamentación de la teoría, se hace necesario caracterizar y analizar mediante simulaciones, las diferencias en el desarrollo y creación de redes SDN y redes Ethernet 802.3, demostrando así las ventajas y desventajas de usar esta nueva arquitectura.

Se hace necesario desarrollar nuevas alternativas que permitan disminuir costos, optimizar los recursos, agilizar los procesos, entre otras ventajas, además de poder lograr un análisis comparativo de los factores tiempo de vida de los paquetes, el uso del ancho de banda y demás factores entre redes SDN gestionadas por OpenFlow y redes Ethernet basadas en el modelo OSI.

Surge además, la necesidad de documentar acerca del tema debido al poco contenido en castellano de las SDN, al cual se tiene acceso dentro de la universidad y a nivel general, teniendo éstas un enfoque muy superficial que no permite tener una visión clara de lo que son las redes SDN, sus beneficios y su justificación mediante emulaciones que muestran cómo operan.

1.3 OBJETIVOS

1.3.1 GENERAL

Realizar un análisis comparativo entre el tiempo de vida de los paquetes y el uso del ancho de banda, en las redes basadas en SDN y las redes basadas en el modelo OSI.

1.3.2 OBJETIVOS ESPECÍFICOS

- Descripción de una red SDN.
- Descripción de una red Ethernet (802.3) usando el modelo OSI.
- Realizar un comparativo de las ventajas, y desventajas de la emulación de redes SDN basados en los resultados obtenidos.

1.4 MARCO REFERENCIAL

En la Universidad Tecnológica de Pereira recientemente se observó un proyecto de grado acerca de las redes SDN, la cual solo realizaba una breve caracterización y definición de lo que estas son, siendo la única referencia textual, sin encontrar más información acerca de ninguna implementación en la cual se pudiese ver en funcionamiento una SDN. Los estudios documentados sobre la materia son relativamente pocos, aunque es un modelo que ha ido cobrando fuerza e interés.

Las redes basadas en el modelo OSI poseen una gran cantidad de contenido, donde se describen diferentes protocolos, así como cursos virtuales que proporcionan información detallada del funcionamiento del modelo a través de todas sus capas. Ethernet opera a través de dos capas del modelo OSI. El modelo proporciona una referencia con la cual puede relacionarse Ethernet, pero en realidad se implementa sólo en la mitad inferior de la capa de enlace de datos, que se conoce como subcapa de Control de acceso al medio (MAC), y en la capa física.

1.5 MARCO CONCEPTUAL

El creciente avance de la tecnología ha contribuido en investigar cómo se gestionan y se distribuyen los diferentes tipos de servicios ofrecidos en la red. El incremento en la utilización de dispositivos móviles, los servicios en la nube, y la convergencia de redes de datos, voz y vídeo, exigen nuevos requerimientos que la arquitectura de red actual no es capaz de satisfacer. Las arquitecturas utilizadas en la actualidad para el transporte de datos no son flexibles y su complejidad es alta, además enfocadas en prestar un servicio de transmisión de datos, minimizando agilidad, flexibilidad y escalabilidad de las mismas, su gestión es poco automatizada, y para realizar políticas de configuración se debe hacer manualmente a cada dispositivo, incrementando los riesgos de afectación del servicio por un error de quien realiza la configuración.

Durante los últimos años, han surgido un conjunto de técnicas para el diseño y gestión de redes, las redes definidas por software. Estas tienen en cuenta dos aspectos fundamentales que son la aplicación y el controlador. El término SDN (Software Defined Network o Red Definida por Software) se ha vuelto más innovador, y hace referencia a una arquitectura de red que permite separar el plano del control del plano de datos, para conseguir redes más programables, automatizables y flexibles. “Es una arquitectura de red emergente donde el control es desacoplado de la transmisión de datos y este es directamente programable” [1].

Para lograr una mejor definición de redes definidas por software debemos compararlas con las redes tradicionales, esto con el fin de ver su comportamiento funcional, allí es donde verdaderamente se podría evaluar o comparar en términos más específicos su funcionalidad.

1

En las redes actuales, la manera como se procesan los paquetes depende de una programación permanente a cada uno de los dispositivos que hacen parte de la red, mientras que en una red SDN está condicionada por una interfaz de programación con un software que gobierna su comportamiento.

¹ Software-Defined Networking: The New Norm for Networks
ONF White Paper April 13, 2012.

1.6 MARCO TECNOLÓGICO

Debido a las necesidades que plantean las SDN en cuestiones tecnológicas, algunos desarrolladores de dispositivos de comunicación se vieron en la obligación de diseñar e implementar equipos que soportaran la tecnología, tales como hp, cisco, huawei, brocade, vmware ofreciendo productos de hardware, controladores y aplicaciones.

Arquitectura one

La arquitectura de redes empresariales Cisco ONE transforma la TI a través de una plataforma abierta y programable para los servicios de red que conecta más estrechamente las redes y aplicaciones. Crea un “bucle de redistribución” entre la infraestructura de la red y las aplicaciones que se ejecutan en ella, lo que permite que las aplicaciones accedan a la inteligencia de la red en diversos niveles y que la red reconozca las aplicaciones que se ejecutan en ella. La arquitectura de redes empresariales Cisco ONE se aplica más allá de la red digital de servicios (SDN) al proporcionar API abiertas y programación para obtener un mayor acceso a los recursos de la red en toda la empresa.

Arquitectura de redes empresariales Cisco ONE: una base automatizada de reconocimiento de aplicaciones para la empresa moderna, informe técnico

MININET

MiniNet es un emulador de red que crea una red de hosts virtuales, conmutadores, reguladores, y acoplamientos. Anfitriones MiniNet ejecutan software de red estándar de Linux, y sus switches soportan OpenFlow para el enrutamiento de encargo altamente flexible y redes definidas por software.

MiniNet apoya la investigación, el desarrollo, el aprendizaje, la creación de prototipos, pruebas, depuración, y cualquier otra tarea que podrían beneficiarse de tener una red experimental completa en una computadora portátil u otro PC.

1.7 DISEÑO METODOLÓGICO

Para el caso concreto de este ejercicio académico se adoptó el modelo descriptivo ya que se hace un registro del comportamiento de diferentes variables, entre ellas el uso del ancho de banda y el tiempo de vida de los paquetes, específicamente. Se enmarca como un estudio documental ya que se encuentra basado en otras investigaciones que aportan un alto grado de veracidad puesto que la información se deriva de los informes e investigaciones existentes en el campo.

Es un estudio cuantitativo porque en él se hace referencia a procedimientos estadísticos, que pretenden reflejar las ventajas que brinda las redes definidas por software frente a los modelos de redes tradicionales, respecto a las variables en estudio uso de ancho de banda y tiempo de vida de los paquetes.

Población

La población o universo al cual está dirigida esta investigación, la constituyen los departamentos de IT de infinidad de empresas y proveedores de servicios de redes y todo el público mundial que hace parte de los requerimientos de telecomunicaciones. Las compañías más importantes de redes y todos aquellos involucrados en el desarrollo de tecnologías que buscan como mejorar la velocidad, calidad y la cobertura de todos los servicios.

Muestra

El estudio ha considerado una muestra de 7 casos concretos:

Ping entre pc1 y pc9

Ping entre pc1 y pc5

Ping entre pc6 y pc4

Ping entre pc11 y pc12

Ping entre pc2 y pc3

Ping entre pc0 y pc7

Ping entre pc3 y pc11

Se ejecutaron ping entre los nodos de las topologías, tanto en Packet Tracer como en Mininet.

TECNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

Técnicas

- Seguimiento de reportajes.
- Entrevistas a profesionales encargados de este tipo de procesos.

Instrumentos

- Guía de entrevistas.

PLAN DE PROCESAMIENTO DE DATOS

En la parte descriptiva se procedió de acuerdo a los informes de investigación de las redes definidas por software y las redes tradicionales basadas en el modelo OSI, además de los estudios de los dos parámetros ancho de banda y tiempo de vida, de lo anterior se hacen comparaciones de cada caso caracterizando los aspectos más importantes.

En el aporte cuantitativo de la investigación, se examinó y presentó la peculiaridad de cada caso, tanto en el uso del ancho de banda y en el tiempo de vida de los paquetes.

La información se presenta en graficas donde se plantea el caso y sus resultados.

CAPITULO 2: REDES DEFINIDAS POR SOFTWARE

2.1 HISTORIA

La historia de las SDN se remonta a las décadas en las que el internet entraba en su auge, donde los primeros pasos se basaban en transferencia de archivos, y donde el correo electrónico era una de las aplicaciones más interesantes y las que representaban mayor uso del público.

Posterior a este evento, la evolución de la tecnología permite que la historia se divida en tres etapas, estas son: redes activas (1995-2000 aprox.), separación del plano de control del plano de datos (2001-2007 aprox.) y la aparición de la interfaz de programación de aplicaciones de Openflow (2007-2010 aprox.). En 2014 Avaya hizo una demostración de redes definidas por software usando Shortest Path Bridging y OpenStack, se eliminó la configuración manual.

Los primeros esfuerzos de las redes definidas por software fueron:

Active networking

Open Signaling

DCAN

4D Project

Netconf

Ethane

2.2 CARACTERÍSTICAS

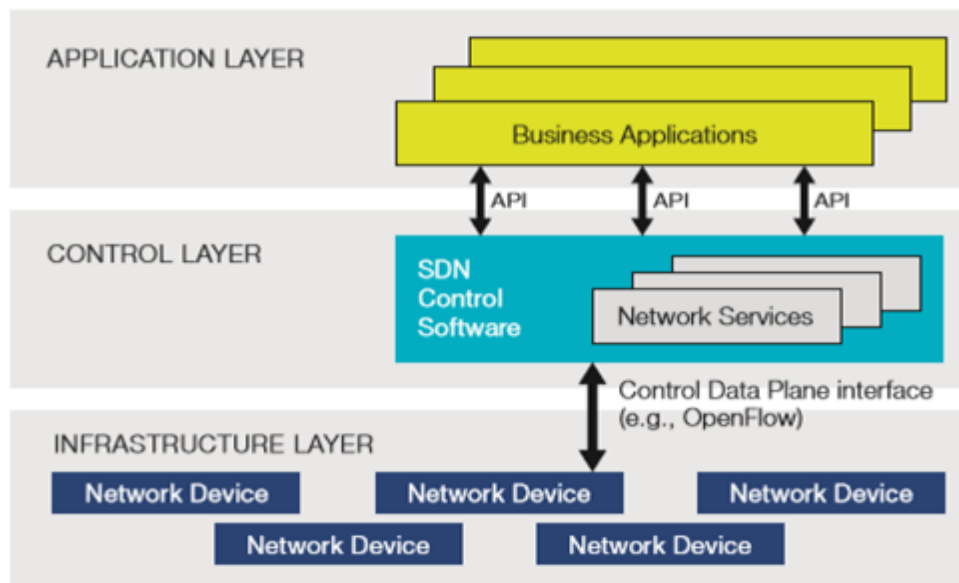
- Separación de la construcción de la red física y configuración de administración por medio de un controlador central.
- Infraestructura de red basada en openflow conectado al centro de datos.
- Procesamiento propio de los flujos.
- Esquemas de arquitectura para esquemas de direccionamiento.

2.3 OBJETIVO

El objetivo de las redes SDN es brindar a los administradores un conjunto de herramientas centralizadas de programación, virtualización y monitorización de la situación en tiempo real para que las redes se puedan adaptar de forma eficiente a las necesidades concretas de cada momento, lo cual provee ser más escalable, permitiendo automatizar recursos y simplificar tareas. Con redes SDN el procesamiento de paquetes ya no deriva de una serie de ficheros de configuración estáticos en cada uno de los dispositivos que conforman una red, sino de una organización dinámica a través de una capa de software que virtualiza la red y la independiza de la infraestructura física subyacente.

2.4 CAPAS ARQUITECTURA SDN

Figura 1. Arquitectura SDN ^[2].



Fuente: The promise of sdn, network world digital spotlight,cisco may 13, 2013.

Capa de infraestructura

Es considerada la capa más baja de SDN, aquí se encuentran los dispositivos virtuales como switch y routers. Estos dispositivos están conectados a través de una interfaz que permite el envío de paquetes en una conexión de red. Entre los medios de transmisión están la fibra óptica, el cobre y las redes inalámbricas.

2

² the promise of sdn, network world digital spotlight,cisco may 13, 20113.

Características de la capa de infraestructura

Tamaño tabla de flujo.

Método de consultas.

Rendimiento y flexibilidad.

Rendimiento CPU.

CAPA DE CONTROL

Está basado en un conjunto de controladores SDN, que mediante un conjunto de APIs supervisan el comportamiento del envío de paquetes a través de una interfaz. Esta capa es el puente entre la capa de aplicación y la capa de infraestructura. Está conformado por cuatro componentes que son:

Condiciones para proceso de sincronización de la red.

