Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros en Telecomunicaciones



Desarrollo de aplicaciones de ingeniería de tráfico en redes WAN basadas en software

Andrés Jorge Muracciole Vázquez

Memoria de master presentada a la Universidad Politécnica de Madrid como requisito parcial para la obtención del Máster en Ingeniería en Redes y Servicios Telemáticos

Tutor: Carlos Mariano Lentisco Sánchez

Madrid, Julio 2020

# Índice

[Índice - 3 -](#_Toc41323699)

[Índice de Figuras - 4 -](#_Toc41323700)

[Glosario - 5 -](#_Toc41323701)

[Capítulo 1 - 7 -](#_Toc41323702)

[Introducción - 7 -](#_Toc41323703)

[1.1 Resumen - 7 -](#_Toc41323704)

[1.2 Problema a abordar - 8 -](#_Toc41323705)

[1.3 Estado del arte - 9 -](#_Toc41323706)

[Capítulo 2 - 10 -](#_Toc41323707)

[Soluciones existentes - 10 -](#_Toc41323708)

[2.1 Nuage - 10 -](#_Toc41323709)

[2.2 Viptela - 12 -](#_Toc41323710)

[2.3 FlexiWAN - 13 -](#_Toc41323711)

[2.3.1 Pruebas de la herramienta flexiWAN - 14 -](#_Toc41323712)

[Referencias - 16 -](#_Toc41323713)

# Índice de Figuras

[Fig. 1 – Red sin interrupciones SD-WAN [i] - 11 -](#_Toc41268244)

[Fig. 2 – Arquitectura Nuage [i] - 12 -](#_Toc41268245)

[Fig. 3 – Arquitectura Viptela [ii] - 13 -](#_Toc41268246)

[Fig. 4 – Diagrama de bloques de alto nivel de la arquitectura flexiWAN [iii] - 14 -](#_Toc41268247)

[Fig. 5 – Visualización el flexiWAN-router desde la interfaz central manage. - 14 -](#_Toc41268248)

[Fig. 6 – Estadísticas de bps y pps en un flexiWAN-router. - 15 -](#_Toc41268249)

# Glosario

*SDN:* Software Defined Network.

*SD-WAN*: Software Defined Wide Area Netkork.

*CPD*: Centro de Procesamiento de Datos

*ISP*: Internet Service Provider

# Capítulo 1

Introducción

## Resumen

Este documento forma parte de un trabajo de fin de master el cual trata acerca del estudio e implementación de una solución de ingeniería de tráfico para una red WAN basada en software. El objetivo de la misma es poder realizar una implementación que sea capaz de optimizar el camino que debe cursar el tráfico en una red WAN en función de las necesidades de las aplicaciones y el estado de la red en tiempo real.

En primer instancia se estudiarán los conceptos fundamentales de una red WAN tradicional identificando sus limitaciones que motivan su transformación hacia redes basadas en software y aplicaciones SDN[[1]](#footnote-1). Luego se mencionará cómo está conformada una red SD-WAN de forma de entender mejor este concepto y así poder adentrarse en la parte central del trabajo que consistirá en el diseño e implementación de una solución de tráfico para este tipo de redes. Aquí se detallarán no solo los conceptos relevantes de la misma sino que también se explicará el funcionamiento y desarrollo de la solución. Con ello se buscará no solo que el lector entienda el trabajo técnico realizado sino también ayudar a reflexionar sobre el potencial de este tipo de soluciones en el mercado actual. Finalmente se realizarán una serie de pruebas prácticas donde se reflejará el funcionamiento de la solución establecida y se verá el camino que toma el tráfico en función de las prestaciones de la red. Por último se mencionarán a una serie de ejemplos prácticos donde se puede ver beneficiado el negocio con una solución de este tipo, finalizando con una serie de conclusiones a modo de resumen general.

## Problema a abordar

El problema principal a tratar será el diseño e implementación de una solución capaz de censar periódicamente el ancho de banda libre de los enlaces de un escenario virtual y realizar configuraciones en tiempo real en el controlador de forma de poder cursar el tráfico de la manera mas óptima en función de las prestaciones requeridas y los criterios previamente definidos por el administrador de la red.

El alcance del proyecto en primer lugar será poder conocer y entender en profundidad el concepto de redes basadas en software SDN, los equipos que la conforman y el tipo de tráfico generado por y sobre ellas. Esto permitirá entender en mayor profundidad el nuevo paradigma a nivel de redes de comunicación y sus ventajas respecto al sistema tradicional. Por otro lado se deberán tomar decisiones bien fundamentadas acerca del tipo de controlador con el cual trabajar, los parámetros de red a medir y cómo se realizará la configuración a nivel de controlador para que este pueda dirigir el tráfico según lo establecido. Dependiendo del controlador de red, la interfaz de comunicación con este puede ser variada, desde CLI, pasando por interfaz gráfica o en algunos casos usando como intermediario API´s propias, lo cual trae como desafío el estudio de las mismas para entender su funcionamiento y poder así sacarles el mayor rédito a las aplicaciones diseñadas.

Al trabajar con escenarios virtualizados será necesario diseñar uno que se asemeje una red real y para ello se planteó uno con “topología de pez”[[2]](#footnote-2), de forma de poder trabajar con un escenario que cuente con multicaminos para poder implementar la ingeniería de trafico deseada. Por último, a partir de la experiencia en este rubro, es muy probable que sea necesario programar scripts en diferentes lenguajes de programación, lo cual trae como consecuencia estar familiarizados con ellos y/o de ser necesario, estudiarlos con detenimiento.

