# ONGOING RESEARCH EFFORTS AND CHALLENGES

Como ya hemos mencionado anteriormente, el Software-Defined Networking (SDN) es una nueva tecnología que está revolucionando la manera de diseñar y gestionar las redes. SDN ha ido ganando popularidad a causa de su capacidad para simplificar la complejidad de la red y mejorar la escalabilidad, flexibilidad y agilidad. Es una herramienta que puede llegar a aportar un nuevo nivel de automatización y control en la red.

Aunque esta tecnología sea tan prometedora, todavía queda mucho camino para hacer. Quedan muchos retos para resolver antes de que se pueda comercializar su uso de manera generalizada. Actualmente, ya está en proceso de investigación por diferentes ámbitos para poder acelerar su aparición; y a medida que esta investigación va madurando, los investigadores exploran ideas para poder hacerlas más accesibles y fiables. Así pues, en esta sección trataremos varios puntos que envuelven los esfuerzos y retos de la investigación de esta tecnología.

Como nuestro trabajo da énfasis a la capa 8 de SDN, esta sección irá ligada a mucha bibliografía para así no darle mucha extensión. Sin embargo, es un tema de investigación bastante interesante y que cada vez crece más rápido.

## A. Switch Designs

Los diseños de Switches son un componente importante del SDN, puesto que son responsables de la transmisión y control del tráfico de red. La investigación sobre los diseños de estos Switches para SDN se ha centrado en mejorar el rendimiento, la escalabilidad y la programabilidad de su hardware. Esto ha incluido el desarrollo de nuevas arquitecturas, como la OpenFlow Switch, así como la exploración de tecnologías de virtualización y la integración de componentes para poder crear un diseño el más heterogéneo posible. Además, también se ha investigado en la mejora de la seguridad y fiabilidad de los diseños de switches SDN.

Actualmente hay una gran variedad de implementaciones diferentes de Switches y esto provoca que pueda acontecer diferentes niveles de rendimiento inesperados [381] o funcionamientos dispares en el mismo protocolo. Los esfuerzos de investigación se han centrado en la creación de diseños que sean más adecuados para gestionar las demandas de las redes modernas y que se establezca una implementación estándar para ellas. Cómo sería el uso de librerías conjuntas [246] o la agrupación de varios protocolos [421].

Un aspecto clave de SDN que también se está estudiando es la capacidad de la tabla de flujo de los switches, que es la cantidad de reglas que se pueden almacenar [422]. El reto que hay actualmente es facilitar esta implementación para que los Switches puedan soportar y

gestionar tablas de dimensiones más grandes, y lo más importante, de manera eficiente. El centro de la investigación reside en que los chips que pueden permitir una resolución de este reto son caros y además consumen grandes cantidades de energía [424]. Por cuya causa, el ciclo de vida del Switch disminuye provocando un proceso poco eficiente [425].

## B. Controller Platforms

La plataforma de control es el componente central de un sistema SDN, proporcionando las funciones de control y gestión necesarias. Es por eso que es de vital importancia dedicar tiempo de investigación en este sector para poder garantizar mejoras en el funcionamiento del sistema. Primeramente, se han centrado en la mejora del rendimiento con la reducción de la cantidad de overhead que genera [462]. Pero todavía quedan aspectos que siguen siendo un reto. A continuación, trataremos los pilares en investigación en relación a las plataformas de control:

- **Modularidad y flexibilidad:** actualmente se están dedicando recursos al conseguir que los controladores sean lo más modulares y flexibles posibles. Uno de los objetivos para resolverlo consiste al implementar el uso de APIO este/oeste para conseguir un diseño jerárquico. Gracias a esta jerarquía, los controladores de cada nivel pueden ofrecer diferentes abstracciones sobre el enrutamiento de datos y así poder generar una mejora de la escalabilidad y modularidad del sistema [218]. A causa de que la modularidad del sistema no está del todo solucionada, muchos desarrolladores tienen que re-implementar operaciones básicas de la red para poderse adaptar a cada nueva aplicación [29]. Por lo tanto, acaba siendo una práctica difícil de mantener y acaban siendo poco actualizables. Así pues, es difícil tener la idea de un sistema escalable si desde un principio no es nada modular.