Colección estado de la red.

Lenguaje de alto nivel.

Reglas para proceso de actualización.

Características de la capa de control

Rendimiento, escalabilidad y fiabilidad.

Seguridad.

Controles de partición.

CAPA DE APLICACIONES

Consiste en un conjunto de aplicaciones dirigidas al usuario final. Aquí las aplicaciones SDN pueden tener una visión general del estado de la red.

2.5 BENEFICIOS

Algunos de los beneficios esperados de una SDN son los siguientes:

- Simplicidad. La red puede ser controlada y gestionada de forma centralizada como una sola entidad. Las tareas de gestión son automatizadas y aisladas de la complejidad de la infraestructura física a través de interfaces fáciles de utilizar.
- Agilidad. Los nuevos servicios y aplicaciones pueden ser provistos en muy poco tiempo, además los administradores de T.I. obtienen la posibilidad de programar funcionalidades y servicios (*programmability*) según lo requieran, eliminando así la dependencia de los fabricantes de hardware.
- Control. A través del plano de control se puede supervisar de forma granular los flujos de la red, contemplando las aplicaciones como un elemento central. Permite lograr mejoras en la confiabilidad y seguridad de la red, centralizando la definición, configuración e implementación de políticas.
- Mejoras en la experiencia del usuario final. Al permitir que las aplicaciones exploten la información centralizada sobre el estado de la red y de la capacidad de adaptación del comportamiento de la misma, se pueden realizar ajustes en tiempo real para que los usuarios finales tengan un tiempo de respuesta de acuerdo a los niveles de servicio establecidos.
- Reducción de costos operativos. Finalmente, todo lo anterior redundando en el principal beneficio que se espera, que consiste en un menor costo operativo de gestión de las redes^[3].

3

³ Copyright © 2013 Magazciturum. All Rights Reserved. Magazine Premium created by c.bavota. [Consulta: 10 octubre 2014]. Disponible en: <http://www.magazciturum.com.mx/?p=2353>.

2.6 OPENFLOW

Es un estándar abierto, proporciona un conjunto de herramientas para ejecutar protocolos experimentales en las redes de datos.

En un conmutador convencional, la transferencia de paquetes y el enrutamiento de alto nivel se dan en el mismo host (dispositivo). Un conmutador OpenFlow separa la trayectoria de datos de la trayectoria de control. La porción de trayectoria de datos reside en el propio conmutador; un controlador separado toma las decisiones de enrutamiento de alto nivel. El conmutador y el controlador se comunican por medio del protocolo de OpenFlow.

OpenFlow es la primera interfaz estándar de comunicaciones definida entre el control y el forwarding de una arquitectura SDN. Openflow permite acceso directo y manipulación de los planes de forwarding de los dispositivos de la red como lo son switches y routers. Identifica el tráfico de la red basándose en reglas predefinidas que pueden ser estáticas o dinámicas, siendo dinámicas programadas por el software de control de las SDN. OpenFlow le permite a la red programar en función de cada flujo, y la arquitectura SDN basada en OpenFlow permite tener un control sobre cada aspecto de la red, permitiendo responder a cambios en tiempo real a nivel de usuarios, aplicación, y sesión. En la actualidad es el único protocolo SDN estandarizado que permite la manipulación directa de reenvío de los dispositivos de red.^[4]

4

Beneficios de OpenFlow con Redes SDN:

- Control sobre cualquier dispositivo de red OpenFlow habilitado (switches, routers, y switches virtuales).
- Reducción de gastos generales de operación, minimización de riesgos a causa de intervención humana, reducción de la complejidad a causa de la automatización.
- Adaptación a los requerimientos necesarios en tiempo real para prestar los servicios.^[5]

5

⁴ Openflow.(2011).Recuperado el 10 de 2014, de <http://www.openflowswitch.org/wk/index.php/OpenFlowTutorial>.

⁵ Erickson, D. (Aug. de 2013). The Beacon OpenFlow controller. In Proc. HotSDN. Recuperado de www.openflow.org

CAPITULO 3: REDES TRADICIONALES, MODELO OSI

3.1 El modelo OSI.

A finales de la década de los setenta, la Organización Internacional para la Normalización ISO inicio a desarrollar un modelo para la conexión en red, a este modelo se le denomino Modelo de referencia de interconexión de sistemas abiertos, bautizado con el nombre de Open Systems inteconnection Reference Model, por sus siglas en ingles OSI. Este modelo ofrece un marco de trabajo conceptual desde 1984.

El proceso de transmisión de la información entre equipos informáticos se divide en 7 capas según el modelo OSI, cada capa ejecuta una función. Dos de las pilas de protocolos que se utilizan para la transmisión de datos son: TCP/IP y AppleTalk.

Durante la comunicación entre sistemas, se producen una serie de eventos abarcados bajo el modelo OSI, estos son:

- El modo en el que el direccionamiento lógico de los paquetes pasa a convertirse en el direccionamiento físico que proporciona la red.
- El modo en que los datos se transmiten entre los distintos dispositivos.
- La forma en que se resuelve la secuencia y comprobación de errores.
- El modo en que los datos se traducen a un formato apropiado para la arquitectura de la red a utilizar.

Las únicas capas con las que interactúa el usuario son con la capa física y la capa de aplicación.^[7]

⁶]. Redes de telecomunicaciones, protocolos, modelado y análisis, Mischa Schwartz.

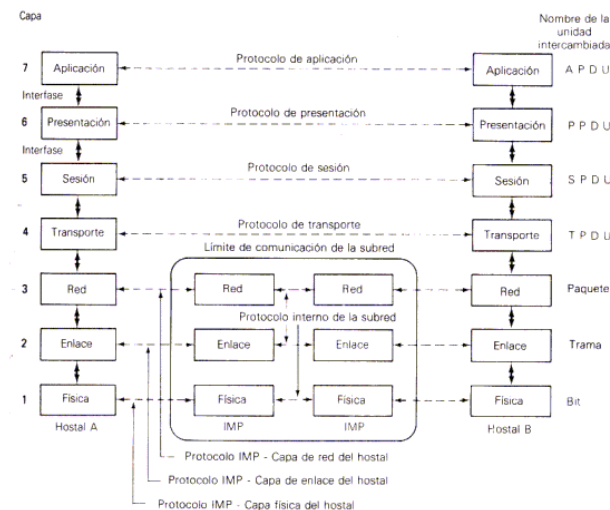
3.2 Arquitectura de comunicación por capas.

La arquitectura de comunicación por capas es quien impone los requerimientos funcionales que deben cumplir las estaciones de usuarios para asegurar que los datos se entreguen y reciban de forma adecuada.

En la comunicación de redes es necesario que las dos partes, proveedor y cliente, en una conversación o sesión a través de la red, estén de acuerdo con el establecimiento de la comunicación, al igual que acordar el formato de los datos que se entregarán a la terminal del cliente. La terminal debe ser capaz de regular la velocidad de la transmisión de datos (tasa de llegada de paquetes), de no ser así, el computador central que controla la base de datos fuente podría causar el desbordamiento de los datos en la terminal, si se usa transmisión sin conexión, los paquetes (datagramas) pueden llegar en desorden, la terminal receptora debe ser capaz de reconstruir la secuencia de paquetes. Estas y otras tareas afines no deben afectar la operación de la red, debe ser capaz de entregar correctamente y al lugar preciso los paquetes, aunque el resto del sistema no funcione en forma apropiada.

Normalmente este tipo de sistemas de control se incorporen por software a cada una de los dispositivos en la red, en el conmutador central, la termina, o cualquier otro sistema de comunicación, realizando de manera organizada la secuencia de actividades requeridas, dando lugar al concepto de arquitectura de comunicación por capas.

Figura 2. Estructura arquitectura de capas.



Fuente: http://www1.frm.utn.edu.ar/comunicaciones/modelo_osi.html

3.3 Protocolos de red

TCP/IP

Los protocolos de red son muy importantes ya que contribuyen a encaminar los paquetes en una conexión entre redes. Uno de los protocolos más conocidos es el TCP/IP, este se ha convertido en uno de los más reconocidos y un estándar para la conexión en red.

TCP/IP proporciona un conjunto de características referidas a la conexión en red, por lo que requiere de una gran carga para poder ejecutarse. También brinda un sistema de direccionamiento lógico y único. Un usuario que se conecte a internet navegara con una dirección IP de 32 bits.

Está compuesto de un conjunto de protocolos que forman la pila TCP/IP. Estos protocolos son:

FTP: protocolo de transferencia de archivos.

SMTP: protocolo de simple de transferencia de correo

TCP: protocolo de control de transporte.

UDP: protocolo de datagrama de usuario

IP: protocolo de internet.

ARP: protocolo de resolución de direcciones

NetBEUI

Interfaz ampliada de usuario para NetBIOS, es un protocolo de red que opera en la capas de transporte y red del modelo OSI. Fue diseñado por IBM y Microsoft para las redes pequeñas. Necesita funcionar con otro protocolo llamado NetBIOS que opera en la capa de sesión del modelo OSI para complementar su proceso. También incluye otros dos componentes que son: el redirector, permite que una computadora cliente perciba los recursos de la red suponiendo que son recursos locales, este opera en la capa de aplicación. Y el bloque de mensajes del servidor, permite comunicación de mismo nivel entre las maquinas clientes y servidor en sus respectivos redirectores, este opera en la capa de presentación del modelo OSI.^[8]

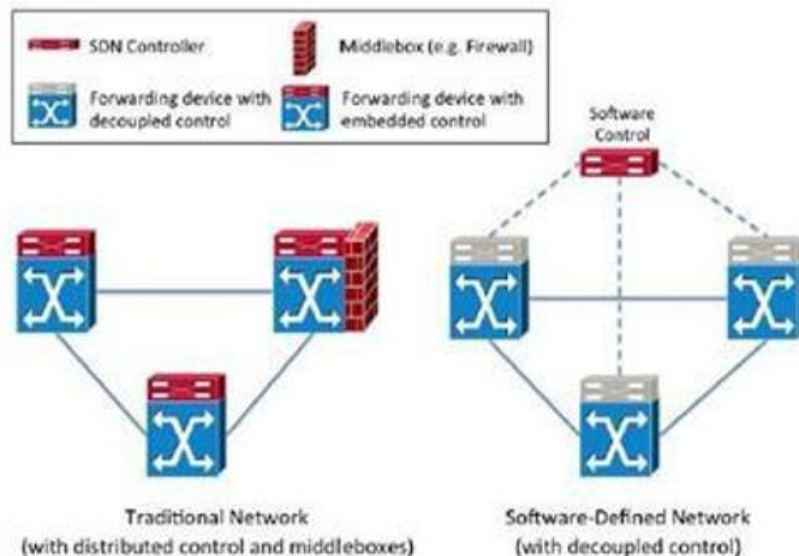
7

⁷ El modelo OSI y los protocolos de red. Disponible en http://blyx.com/public/docs/pila_OSI.pdf

CAPITULO 4: COMPARATIVO

El incremento en los costos de administración y operación que se observan en las distintas limitaciones de las redes actuales es una de las razones por las cuales se han desarrollado avances en las tecnologías de redes de computación durante varios años. Uno de esos avances son las Redes Definidas por Software (SDN), las redes SDN utilizan software para controlar los dispositivos, a diferencia de una red tradicional.

FIGURA 3. Comparación de una red SDN y una red tradicional.



Fuente:<http://ribuc.ucp.edu.co:8080/jspui/bitstream/handle/10785/2788/CDMIST97.pdf?sequence=2>

Las soluciones SDN mediante la automatización basada en políticas de servicios y mediante mecanismos de abstracción basados en estándares, permitirán la provisión y exportación respectivamente.

Una diferencia clave es el **ancho de banda**, los centros de datos gozan de un ancho de banda ilimitado, las redes tradicionales (como las MAN y las WAN) están sujetas a restricciones tecnológicas y de ancho de banda. Actualmente los proveedores de servicios satisfacen estas necesidades con procesos de planificación de red, están soportadas bajo los patrones de tráfico estáticos y predecibles. En las SDN las necesidades de ancho de banda que provoquen que estos patrones sean menos predecibles, se requerirá que las redes definidas por software brinden un modo de ajustar los recursos de red en tiempo real. ^[9]

8

TIEMPO DE VIDA (TTL)

Es un valor en el protocolo de internet (IP). En el uso de la multicast IP, el valor TTL indica el alcance o rango en el que un paquete puede ser enviado. Por convención:

- 0 se limita al mismo host
- 1 se limita a la misma subred
- 32 se restringe al mismo sitio
- 64 se limita a la misma región
- 128 se limita a un mismo continente
- 255 es sin restricciones

ANCHO DE BANDA

Es la cantidad de datos que se pueden enviar a través de la red en un lapso de tiempo determinado. Se expresa generalmente en bites por segundo (BPS), también se puede expresar en kilobites por segundo y en megabites por segundo. Generalmente el lapso de tiempo es un segundo y lo que se pretende conocer es medir la tasa de transferencia de datos que se pueden llevar de un punto a otro.

