

# EVALUACIÓN DE DESEMPEÑO Y CONFIGURACIONES DE LAS SDN MEDIANTE LA SIMULACIÓN

**Por:** Ing. Liz Gámez Picó; Dra. Caridad Anías Calderón; Ing. Susana Ballester Macías, Profesores del Departamento de Telecomunicaciones y Telemática, ISPJAE; Ing. Carlos M. Rodríguez Vergel, CNIC.  
[liz.gp@electrica.cujae.edu.cu](mailto:liz.gp@electrica.cujae.edu.cu); [cache@tesla.cujae.edu.cu](mailto:cache@tesla.cujae.edu.cu); [sballesterm@electrica.cujae.edu.cu](mailto:sballesterm@electrica.cujae.edu.cu); [carlos.vergel@cnic.edu.cu](mailto:carlos.vergel@cnic.edu.cu)

## RESUMEN

El desarrollo gradual de las SDN requiere de herramientas que permitan evaluar sus elementos, siendo Mininet la más recomendable. El objetivo de este trabajo es destacar las métricas recientemente definidas por la ONF que se deben considerar para evaluar los elementos de la arquitectura SDN, así como definir diferentes pruebas de configuración y desempeño que se puedan realizar en Mininet para obtener dichas métricas. Como resultado de la investigación se realiza un ejemplo práctico de una simulación de una red de campus SDN, aplicando las pruebas definidas.

**Palabras clave:** Redes Definidas por Software, Simulación, Mininet, Métricas, Pruebas

## ABSTRACT

*SDN gradual evolution requires tools for elements assessment, being Mininet strongly recommended. This paper aims at highlighting the metrics recently defined by ONF that should be taken into account for assessing elements of SDN architecture, as well as for defining different configuration and performance tests that can be carried out in Mininet to obtain such metrics. As a result of this research, a practical sample of SDN campus network simulation applying defined tests is done.*

**Key words:** Software-Defined Networks, simulation, Mininet, metrics, tests

## Introducción

Las SDN —*Software Defined Network*— continúan creciendo en escala, complejidad e importancia, por lo que resulta necesario estudiar y evaluar el comportamiento de las aplicaciones, protocolos y servicios que se implementen para estas redes. Los investigadores requieren de herramientas que les faciliten estudiar el comportamiento de estas redes y trabajar en su desarrollo. Las herramientas de simulación con soporte para las SDN brindan un entorno virtual para pruebas que reproducen el comportamiento de una red real y que permiten evaluar el desempeño de estas redes y desarrollar nuevas aplicaciones, protocolos, controladores y servicios.

Para evaluar los elementos de las SDN es necesario conocer las métricas que brindan información relevante del desempeño de la red, relacionadas con los mensajes *OpenFlow* del protocolo, del controlador o de los switches. También es importante definir las principales pruebas que se pueden realizar en un escenario de simulación para extraer dichas métricas.

En este artículo se detallarán las más importantes para las SDN, se analizarán algunas herramientas que permiten

simular estas redes y se generalizarán algunas pruebas de configuraciones y desempeño que se pueden realizar en Mininet. Por último, se realizará la simulación de un escenario empleando Mininet, con el objetivo de obtener mediante las pruebas definidas, algunas de las métricas relevantes para las SDN.

Existen muchas métricas definidas para las redes tradicionales que también son válidas para las SDN como es el caso de la razón de pérdidas de paquetes, la latencia y variación de la latencia y el tiempo de activación del servicio; aunque también se han definido por la ONF métricas [1] que son específicas para ese tipo de redes, relacionadas con los mensajes del protocolo *OpenFlow*, con el controlador *OpenFlow* o con los switches *OpenFlow*, a continuación se detallan las más importantes y otras incluidas por los autores.

Entre las métricas cuantitativas para el protocolo *OpenFlow* se destacan: tiempo de respuesta del protocolo *OpenFlow*, tiempo de activación del protocolo y duración de una entrada. Algunas de las métricas que se pueden obtener para cada controlador *OpenFlow* son: disponibilidad, confiabilidad, capacidad,



exactitud, seguridad y rendimiento. También se puede incluir la eficiencia de las entradas instaladas en las tablas de flujo que es el porcentaje de las consultas a estas de los flujos que coincidieron con alguna de las entradas y la escalabilidad que es el número máximo de switches que un controlador puede manejar sin que disminuya su rendimiento. [2]

### Herramientas de simulación con soporte para las SDN

Actualmente, algunas herramientas de simulación se han creado y otras se han complementado para permitir la simulación de las SDN, entre las cuales se pueden destacar [3]: EstiNet, IxNetwork, ns-3, OMNeT++, OpenNet, fs-sdn y Mininet [4]. Teniendo en cuenta un conjunto de criterios que deben ser considerados en la herramienta que se utilice para simular las SDN, como la versión *OpenFlow* que soporta, el empleo de controladores, las métricas que permite obtener, si es un software de código abierto, la documentación que presenta para trabajar con las SDN y las herramientas con las que puede integrarse, se concluye que Mininet es actualmente el mejor y más accesible simulador para el desarrollo de experimentos y pruebas sobre las SDN.

