RELIGACIÓN
REVISTA

Aplicación de políticas de seguridad en SDN (OpenDaylight): mTLS y GBP frente a ataques DoS/MITM en simulación

Enforcing security policies in SDN (OpenDaylight): mTLS and GBP against DoS/MITM in simulation

Gabriel Gustavo Narváez Chiliguano, Roberto Omar Andrade Paredes, Juan Pablo Cuenca Tapia

Resumen

Las redes definidas por software (SDN) centralizan y programan el control del tráfico, pero introducen vulnerabilidades críticas en la controladora. En un entorno simulado con Mininet y OpenDaylight, se identificaron debilidades (credenciales por defecto "admin", comunicación sin cifrado y manipulación de reglas de flujo), se aplicaron políticas de seguridad (reglas OpenFlow vía RESTCONF, Group-Based Policy y mTLS con certificados X.509) y se evaluó su efecto frente a ataques DoS y MITM mediante Wireshark, Ettercap y registros del controlador. En cuatro periodos (T1–T4) se observó disminución de la latencia relativa (~45%→~30%), disponibilidad estable (≈90–95%) y aumento de la tasa de detección (~40%→~45%). Un t-test para muestras relacionadas (α=0,05) indicó mejoras estadísticamente significativas. Los resultados muestran que mTLS + GBP + OpenFlow fortalecen confidencialidad, integridad y disponibilidad en la red SDN simulada.

Palabras clave: Redes definidas por software; OpenDaylight; OpenFlow; Política Basada en Grupos; mTLS; Denegación de servicio; Detección de intrusiones.

Gabriel Gustavo Narváez Chiliguano
Universidad Católica de Cuenca | Cuenca | Ecuador | gnarvaez@ucacue.edu.ec
https://orcid.org/0009-0008-7223-3432
Roberto Omar Andrade Paredes
Universidad Católica de Cuenca | Cuenca | Ecuador | roberto.andrade@ucacue.edu.ec
https://orcid.org/0000-0002-7120-281X
Juan Pablo Cuenca Tapia
Universidad Católica de Cuenca | Cuenca | Ecuador | jcuenca@ucacue.edu.ec
https://orcid.org/0000-0001-5982-634X

http://doi.org/10.46652/rgn.v11i49.1587
ISSN 2477-9083
Vol. 11 No. 49, enero-marzo, 2026, e2601587
Quito, Ecuador

Enviado: julio 30, 2025
Aceptado: septiembre 04, 2025
Publicado: noviembre 21, 2025
Publicación Continua

Abstract

Software-defined networking (SDN) centralizes and programs network control but exposes controller-centric vulnerabilities. In a Mininet + OpenDaylight simulated environment, we identified weaknesses (default "admin" credentials, unencrypted controller–switch traffic, flow rule manipulation), deployed security policies (OpenFlow rules via RESTCONF, Group-Based Policy, mTLS with X.509 certificates) and assessed their effect against DoS and MITM using Wireshark, Ettercap and controller logs. Across four periods (T1–T4), we observed lower relative latency (~45%→~30%), stable availability (≈90–95%), and improved detection rate (~40%→~45%). A paired t-test (α=0.05) indicated statistically significant improvements. Findings show that mTLS + GBP + OpenFlow strengthen confidentiality, integrity and availability in the simulated SDN.

Keywords: SoftwareDefined Networking; OpenDaylight; OpenFlow; GroupBased Policy; mTLS; DenialofService; Intrusion Detection.

Introducción

Antecedentes. SDN utiliza el protocolo OpenFlow para la gestión de la red, habilitando administración centralizada y programable, pero concentra el riesgo en la controladora y su canal de comunicación, con exposición a DoS, MITM, credenciales por defecto e inyección de flujos maliciosos.