⁸ Soluciones SDN en las redes MAN/WAN: ¿revolución o evolución? – Tony Kourlas, enero 15 de 2014. Disponible en: <https://techzine.alcatel-lucent.com/es/soluciones-sdn-en-las-redes-manwan-revolucion-o-evolucion>

CONSIDERACIONES

A la hora de hacer el análisis comparativo se debe tener en cuenta:

- Los paquetes de datos que se envían en las pruebas son de diferentes tamaños, en Packet Tracer son de 32 bytes y en Mininet 64 bytes.
- El tiempo de vida en Packet Tracer es 126, es decir se limita a un mismo continente y en Mininet es 64, es decir se limita a la misma región.
- El tiempo de vida, es el tiempo en que tarda el paquete en ser recibido.

CÓDIGO EN MININET.

Código en Mininet para la elaboración de la topología con la que se realiza el análisis comparativo.

```
#!/usr/bin/phyton
```

```
From mininet.topo import Topo  
From mininet.net import Mininet
```

```
Class Mytopo(Topo):
```

```
    Def __init__(self,enable_all = True):
```

```
        Topo.__init__(self)
```

```
        Switch1 = self.addSwitch('s1')
```

```
        Switch2 = self.addSwitch('s2')
```

```
        Switch3 = self.addSwitch('s3')
```

```
        Switch4 = self.addSwitch('s4')
```

```
        Host0 = self.addHost('h0', ip = "192.168.0.1", defaultRoute =  
"192.168.0.7")
```

```
        Host1 = self.addHost('h1', ip = "192.168.0.2", defaultRoute =  
"192.168.0.7")
```

```
        Host2 = self.addHost('h2', ip = "192.168.0.3", defaultRoute =  
"192.168.0.7")
```

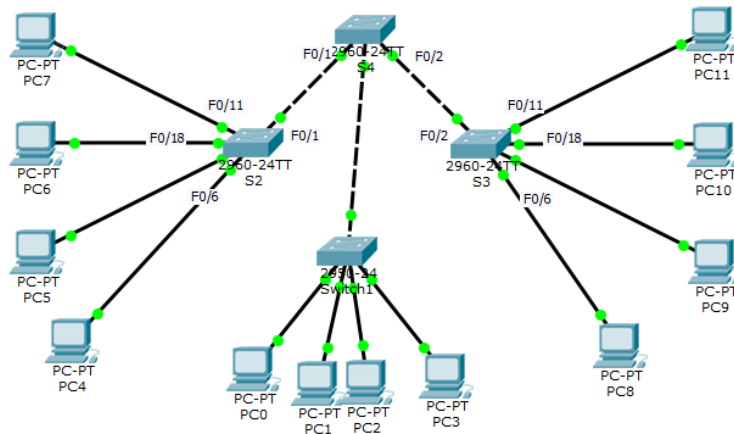
```
Host3 = self.addHost('h3', ip = "192.168.0.4", defaultRoute =
"192.168.0.7")
Host4 = self.addHost('h4', ip = "192.168.0.9", defaultRoute =
"192.168.0.15")
Host5 = self.addHost('h5', ip = "192.168.0.10", defaultRoute =
"192.168.0.15")
Host6 = self.addHost('h6', ip = "192.168.0.11", defaultRoute =
"192.168.0.15")
Host7 = self.addHost('h7', ip = "192.168.0.12", defaultRoute =
"192.168.0.15")
Host8 = self.addHost('h8', ip = "192.168.0.17", defaultRoute =
"192.168.0.23")
Host9 = self.addHost('h9', ip = "192.168.0.18", defaultRoute =
"192.168.0.23")
Host10 = self.addHost('h10', ip = "192.168.0.19", defaultRoute =
"192.168.0.23")
Host11 = self.addHost('h11', ip = "192.168.0.20", defaultRoute =
"192.168.0.23")
```

```
Self.addLink(host0,switch1)
Self.addLink(host1,switch1)
Self.addLink(host2,switch1)
Self.addLink(host3,switch1)
Self.addLink(host4,switch2)
Self.addLink(host5,switch2)
Self.addLink(host6,switch2)
Self.addLink(host7,switch2)
Self.addLink(host8,switch3)
Self.addLink(host9,switch3)
Self.addLink(host10,switch3)
Self.addLink(host11,switch3)
Self.addLink(switch1,switch4)
Self.addLink(switch2,switch4)
Self.addLink(switch3,switch4)
```

```
Topos = {'Mytopo' : (lambda:Mytopo())}
```

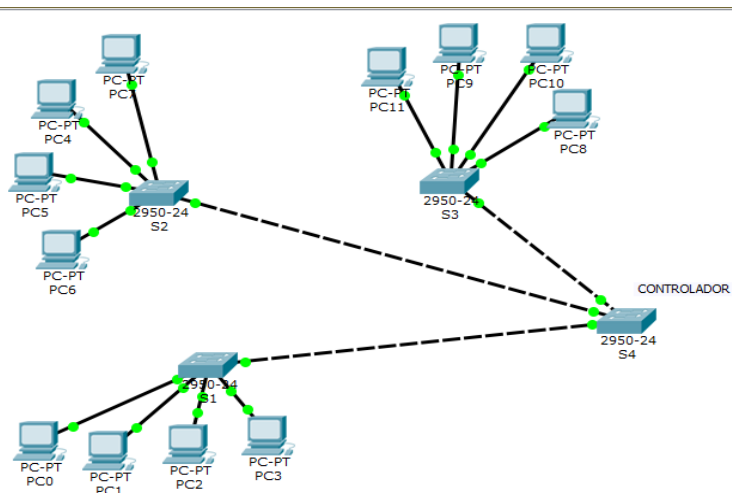
A continuación presentamos las dos redes utilizadas para la emulación, evaluando el uso del ancho de banda y tiempo de vida de los paquetes

Figura 4. Topología en Packet Tracer.



Representación de la topología en Packet Tracer. La red consta de 3 switches. Cada switch tiene conectados cuatro equipos, para un total de 12 equipos. El tipo de cables para la conexión fueron: de switch a switch cable cruzado, de pc a switch cable directo.

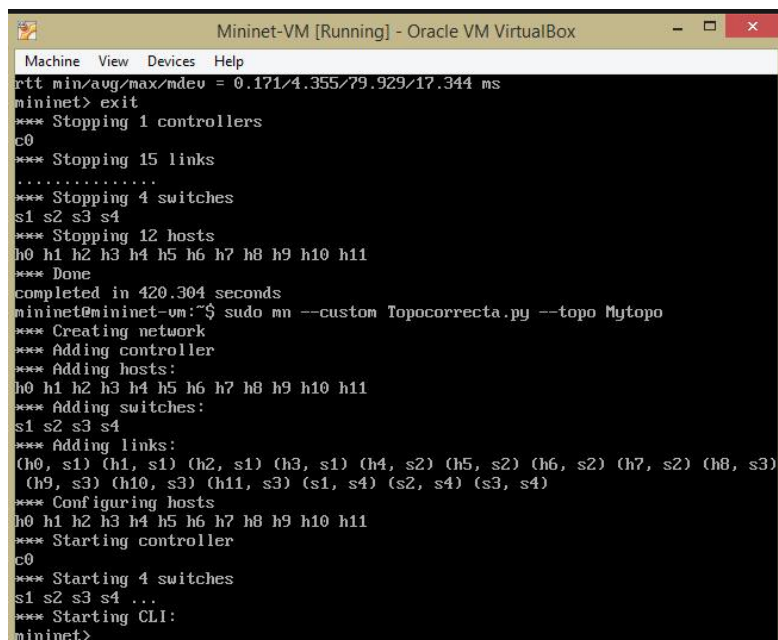
Figura 5. Topología en Mininet.



Representación de la topología en Mininet. La red consta de 4 switches, el switch 4 es el controlador, responsable de gestionar la información de reenvío de los conmutadores. Cada switch tiene conectados cuatro equipos, para un total de 12 equipos. El tipo de cables para la conexión fueron: de switch a switch cable cruzado, de pc a switch cable directo.

Luego de ejecutar el código en Mininet con el comando `sudo mn --custom topocorrecta.py --topo Mytopo`, obtenemos el siguiente resultado:

Figura 6. Ejecución topología en Mininet.



```
Mininet-VM [Running] - Oracle VM VirtualBox
Machine View Devices Help
rft min/aug/max/mdev = 0.171/4.355/79.929/17.344 ms
mininet> exit
*** Stopping 1 controllers
c0
*** Stopping 15 links
.....
*** Stopping 4 switches
s1 s2 s3 s4
*** Stopping 12 hosts
h0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11
*** Done
completed in 420.304 seconds
mininet@mininet-vm:~$ sudo mn --custom Topocorrecta.py --topo Mytopo
*** Creating network
*** Adding controller
*** Adding hosts:
h0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11
*** Adding switches:
s1 s2 s3 s4
*** Adding links:
(h0, s1) (h1, s1) (h2, s1) (h3, s1) (h4, s2) (h5, s2) (h6, s2) (h7, s2) (h8, s3)
(h9, s3) (h10, s3) (h11, s3) (s1, s4) (s2, s4) (s3, s4)
*** Configuring hosts
h0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11
*** Starting controller
c0
*** Starting 4 switches
s1 s2 s3 s4 ...
*** Starting CLI:
mininet>
```

La imagen muestra la creación de la red definida por software, esta consta de 4 switches y 12 host, además de un controlador.

4.1 Resultados

Prueba #1

En esta prueba, se realizaron diferentes PING entre pc1 y pc9 con un ancho de banda de 2gb, con el comando PING seguido de la IP asignada a diferentes nodos para chequear si está bien realizada la conexión y conocer la transmisión de datos en el tiempo.

También se realizó el PING para conocer a qué velocidad se están enviando y recibiendo archivos, además identificar el tamaño del paquete, su tiempo de vida y el tiempo transcurrido para llegar a su destino.

Figura 7. Ping entre pc1 y pc9 en Packet Tracer.

```
PC>ping 172.17.20.22

Pinging 172.17.20.22 with 32 bytes of data:

Reply from 172.17.20.22: bytes=32 time=20ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.22:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 20ms, Average = 5ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 8. Ping entre pc1 y pc9 en Mininet.

```
rtt min/avg/max/mdev = 0.239/27.735/107.258/45.927 ms
mininet> h1 ping -c 4 h9
PING 192.168.0.18 (192.168.0.18) 56(84) bytes of data.
64 bytes from 192.168.0.18: icmp_seq=1 ttl=64 time=2.01 ms
64 bytes from 192.168.0.18: icmp_seq=2 ttl=64 time=0.171 ms
64 bytes from 192.168.0.18: icmp_seq=3 ttl=64 time=0.217 ms
64 bytes from 192.168.0.18: icmp_seq=4 ttl=64 time=0.240 ms

--- 192.168.0.18 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.171/0.661/2.018/0.783 ms
mininet>
```

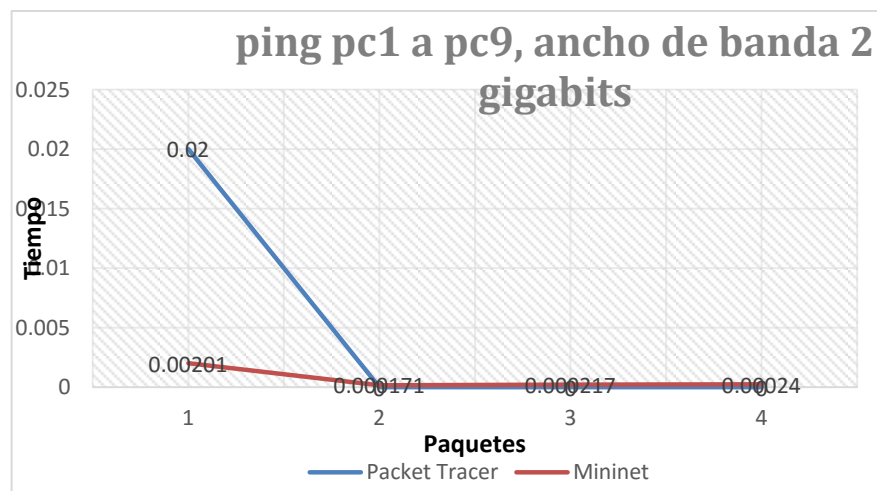
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 1. Datos ping entre pc1 y pc9.

Paquete	Tiempo(seg)	
	Packet Tracer	Mininet
1	0.02	0.00201
2	0	0.000171
3	0	0.000217
4	0	0.00024

Gráfica 1. Packet Tracer vs Mininet ping entre pc1 y pc9.



De la gráfica se puede observar que el tiempo de envío de los paquetes en las redes SDN, es más acelerado debido a que los switches consultan sus tablas de flujo antes de hacer el forwarding del tráfico y lo realizarán conforme a ellas. Si se da la circunstancia de no saber qué hacer con el tráfico, preguntaran al controlador y este les dirá que hacer, lo que la convertirá en una red gestionada de manera centralizada. En las redes Ethernet consultarán las tablas MAC, en caso de no conocer la ruta, se envía una solicitud de broadcast ARP a todos los nodos esperando respuesta lo cual genera un retardo inicial para el envío de las tramas.

Prueba #2

En esta prueba, se realizaron diferentes PING extendidos entre pc1 y pc9 con un ancho de banda de 2gb, enviando 22 solicitudes a 32 bytes cada una. Se uso la opción `-t` que hace un ping al host especificado hasta que se detenga para ver las estadísticas y continuar. Para detener se hace uso de las teclas control-C.

Figura 9. Ping extendido entre pc1 y pc9 en Packet Tracer.