Mininet es un emulador de código abierto desarrollado para crear, personalizar e interactuar con prototipos SDN. Las distribuciones de Mininet incluyen muchas aplicaciones de ejemplo, *scripts* y un editor gráfico de red, denominado MiniEdit, útil para la creación de topologías y otras configuraciones. El simulador Mininet puede ser integrado con un gran número de herramientas de código abierto que permiten complementar sus prestaciones, como es el caso de los analizadores de protocolos, interfaces gráficas de usuario, y *benchmarks* entre los que se destacan el Disector de Wireshark de *OpenFlow*, ping, pingall y Cbench. [11]

### Pruebas de configuraciones y desempeño en Mininet

En la simulación de las SDN se pueden definir dos categorías de pruebas: de configuraciones y de desempeño. Las pruebas de configuraciones tienen el objetivo de comprobar el correcto funcionamiento de las características básicas de la SDN simulada como, por ejemplo, la conmutación, el establecimiento de los flujos y la comunicación entre el controlador y los switches. Las pruebas de desempeño permiten evaluar, entre otros, la escalabilidad, la utilización de los recursos y la confiabilidad del escenario, y deben estar basadas en las métricas cuantitativas definidas por la ONF analizadas anteriormente. A continuación, se generalizarán algunas de estas pruebas.

### Pruebas de configuración

Las principales pruebas de configuración que se pueden llevar a cabo para evaluar y caracterizar el funcionamiento de un escenario de SDN mediante su simulación en Mininet son:

- 1) Conectividad entre nodos: Ejecutando los comandos *ping*, *pingall*, *iperf*, etc.
- 2) Verificación del intercambio de mensajes *OpenFlow*: Mediante la herramienta Wireshark aplicando el filtro para paquetes *OpenFlow*.
- 3) Integración con dispositivos tradicionales.

### Pruebas de desempeño

#### Métricas típicas aplicadas a las SDN

- 1) Razón de pérdidas de paquetes: Se puede utilizar el comando ping, que muestra esta información, en función de la cantidad de mensajes ICMP intercambiados.
- 2) Latencia y variación de la latencia: Se puede emplear el comando ping, que muestra una estadística con los valores mínimo, promedio, máximo y la desviación estándar del parámetro RTT durante el intercambio de mensajes ICMP.
- 3) *Throughput*: Se puede utilizar la herramienta de línea de comando Iperf.

#### Métricas cuantitativas para el protocolo OpenFlow

- 1) Tiempo de respuesta del protocolo *OpenFlow*: Se definirá como Tresp\_OF y está dado por [12]:

$$\text{Tresp\_OF} = 2 \times \text{Tprop} + \text{Ttx\_mensaje\_OF} + \text{Tprocesamiento} + \text{Ttx\_respuesta\_OF} \quad (1)$$

- 2) Tiempo de activación del protocolo: El tiempo transcurrido entre el envío de un mensaje OFPT\_PACKET\_IN y el cambio en la entrada en la tabla de flujo, que también puede ser considerado como el tiempo de establecimiento de un flujo, se definirá como Tact\_OF y está dado por [13]:

$$\text{Tact\_OF} = 2 \times \text{Tprop} + \text{Ttx\_packet\_in} + \text{Tprocesamiento\_controlador} + \text{Ttx\_packet\_out} + \text{Tproc\_sw} \quad (2)$$

Se recomienda utilizar la herramienta Cbench en modo latencia, para lo cual se debe activar alguna aplicación en el controlador, y ejecutar Cbench desde otra sesión.

- 3) Duración de una entrada en la tabla de flujo: La duración de una entrada en la tabla de flujo está determinada por los campos *idle\_timeout* y *hard\_timeout*. El valor de estos campos para cada entrada en la tabla de flujo se obtiene ejecutando la herramienta Dpctl.

#### Métricas cuantitativas para el controlador OpenFlow

- 1) Disponibilidad: Se puede utilizar la herramienta top y analizar el parámetro *uptime*.
- 2) Capacidad: Se puede utilizar el comando top, que muestra la cantidad de memoria disponible, así como el porcentaje de memoria y de CPU utilizada por el controlador.