En cuanto al alcance, se espera poder lograr una aplicación flexible pero a la vez robusta que permita medir los diferentes enlaces entre un origen y destino en tiempo real y tomar decisiones a partir de información dada por el usuario o de forma automática mediante recopilación de estados del sistema. Este deberá ser capaz de trabajar de forma programable, haciendo mínima la interacción del usuario administrador pero que a la vez sea amigable con este y le permita realizas configuraciones de forma rápida y sencilla si así lo requiera. Se prevé realizar una interfaz gráfica amigable mediante la cual el operador pueda realizar configuraciones en el controlador y visualizar gráficos o configuración de la red, lo que lo ayudará a corroborar el correcto funcionamiento.

## Estado del arte

Las redes LAN (*Local Area Network*) son sin duda las mas utilizadas a nivel empresarial. Allí radican los servicios internos de una empresa así como también los activos tales como impresoras, registros y servidores de información, los cuales en un principio solo interesa que se tenga acceso a ellos desde dentro de la corporación. Sin embargo, en muchos casos también es requerido el acceso desde fuera de dicha red, ya sea para obtener cierta información interna como también para utilizar recursos de la misma. Es aquí donde entra en juego el rol de las redes WAN (*Wide Arde Network*). Estas redes permiten interconectar redes LAN que en muchos casos están físicamente a cientos o miles kilómetros de distancia y simular como si estuvieran directamente conectadas entre sí. De no existir estas, las empresas que cuentan con mas de una sucursal deberían tener replicado su equipamiento en cada una de los sitios haciendo así poco escalable el negocio. De esta forma un usuario en América puede acceder al registro de facturación alojado en las oficinas de Europa, o viceversa sin siquiera saber que los paquetes se encuentran cruzando miles de kilómetros de distancia. Sin embargo, las redes WAN son de acceso público y por ende en ellas viaja información tanto propia como de muchas otras y es por esto que es imposible asegurar y estimar la calidad de servicio a la que están expuestos. A raíz de esto es que las empresas suelen contratar a los proveedores de servicios unas conexiones dedicadas de forma de asegurarse un nivel de QoS aceptable para sus aplicaciones. Ejemplo de estas son las conocidas MPLS (*Multiprotocol Label Switching*) las cuales se implementan sobre redes WAN y a diferencia de redes IP, el protocolo de encaminamiento de paquetes es mas ágil y sencillo debido a que se hace mediante el uso de etiquetas y no de direcciones. De este modo una empresa puede tener dos o mas enlaces WAN (muchas veces provistos por diferentes ISP`s [[3]](#footnote-3)) y decidir que tipo de tráfico es mas sensible y en función de ello enviarlo por el canal con mayores prestaciones. A pesar de ello, estos servicios suelen ser bastante costosos para las empresas y poco flexibles.

Por otra parte dese hace ya unos años la demanda de recursos y cómputo viene en constante crecimiento al punto de que aplicaciones que antes residían en CPD`s[[4]](#footnote-4) propios, hoy se ejecutan en la nube pública. Esto trae consigo un aumento en el tráfico hacia internet, haciendo que este se vea saturado y como internet se basa en el principio de “*best effort*”, no es posible asegurar calidad de servicio en la misma.

Es por tal motivo que las redes SD-WAN vienen a simplificar estos asuntos dando mas flexibilidad de una manera mas simple, ágil y económica. Con este nuevo paradigma un operador puede al cabo de unos pocos minutos levantar un nuevo sitio remoto, realizar configuraciones y establecer criterios de flujo para decidir bajo demanda el camino mas óptimo por el cual enviar los paquetes en función de las prestaciones que se requieran. Por otra parte, muchas veces los enlaces WAN son contratados a diferentes empresas; sin embargo, las soluciones SD-WAN son independientes a esto, permitiendo trabajar con ellas sin importar el proveedor o mismo la solución SD-WAN sea contratada a una empresa diferentes a los ISP´s de las redes WAN. En definitiva permite unificar las redes WAN y tratarla como si fuera “una sola” y bajo demanda.

# Capítulo 2

Soluciones existentes

Con la incursión del las nuevas tecnologías y el cambio de paradigma respecto a las redes de comunicación tradicionales con SDN y SD-WAN, varios proveedores comenzaron a ver la importancia de las mismas y el potencial que podrían llegar a alcanzar en caso de migrar su red hacia estas. Es por ello que existen tantas soluciones como operadores haya, con la desventaja de que son en su totalidad servicios de pago con lo cual no es posible realizar pruebas de laboratorio sin tener que pagar por ello. Algunas implementaciones cuentan con licencias trial pero están muy limitadas en cuanto a su uso y no permiten realizar lo que se espera en este trabajo. A continuación se procederá a describir tres soluciones comerciales de servicio SD-WAN para entender un poco mejor de que tratan, haciendo mayor hincapié en *Flexiwan* ya que si bien es solución paga, esta es de código abierto, por lo que fue posible realizar una serie de pruebas que se detallarán en el punto 2.3.1 de este capítulo.