```
Command Prompt
PC>ping -t 172.17.20.22

Pinging 172.17.20.22 with 32 bytes of data:

Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=2ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=8ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=3ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=3ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
```

Capturas realizadas el 1 de diciembre de 2015

Figura 10. Ping extendido entre pc1 y pc9 con un ancho de banda de 2gb en Mininet.

```
Mininet-VM [Running] - Oracle VM VirtualBox
Machine View Devices Help
mininet> h1 ping -c 23 h9
PING 192.168.0.18 (192.168.0.18) 56(84) bytes of data:
64 bytes from 192.168.0.18: icmp_seq=1 ttl=64 time=87.6 ms
64 bytes from 192.168.0.18: icmp_seq=2 ttl=64 time=10.8 ms
64 bytes from 192.168.0.18: icmp_seq=3 ttl=64 time=1.84 ms
64 bytes from 192.168.0.18: icmp_seq=4 ttl=64 time=0.181 ms
64 bytes from 192.168.0.18: icmp_seq=5 ttl=64 time=0.416 ms
64 bytes from 192.168.0.18: icmp_seq=6 ttl=64 time=0.200 ms
64 bytes from 192.168.0.18: icmp_seq=7 ttl=64 time=1.91 ms
64 bytes from 192.168.0.18: icmp_seq=8 ttl=64 time=0.242 ms
64 bytes from 192.168.0.18: icmp_seq=9 ttl=64 time=0.371 ms
64 bytes from 192.168.0.18: icmp_seq=10 ttl=64 time=0.154 ms
64 bytes from 192.168.0.18: icmp_seq=11 ttl=64 time=0.344 ms
64 bytes from 192.168.0.18: icmp_seq=12 ttl=64 time=0.331 ms
64 bytes from 192.168.0.18: icmp_seq=13 ttl=64 time=0.291 ms
64 bytes from 192.168.0.18: icmp_seq=14 ttl=64 time=0.644 ms
64 bytes from 192.168.0.18: icmp_seq=15 ttl=64 time=0.279 ms
64 bytes from 192.168.0.18: icmp_seq=16 ttl=64 time=0.347 ms
64 bytes from 192.168.0.18: icmp_seq=17 ttl=64 time=0.379 ms
64 bytes from 192.168.0.18: icmp_seq=18 ttl=64 time=0.452 ms
64 bytes from 192.168.0.18: icmp_seq=19 ttl=64 time=0.169 ms
64 bytes from 192.168.0.18: icmp_seq=20 ttl=64 time=0.651 ms
64 bytes from 192.168.0.18: icmp_seq=21 ttl=64 time=0.307 ms
64 bytes from 192.168.0.18: icmp_seq=22 ttl=64 time=0.148 ms
64 bytes from 192.168.0.18: icmp_seq=23 ttl=64 time=0.307 ms

--- 192.168.0.18 ping statistics ---
23 packets transmitted, 23 received, 0% packet loss, time 22044ms
rtt min/avg/max/ndev = 0.148/4.713/87.603/17.803 ms
mininet>
```

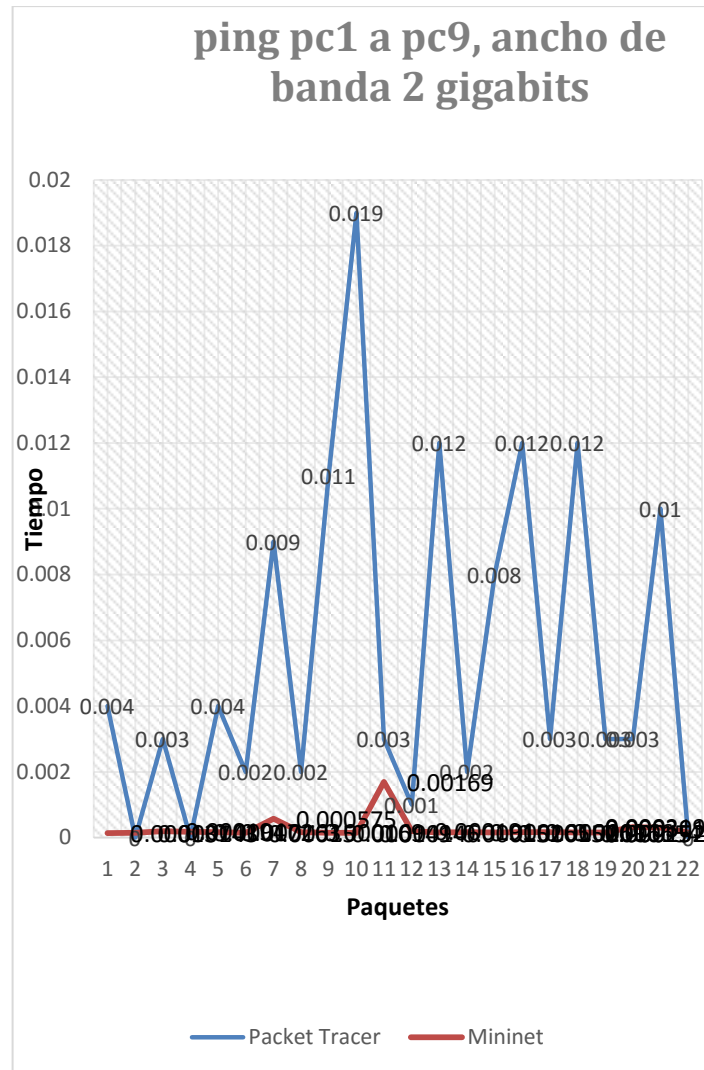
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 2. Datos ping extendido entre pc1 y pc9.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0	0.00201
2	0	0.000171
3	0.001	0.000217
4	0	0.00024
5	0	0.0876
6	0	0.0108
7	0	0.00184
8	0	0.000181
9	0	0.000416
10	0	0.0002
11	0	0.00191
12	0	0.000242
13	0	0.000371
14	0	0.000154
15	0	0.000344
16	0	0.000331
17	0.007	0.000291
18	0.019	0.000644
19	0	0.000279
20	0	0.000347
21	0	0.000379
22	0.001	0.000452

Gráfica 2. Packet Tracer vs Mininet ping extendido entre pc1 y pc9.



De la gráfica se puede evidenciar que entre más paquetes se envían, se necesita más uso de ancho de banda en las redes SDN. En prueba limite Mininet muestra un ancho de banda máximo de 30 Gbits/segundo condicionado al número de host virtualizados. Los picos y las oscilaciones representados en la gráfica, representan como las redes Ethernet varían en el tiempo. El primer pico representa el retardo que tienen las redes convencionales al no tener la ruta de envío de paquetes identificada porque no están las tablas ARP.

Prueba #3

En esta prueba, se realizaron diferentes PING entre pc1 y pc5 con un ancho de banda de 3gb. Se realizaron los PING para conocer a qué velocidad se está enviando y recibiendo archivos, además identificar el tamaño del paquete, su tiempo de vida y el tiempo trascurrido para llegar a su destino.

Figura 11. Ping entre pc1 y pc5 en Packet Tracer.

```
PC>
PC>ping 172.17.20.25

Pinging 172.17.20.25 with 32 bytes of data:

Reply from 172.17.20.25: bytes=32 time=24ms TTL=128
Reply from 172.17.20.25: bytes=32 time=14ms TTL=128
Reply from 172.17.20.25: bytes=32 time=7ms TTL=128
Reply from 172.17.20.25: bytes=32 time=11ms TTL=128

Ping statistics for 172.17.20.25:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 7ms, Maximum = 24ms, Average = 14ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 12. Ping entre pc1 y pc5 con un ancho de banda de 3 gb en Mininet.

```
rtt min/avg/max/mdev = 0.229/18.041/69.073/29.480 ms
mininet> h1 ping -c 4 h5
PING 192.168.0.10 (192.168.0.10) 56(84) bytes of data.
64 bytes from 192.168.0.10: icmp_seq=1 ttl=64 time=2.01 ms
64 bytes from 192.168.0.10: icmp_seq=2 ttl=64 time=0.418 ms
64 bytes from 192.168.0.10: icmp_seq=3 ttl=64 time=0.454 ms
64 bytes from 192.168.0.10: icmp_seq=4 ttl=64 time=0.445 ms

--- 192.168.0.10 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 0.418/0.832/2.014/0.683 ms
mininet>
```

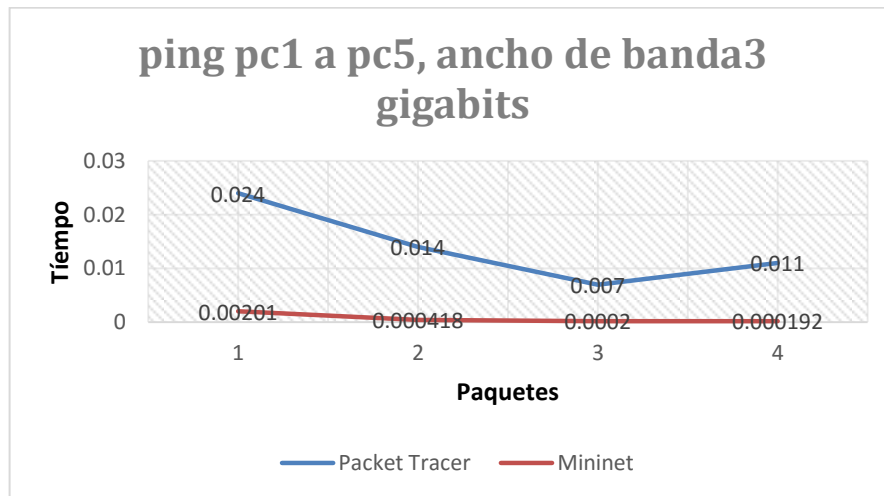
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 3. Datos ping entre pc1 y pc5.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.024	0.00201
2	0.014	0.000418
3	0.007	0.0002
4	0.011	0.000192

Gráfica 3. Packet Tracer vs Mininet ping entre pc1 y pc5.



De la grafica se puede evidenciar que las redes SDN realizan una búsqueda para trazar la ruta y guardarla. Las redes SDN evitan que el tiempo entre los paquetes sea mayor, logrando disminuir el tiempo en el recorrido de los paquetes por sus rutas. Cuando se cambia de segmento en la red, en las redes Ethernet se produce una demora debido a la consulta de la ruta indicada mediante las tablas ARP para el envío de los paquetes de datos.

Prueba #4

En esta prueba, se realizaron diferentes PING extendidos entre pc1 y pc5 con un ancho de banda de 3gb, enviando 6 solicitudes a 32 bytes cada una. Se uso la opción -n donde n es el número de solicitudes de eco para enviar.

Figura 13. Ping extendido entre pc1 y pc5 en Packet Tracer.

```
PC>ping -n 6 172.17.20.25

Pinging 172.17.20.25 with 32 bytes of data:

Reply from 172.17.20.25: bytes=32 time=4ms TTL=128
Reply from 172.17.20.25: bytes=32 time=1ms TTL=128
Reply from 172.17.20.25: bytes=32 time=5ms TTL=128
Reply from 172.17.20.25: bytes=32 time=14ms TTL=128
Reply from 172.17.20.25: bytes=32 time=2ms TTL=128
Reply from 172.17.20.25: bytes=32 time=9ms TTL=128

Ping statistics for 172.17.20.25:
    Packets: Sent = 6, Received = 6, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 14ms, Average = 5ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 14. Ping extendido entre pc1 y pc5 con un ancho de banda de 3 gb en Mininet.