- 3) Rendimiento: Para cuantificar el rendimiento del controlador dado por (3) se puede utilizar la herramienta Cbench, en modo *throughput*.

Razón de establecimiento de flujos (i)=(Cantidad de respuestas OF (i))/(Duración de la prueba(i))

- 4) Eficiencia de las entradas instaladas en las tablas de flujo: Para conocer la cantidad de paquetes que coincidieron con alguna entrada en la tabla de flujos, se deben examinar estas tablas mediante el comando:

```
$: dpctl dump-tables tcp:<IP del controlador>:<puerto>.
```

El campo *lookup* de las tablas de flujo especifica la cantidad de paquetes que fueron recibidos en el plano de datos y el campo *match field*, la cantidad de paquetes que coincidieron con las entradas existentes en la tabla de flujos, con estos datos se puede calcular la eficiencia como:

Eficiencia= (Paquetes que coincidieron con alguna entrada)/(Paquetes que fueron recibidos)×100%

- 5) Escalabilidad (en función de la cantidad de nodos): Para obtener la máxima cantidad de switches y hosts que el controlador puede manejar se debe evaluar el impacto de aumentar la cantidad de nodos en la red, para lo cual se puede utilizar la herramienta Cbench en modo *throughput* ejecutando el comando:

```
$: cbench -c <IP del controlador> -p <puerto> -s <cantidad de switch> -M <número de host> -m <duración de la prueba en milisegundos> -l <cantidad de pruebas> -t
```

### Otras pruebas

Las métricas cuantitativas para el *switch* OpenFlow definidas anteriormente se pueden obtener principalmente con las opciones del comando dpctl. Es necesario destacar que en los reportes técnicos del *Grupo de trabajo para la evaluación y verificación de la ONF* [15] se destacan otros aspectos que se pueden evaluar y mejorar en los switches y controladores OpenFlow, los cuales podrían apoyar las pruebas descritas anteriormente.

### Simulación del prototipo de una red de campus SDN

En este apartado se realizará la simulación de un caso de uso de las SDN para evaluar el funcionamiento básico de los elementos que conforman la arquitectura SDN y en particular, el desempeño del controlador OpenDaylight que actualmente es uno de los proyectos de mayor impacto y desarrollo, integrado en un escenario con *switches* tradicionales, Open Switch y con la versión 1.3 del protocolo OpenFlow. Se trabajó con una máquina virtual con imagen de Ubuntu 14.04 de 64 bits disponible en [11].

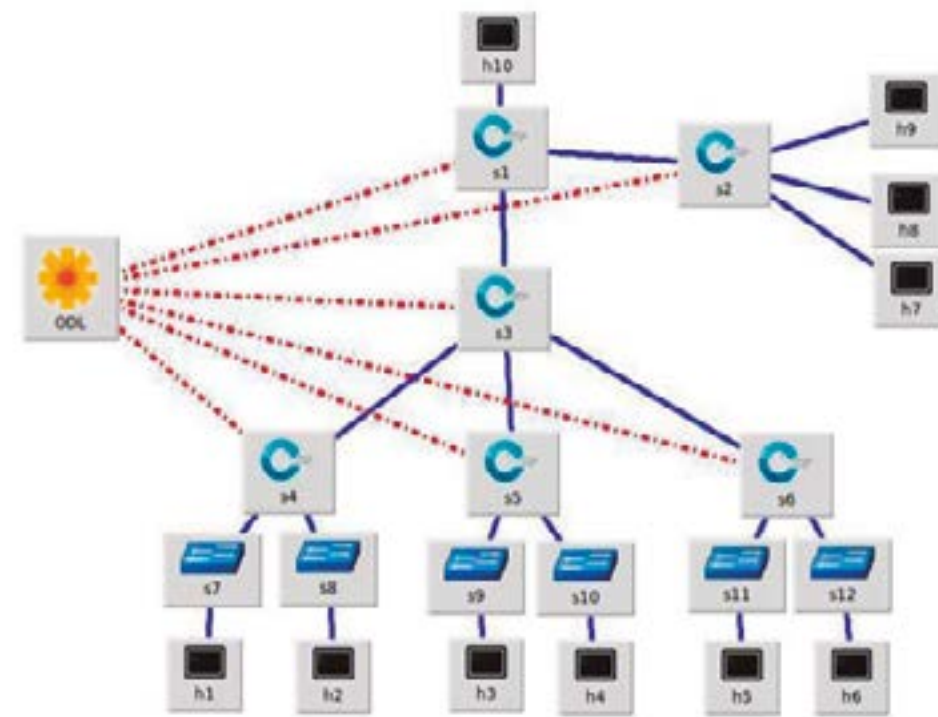


Figura 1. Escenario de la simulación Fuente: Elaboración propia.