## Nuage

Nuage Networks [i] es la solución SD-WAN implementada por Nokia que sirve para automatizar la conectividad de sitios remotos en cualquier red con hardware. Permite entregar y organizar servicios de TI corporativos en centros de datos, nubes públicas o privadas y gestionar las conexiones WAN mediante la utilización de políticas de forma de optimizar el uso de la red. Esto permite disminuir los costes de infraestructura satisfaciendo las necesidades requeridas y al mismo tiempo brindando seguridad a las aplicaciones.

Con VNS (*Virtualized Network Services*) de Nuage, la red SD-WAN se puede optimizar dinámicamente para enrutar tráfico por la red mas rentable. En muchos casos las empresas suelen tener redes IP, MPLS, 3G y/o LTE de diferentes proveedores para transportar sus servicios, con lo cual se podría aprovechar por ejemplo internet para las aplicaciones que no requieran prestaciones exigentes dejando las demás conexiones para aquellos servicios críticos. El hecho de que estos enlaces sean provistos por diferentes ISP es indiferentes para esta solución SD-WAN ya que es capaz de gestionarlos desde una plataforma unificada.

Por otra parte esta Nuage permite ocultar la complejidad de la red WAN empresarial estableciendo una WAN extremo a extremo que posibilita conectar centros de datos privados, sucursales y servicios de nube pública o privada como se puede ver en la Figura 1.

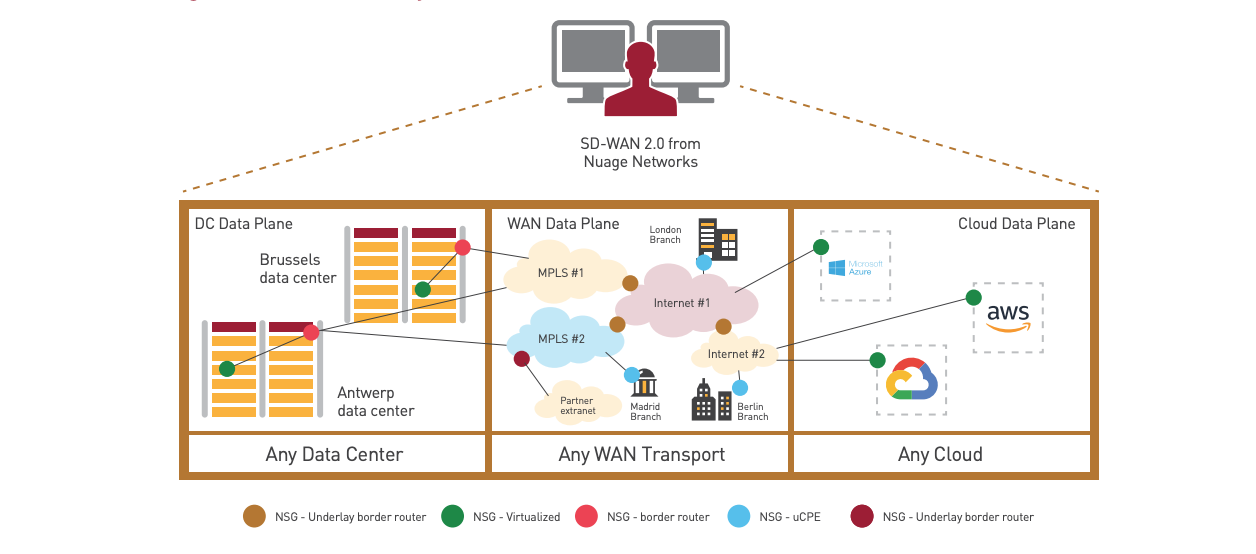


Fig. 1 – Red sin interrupciones SD-WAN [i]

La Figura 2 muestra la arquitectura de Nuage. El VSD (Virtualized Services Directory) permite al administrados SD-WAN definir y aplicar políticas a la red de una manera fácil y de forma gestionada y centralizada. A su vez implementa la funcionalidad de recolección de datos de forma periódica para la creación de informes y reportes de la red, así como también alertas en caso de que un tráfico haya sobrepasado un umbral determinado. Todas estas funciones se pueden desplegar en un portal personalizable con widgets. Las funciones de red se pueden seleccionar mediante el catálogo de VSD, pudiendo optar entra VNFs de firewalls, IPSec, NAT, balanceo de carga y gestión de dominios, entre otros.