- **Interoperabilidad y portabilidad de la aplicación:** La interoperabilidad y la portabilidad de la aplicación del SDN se han convertido en temas clave de investigación a medida que la tecnología continúa evolucionando. Los investigadores están explorando maneras de hacer que el SDN sea más interoperable y portable, de forma que las aplicaciones se puedan desarrollar para funcionar en múltiples plataformas SDN. Primeramente, se escogieron unos lenguajes portables y una serie de interfaces para poder garantizar la portabilidad deseada; pero con el paso del tiempo, los investigadores se dieron cuenta de que no era suficiente. Es por eso que Statesman [465] estipula un marco donde diferentes aplicaciones de la red podían coexistir en el mismo plano de control sin que haya ningún tipo de problema en verso a la seguridad o rendimiento del sistema. Así pues, se puede crear una composición de instrucciones o acciones coordinadas entre todas las aplicaciones que hay en el mismo plan y garantizar que el sistema pueda operar sin problemas. El otro horizonte que se está tratando consiste en poder facilitar que el uso de diferentes lenguajes no afecte al rendimiento del sistema. Esto se está solucionando gracias a unos “Hypervisors” [181] que permiten integrar los lenguajes gracias a un conjunto de reglas OpenFlow.

- **High availability:** Actualmente, los investigadores están investigando diferentes enfoques para conseguir una alta disponibilidad de SDN, como por ejemplo el uso de mecanismos de redundancia, enfoques de sistema distribuido y algoritmos de busca de caminos resistentes. Estos enfoques se están evaluando por su capacidad de reducir el impacto de los fallos de la red, minimizar el tiempo de inactividad, y asegurar que los flujos de datos se encaminan correctamente [359]. Este objetivo es clave, sobre todo en el caso de sistemas distribuidos. Mantener todos los datos de control en un solo punto del sistema generaría penalizaciones graves en el rendimiento. Es por eso que se están desarrollando sistemas SDN que soportan soluciones híbridas donde se tiene que decidir qué están dispuestos a dar prioridad de manera distribuida. Un ejemplo claro de un sistema híbrido funcional como sería el de Onix [7] dónde ha acabado siendo un modelo bastante consistente.

  Una otra dirección de investigación hacia el alta disponibilidad consiste en conseguir una mejora de las API con dirección sur gracias a algoritmos heurísticos de posicionamiento del controlador [468].

- **Delegation of control:** la delegación de control consiste en que podemos delegar funciones de control como respuesta de cualquier alteración o aviso que se ha implementado para garantizar un buen nivel de eficiencia operacional. Es decir, si el sistema detecta algún problema, como algún cambio de estado o algún fallo, el sistema lo notifica para hacer los cambios necesarios para garantizar un buen control. Así pues, también se garantiza resiliencia y fiabilidad [471]. Algunas de las razones documentadas en el papel serían problemas de latencia, overheads generados, tolerancia a fallos y entre otros casos.

## C. Resiliencia

La investigación sobre la resiliencia de SDN está todavía en curso, con un enfoque en la capacidad de la red para mantener operaciones ante diferentes tipos de fallos. Puesto que la resiliencia de una red OpenFlow depende de la tolerancia a errores en el plan de datos y también en la disponibilidad de las funciones.

Los investigadores están explorando el uso de diferentes técnicas para poder resolver estos principales retos. Una de ellas es el uso de protocolos de enrutamiento que toleran fallos, para asegurar que la red pueda continuar operando aunque aparezcan posibles fallos o problemas. Así pues, se resolverán los retos sobre la disponibilidad. Pero actualmente, todavía no hay suficiente información al respecto y es un ámbito que todavía está en exploración [477].