Para crear el escenario se utilizó el editor gráfico MiniEdit y se ejecutó una aplicación SDN en ODL que permite trabajar con la interfaz Web del controlador e inicializar funciones básicas de red como el descubrimiento de topología, estadísticas de los puertos y las tablas de flujo y la gestión de los switches. (Figura 1)

En el escenario anterior se aplicaron las pruebas de configuraciones y desempeño definidas anteriormente, las cuales

confirmaron el correcto comportamiento del escenario y del controlador *OpenDaylight*. Los resultados obtenidos para el rendimiento del controlador son los esperados bajo las condiciones de la prueba. Las figuras 2, 3 y 4 muestran los resultados de la razón de establecimiento de flujos aumentando la cantidad de switches y la cantidad de host, lo que permite llegar a conclusiones importantes de la escalabilidad y del rendimiento del controlador.



Figura 2. Interfaz Web de OpenDaylight. Fuente: Elaboración propia.

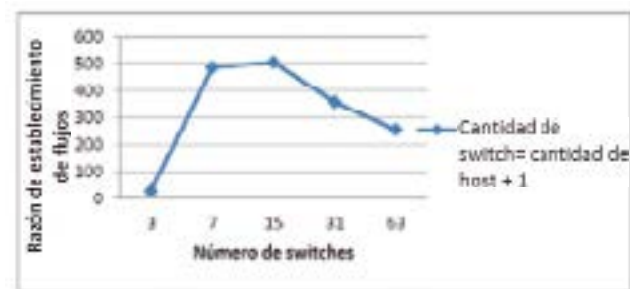


Figura 3. Razón de establecimiento de flujos aumentando la cantidad de hosts. Fuente: Elaboración propia.



Figura 4. Razón de establecimiento de flujos aumentando la cantidad de switch. Fuente: Elaboración propia.

## Conclusiones

Las SDN significan una revolución en el mundo de las redes. El carácter aún novedoso de esta arquitectura exige profundizar en el estudio detallado de este tema, principalmente en su evaluación. La simulación de las SDN constituye un método muy útil para esto pues permite evaluar el desempeño y la funcionalidad de estas redes, para lo cual Mininet es la herramienta de simulación recomendada. En función de las pruebas que se realicen en cada simulación se podrá extraer determinada información relevante del escenario. La

simulación de un escenario recreando un prototipo de una red de campus SDN, empleando Mininet, permitió obtener las métricas relevantes para las SDN y el desempeño del controlador OpenDaylight. Los resultados obtenidos permitieron analizar aspectos importantes como el rendimiento y la escalabilidad del controlador y se pudo comprobar la importancia de la simulación para estudiar y evaluar el comportamiento de una red SDN, por lo cual se consideran cumplidos los objetivos propuestos.

## Referencias bibliográficas:

- [1] ONF. "Migration Tools and Metrics". 2014.
- [2] S. G. Amin Tootoonchian, Yashar Ganjali y Martin Casado, Sherwood Rob. "On Controller Performance in Software-Defined Networks". 2012.
- [3] E. B. Y. Mohammed Basheer Al-Somaidai. "Survey of software components to emulate OpenFlow protocol as an SDN implementation". American Journal of Software Engineering and Applications, December 16, 2014.
- [4] Lantz, Bob B. H. y McKeown, Nick. "A Network in a Laptop: Rapid Prototyping for Software-Defined Networks". Hotnets '10, October 20–21, 2010.
- [5] Wireshark. Acceso: Noviembre 2015. Disponible en: <http://www.wireshark.org/>
- [6] Shalimov, D. Z. A.; Zimarina, D.; Pashkov, V. y Smeliansky, R. "Advanced study of SDN/OpenFlow controllers". Central & Eastern European Software Engineering Conference in Russia, noviembre 2013.
- [7] Sunnen, D. "Performance Evaluation of OpenFlow Switches". Information Technology and Electrical Engineering, 2011.
- [8] Azodolmolky, S. *Software Defined Networking with OpenFlow*, 2013.
- [9] Vengainathan, B. "Benchmarking Methodology for OpenFlow SDN Controller Performance". 2014.
- [10] ONF. "Interoperability Event Technical Issues". 2013.
- [11] I.C.M.R.V. Ballester Macías, Susana; García Centeno, Alejandro; Casmartiño Bondarenko, Frank Camilo y Anías Calderón, Caridad. *Monografía: Redes Definidas por Software (SDN)*. ed. 2014.

(Artículo recibido en noviembre de 2015 y aprobado en enero de 2016)