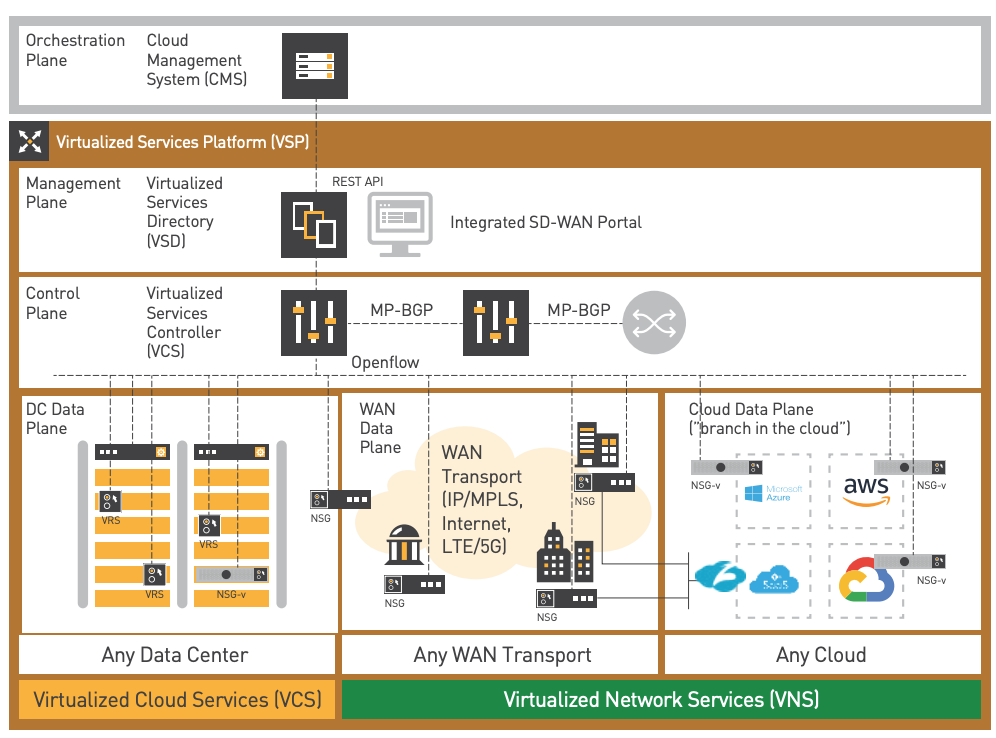


Fig. 2 – Arquitectura Nuage [i]

## Viptela

Viptela [i[[5]](#endnote-1)] es la solución SD-WAN adquirida por Cisco que permite abstraerse del medio de transporte y tener visibilidad en tiempo real del estado de la red para así poder por ejemplo tomar decisiones de enrutamiento. Viptela está conformada por cuatro componentes, cada uno asociado a un plano diferente tal cual se ve en la Figura 3. Por un lado se encuentra el vEdge que representa el router en las oficinas. Este es quien se encarga de establecer comunicaciones seguras con los demás vEdges (virtualizados o físicos) mediante túneles IPSec y son quienes ejecutan las políticas implementadas por el vSmart.

El vSmart es el controlador de la red encargado del plano de control y es quien establece conexiones SSL con los demás componentes que forman la arquitectura SD-WAN mediante un protocolo propio llamado OMP (*Overlay Management Protocol*). Brinda seguridad, control de acceso y políticas de encaminamiento sin la utilización de protocolos tradicionales como OSPF y BGP. En cuanto a las políticas estas pueden ser centralizadas, aplicándose sobre todo el *Fabric* y actuando en varios sites, o bien localizadas en un determinado vEdge.

El tercer elemento de la topología de Viptela es el bashboard centralizado llamado vManage. Este es el encargado de desplegar la información de la red el tiempo real y permitir al administrador aplicar reglas y monitorizar la red SD-WAN. Soporta diversos protocolos de gestión como SNMP, NETCONF o Syslog, así como también la capacidad que tienen los vEdges de realizar Deep Packet inspection le servirá al vManage para obtener analíticas a nivel de aplicación.

Por último el vBond cumple el rol de orquestador SD-WAN y se encarga de realizar la autenticación y autorización de los elementos de la red gestionando las comunicaciones entre lo vEdges y los controladores.

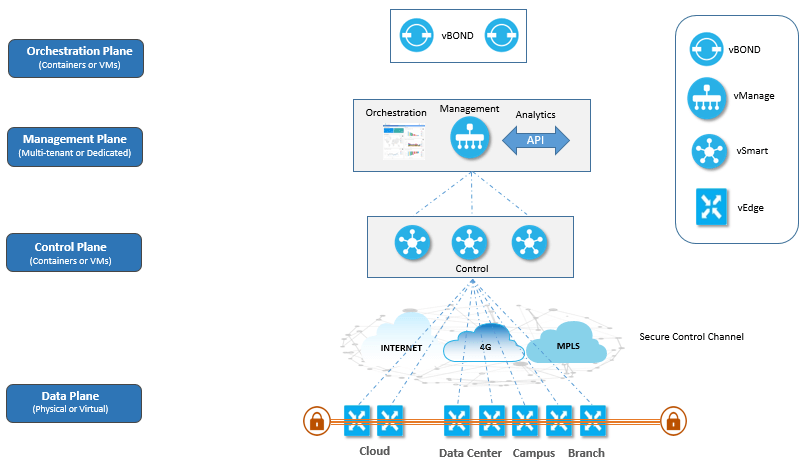


Fig. 3 – Arquitectura Viptela [ii]

## FlexiWAN

FlexiWAN [iii] es otra solución SD-WAN que a diferencia de las anteriores, es de código abierto la cual permite instalar hasta 3 instancias de forma gratuita aunque con ciertas limitaciones. La infraestructura SD-WAN de FlexiWAN está formada principalmente por dos actores principales tal como se ve en la Figura 4: flexiEdge y flexiManage. El flexiEdge es el dispositivo que se ejecuta el punto de presencia el cual tiene incorporado VPP (*Vector Packet Processor*) y es utilizado para aumentar la velocidad de procesamiento ya que puede tratar varios paquetes el simultaneo. Por otro lado cuenta con FRR (*Free Range Routing*) que implementa diferentes tipos de protocolos de enrutamiento como OSPF, RIP, BGP, IS-IS, entre otros y se ejecuta en plataformas similares a Unix. Estos se conectan al flexiManage, quien se encuentra ejecutado sobre un servidor web y es quien se encarga de la administración de la red. A través de él un operario puede gestionar y recopilar información de los dispositivos flexiEdge para luego analizarla y proporcionar informes de la red. Cabe destacar que la plataforma está fuertemente orientada a facilitar y centralizar las configuraciones y gestionar los equipos. Por tal motivo es que cuenta con funcionalidades capaces de desplegar alarmas y avisar al administrador de posibles fallos en la red para que pueda actuar en el menor tiempo posible.