```
--- 192.168.0.18 ping statistics ---
23 packets transmitted, 23 received, 0% packet loss, time 22044ms
rtt min/avg/max/mdev = 0.148/4.713/87.603/17.803 ms
mininet> h1 ping -c 6 h5
PING 192.168.0.10 (192.168.0.10) 56(84) bytes of data.
64 bytes from 192.168.0.10: icmp_seq=1 ttl=64 time=85.2 ms
64 bytes from 192.168.0.10: icmp_seq=2 ttl=64 time=9.03 ms
64 bytes from 192.168.0.10: icmp_seq=3 ttl=64 time=0.158 ms
64 bytes from 192.168.0.10: icmp_seq=4 ttl=64 time=0.285 ms
64 bytes from 192.168.0.10: icmp_seq=5 ttl=64 time=0.295 ms
64 bytes from 192.168.0.10: icmp_seq=6 ttl=64 time=0.327 ms

--- 192.168.0.10 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5006ms
rtt min/avg/max/mdev = 0.158/15.893/85.262/31.187 ms
mininet>
mininet>
mininet>
mininet>
mininet>
```

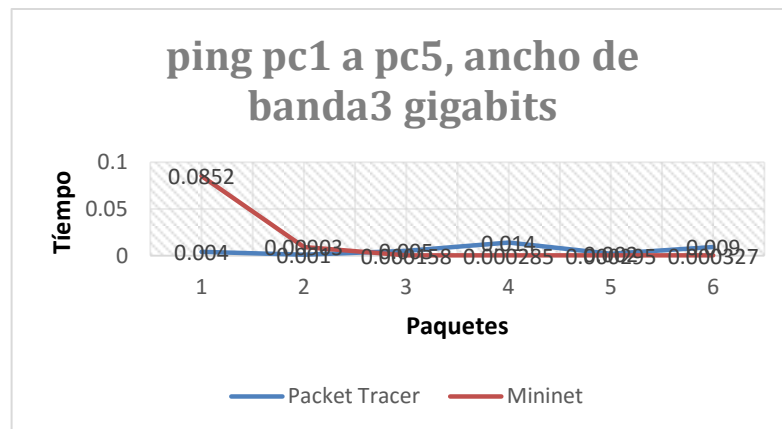
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 4. Datos ping extendido entre pc1 y pc5.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.04	0.0852
2	0.001	0.00903
3	0.005	0.000158
4	0.014	0.000285
5	0.002	0.000295
6	0.009	0.000327

Gráfica 4. Packet Tracer vs Mininet ping extendido entre pc1 y pc5.



En la gráfica se puede evidenciar la variación del tiempo en el envío de los paquetes en cada uno de los aplicativos que se usaron para las pruebas (Packet Tracer, Mininet). Las redes SDN evitan que la cantidad de saltos sea de un rango mayor. En las redes SDN el envío del primer paquete tiene un retardo en el tiempo, debido al tiempo de consulta al controlador. En las redes Ethernet como ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes, estabilizando el tiempo de envío de datos.

Prueba #5

En esta prueba, se realizaron diferentes PING entre pc6 y pc4 con un ancho de banda de 3gb, con el comando PING seguido de la IP asignada a diferente nodo para chequear si está bien realizada la conexión y conocer la transmisión de datos en tiempo.

Figura 15. Ping entre pc6 y pc4 en Packet Tracer.

```
PC>ping 172.17.20.21

Pinging 172.17.20.21 with 32 bytes of data:

Reply from 172.17.20.21: bytes=32 time=0ms TTL=128
Reply from 172.17.20.21: bytes=32 time=3ms TTL=128
Reply from 172.17.20.21: bytes=32 time=1ms TTL=128
Reply from 172.17.20.21: bytes=32 time=1ms TTL=128

Ping statistics for 172.17.20.21:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 1ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 16. Ping entre pc6 y pc4 en Mininet.

```
mininet> h6 ping -c 4 h4
PING 192.168.0.9 (192.168.0.9) 56(84) bytes of data:
64 bytes from 192.168.0.9: icmp_seq=1 ttl=64 time=1.23 ms
64 bytes from 192.168.0.9: icmp_seq=2 ttl=64 time=1.36 ms
64 bytes from 192.168.0.9: icmp_seq=3 ttl=64 time=0.200 ms
64 bytes from 192.168.0.9: icmp_seq=4 ttl=64 time=0.192 ms

--- 192.168.0.9 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.192/0.747/1.361/0.552 ms
mininet>
```

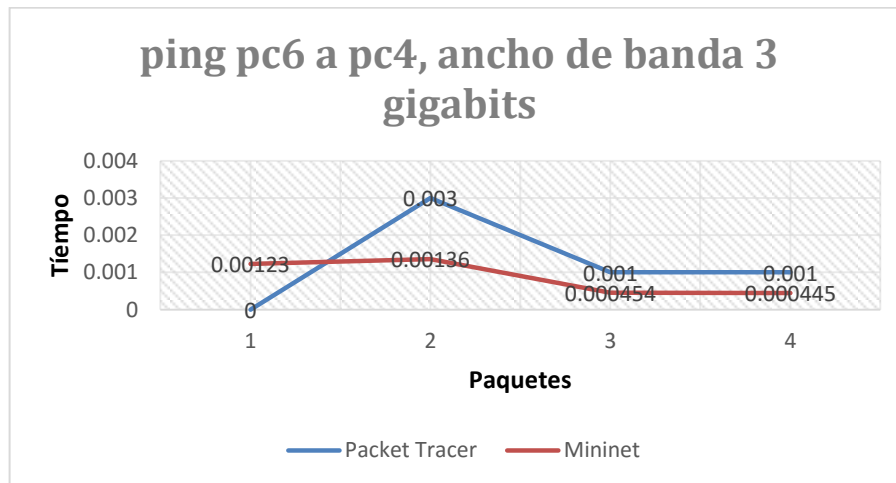
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 5. Datos ping entre pc6 y pc4.

Paquete	tiempo	
	Packet Tracer	Mininet
1	0	0.00123
2	0.003	0.00136
3	0.001	0.000454
4	0.001	0.000445

Gráfica 5. Packet Tracer vs Mininet entre pc6 y pc4.



De la gráfica se puede concluir que en las redes definidas por software el primer paquete utiliza una mayor cantidad de tiempo en ser enviado, debido a que estas realizan una consulta al controlador para que este les devuelva la ruta indicada. Las redes ethernet logran disminuir el tiempo en el recorrido de los paquetes estabilizando el tiempo, debido a que ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes de datos.

Prueba 6

En esta prueba, se realizaron diferentes PING extendidos entre pc6 y pc4 con un ancho de banda de 3gb, enviando 6 solicitudes a 32 bytes cada una. Se uso la opción -n donde n es el número de solicitudes de eco para enviar.

Figura 17. Ping extendido entre pc6 y pc4 con un ancho de banda de 3 gb en Packet Tracer.

```
PC>ping -n 6 172.17.20.22

Pinging 172.17.20.22 with 32 bytes of data:

Reply from 172.17.20.22: bytes=32 time=11ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.22:
    Packets: Sent = 6, Received = 6, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 11ms, Average = 1ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 18. Ping extendido entre pc6 y pc4 con un ancho de banda de 3 gb en Mininet.

```
Mininet-VM [Running] - Oracle VM VirtualBox
Machine View Devices Help
--- 192.168.0.10 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5006ms
rtt min/avg/max/mdev = 0.158/15.893/85.262/31.187 ms
mininet>
mininet>
mininet>
mininet> h6 ping -c 6 h4
PING 192.168.0.9 (192.168.0.9) 56(84) bytes of data:
64 bytes from 192.168.0.9: icmp_seq=1 ttl=64 time=142 ms
64 bytes from 192.168.0.9: icmp_seq=2 ttl=64 time=1.17 ms
64 bytes from 192.168.0.9: icmp_seq=3 ttl=64 time=0.154 ms
64 bytes from 192.168.0.9: icmp_seq=4 ttl=64 time=0.431 ms
64 bytes from 192.168.0.9: icmp_seq=5 ttl=64 time=0.145 ms
64 bytes from 192.168.0.9: icmp_seq=6 ttl=64 time=0.158 ms

--- 192.168.0.9 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5055ms
rtt min/avg/max/mdev = 0.145/24.030/142.121/52.813 ms
mininet>
mininet>
mininet> h11 ping -c 2 h12
PING 192.168.0.25 (192.168.0.25) 56(84) bytes of data:
64 bytes from 192.168.0.25: icmp_seq=1 ttl=64 time=95.0 ms
64 bytes from 192.168.0.25: icmp_seq=2 ttl=64 time=1.80 ms

--- 192.168.0.25 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1005ms
rtt min/avg/max/mdev = 1.809/48.427/95.045/46.618 ms
mininet>
```

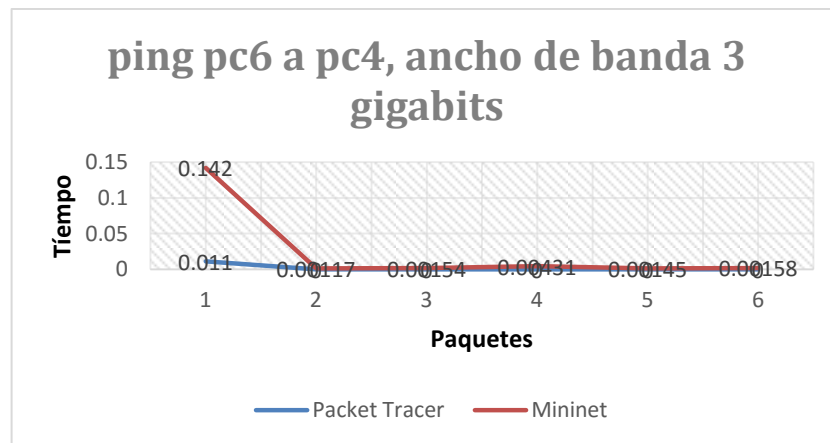
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 6. Datos ping extendido entre pc6 y pc4.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.011	0.142
2	0	0.00117
3	0	0.000154
4	0	0.000431
5	0	0.000145
6	0	0.000158

Gráfica 6. Packet Tracer vs Mininet ping extendido entre pc6 y pc4.



En la gráfica se muestra un ping realizado entre dos host ubicados bajo un mismo switch, esto hace que las redes Ethernet no toma valores de tiempo para el envío de los paquetes ya que no tiene saltos significativos, por el contrario las redes SDN deben realizar una búsqueda de las rutas que le puedan redireccionar a su destino, por esta razón tiene sus respectivos tiempos para los paquetes enviados. Para redes locales las redes SDN no plantean una buena eficacia ya que busca por todos los nodos posibles el camino más eficiente para llegar a su destino, manejando la red por toda su dimensión.

Prueba #7

En esta prueba, se realizaron diferentes PING entre pc11 y pc12 con un ancho de banda de 2gb, identificando el tamaño del paquete, su tiempo de vida y el tiempo transcurrido para llegar a su destino.

Figura 19. Ping entre pc11 y pc12 con un ancho de banda de 2 gb en Packet Tracer.

```
PC>ping 172.17.20.22

Pinging 172.17.20.22 with 32 bytes of data:

Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.22:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 20. Ping entre pc11 y pc12 en Mininet.

```
mininet> h11 ping -c 4 h12
PING 192.168.0.25 (192.168.0.25) 56(84) bytes of data.
64 bytes from 192.168.0.25: icmp_seq=1 ttl=64 time=0.141 ms
64 bytes from 192.168.0.25: icmp_seq=2 ttl=64 time=3.10 ms
64 bytes from 192.168.0.25: icmp_seq=3 ttl=64 time=0.240 ms
64 bytes from 192.168.0.25: icmp_seq=4 ttl=64 time=0.227 ms

--- 192.168.0.25 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.141/0.927/3.101/1.255 ms
mininet>
```

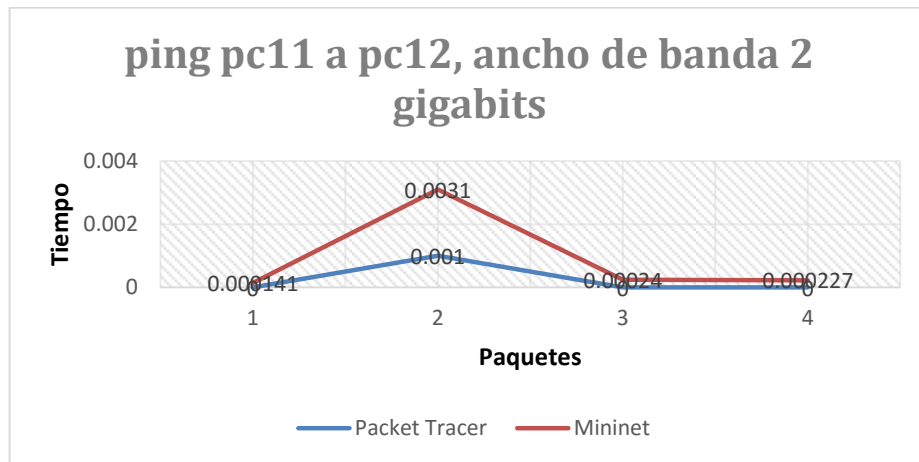
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 7. Datos ping pc11 a pc12.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0	0.000141
2	0.001	0.0031
3	0	0.00024
4	0	0.000227

Grafica 7. Packet Tracer vs Mininet ping entre pc11 y pc12.