Soluciones actuales para asegurar ciertos niveles resiliencia consiste en que, en una red OpenFlow, si el switch detecta un problema de conexión entre dos puntos, enviará una notificación al controlador y él mismo actualizará las tablas de flow para que la conexión se pueda llevar a cabo y así garantizar un funcionamiento continuo. El problema se resuelve

pero no del todo, este proceso salva al sistema de un fallo pero en cambio genera un tiempo de delay importando provocando directamente una disminución del rendimiento [478]. La intención está en poder reducir este tiempo de delay implementando nuevos métodos.

Una de las nuevas propuestas que está cogiendo bastante actualmente es la de INFLEX [483] que basa la resiliencia de la red SDN al proporcionar recuperación de ruta mediante hashtags en los paquetes que contienen información sobre un plan de enrutamiento virtual. Al detectar un fallo, los routers utilizan esta estrategia para enrutar paquetes modificando las etiquetas de los paquetes.

## D. Scalability

La escalabilidad es uno de los principales retos en la investigación de los SDN. Para poner en contexto, la escalabilidad de red es la capacidad de un sistema SDN para redimensionar su capacidad según sean las necesidades cambiantes de la red [11]. La escalabilidad SDN es un factor crítico en el diseño y despliegue de redes SDN porque afecta el número de usuarios, servicios y aplicaciones que la red puede soportar.

La escalabilidad en la investigación SDN es complicada por la complejidad de la infraestructura de red. Es decir, la red puede llegar a incluir múltiples routers, switches y otros dispositivos. Añadir más dispositivos en la red aumenta la complejidad de gestionar y mantener la red, lo que puede limitar la escalabilidad del sistema. Por ejemplo, en redes de grandes dimensiones se tiene que garantizar se un sistema muy escalable para que sea capaz de soportar grandes cantidades de flujos en poco tiempo, y así garantizar calidad de servicio (QoS) [488].

La mayor parte de la investigación se puede dividir en tres sectores. El data plane, control plane y el hybrid.

Hablando de data plane, lo que se busca es sobre todo distribuir bien el trabajo para no sobrecargar el plan de control. Básicamente consistiría en que los switches sean capaces de identificar los flujos selectivamente sin tener que necesitar el plan de control. Esta idea surge de DevoFlow [418] y SDC [434].

Para el plan de control, la investigación ha consistido en explorar ámbitos ya muy conocidos como arquitecturas de computadores o el paralelismo para así poder aumentar el throughput y garantizar una arquitectura más elástica.

En el caso del hybrid, DIFANE [489] propone switches autoritativos para mantener todo el tráfico en el data plane, con el fin de que el control plane tenga menos carga de trabajo y sea más escalable y eficiente. Estos switches serían los encargados de instalar reglas en el resto de switches, pero el trabajo de generar estas reglas sigue siendo del controlador. Con esta estrategia, lo que se busca es dividir el trabajo para que todo el conjunto sea más escalable y obtenga un mejor rendimiento.

## E. Performance evaluation

No entraremos en mucho detalle en esta sección, pero si recalcamos las ideas principales de este objetivo de investigación. Actualmente, ya hay varios implementaciones OpenFlow a redes de CPD de diferentes escalas y hay una previsión de que este número crezca. De este modo, nuevos retos y experimentos surgirán con el fin de poder profundizar sobre este punto. La evaluación del rendimiento es clave en un sistema SDN puesto que se pueden entender con más facilidad sus limitaciones. Para evaluar el rendimiento de las redes SDN, se han desarrollado una variedad de métricas y técnicas. Estas métricas se utilizan para medir el rendimiento de las redes SDN en comparación con las redes tradicionales, incluyendo el rendimiento de la red, latencia, la pérdida de paquetes, escalabilidad y seguridad.