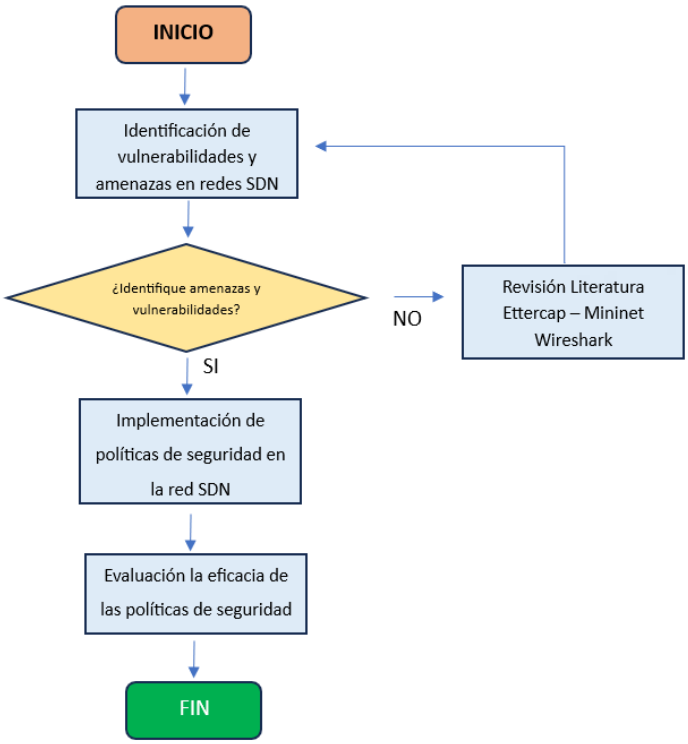
Marco Conceptual. La arquitectura SDN separa los planos de datos y control, unificando la inteligencia de red en un controlador de software. OpenFlow es un protocolo estándar que permite la comunicación entre el controlador – dispositivos de red, mientras Mininet permite emular redes virtuales, para probar políticas y medir métricas de desempeño. La seguridad en SDN enfrenta nuevos vectores de ataque debido a su arquitectura centralizada (tablas de flujo, canal controlador-switch), requiriendo autenticación robusta y cifrado.

Problema y justificación. Las carencias de autenticación robusta y cifrado afectan confidencialidad, integridad y disponibilidad. La simulación permite probar políticas sin afectar producción y medir su impacto en métricas operativas.

Aporte del estudio. (i) Identificación de vulnerabilidades en SDN; (ii) implementación de políticas (OpenFlow/RESTCONF, GBP, mTLS); (iii) evaluación bajo DoS/MITM con análisis estadístico t-test sobre series T1–T4. (Figura 1).

Objetivo. Implementar políticas de seguridad en un entorno SDN simulado para fortalecer la protección contra amenazas cibernéticas, mediante la identificación de vulnerabilidades críticas, la aplicación de mecanismos de control de tráfico y la evaluación de su eficacia con ataques controlados y análisis de tráfico.

Figura 1. Fases del Trabajo de Investigación



Fuente: elaboración propia

Metodología

El diseño del estudio del presente trabajo adoptó un diseño experimental, cuantitativo, y con un componente exploratorio–descriptivo inicial basado en la revisión de literatura especializada. Este enfoque permitió establecer relaciones de causa–efecto entre la implementación de políticas de seguridad (variable independiente) y el impacto de los riesgos en la red SDN (variable dependiente).

El entorno experimental, se configuró utilizando una controladora OpenDaylight conectada a tres switches virtuales y dos hosts, además de un servidor Triple Play que generó tráfico de fondo para emular condiciones de red realistas (Figura 2).

Se emplearon diversas herramientas de software para la simulación, monitoreo y análisis de la red. Mininet permitió emular la interacción entre el controlador SDN y los switches sin requerir hardware físico, facilitando la creación de topologías. Wireshark se utilizó para analizar y monitorear el tráfico, detectar anomalías y validar la efectividad de las políticas de seguridad implementadas. Ettercap ejecutó ataques controlados, como Man in the Middle (MITM) e ICMP flood. Nmap identificó hosts activos, puertos abiertos y posibles vulnerabilidades. Finalmente, RESTCONF permitió la gestión de políticas de seguridad mediante una interfaz API RESTful, facilitando la comunicación directa y centralizada con el controlador SDN.

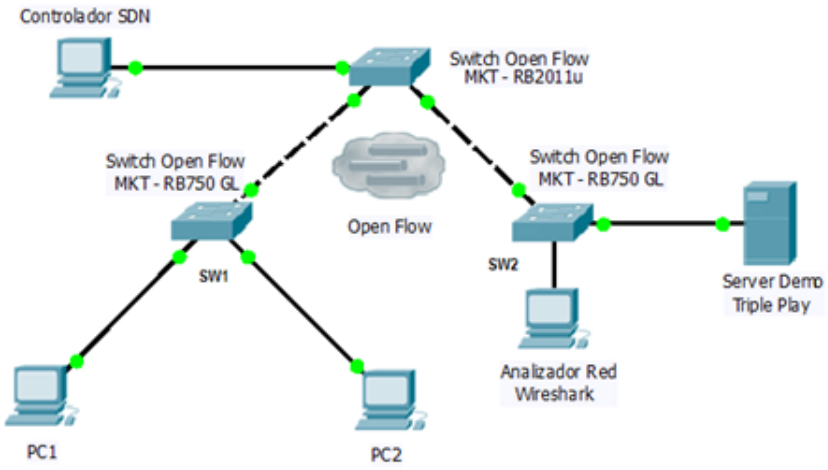
Durante la fase experimental se implementaron dos tipos de escenarios de ataques. El primero, SYN/ICMP Flood, generó una sobrecarga en el controlador SDN y en los switches, provocando alta latencia, pérdida de paquetes y degradación del servicio. El segundo, MITM (ARP Poisoning) este ataque alteró las tablas ARP para interceptar y redirigir tráfico, afectando la confidencialidad e integridad de los datos.