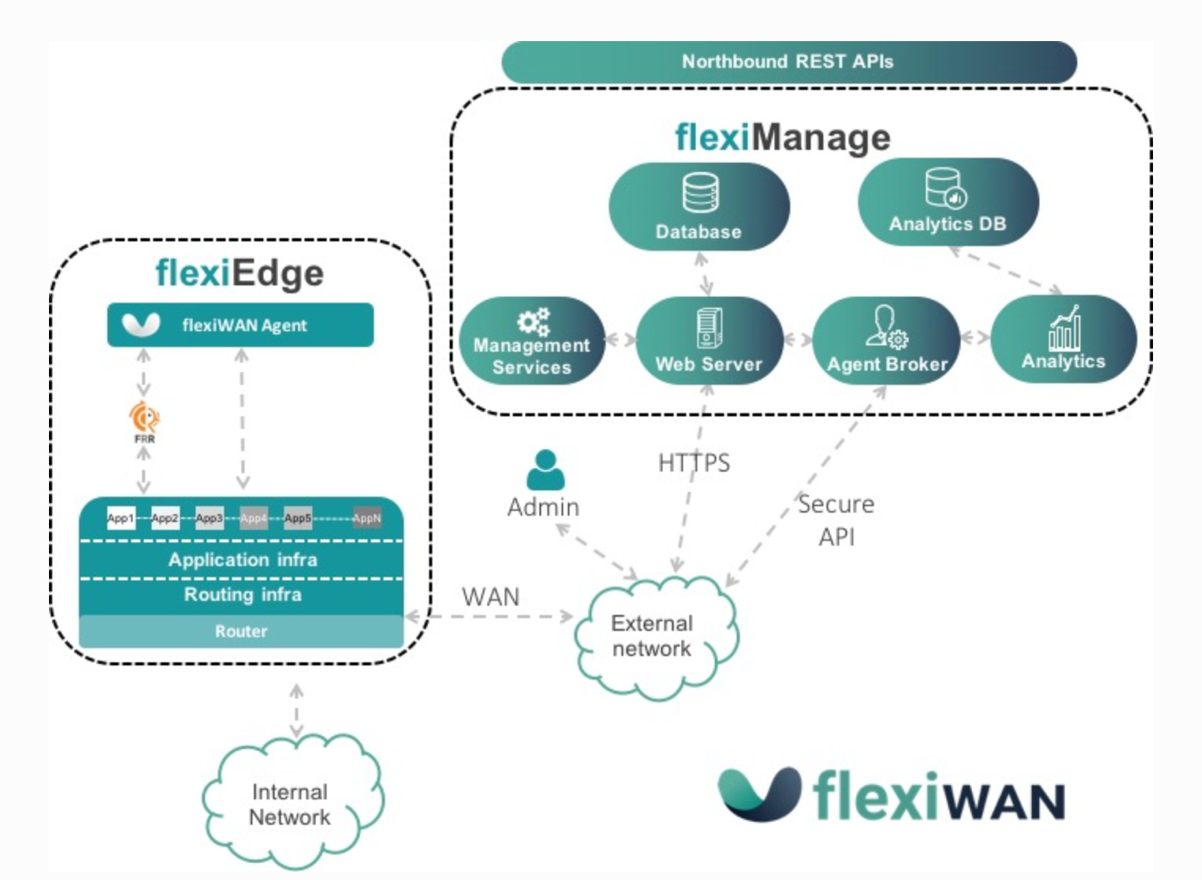


Fig. 4 – Diagrama de bloques de alto nivel de la arquitectura flexiWAN [iii]

### Pruebas de la herramienta flexiWAN

Una vez estudiada la teoría de procedió a implementar un escenario para ver en funcionamiento de las prestaciones que ofrece la plataforma en la versión gratuita. En primer lugar se crearon y configuraron dos máquinas virtuales Ubuntu con conexión a internet para que tengan conectividad con el flexiManage las cuales simulan ser dos sitios remotos. En ambas fue necesario asociarles un token de autenticación (creado en el flexiManage) para que los reconozca como sitios seguros y establecer una canal de comunicación contra ellos. En este punto ya deberían de aparecer los routers virtuales en el dashboard dentro de la sección *Inventory/Devices and manage* como se aprecia en la Figura 5 y es desde aquí donde se termina de dar de alta el sitio en la plataforma SD-WAN.

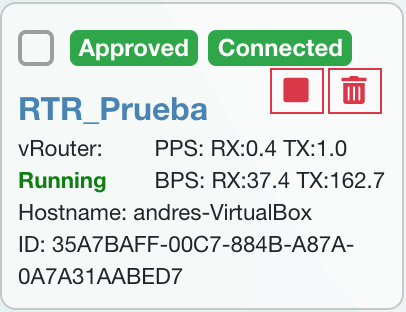


Fig. 5 – Visualización el flexiWAN-router desde la interfaz central manage

Teniendo los sitios remotos configurados y activos en el flexiManage es que se puede comenzar a realizar las visualizaciones y configuraciones que se deseen como por ejemplo cambiar de forma remota las direcciones IP públicas y privadas del router, así como también crear nuevas interfaces. Todo esto se hace de forma centralizada y en pocos minutos lo cual proporciona una flexibilidad y rapidez en la solución que no se obtendría con una solución tradicional. Por otro lado se probó crear túneles virtuales entre los flexiEdges de forma de darle conectividad entre ellos medir también la latencia y los paquetes perdidos en dicho canal. Esto se hace dese la sección Inventory/Tunneles mediante un clic, lo cual muestra otra vez una gran ventaja respecto a sistemas tradicionales donde es necesario realizar configuraciones en ambos extremos para establecer un túnel.