De la gráfica se puede evidenciar que la transmisión de datos es más rápida en las redes Ethernet. Para esta prueba, los equipos se encuentran en un mismo segmento de red y los hosts están conectados a un mismo switch. En el envío de datos no se realiza búsqueda, por eso los intervalos entre los tiempos son cercanos, tienden a ser una línea recta. En las redes SDN, se evidencia un pico, representa el retardo en tiempo que existe cuando se hace una consulta al controlador.

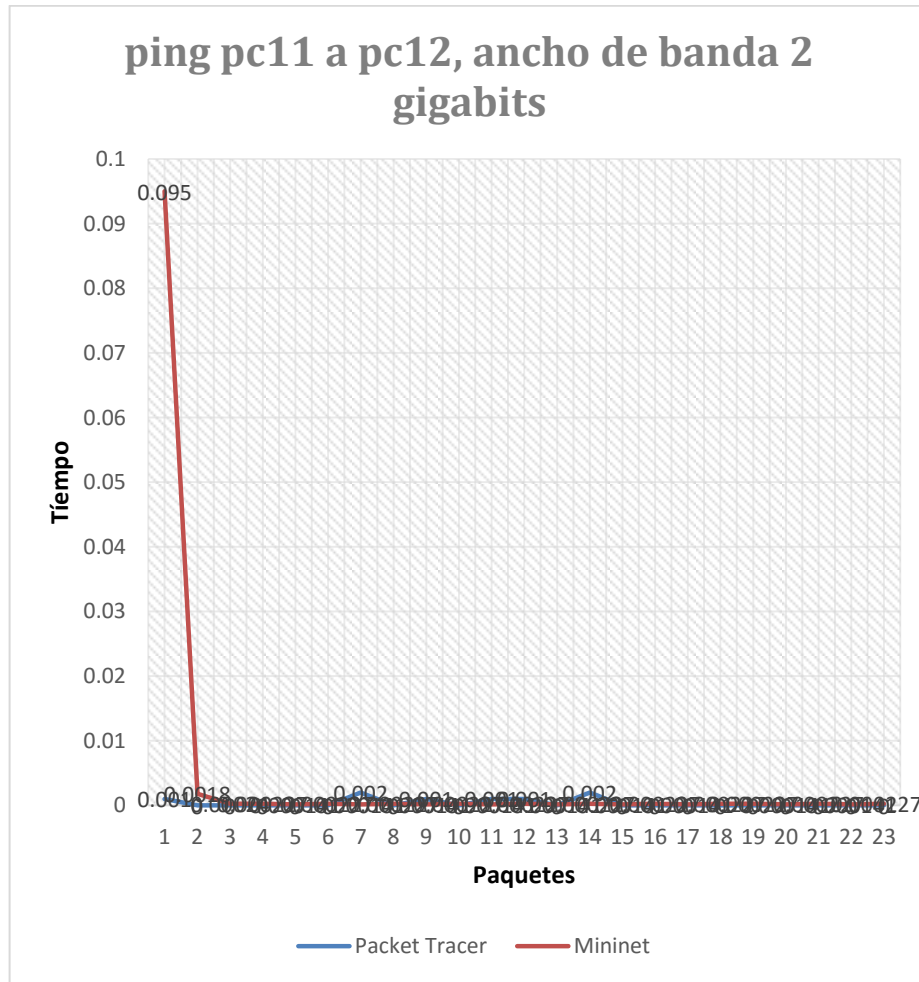
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 8. Datos ping extendido pc11 a pc12.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.001	0.095
2	0	0.0018
3	0	0.00024
4	0	0.000227
5	0	0.000141
6	0	0.000227
7	0.002	0.000141
8	0	0.000227
9	0.001	0.000141
10	0	0.000227
11	0.001	0.000141
12	0.001	0.000227
13	0	0.000141
14	0.002	0.000227
15	0	0.000141
16	0	0.000227
17	0	0.000141
18	0	0.000227
19	0	0.000227
20	0	0.000141
21	0	0.000227
22	0	0.000141
23	0	0.000227

Gráfica 8. Packet Tracer vs Mininet ping extendido entre pc11 y pc12.



De la grafica se puede evidenciar que los tiempos tienden a ser constantes, representándose en una línea recta. Un claro ejemplo de ellos es en red de área local donde el tiempo de envío de los paquetes tiende a ser cero. En las redes SDN el envío del primer paquete tiene un retardo en el tiempo, debido al tiempo de consulta al controlador. En las redes Ethernet como ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes, estabilizando el tiempo de envío de datos.

Prueba #9

En esta prueba, se realizaron diferentes PING entre pc2 y pc3 con un ancho de banda de 3.5gb, identificando el tiempo transcurrido de cada paquete para llegar a su destino.

Figura 23. Ping entre pc2 y pc3 y en Packet Tracer.

```
Command Prompt

Packet Tracer PC Command Line 1.0
PC>ping 172.17.20.21

Pinging 172.17.20.21 with 32 bytes of data:

Reply from 172.17.20.21: bytes=32 time=1ms TTL=128
Reply from 172.17.20.21: bytes=32 time=1ms TTL=128
Reply from 172.17.20.21: bytes=32 time=0ms TTL=128
Reply from 172.17.20.21: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.21:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 24. Ping entre pc2 y pc3 en Mininet.

```
mininet> h2 ping -c 4 h3
PING 192.168.0.4 (192.168.0.4) 56(84) bytes of data:
64 bytes from 192.168.0.4: icmp_seq=1 ttl=64 time=1.19 ms
64 bytes from 192.168.0.4: icmp_seq=2 ttl=64 time=0.192 ms
64 bytes from 192.168.0.4: icmp_seq=3 ttl=64 time=0.135 ms
64 bytes from 192.168.0.4: icmp_seq=4 ttl=64 time=0.191 ms

--- 192.168.0.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.135/0.427/1.192/0.442 ms
mininet>
```

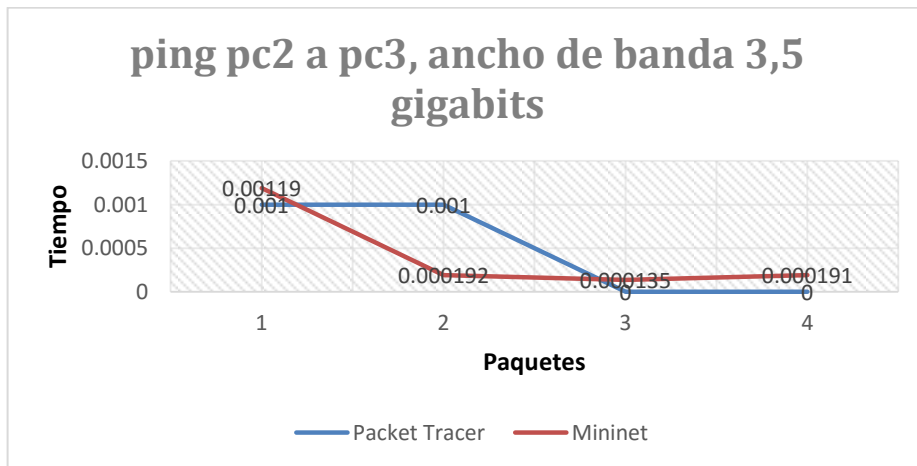
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 9. Datos entre pc2 y pc3.

Paquete	Tiempo	
	Packet Tracer	Mininet
1	0.001	0.00119
2	0.001	0.000192
3	0	0.000135
4	0	0.000191

Gráfica 9. Packet Tracer vs Mininet ping entre pc2 y pc3.

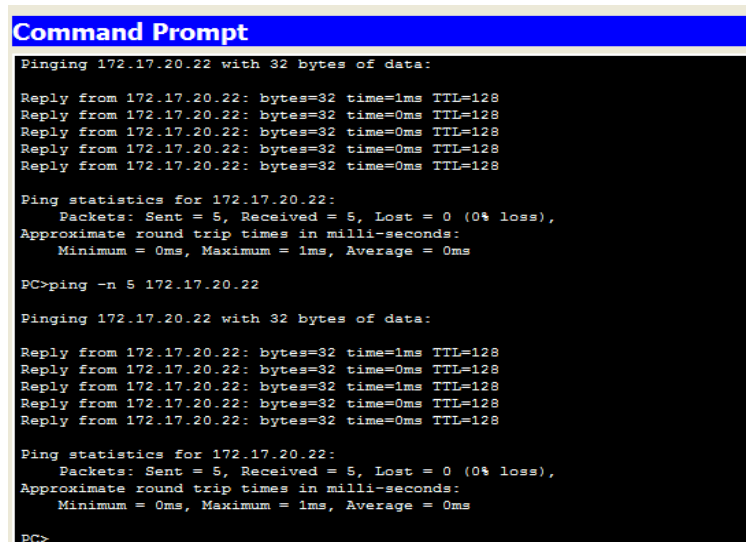


De la grafica se puede concluir que el tiempo de vida de los paquetes en Packet Tracer se limita a un mismo continente y en Mininet se limita a la misma región. El valor en el campo TTL en la ejecución del ping determinan la región. En las redes SDN el envío del primer paquete tiene un retardo en el tiempo, debido al tiempo de consulta al controlador. En las redes Ethernet como ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes, estabilizando el tiempo de envío de datos.

Prueba # 10

En esta prueba, se realizaron diferentes PING extendidos entre pc2 y pc3 con un ancho de banda de 3.5gb, enviando 5 solicitudes a 32 bytes cada una. Se uso la opción -n donde n es el número de solicitudes de eco para enviar.

Figura 25. Ping extendido entre pc2 y pc3 en Packet Tracer.



```
Command Prompt
Pinging 172.17.20.22 with 32 bytes of data:
Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.22:
    Packets: Sent = 5, Received = 5, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>ping -n 5 172.17.20.22

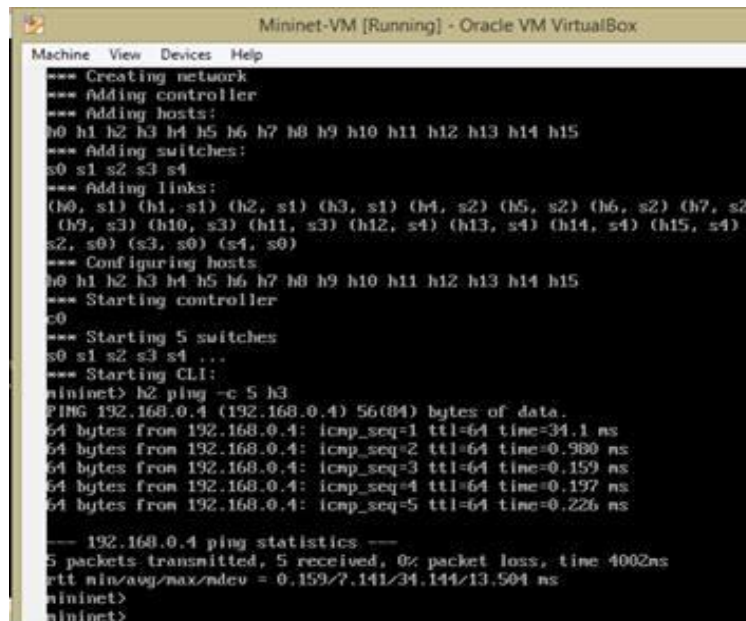
Pinging 172.17.20.22 with 32 bytes of data:
Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=1ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128
Reply from 172.17.20.22: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.22:
    Packets: Sent = 5, Received = 5, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015

Figura 26. Ping extendido entre pc2 y pc3 en Mininet.



```
Mininet-VM [Running] - Oracle VM VirtualBox
Machine View Devices Help
*** Creating network
*** Adding controller
*** Adding hosts:
h0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15
*** Adding switches:
s0 s1 s2 s3 s4
*** Adding links:
(h0, s1) (h1, s1) (h2, s1) (h3, s1) (h4, s2) (h5, s2) (h6, s2) (h7, s2)
(h8, s3) (h9, s3) (h10, s3) (h11, s3) (h12, s4) (h13, s4) (h14, s4) (h15, s4)
(s2, s0) (s3, s0) (s4, s0)
*** Configuring hosts
h0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15
*** Starting controller
c0
*** Starting 5 switches
s0 s1 s2 s3 s4 ...
*** Starting CLI:
mininet> h2 ping -c 5 h3
PING 192.168.0.4 (192.168.0.4) 56(84) bytes of data:
64 bytes from 192.168.0.4: icmp_seq=1 ttl=64 time=34.1 ms
64 bytes from 192.168.0.4: icmp_seq=2 ttl=64 time=0.980 ms
64 bytes from 192.168.0.4: icmp_seq=3 ttl=64 time=0.159 ms
64 bytes from 192.168.0.4: icmp_seq=4 ttl=64 time=0.197 ms
64 bytes from 192.168.0.4: icmp_seq=5 ttl=64 time=0.226 ms

--- 192.168.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 0.159/?.141/34.144/13.504 ms
mininet>
mininet>
```