Se recopilaron métricas claves para evaluar la efectividad de las políticas de seguridad implementadas. Se midió la latencia relativa, entendida como el tiempo promedio de respuesta de la red ante solicitudes de comunicación; la disponibilidad del servicio, determinada por la capacidad de mantener la conectividad y estabilidad del sistema bajo condiciones de ataque; y la tasa de detección, calculada como el porcentaje de eventos maliciosos correctamente identificados por los mecanismos de seguridad implementados.

Cada escenario de prueba fue replicado tres veces con el fin de obtener valores promedio y reducir posibles sesgos experimentales. Esta replicación permitió garantizar la consistencia de los resultados y mejorar la validez estadística del análisis comparativo.

El análisis estadístico fue realizado con datos que se procesaron mediante métodos descriptivos y comparativos. Se calcularon medias y desviaciones estándar. Para contrastar los resultados antes y después de la implementación de las políticas de seguridad, se aplicó una prueba t-test pareada ($\alpha = 0{,}05$), determinando la existencia de diferencias significativas en las métricas de tiempo de respuesta y tasa de detección.

Figura 2. Diseño de la topología red SDN



Fuente: elaboración propia

Políticas implementadas:

- OpenFlow + RESTCONF: bloqueo ICMP, segmentación de subredes, denegar Telnet / permitir SSH, ACL por MAC/IP, limitación de tráfico

- Group-Based Policy (GBP): grupos lógicos (“Servidor”, “Clientes”) y contratos (permitir HTTP y denegar por defecto el resto)
- mTLS: autenticación mutua con certificados X.509 entre controlador y switches; canal cifrado
- Las configuraciones/políticas se documentaron en un repositorio (Figuras 3 y 4 presentan scripts).

Figura 3. Script generación Certificado Controlador y switches SDN para mTLS

```
#!/bin/bash
# Script para generar certificados autofirmados para controlador y switches SDN para mTLS

# === 1. Generar clave privada del controlador (2048 bits) ===
openssl genrsa -out controller.key 2048

# === 2. Generar certificado X.509 autofirmado del controlador (365 días) ===
openssl req -x509 -new -nodes \
  -key controller.key \
  -sha256 -days 365 \
  -out controller.crt \
  -subj "/C=EC/ST=Cuenca/O=SDN-red/CN=controller.sdn.local"

# === 3. Generar certificados para switches ===
switches=("switch1" "switch2" "switch3")

for sw in "${switches[@]}"; do
  CN="${sw}.sdn.local"
```

Fuente: elaboración propia

Figura 4. Script Group Based Policy (GBP)

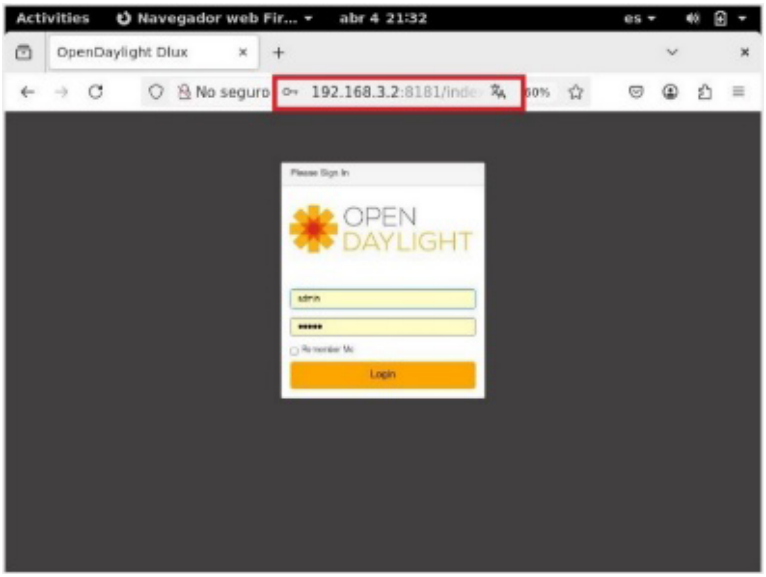
```
#!/bin/bash
# Script para configurar políticas y grupos de endpoints en OpenDaylight

# === 1. Configurar política de tráfico HTTP ===
curl -u admin:admin -H "Content-Type: application/json" \
  -X PUT \
  http://192.168.3.2:8181/restconf/config/policy:policy/contract/permit-http \
  -d '{
    "contract": [
      {
        "id": "permit-http",
        "subject": [
          {
            "name": "http-traffic",
            "rule": [
              {
                "name": "allow-http",
                "classifier-ref": [
                  {
                    "name": "tcp-destination-port",
                    "value": "80"
                  }
                ],
                "action-ref": [
                  {
                    "name": "allow"
                  }
                ]
              }
            ]
          }
        ]
      }
    ]
  }'
```