Para acabar de entender por qué es importante este objetivo de investigación ponemos el siguiente ejemplo. Por un lado tenemos un sistema que ignora las inconsistencias y por otro qué las tiene en cuenta. Según un estudio [466], un sistema que considera inconsistencias funciona de manera significativamente más óptima y a consecuencia el sistema acaba siendo mucho más robusto en comparación con el otro caso.

## F. Security and Dependability

La tecnología de la SDN ha ido aumentado durante el paso del tiempo y es por eso que la seguridad y la fiabilidad es uno de los centros de investigación más necesarios [359]. A medida que las redes se vuelven más complejas e interconexionadas, los riesgos de seguridad, violaciones de datos y ataques maliciosos aumentan significativamente [499]-[504]. Para garantizar la seguridad de las redes SDN, las organizaciones tienen que entender las posibles amenazas y vulnerabilidades asociadas con SDN y desarrollar medidas de seguridad efectivas para proteger la red.

Para garantizar la seguridad y la fiabilidad de SDN, las organizaciones tienen que analizar primero las amenazas que plantea la arquitectura de red. Esto incluye evaluar la seguridad del software y hardware subyacentes, así como los protocolos utilizados para controlar y gestionar la red. Las organizaciones también tienen que evaluar los riesgos potenciales que presentan los actores maliciosos, como los piratas informáticos y los maliciosos. Finalmente, las organizaciones tienen que decidir sobre las mejores estrategias de seguridad para mitigar estos riesgos.

Actualmente ya se han identificado diferentes vectores de amenaza en las arquitecturas SDN [359] debidos a problemas de seguridad o debilidades en la red. No solo vienen directamente por la naturaleza de la SDN, sino que provienen de las redes ya existentes. Con las investigaciones vigentes, se han clasificado 7 posibles vectores de amenazas y también los ataques más comunes [509]: Spoofing, Tampering, Repudiation, Information Disclosure,

Denial of Service y Elevation of privilege. Estos son los principales vectores de amenaza pero en relación a la arquitectura de la SDN, los más castigadores son los numerados como 3, 4 y 5, que son los que atacan directamente el plan de control. Esto provocaría que, si se da un ataque, el atacante tendría control total de la red. El resto de vectores ya estaban presentes en redes tradicionales.

Se han desarrollado varias estrategias y contramedidas como prevención y respuesta de toda posible amenaza. En la mesa ya se ven métodos que ya se usan en las redes tradicionales. El problema está en que todavía es un sector que queda bastante para explorar y la mayoría de métodos todavía no están del todo implementados o soportados por todas las arquitecturas. Además que implementar algunos de estos métodos suelen añadir todavía más complejidad al sistema haciendo que si no es del todo robusto, el rendimiento pueda salir afectado.

## G. Migration and Hybrid Deployments

Esta área de investigación consiste en el desarrollo de estrategias de migración de redes tradicionales a SDNs. Esto implica el análisis de las redes existentes, la identificación de los pasos necesarios para la transición a los SDNs, y el desarrollo de herramientas y técnicas para facilitar el proceso de migración. Además, los investigadores están buscando la mejor manera de gestionar redes híbridas de componentes tradicionales y SDN. Esto implica la integración de elementos de plan de control y de plan de datos, así como el desarrollo de técnicas para la interoperabilidad entre los dos.

Otras investigaciones se centran en el desarrollo de nuevas aplicaciones y servicios que puedan aprovechar la flexibilidad y escalabilidad de la SDN.

Actualmente se están haciendo pruebas de ir integrando poco a poco aplicaciones que se acercan al que entendemos como SDN en infraestructuras de redes tradicionales sin afectar directamente en el control total [219]. Básicamente coge el firmware y hardware que se dispone y lo usa para actuar como si fuera un sistema SDN y obtener, por ejemplo, más flexibilidad y dinamismo en la programabilidad del data plane. El paso siguiente sería ir introduciendo controladores de OpenFlow que ayudarán a los actuales controladores y seguir dando más interoperabilidad. De este modo, cada vez estaríamos más cerca de una transición total del que conocemos como red tradicional a una de SDN.

## H. Meeting Carrier-Grade and Cloud Requirements

A medida que la tecnología SDN ha avanzado, también tiene la necesidad de cumplir los requisitos para la transición hacia el cloud. La investigación para cumplir este requisitos en las redes SDN se ha centrado en mejorar la fiabilidad, escalabilidad, seguridad y rendimiento de estas redes. Es un foco de investigación muy importante puesto que si esta transición se acaba realizando, se optimizarán y simplificarán todo el que rodea la gestión de la red de

cualquier sistema. A continuación, enumeramos los cuatro pilares los cuales se centra la investigación:

- Comunicación mejorada entre los planes de la red pudiendo utilizar, de forma dinámica, altos anchos de banda sin empeorar la calidad de servicio [8].
- Conexión más eficiente entre planes [478].
- Un aumento del nivel de fiabilidad y tolerancia a fallos [478].
- Reducción de hardware y sustitución por otras de más simples y capaces de soportar todo el que SDN favorece [8].

Cosas a decir:

Canvi de paradigma fins al punt de que la majoria de protocols que coneixem actualment acabaran substituits.

Es un paper del 2014, es punt de partida molt interessant per veure una mica la evolucio. Pero encara queden coses per pulir.

En el 2014 habian 2000 papers y ahora en 2022 hay 14000 (IEEE). Dice mucho de que realmente hay mucha investigacion al respecto i bla bla.

—

**Intro:**
Como ya hemos ido viendo a lo largo de esta presentación, la tecnología de las SDN puede acabar siendo un cambio de paradigma, hasta el punto de que la mayoría de protocolos que conocemos actualmente acabarán siendo sustituidos.

Pero antes de llegar a este punto, primero se tienen que resolver los principales retos, que son los que tenemos en pantalla, para así asegurarnos de que es una tecnología accesible y sobre todo que sea fiable.

**Switch Designs:**
Esta investigación se ha centrado en la creación y estandarización de diseños hardware que puedan gestionar los flujos que forman la SDN y que además sigan dando soporte a las redes tradicionales para que se puedan ir integrando poco a poco y así ir acelerando su aparición.

**Controller Platforms:**
Esta se ha centrado en básicamente mejorar el funcionamiento del sistema (controlador). Es decir, aumentar la modularidad y flexibilidad gracias a la jerarquía que se consigue con las API y además mejorar la disponibilidad del sistema con mecanismos de redundancia.

**Resilience:**
Este campo de investigación consiste en conseguir que el sistema siga funcionando correctamente ante cualquier fallada. Por ejemplo, cuando hay un problema de conexión entre dos puntos de la red, éstos notifican al controlador y para solventarlo les actualiza las tablas y así enrutar los flujos para que el sistema no caiga. Este tipo de soluciones implementadas generan un cierto delay. Entonces, uno de los focos de investigación es que cualquier tipo de fallada, como la que he mencionado, provoque cada vez menos delay.

**Scalability:**
Mejorar la capacidad de un sistema SDN para redimensionar su capacidad dependiendo de las necesidades de la red.
Un ejemplo de la necesidad de investigación de este reto sería que en redes de grandes dimensiones se tiene que garantizar un sistema muy escalable para que sea capaz de soportar grandes cantidades de flujos en poco tiempo sin que haya problemas en el servicio.

**Performance evaluation:**
Se han desarrollado varias técnicas y métricas  para medir el rendimiento de las redes SDN en comparación con las redes tradicionales. Estas métricas son las que ya hemos ido viendo a lo largo del curso, como latencia, pérdida de paquetes, el ancho de banda, entre otras. Es necesario tener bien implementado esto ya que así estamos al tanto de cualquier inconsistencia del sistema.

**Security and Dependability:**
El reto en esta investigación consiste en identificar cuáles son los riesgos más fatales de una

SDN, para así luego poder decidir cual es la mejor estrategia para mitigarlos. Ya se identificaron los 7 vectores de más riesgo, de los cuales 3 son específicos de las SDN. Estos consisten en atacar directamente al plano de control comprometiendo toda la red. El resto de vectores ya estaban presentes en las redes tradicionales.