En cuanto a las visualizaciones que permite la plataforma, estas son bastante limitadas pero aceptables. Por un lado es posible obtener las tablas de ruteo de cada flexiEdge y modificarlas o agregar nuevas rutas de ser necesario. También permite obtener estadísticas en tiempo real del tráfico cursado tanto en BPS (Bits por segundo) o PPS (Paquetes por segundo) como se muestra en la Figura 6. Todas las indicaciones fueron tomadas del manual de flexiWAN [iii].

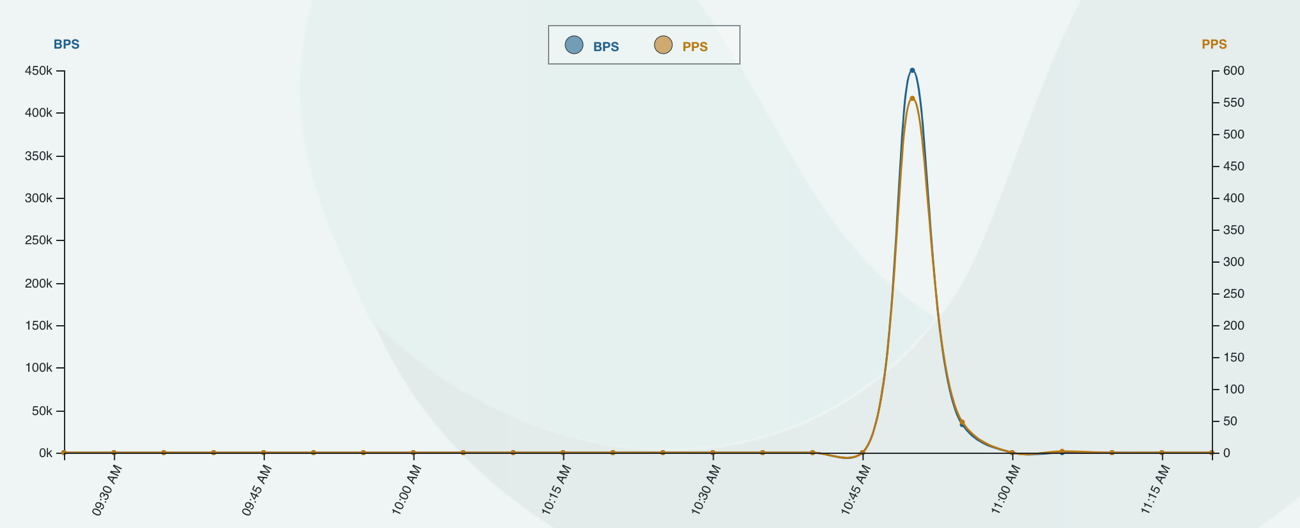


Fig. 6 – Estadísticas de bps y pps en un flexiWAN-router

# Capítulo 3

Desarrollo de la aplicación SD-WAN

En esta sección se procede a detallar en profundidad la solución implementada, las decisiones tomadas y las pruebas técnicas realizadas para dejar constancia del alcance de la misma.

En problema a abordar consiste en el desarrollo e implementación de una solución de ingeniería de tráfico SD-WAN que sea capaz de poder seleccionar bajo demanda el camino que debe tomar un tipo de tráfico determinado a partir de dos componentes esenciales: estado de la red en tiempo real y parámetros de QoS. El escenario a tratar busca parecerse lo mas posible a la realidad donde para llegar de un origen a un destino puede haber mas de un camino posible y es aquí donde el controlador debe decidir cual de todos es el más óptimo para el tipo de tráfico a enviar. Se entiende por “tráfico sensible” todo aquel que previamente haya definido el administrador de red como tal y este se especifica en la sección 3.3.

3.1 Controlador Ryu

SDN cuenta con una arquitectura muy diferente a la tradicional ya que se busca desacoplar de forma lógica y en algunos casos física el plano de control[[6]](#footnote-5) del plano de datos[[7]](#footnote-6) haciendo así la red mas programable, automatizada y controlable. Con estas soluciones nace el concepto de *controlador*, el cual oficia como “cerebro” de la red, teniendo una visión general de la red y comunicándose con los equipos mediante protocolos estandarizados como OpenFlow. Este al ser estándar permite gestión y control centralizado de equipamiento multivendor, utilización de API´s comunes y control granular mediante políticas de sesión, usuarios, dispositivos y aplicaciones.

En la figura 7 se puede apreciar una visión lógica de este tipo de arquitecturas SDN donde queda en evidencia el desacople del plano de datos (capa de infraestructura) y el plano de control (capa de control). En este último se encuentra el controlador SDN el cual se comunica hacia el norte (capa de aplicaciones) mediante API´s haciendo así muy gestionable y programable en función del uso requerido, así como también permitiendo la integración con aplicaciones de negocio, calidad de servicio, entre otras. Por otro lado, la comunicación hacia los dispositivos de red, comúnmente denominados conmutadores, se hace como se mencionó anteriormente por medio de protocolos estándares como por ejemplo OpenFlow mediante paquetes *packet-in[[8]](#footnote-7)* y *packet-out[[9]](#footnote-8).* Mediante estos, el controlador se comunica con los routers y switches permitiendo manipular por ejemplo sus tablas de *forwarding* ya que estos delegan su inteligencia al controlador y pasan a ser simples unidades de conmutación de red.

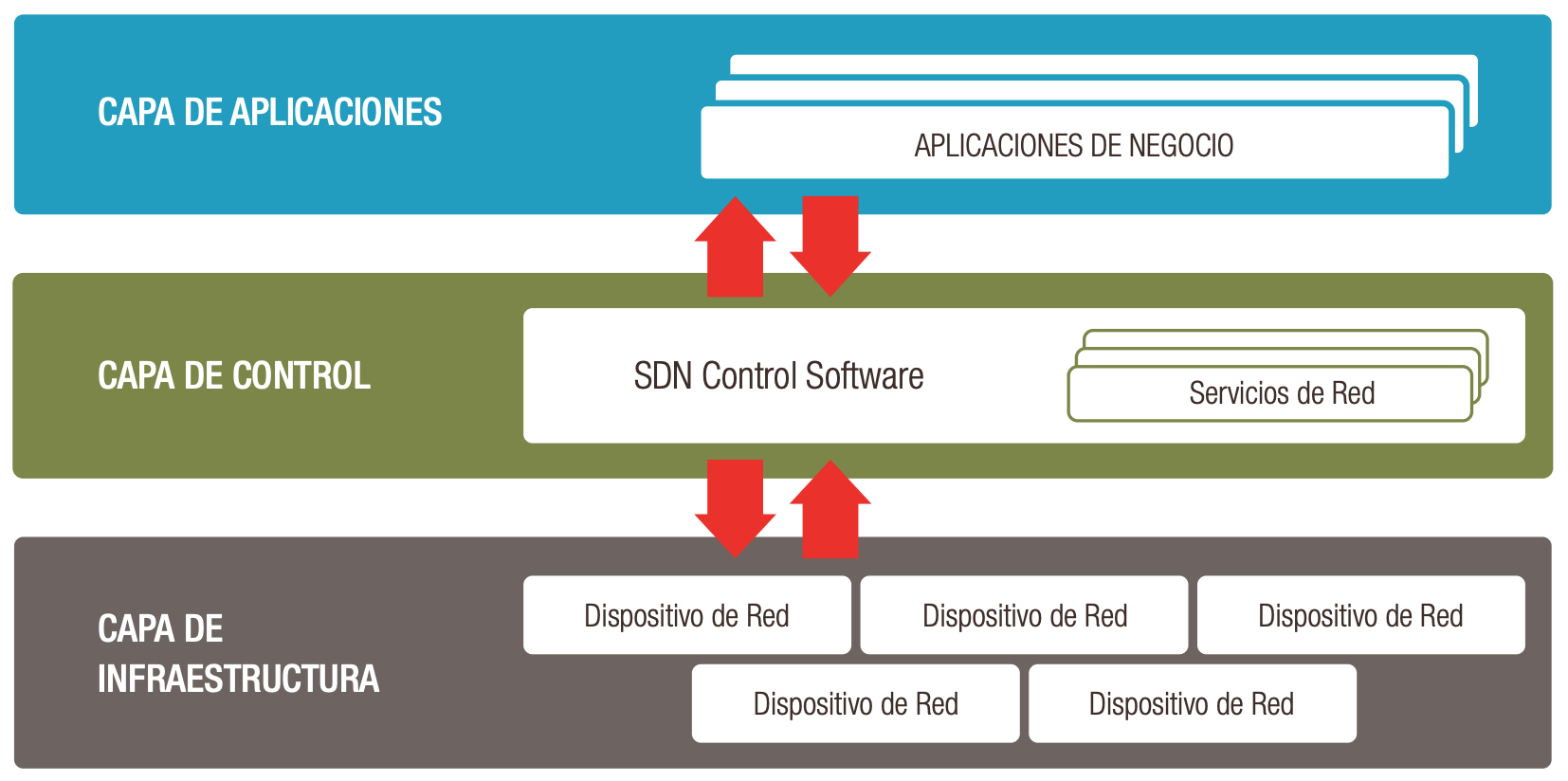


Fig. 7 – Arquitectura SDN [iv]

El controlador elegido para esta implementación es Ryu[[10]](#footnote-9) debido a una serie de factores que lo diferencian del resto. En primer lugar es una solución *Open Source* *SDN* diseñada para incrementar la agilidad de la red facilitando la administración y adaptación de cómo se maneja el tráfico [v]. Cuenta con una gran cantidad de aplicaciones creadas por desarrolladores y publicadas en diversos portales abiertos y foros reconocidos. De esta forma se potencializa el uso de la plataforma ya que se genera un clima acorde para la transferencia de conocimientos y potenciamiento de la herramienta aumentando así el alcance de la misma. En segundo lugar soporta OF-config, NETCONF, extensiones Nicira y diversas versiones de OF (OpenFlow) [v], en especial la 1.3 que es la utilizada en este trabajo. Provee una API REST básica pero a la vez poderosa que ayuda a diseñar soluciones acordes para cada desarrollador, a parte de una interfaz web muy limitada pero que al ser de código abierto permite su adaptación. Este framework está escrito en lenguaje Python y apoyado sobre plataforma Linux lo cual permite la virtualización con Mininet y OpenvSwitch. Esta característica permite a los administradores crear redes virtuales de forma dinámica, disociadas de las redes físicas para por ejemplo satisfacer la ampliación de la capacidad sin afectar los flujos existentes [vi].