Fuente: elaboración propia

Resultados

Vulnerabilidades observadas. Credenciales por defecto "admin" permitieron manipular reglas (Figura 5); el tráfico no cifrado controlador–switch expuso la red a MITM; se lograron DoS que incrementaron latencia y afectaron disponibilidad.

Figura 5. Credenciales predeterminadas en la controladora "admin"



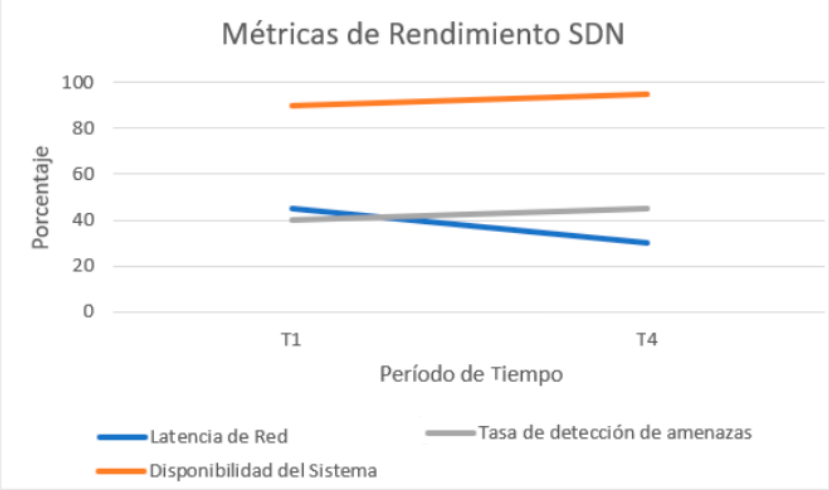
Fuente: elaboración propia

Efecto de las políticas

- - Latencia relativa: tendencia decreciente ~45%→30% (T1→T4)
- - Disponibilidad: alta y estable ~90–95%
- - Detección: incremento ~40%→~45%

La prueba t para muestras relacionadas (α=0,05) indicó diferencias significativas entre condiciones (Figura 6).

Figura 6. Métricas de rendimiento SDN



Fuente: elaboración propia

Discusión

La literatura reciente identifica los ataques de *flow table overflow* como amenazas críticas en SDN. Estudios como " *FloRa: Flow Table Low-Rate Overflow Reconnaissance and Detection in SDN*" (2024) y " *Manipulating OpenFlow Link Discovery Packet Forwarding for Topology Poisoning*" (2024), revelan técnicas de manipulación de flujos, inyección de paquetes y ataques DoS/MITM, que alteran la percepción del controlador sobre la topología y redirigen tráfico de forma encubierta. Los dos trabajos coinciden en la necesidad de implementar defensas avanzadas, como el aprendizaje automático, para mitigar estos riesgos de manera eficiente y adaptativa.

Los resultados experimentales validaron los hallazgos reportados en la literatura especializada: las amenazas analizadas (inyección de paquetes, DoS y manipulación de flujos) se reprodujeron exitosamente en el entorno simulado, mientras que las contramedidas implementadas —reglas OpenFlow, políticas basadas en grupos (GBP) y autenticación mutua TLS (mTLS)— demostraron alinearse con las mejores prácticas actuales en seguridad SDN.

Conclusiones

La combinación de mTLS, GBP y reglas OpenFlow fortaleció la confidencialidad, integridad y disponibilidad en la red SDN simulada.

Se observaron reducciones de latencia relativa, mejoras en detección y estabilidad de la disponibilidad, con significancia estadística (t-test). Este enfoque ofrece una base práctica para endurecer SDN en escenarios controlados.