**Migration and hybrid solutions:**
Esta investigación consiste en identificar cuáles son los pasos que hay que seguir para la transición de las redes tradicionales a las SDN. (Cómo sería desarrollando nuevas aplicaciones para redes tradicionales que simulan o se acercan a una SDN)

**Meeting Cloud-requirements:**

La tecnología SDN también aporta nuevas posibilidades a los proveedores de cloud. Ya que, aprovechando el control centralizado de los recursos, es posible simplificar y optimizar la gestión de la red.  Sobre este foco de investigación aún hay varios puntos que resolver pero ya se están empezando a implementar en empresas. Es el caso de Google Cloud que ya tienen implementado SDN.

**Final:**
Para concluir este apartado quería destacar que en 2014 habían 2000 documentos de investigación sobre las SDN y en 2022 van alrededor de los 14000. Esto da entender que cada vez hay más investigación al respecto por lo tanto podríamos decir que cada vez estamos más cerca de llegar a su uso generalizado.

**Conclusiones:**

SDN es un cambio de paradigma muy prometedor pero que aún necesita un tiempo para llegar a su uso generalizado. Aunque cada vez estamos más cerca.

# Bibliografía


[381] C. Rotsos, N. Sarrar, S. Uhlig, R. Sherwood, and A. W. Moore, ‘‘OFLOPS: An open framework for OpenFlow switch evaluation,’’ in Proc. 13th Int. Conf. Passive Active Meas., 2012, pp. 85–95.

[246] M. Yu, A. Wundsam, and M. Raju, ‘‘NOSIX: A lightweight portability layer for the SDN OS,’’ SIGCOMM Comput. Commun. Rev., vol. 44, no. 2, pp. 28–35, Apr. 2014.

[421] A. Vidal, C. E. Rothenberg, and F. L. Verdi, ‘‘The libfluid OpenFlow driver implementation,’’ in Proc. 32nd Brazilian Symp. Comp. Netw. (SBRC), May 2014, pp. 1029–1036.

[422] M. Appelman and M. D. Boer, ‘‘Performance analysis of open-flow hardware,’’ Univ. Amsterdam, Amsterdam, The Netherlands, Tech. Rep., Feb. 2012.

[424] J. Liao, ‘‘SDN system performance,’’ Jun. 2012. [Online]. Available: http://pica8.org/blogs/?p=201

[425] B. Agrawal and T. Sherwood, ‘‘Modeling TCAM power for next generation network devices,’’ in Proc. IEEE Int. Symp. Performance Anal. Syst. Softw., 2006, pp. 120–129.

[462] Y. Zhang, S. Natarajan, X. Huang, N. Beheshti, and R. Manghirmalani, ‘‘A compressive method for maintaining forwarding states in SDN controller,’’ in Proc. 3rd Workshop Hot Topics Softw. Defined Netw., 2014, pp. 139–144.

[218] Open Networking Foundation (ONF), ‘‘SDN architecture,’’ Jun. 2014. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf.

[29] M. Casado, N. Foster, and A. Guha, ‘‘Abstractions for software-defined networks,’’ ACM Commun., vol. 57, no. 10, pp. 86–95, Sep. 2014.

[465] P. Sun et al., ‘‘A network-state management service,’’ in Proc. ACM Conf. SIGCOMM, 2014, pp. 563–574.

[181] X. Jin, J. Rexford, and D. Walker, ‘‘Incremental update for a compositional SDN hypervisor,’’ in Proc. 3rd Workshop Hot Topics Softw. Defined Netw., 2014, pp. 187–192.

[7] T. Koponen et al., ‘‘Onix: A distributed control platform for large-scale production networks,’’ in Proc. 9th USENIX Conf. Oper.