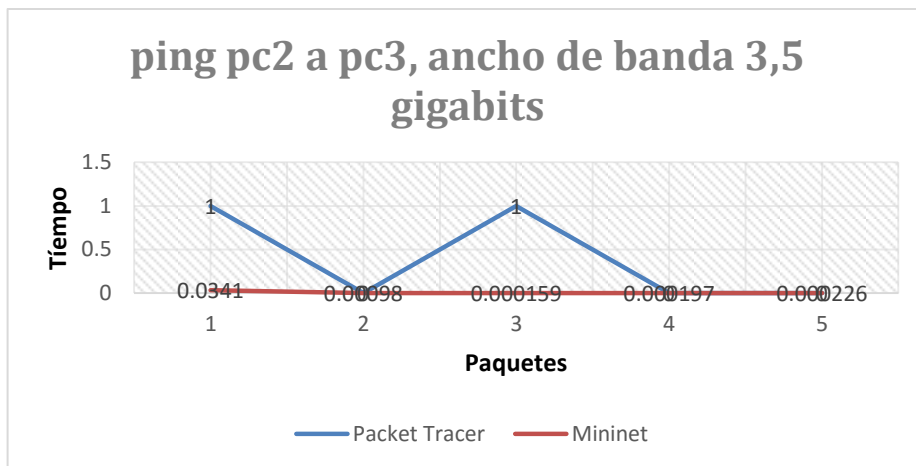
Capturas realizadas el 1 de diciembre de 2015

A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 10. Datos ping extendido entre pc2 y pc3.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	1	0.0341
2	0	0.00098
3	1	0.000159
4	0	0.000197
5	0	0.000226

Gráfica 10. PACKET Tracer vs Mininet ping extendido entre pc2 y pc3.



De la gráfica se puede concluir que los paquetes de datos que se envían en las pruebas son de diferentes tamaños, en las redes Ethernet son de 32 bytes y en las redes definidas por software de 64 bytes. Aún así las redes SDN utilizando paquetes de datos con el doble de tamaño estabilizan los tiempos en la transmisión de datos. En un mismo segmento de red la transmisión de datos es más rápida. En las redes Ethernet, el primer paquete sufre un retardo en el tiempo de envío, debido a que no se ha identificado la ruta de transmisión mediante las tablas ARP.

Prueba #11

En esta prueba, se realizaron diferentes PING extendidos entre pc0 y pc7 con un ancho de banda de 3.5gb, enviando 32 solicitudes a 32 bytes cada una. Se uso la opción -n donde n es el número de solicitudes de eco para enviar.

El objetivo de la prueba, es determinar cómo los tiempos aumentan cuando el uso de ancho de banda incrementa porque la red tiene un tráfico mayor. Durante el ping, se realizaron otros ping para incrementar el tráfico de la red.

Figura 27. Ping extendido entre pc0 y pc7 en Packet Tracer.

```
Command Prompt
PC>ping -n 32 172.17.20.25

Pinging 172.17.20.25 with 32 bytes of data:

Reply from 172.17.20.25: bytes=32 time=4ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=4ms TTL=128
Reply from 172.17.20.25: bytes=32 time=2ms TTL=128
Reply from 172.17.20.25: bytes=32 time=9ms TTL=128
Reply from 172.17.20.25: bytes=32 time=2ms TTL=128
Reply from 172.17.20.25: bytes=32 time=11ms TTL=128
Reply from 172.17.20.25: bytes=32 time=19ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=1ms TTL=128
Reply from 172.17.20.25: bytes=32 time=12ms TTL=128
Reply from 172.17.20.25: bytes=32 time=2ms TTL=128
Reply from 172.17.20.25: bytes=32 time=8ms TTL=128
Reply from 172.17.20.25: bytes=32 time=12ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=10ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=1ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=18ms TTL=128
Reply from 172.17.20.25: bytes=32 time=18ms TTL=128
```

Figura 28. Complementación Ping extendido entre pc0 y pc7 en Packet Tracer.

```
Command Prompt
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=1ms TTL=128
Reply from 172.17.20.25: bytes=32 time=12ms TTL=128
Reply from 172.17.20.25: bytes=32 time=2ms TTL=128
Reply from 172.17.20.25: bytes=32 time=8ms TTL=128
Reply from 172.17.20.25: bytes=32 time=12ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=3ms TTL=128
Reply from 172.17.20.25: bytes=32 time=10ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=1ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=18ms TTL=128
Reply from 172.17.20.25: bytes=32 time=18ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128
Reply from 172.17.20.25: bytes=32 time=5ms TTL=128
Reply from 172.17.20.25: bytes=32 time=22ms TTL=128
Reply from 172.17.20.25: bytes=32 time=5ms TTL=128
Reply from 172.17.20.25: bytes=32 time=9ms TTL=128
Reply from 172.17.20.25: bytes=32 time=4ms TTL=128
Reply from 172.17.20.25: bytes=32 time=25ms TTL=128
Reply from 172.17.20.25: bytes=32 time=0ms TTL=128

Ping statistics for 172.17.20.25:
    Packets: Sent = 32, Received = 32, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 25ms, Average = 6ms

PC>
```

Capturas realizadas el 1 de diciembre de 2015.

Figura 29. Ping extendido entre pc0 y pc7 en Mininet.

```

--- 192.168.0.12 ping statistics ---
30 packets transmitted, 30 received, 0% packet loss, time 29007ms
rtt min/avg/max/mdev = 0.130/0.194/0.342/0.064 ms
mininet> h0 ping -c 32 h7
PING 192.168.0.12 (192.168.0.12) 56(84) bytes of data:
64 bytes from 192.168.0.12: icmp_seq=1 ttl=64 time=0.137 ms
64 bytes from 192.168.0.12: icmp_seq=2 ttl=64 time=0.143 ms
64 bytes from 192.168.0.12: icmp_seq=3 ttl=64 time=0.194 ms
64 bytes from 192.168.0.12: icmp_seq=4 ttl=64 time=0.172 ms
64 bytes from 192.168.0.12: icmp_seq=5 ttl=64 time=0.163 ms
64 bytes from 192.168.0.12: icmp_seq=6 ttl=64 time=0.150 ms
64 bytes from 192.168.0.12: icmp_seq=7 ttl=64 time=0.575 ms
64 bytes from 192.168.0.12: icmp_seq=8 ttl=64 time=0.153 ms
64 bytes from 192.168.0.12: icmp_seq=9 ttl=64 time=0.149 ms
64 bytes from 192.168.0.12: icmp_seq=10 ttl=64 time=0.146 ms
64 bytes from 192.168.0.12: icmp_seq=11 ttl=64 time=0.169 ms
64 bytes from 192.168.0.12: icmp_seq=12 ttl=64 time=0.191 ms
64 bytes from 192.168.0.12: icmp_seq=13 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=14 ttl=64 time=0.165 ms
64 bytes from 192.168.0.12: icmp_seq=15 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=16 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=17 ttl=64 time=0.169 ms
64 bytes from 192.168.0.12: icmp_seq=18 ttl=64 time=0.166 ms
64 bytes from 192.168.0.12: icmp_seq=19 ttl=64 time=0.154 ms
64 bytes from 192.168.0.12: icmp_seq=20 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=21 ttl=64 time=0.313 ms
64 bytes from 192.168.0.12: icmp_seq=22 ttl=64 time=0.301 ms
64 bytes from 192.168.0.12: icmp_seq=23 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=24 ttl=64 time=0.166 ms
64 bytes from 192.168.0.12: icmp_seq=25 ttl=64 time=0.152 ms
64 bytes from 192.168.0.12: icmp_seq=26 ttl=64 time=0.164 ms
64 bytes from 192.168.0.12: icmp_seq=27 ttl=64 time=0.164 ms
64 bytes from 192.168.0.12: icmp_seq=28 ttl=64 time=0.840 ms
64 bytes from 192.168.0.12: icmp_seq=29 ttl=64 time=0.150 ms
64 bytes from 192.168.0.12: icmp_seq=30 ttl=64 time=0.152 ms
--- 192.168.0.12 ping statistics ---
30 packets transmitted, 30 received, 0% packet loss, time 28997ms
rtt min/avg/max/mdev = 0.137/0.206/0.840/0.143 ms
mininet> █

```

Capturas realizadas el 1 de diciembre de 2015

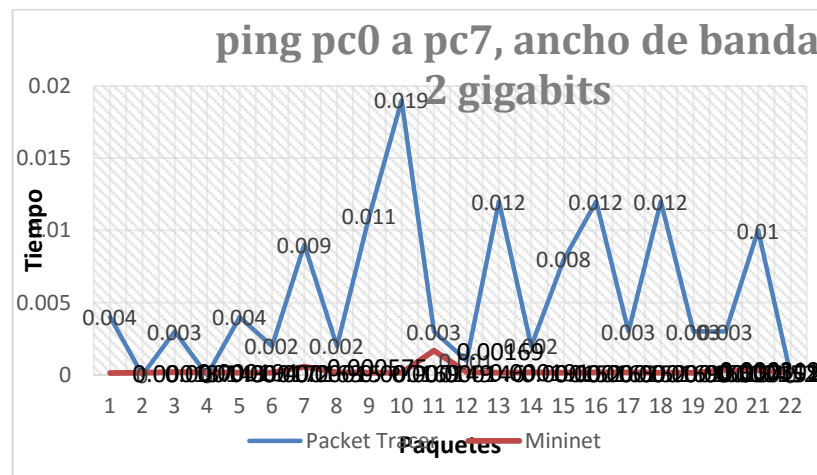
A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 11. Datos ping extendido entre pc0 y pc7.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.004	0.000137
2	0	0.000143
3	0.003	0.000194
4	0	0.000172
5	0.004	0.000163
6	0.002	0.00015
7	0.009	0.000575
8	0.002	0.000169
9	0.011	0.000149
10	0.019	0.000146
11	0.003	0.00169
12	0.001	0.000191
13	0.012	0.000152
14	0.002	0.000165
15	0.008	0.000152

16	0.012	0.000169
17	0.003	0.000166
18	0.012	0.000154
19	0.003	0.000152
20	0.003	0.000319
21	0.01	0.000301
22	0	0.000152
23	0.001	0.000166
24	0	0.000152
25	0.018	0.000164
26	0.018	0.000164
27	0	0.00084
28	0.005	0.00015
29	0.009	0.000152
30	0.004	0.000152
31	0.025	0.000152
32	0	0.000152

Gráfica 11. Packet Tracer vs Mininet ping extendido entre pc0 y pc7.



En la prueba se empieza a saturar el switch raíz para determinar el comportamiento de la red. De la gráfica se puede concluir que cuando el ancho de banda está siendo consumido por el envío continuo de datos las redes Ethernet sufren cambios más drásticos demostrados en las oscilaciones y en los picos de la gráfica, las redes SDN estabilizan sus tiempos en el envío de los paquetes de datos y hacen uso más eficiente del ancho de banda.