Se identificaron requisitos específicos de versión entre OpenDaylight y OVS (1.5+), destacando la necesidad de una gestión rigurosa de dependencias en entornos SDN.

El bloqueo de puertos 6633/6653 a nivel de firewall, evidenciaron la importancia de la preparación del entorno antes del despliegue.

Las métricas obtenidas en el entorno simulado presentan carácter relativo y un tamaño muestral reducido (3 réplicas), por lo que la generalización de resultados a entornos productivos requiere validación adicional y ampliación de pruebas.

Referencias

Arévalo-Herrera, J., Camargo Mendoza, J., & Martínez Torre, J. I. (2025). Assessing SDN controller vulnerabilities: A survey on attack typologies, detection mechanisms, controller selection, and dataset application in machine learning. *Wireless Personal Communications, 140, 739–775*. https://doi.org/10.1007/s11277-025-11748-w

Mabood, N., Tariq, N., Khan, F. A., & Ashraf, M. (2025). A comprehensive survey on software defined networking (SDN) security. En M. Arif, A. Jaffar, & O. Geman, (eds.). *Computing and emerging technologies (ICCET 2023)*. Springer, Cham. https://doi.org/10.1007/978-3-031-77617-5_18

Mejía Viteri, J. T., Gonzales Valero, M. I., Fernández Torres, A. del R., & Crespo Torres, N. M. (2024). Seguridad contra ataques DDoS en los entornos SDN con inteligencia artificial. *Revista Metropolitana de Ciencias, 7*(3). https://doi.org/10.33262/rmc.v7i3.2844

Mudgal, A., Verma, A., Singh, M., Sahoo, K. S., Elmroth, E., & Bhuyan, M. (2024). FloRa: Flow table low-rate overflow reconnaissance and detection in SDN. *IEEE Transactions on Network and Service Management, 21*(6), 6670–6683. https://doi.org/10.1109/TNSM.2024.3446178

Ohri, P., & Guha Neogi, S. (2022). Software-defined networking security challenges and solutions: A comprehensive survey. *International Journal of Computing and Digital Systems, 12*(1), 383–400. https://doi.org/10.12785/ijcds/120131

Open Networking Foundation. (2025). *OpenFlow 2.0*. https://n9.cl/rw7o3

Ouedraogo Rakissaga, W. A., Omar, H. H., & Kouraogo, P. J. (2025). Software Defined Networks: Strengths, weaknesses, and resilience to failures. *Engineering, 17*(1), 20–35. https://doi.org/10.4236/eng.2025.171002

Shahzad, M., Rizvi, S., Khan, T. A., Ahmad, S., Siddiqui, M. S., Chen, J., Li, X., Wang, Y., Kumar, N., Patel, R., García, S., Fernández, L., Müller, H., Schmidt, P., Ivanov, A., Silva, C., Kim, Y., & Al-Mistarihi, M. (2025). An exhaustive parametric analysis for securing SDN through traditional, AI/ML, and blockchain approaches: A systematic review. *International Journal of Networked and Distributed Computing, 13*(1). https://doi.org/10.1007/s44227-024-00055-8

Swileh, M. N., & Zhang, S. (2024). Unseen attack detection in software-defined networking using a BERT-based large language model. *arXiv.* https://arxiv.org/abs/2412.06239

Wang, H. (2025). A zero-trust security model applied in SDN intelligent network. *SPIE Digital Library.* https://doi.org/10.1117/12.3058610

Autores

Gabriel Gustavo Narváez Chiliguano. Graduado en la Escuela Politécnica del Ejército obteniendo el título de Ingeniero en Electrónica y Telecomunicaciones. Graduado en la Universidad de Buenos Aires, obteniendo el título de Especialista en Redes y Servicios de Telecomunicaciones. Docente del Instituto Superior Tecnológico Ismael Pérez Pazmiño.

Roberto Omar Andrade Paredes. Ingeniero Electrónico, con una Maestría en Gerencia de Redes y Telecomunicaciones, además, cuento con una Maestría en Sistemas de Información con Mención en Inteligencia de Negocios y Analítica de Datos Masivos. Poseo también un Doctorado en el programa oficial de Doctorado en Software, Sistemas y Computación.

Juan Pablo Cuenca Tapia. Ingeniero en sistemas con Maestría en Sistemas de Información Gerencial y una Maestría en Tecnologías de la Información.

Declaración

Conflicto de interés

No tenemos ningún conflicto de interés que declarar.

Financiamiento

Sin ayuda financiera de partes externas a este artículo.

Nota

El artículo es original y no ha sido publicado previamente.