[468] F. J. Ros and P. M. Ruiz, ‘‘Five nines of southbound reliability in software-defined networks,’’ in Proc. 3rd Workshop Hot Topics Softw. Defined Netw., 2014, pp. 31–36.

[471] D. Kreutz, A. Casimiro, and M. Pasin, ‘‘A trustworthy and resilient event broker for monitoring cloud infrastructures,’’ in Proc. 12th IFIP WG 6.1 DAIS, 2012, pp. 87–95.

[477] H. Kim et al., ‘‘Coronet: Fault tolerance for software defined networks,’’ in Proc. 20th IEEE Int. Conf. Network Protocols, Oct. 2012, DOI: 10.1109/ICNP.2012.6459938.

[478] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester, ‘‘OpenFlow: Meeting carrier-grade recovery requirements,’’ Comput. Commun., vol. 36, no. 6, pp. 656–665, Mar. 2013.

[483] J. T. Arau´jo, R. Landa, R. G. Clegg, and G. Pavlou, ‘‘Software-defined network support for transport resilience,’’ in Proc. IEEE Netw. Oper. Manage. Symp., 2014, DOI: 10.1109/NOMS.2014.6838243.

[11] S. Yeganeh, A. Tootoonchian, and Y. Ganjali, ‘‘On scalability of software-defined networking,’’ IEEE Commun. Mag.,

[488] T. Benson, A. Akella, and D. A. Maltz, ‘‘Network traffic characteristics of data centers in the wild,’’ in Proc. 10th ACM SIGCOMM Conf. Internet

[418] A. R. Curtis et al., ‘‘DevoFlow: Scaling flow management for high-performance networks,’’ Comput. Commun. Rev., vol. 41, no. 4, pp. 254–265, Aug. 2011.

[434] J. C. Mogul and P. Congdon, ‘‘Hey, you darned counters! Get off my asic!’’ in Proc. 1st Workshop Hot Topics Softw. Defined Netw., 2012, pp. 25–30.

[489] M. Yu, J. Rexford, M. J. Freedman, and J. Wang, ‘‘Scalable flow-based networking with difane,’’ SIGCOMM Comput. Commun. Rev., vol. 40, no. 4, pp. 351–362, Aug. 2010.

[466] D. Levin, A. Wundsam, B. Heller, N. Handigol, and A. Feldmann, ‘‘Logically centralized? State distribution trade-offs in software defined networks,’’ in Proc. 1st Workshop Hot Topics Softw. Defined Netw., 2012, DOI: 10.1145/2342441.2342443.

[359] D. Kreutz, F. M. Ramos, and P. Verissimo, ‘‘Towards secure and dependable software-defined networks,’’ in Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw., 2013, pp. 55–60.

[499] M. Marchetti, M. Colajanni, M. Messori, L. Aniello, and Y. Vigfusson, ‘‘Cyber attacks on financial critical infrastructures,’’ in Collaborative Financial Infrastructure Protection, R. Baldoni and G. Chockler, Eds. Berlin, Germany.

[504] R. Perez-Pena, ‘‘Universities face a rising barrage of cyberattacks,’’ New York Times, Jul. 2013. [Online].

[509] R. Kloti, ‘‘OpenFlow: A security analysis,’’ M.S. thesis, Dept. Inf. Tech. Elec. Eng., Swiss Fed. Inst. Technol. Zurich (ETH), Zurich, Switzerland, 2013.

[219] R. Hand and E. Keller, ‘‘ClosedFlow: OpenFlow-like control over proprietary devices,’’ in Proc. 3rd Workshop Hot Topics Softw. Defined Netw., 2014, pp. 7–12.

[8] S. Jain et al., ‘‘B4: Experience with a globally-deployed software defined wan,’’ in Proc. ACM SIGCOMM Conf., 2013, pp. 3–14.

[478] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester, ‘‘OpenFlow: Meeting carrier-grade recovery requirements,’’ Comput. Commun., vol. 36, no. 6, pp. 656–665, Mar. 2013.