Otro motivo por el cual se eligió este controlador es que permite configurar flujos de forma proactiva y reactiva. La manera proactiva consiste en configurar la regla de flujo antes de que el paquete llegue al switch OpenFlow. De esta forma una vez que un paquete lleue al mismo, este ya conoce el tratamiento que debe realizar, reduciendo así el tiempo de encaminamiento. Esto se puede hacer mediante la API REST propia de Ryu. Por otro lado, una configuración reactiva se produce cuando el conmutador recibe un paquete que no coincide con ninguna de sus entradas en la tabla de flujo y debe realizarle una consulta mediante *packet-in* al controlador, respondiéndole este con un *packet-out* la regla de como tratarlo. Esto a gran escala genera un tráfico considerable en la red así como también un retardo (tiempo de paquetes del controlador al conmutador y viceversa, tiempos de procesamiento de los paquetes y tiempo de llenado de tablas de flujo) si se lo compara con el método proactivo [vi].

El código del controlador Ryu se encuentra configurado el Python y está alojado el Github[[11]](#footnote-10). La comunidad Ryu es quien se encarga de efectuarle mejoras, solucionando los posibles fallos y agregando nuevas funcionalidades.

3.2 Escenario virtual

El escenario planteado para la realización de este trabajo consiste en uno que cuenta principalmente con dos condiciones:

* Más de un camino posible para llegar de un origen a un destino
* Red multicamino con “topología de pez”

Estas dos premisas son importantes para poder cumplir con el objetivo primario del trabajo y así decidir en tiempo real el camino que deben tomar los paquetes en cuestión. En cuanto a la “topología de pez” esta se puede apreciar en la figura 8 con los conmutadores A,B,C,D y E. Aquí se puede ver que para ir de R1\_edge a R2\_edge por la red de conmutadores es posible hacerlo por dos caminos diferentes. Uno pasando por CONM\_A, CONM\_C y CONM\_E, al cual llamaremos de aquí en más *“short path”* y otro conformado por CONM\_A, CONM\_B, CONM\_D y CONM\_E al cual se lo llamará *“long path”.* Lo interesante aquí no radica en la existencia de multicaminos sino en que uno de ellos es mas largo que el otro, si se toma como medida la cantidad de saltos. En un escenario con ruteo clásico y suponiendo, como es el caso, que los enlaces son todos iguales de 100 Mbps, el encaminamiento se dará por el camino mas corto ya que tiene menor métrica. Sin embargo la dificultad de este trabajo radica en cómo optimizar la solución para poder en función del tráfico y las condiciones instantáneas de los mismos elegir por cual enviar el tráfico evitando las pérdidas y priorizando aquel que requiera mayores estándares de calidad de servicio



Fig. 8 – Escenario virtual con red de conmutadores en topología de pez.

# Referencias

1. SDN: Software Define Network [↑](#footnote-ref-1)
2. Topología donde existe dos posibles camino para llegar al destino y donde uno de ellos tiene mas enrutadores que el otro y por ende mayor costo de encaminamiento. [↑](#footnote-ref-2)
3. ISP: Proveedor de servicios de intenet [↑](#footnote-ref-3)
4. CPD: Centro de procesamiento de datos [↑](#footnote-ref-4)
5. **[]** Nuage Networks, *Virtualized Network Services*. [Online] de https://onestore.nokia.com/asset/183178

   **[i]** Cisco, *Cisco SD-WAN Solution Overview*. [Online] de https://www.cisco.com/c/en/us/solutions/collateral/enterprise-networks/sd-wan/nb-06-sd-wan-sol-overview-cte-en.html

   **[ii]** FlexiWAN, *flexiWAN documentation*. [Online] de <https://docs.flexiwan.com/>

   **[iv]** Logicalis, Software Defined Networks: el future de las arquitecturas de red. [Online] de https://www.la.logicalis.com/globalassets/latin-america/logicalisnow/revista-20/lnow20-nota-42-45.pdf

   **[v]** sdxcentral, What is Ryu Controller?. [Online] de https://www.sdxcentral.com/networking/sdn/definitions/what-is-ryu-controller/

   **[vi]** ResearchGate, Controladores SDN, elementos para su selección y evaluación. [Online] de https://www.researchgate.net/publication/320711755\_Controladores\_SDN\_elementos\_para\_su\_seleccion\_y\_evaluacion [↑](#endnote-ref-1)
6. Orientado a tratar el tráfico orientado a la gestión, mantenimiento y modificación de la red y sus componentes [↑](#footnote-ref-5)
7. Orientado a tratar el tráfico destinado a los servicios [↑](#footnote-ref-6)
8. Paquetes del conmutador al controlador [↑](#footnote-ref-7)
9. Paquetes del controlador al conmutador [↑](#footnote-ref-8)
10. <https://ryu-sdn.org/index.html> [↑](#footnote-ref-9)
11. <https://github.com/faucetsdn/ryu> [↑](#footnote-ref-10)