Figura 32. Ping extendido entre pc3 y pc11 en Mininet.

```

--- 192.168.0.25 ping statistics ---
32 packets transmitted, 32 received, 0% packet loss, time 31006ms
rtt min/avg/max/mdev = 0.152/0.351/2.742/0.557 ms
mininet> h3 ping -c 32 h11
PING 192.168.0.20 (192.168.0.20) 56(84) bytes of data:
64 bytes from 192.168.0.20: icmp_seq=1 ttl=64 time=0.128 ms
64 bytes from 192.168.0.20: icmp_seq=2 ttl=64 time=1.95 ms
64 bytes from 192.168.0.20: icmp_seq=3 ttl=64 time=0.181 ms
64 bytes from 192.168.0.20: icmp_seq=4 ttl=64 time=0.222 ms
64 bytes from 192.168.0.20: icmp_seq=5 ttl=64 time=0.179 ms
64 bytes from 192.168.0.20: icmp_seq=6 ttl=64 time=0.146 ms
64 bytes from 192.168.0.20: icmp_seq=7 ttl=64 time=1.68 ms
64 bytes from 192.168.0.20: icmp_seq=8 ttl=64 time=0.152 ms
64 bytes from 192.168.0.20: icmp_seq=9 ttl=64 time=0.175 ms
64 bytes from 192.168.0.20: icmp_seq=10 ttl=64 time=0.175 ms
64 bytes from 192.168.0.20: icmp_seq=11 ttl=64 time=0.152 ms
64 bytes from 192.168.0.20: icmp_seq=12 ttl=64 time=0.334 ms
64 bytes from 192.168.0.20: icmp_seq=13 ttl=64 time=0.162 ms
64 bytes from 192.168.0.20: icmp_seq=14 ttl=64 time=0.163 ms
64 bytes from 192.168.0.20: icmp_seq=15 ttl=64 time=0.305 ms
64 bytes from 192.168.0.20: icmp_seq=16 ttl=64 time=0.211 ms
64 bytes from 192.168.0.20: icmp_seq=17 ttl=64 time=0.164 ms
64 bytes from 192.168.0.20: icmp_seq=18 ttl=64 time=0.306 ms
64 bytes from 192.168.0.20: icmp_seq=19 ttl=64 time=0.175 ms
64 bytes from 192.168.0.20: icmp_seq=20 ttl=64 time=0.161 ms
64 bytes from 192.168.0.20: icmp_seq=21 ttl=64 time=0.174 ms
64 bytes from 192.168.0.20: icmp_seq=22 ttl=64 time=0.299 ms
64 bytes from 192.168.0.20: icmp_seq=23 ttl=64 time=0.163 ms
64 bytes from 192.168.0.20: icmp_seq=24 ttl=64 time=0.246 ms
64 bytes from 192.168.0.20: icmp_seq=25 ttl=64 time=0.159 ms
64 bytes from 192.168.0.20: icmp_seq=26 ttl=64 time=0.197 ms
64 bytes from 192.168.0.20: icmp_seq=27 ttl=64 time=0.171 ms
64 bytes from 192.168.0.20: icmp_seq=28 ttl=64 time=0.167 ms
64 bytes from 192.168.0.20: icmp_seq=29 ttl=64 time=0.171 ms
64 bytes from 192.168.0.20: icmp_seq=30 ttl=64 time=0.190 ms
64 bytes from 192.168.0.20: icmp_seq=31 ttl=64 time=0.170 ms
64 bytes from 192.168.0.20: icmp_seq=32 ttl=64 time=0.159 ms
--- 192.168.0.20 ping statistics ---
32 packets transmitted, 32 received, 0% packet loss, time 31021ms
rtt min/avg/max/mdev = 0.146/4.302/128.378/22.288 ms
mininet>

```

Capturas realizadas el 1 de diciembre de 2015.

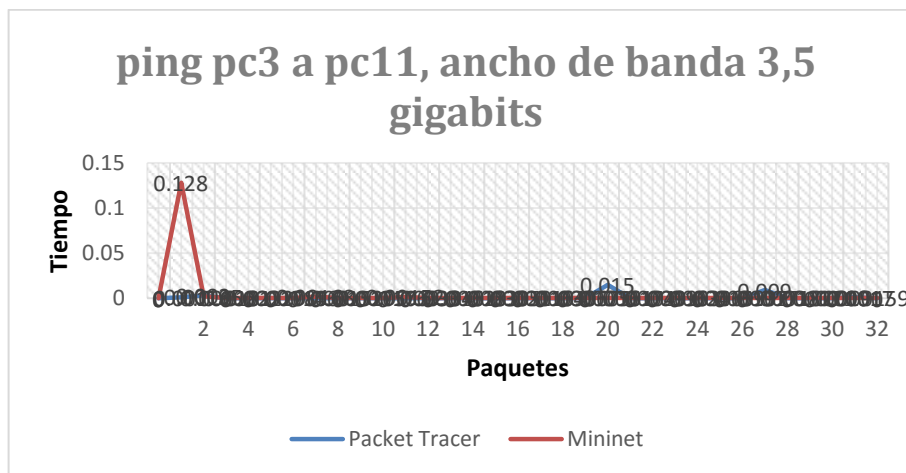
A continuación se presentan los datos arrojados por el ping, clasificados en la tabla por el tiempo, tanto en Packet Tracer como en Mininet.

Tabla 12. Datos ping extendido entre pc3 y pc11.

Paquete	tiempo(seg)	
	Packet Tracer	Mininet
1	0.001	0.128
2	0.003	0.00195
3	0	0.000181
4	0	0.000222
5	0	0.000178
6	0	0.000146
7	0	0.00168
8	0	0.000152
9	0	0.000175
10	0	0.000176
11	0	0.00152
12	0	0.000334
13	0	0.000162
14	0	0.000183
15	0	0.000305

16	0	0.000211
17	0	0.000164
18	0	0.000306
19	0	0.000175
20	0.015	0.000161
21	0	0.000174
22	0	0.000299
23	0	0.000163
24	0	0.000246
25	0	0.000159
26	0	0.000197
27	0.009	0.000171
28	0	0.000187
29	0	0.000171
30	0	0.00019
31	0.001	0.00017
32	0	0.000159

Gráfica 12. Packet Tracer vs Mininet ping extendido entre pc3 y pc11.



De la gráfica se puede concluir que los tiempos aumentan cuando el uso de ancho de banda incrementa porque la red tiene un tráfico mayor. Debido al incremento de ancho de banda, los intervalos entre los tiempos son más lejanos. En las redes SDN el envío del primer paquete tiene un retardo en el tiempo, debido al tiempo de consulta al controlador. En las redes Ethernet como ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes, estabilizando el tiempo de envío de datos.

CAPITULO 5: CONCLUSIONES

5.1 Conclusiones

1. El propósito de las redes definidas por software SDN, consiste en proporcionar una configuración más eficiente, de mayor flexibilidad y rendimiento, dando así apertura a nuevos diseños innovadores de las redes.
2. Mediante el uso de SDN, los administradores de la red poseen la capacidad para controlar el flujo de datos, es decir, se puede alterar las características de los dispositivos de enrutamiento desde una ubicación central.
3. Con todas las ventajas que permite el uso de las redes definidas por software frente a los núcleos de red tradicionales, se puede establecer que la transmisión de datos en una red definida por software SDN en diferentes situaciones como en redes de área amplia, es significativamente más rápida que en una red convencional.
4. El tamaño de los paquetes de datos que se envían en las pruebas son diferentes: en las redes Ethernet son de 32 bytes y en las redes SDN 64 bytes. Aún así las redes SDN utilizando paquetes de datos con el doble de tamaño, la velocidad de transmisión es más eficiente que la utilizada en las redes Ethernet.
5. El tiempo de vida de los paquetes en las redes Ethernet es 126, es decir se limita a un mismo continente y en las redes SDN es 64, es decir se limita a la misma región. La cantidad de saltos que puede utilizar el uno respecto al otro es menor, lo que permite observar, Mininet evita que la cantidad de saltos sea de un rango mayor, logrando disminuir el tiempo en el recorrido de los paquetes por sus rutas.
6. El tiempo de vida de los paquetes, es el tiempo que indica por cuántos nodos puede pasar un paquete antes de ser descartado por la red o devuelto a su origen. Durante la emulación y simulación en ambas topologías, se presentan ventajas en las redes SDN tendiendo a ser más acelerado.

7. Como el SDN\Openflow es un campo relativamente nuevo, muchos temas de diseño práctico están a las espera de otras investigaciones, estas podrían ayudar a comprender el estado del arte de las SDN.
8. Para futuras estudios en redes se debe tener en cuenta que la variable ancho de banda tiende a ser estática y es determinada por el prestador de servicio de internet, debido a esto puede sufrir cambios en la eficacia del envío de paquetes en las redes SDN.
9. Esta nueva propuesta de infraestructura de red se integra con las grandes tendencias que hoy mueven los entornos TIC de las empresas como automatización, virtualización, cloud computing y centros de datos de nueva generación.
10. Se puede concluir que cuando el ancho de banda está siendo consumido por el envío continuo de datos las redes Ethernet sufren cambios más drásticos, por el contrario las redes SDN estabilizan sus tiempos en el envío de los paquetes de datos y hacen uso más eficiente del ancho de banda.
11. En un mismo segmento de red, la transmisión es más rápida en las redes Ethernet, ya que no existe el tiempo de consulta al controlador, caso contrario con las redes SDN.
12. En las redes SDN el envío del primer paquete tiene un retardo en el tiempo, debido al tiempo de consulta al controlador. En las redes Ethernet como ya se conocen las tablas ARP, la red tiene la ruta para el envío de paquetes, estabilizando el tiempo de envío de datos.

5.2 Recomendaciones

- Para las empresas que manejan un alto flujo de información, en la medida que se necesite mayor velocidad en la transmisión de los datos, se recomienda utilizar las redes SDN porque proveen una mayor agilidad en el envío de paquetes.
- Para futuros estudios en redes tener en cuenta que la variable de estudio ancho de banda es poco dinámica.
- Se recomienda fomentar los estudios investigativos a cerca de esta temática, de modo que estos sirvan como ejemplo y soporte teórico que lo fundamenten y le den validez.
- Se recomienda incentivar este tipo de investigaciones con el fin de identificar en esta temática los vacíos o saturación del conocimiento que existen en la actualidad.
- Se recomienda consultar si existen más aplicativos para emulación de las redes definidas por software.

5.3 Referencias bibliográficas

- [1]. Software-Defined Networking: The New Norm for Networks
ONF White Paper April 13, 2012.
- [2].the promise of sdn, network world digital spotlight,cisco may 13, 20113.
- [3]. Copyright © 2013 Magazcitum. All Rights Reserved. Magazine Premium created by c.bavota. [Consulta: 10 octubre 2014]. Disponible en: <http://www.magazcitum.com.mx/?p=2353>.
- [4] Openflow.(2011).Recuperado el 10 de 2014, de <http://www.openflowswitch.org/wk/index.php/OpenFlowTutorial>.
- [5] Erickson, D. (Aug. de 2013). The Beacon OpenFlow controller. In Proc. HotSDN.Recuperado de www.openflow.org
- [6] Revista Telem@tica. Vol. 12. No. 3, septiembre-diciembre, 2014. ISSN 1729-3804, Alejandro García Centeno, Carlos Manuel Rodríguez Vergel, Caridad Anías Calderón , Frank Camilo Casmartiño Bondarenko. Disponible en: <http://revistatelematica.cujae.edu.cu/index.php/tele/article/viewFile/164/153>
- [7]. Redes de telecomunicaciones, protocolos, modelado y análisis, Mischa Schwartz.
- [8] El modelo OSI y los protocolos de red. Disponible en http://blyx.com/public/docs/pila_OSI.pdf
- [9] Soluciones SDN en las redes MAN/WAN: ¿revolución o evolución? – Tony Kourlas, enero 15 de 2014. Disponible en: <https://techzine.alcatel-lucent.com/es/soluciones-sdn-en-las-redes-manwan-revolucion-o-evolucion>
- [10]. Copyright 2015 IEEE – All rights reserved. Use of this Web site signifies your agreement to the IEEE Terms and Conditions.
A not-for-profit organization, IEEE is the world's largest professional association for the advancement of technology.[Consulta: 15 noviembre 2014]. Disponible en: <https://www.ieee.org/index.html>
- [11] Estado del arte redes definidas por software (sdn), Andrés Felipe Ruiz. Disponible en: <http://ribuc.ucp.edu.co:8080/jspui/bitstream/handle/10785/2788/CDMIST97.pdf?sequence=2>
- [12] Indigo Open Source OpenFlow Switches online. (08 de 2011). Recuperado el 10 de 2014, de openflow hub: <http://www.openflowhub.org/display/Indigo/>
- [13] Open Networking Foundation. (10 de 2011). Recuperado el 10 de 2014, de ONF:<https://www.opennetworking.org>

- [14] Mininet. (2014). Obtenido de <https://github.com/mininet/mininet/wiki/Documentation>.
- [15] H. Yin, H. X. (SDNi: A Message Exchange Protocol for Software Defined Networks (SDNS) across Multiple Domains de jun de 2012). Internet Research Task Force. Recuperado el 10 de 2014, de <http://tools.ietf.org/id/draft-yin-sdn-sdni-00.txt>
- [16] Global Environment for Network Innovations. (agost de 2014). Recuperado el 10 de 2014, de <http://www.geni.net/>
- [17] telcom system. (oct de 2013). Recuperado el oct de 2014, de <http://www.telco.com/index.php?page=comunicados-de-prensa&article=156>
- [18] Red_de_computadoras. (2013). Recuperado el oct de 2014, de http://www.ecured.cu/index.php/Red_de_computadoras
- [19] mundo contacto. (2013). Recuperado el OCT de 2014, de <http://mundocontact.com/mx13/temas.html>
- [20] Beacon. (octubre de 2013). Recuperado el octubre de 2014, de <https://openflow.stanford.edu/display/Beacon/Home>
- [21] Simple Network Access Contro I (SNAC). (09 de 2012). Recuperado el 10 de 2014, de <http://www.openflow.org/wp/snac/>
- [22] Big Switch networks. (oct de 2012). Recuperado el 10 de 2014, de <http://www.bigswitch.com/sites/default/files/sdn>
- [23] The OpenFlow Switch Specification. (2011). Recuperado el 10 de 2014, de <http://OpenFlowSwitch.org>
- [24] Software-Defined Networking: The New Norm for Networks. <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf>