Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi
P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

Meningkatkan Keamanan dan Mitigasi pada Arsitektur Software Defined Network

Wahyudi[1], Wafiqah Yasmin Azhar[2], Arif Budimansyah Purba[3] . Anwar Hilman[4], Vivi Ayu Lestari[5], Ahmad Najib Mutawally[6]
[1,2,3,4,5,6] Universitas Horizon Indonesia
Email: wahyudi008@gmail.com

Abstract
Software Defined Network (SDN) is a networking paradigm that offers flexibility and efficiency in network management through the separation of the control plane and data plane. However, the centralized architecture that characterizes SDN also introduces significant security vulnerabilities, such as Distributed Denial of Service (DDoS) attacks, spoofing, controller manipulation, and multi-vector threats. This research aims to conduct a systematic literature review to identify the main security challenges in SDN and the solutions proposed by previous researchers. The method used involves an in-depth analysis of scientific publications related to SDN security, including mitigation approaches such as encryption, authentication, the use of machine learning, and network behavior analysis. The results of this study include a comprehensive mapping of SDN security threats, an evaluation of the effectiveness of existing solutions, and the identification of research gaps that require further attention. Thus, this research is expected to make a significant contribution to the development of a more holistic, proactive, and adaptive SDN security framework to address modern security challenges.
Keywords: *Software Defined Network (SDN), Network Security, Cyber Attacks, Machine Learning, Encryption*

Abstrak
Software Defined Network (SDN) merupakan pendekatan jaringan yang menawarkan kemudahan dalam pengelolaan dan efisiensi operasional melalui pemisahan antara control plane dan data plane. Meskipun demikian, desain arsitektur terpusat yang menjadi ciri utama SDN juga membawa risiko keamanan yang cukup besar, seperti serangan *Distributed Denial of Service* (DDoS), *spoofing*, peretasan kontroler, hingga ancaman multi-vektor. Penelitian ini bertujuan untuk melakukan kajian literatur secara sistematis guna mengidentifikasi tantangan keamanan utama pada SDN serta solusi yang telah diusulkan oleh peneliti sebelumnya. Metode yang digunakan melibatkan analisis mendalam terhadap berbagai publikasi ilmiah terkait keamanan SDN, termasuk strategi mitigasi seperti enkripsi, autentikasi, penerapan *machine learning*, dan analisis pola perilaku jaringan. Hasil dari penelitian ini mencakup pemetaan menyeluruh terhadap potensi ancaman keamanan pada SDN, evaluasi terhadap efektivitas solusi yang telah ada, serta identifikasi celah penelitian yang membutuhkan eksplorasi lebih lanjut. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi penting dalam pengembangan kerangka kerja keamanan SDN yang lebih komprehensif, antisipatif, dan responsif terhadap tantangan keamanan masa kini.
Kata Kunci: *Software Defined Network* (SDN), Keamanan Jaringan, Serangan Siber, *Machine Learning*, Enkripsi.

Article History :

Received 05 April 2025
Revised 12 April 2025
Accepted 21 April 2025

Corresponding Author:

Nama Penulis, : Wahyudi
Departemen, : Sistem Informasi
Instansi : Universitas Horizon Indonesia
Alamat : Jl. Pangkal Perjuangan By Pass No.KM.1, Tanjungpura, Karawang, Jawa Barat 41316
Email Penulis.: wahyudi008@gmail.com

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

1. Pendahuluan

Software Defined Network (SDN) telah muncul sebagai paradigma jaringan yang revolusioner dengan memisahkan *control plane* dari *data plane*, sehingga memungkinkan pengelolaan jaringan yang lebih fleksibel dan terpusat [1]. Arsitektur ini memungkinkan pengelolaan jaringan secara dinamis melalui kontroler pusat, yang bertindak sebagai otak dari seluruh operasi jaringan. Namun, seperti yang dijelaskan oleh Kreutz et al. (2015), desain terpusat ini juga menimbulkan kerentanan keamanan yang signifikan, karena kontroler menjadi titik tunggal kegagalan (*single point of failure*) yang sangat rentan terhadap serangan siber [1]. Ancaman keamanan pada SDN mencakup serangan seperti *Distributed Denial of Service* (DDoS), *spoofing*, manipulasi kontroler, hingga ancaman multi-vektor yang semakin kompleks [2].

Menurut penelitian oleh Dhawan et al. (2015), serangan pada lapisan kontrol SDN dapat menyebabkan gangguan jaringan yang parah, termasuk pemadaman layanan dan kebocoran data [3]. Sebagai contoh, serangan DDoS dapat membanjiri kontroler dengan permintaan palsu, sehingga menurunkan kinerja keseluruhan jaringan [3]. Selain itu, studi oleh Alkadi et al. (2020) menunjukkan bahwa ancaman keamanan SDN terus berkembang seiring dengan meningkatnya kompleksitas jaringan, termasuk serangan canggih seperti *zero-day exploits* yang sulit dideteksi dan dicegah [4].

Meskipun banyak penelitian telah dilakukan untuk mengatasi masalah keamanan SDN, masih ada celah dalam hal integrasi solusi yang komprehensif dan efektif. Sebagai contoh, solusi berbasis enkripsi dan autentikasi yang diusulkan oleh Xie et al. (2016) efektif dalam mencegah jenis-jenis serangan tertentu, tetapi kurang mampu menghadapi ancaman canggih seperti *zero-day exploits* [5]. Sementara itu, pendekatan *machine learning* yang diusulkan oleh Tang et al. (2019) menunjukkan potensi dalam mendeteksi serangan secara real-time dengan menganalisis pola lalu lintas jaringan [6]. Namun, model ini masih menghasilkan tingkat *false positive* yang signifikan, yang dapat mengganggu operasi jaringan [6]. Selain itu, penelitian sebelumnya cenderung fokus pada ancaman individu tanpa mempertimbangkan dampak kumulatif dari serangan multi-vektor, seperti kombinasi antara DDoS dan *spoofing*, yang dapat meningkatkan efektivitas serangan secara signifikan [7].

Berdasarkan tinjauan terhadap penelitian sebelumnya, penelitian ini bertujuan untuk melakukan analisis studi litelatur sistematis terhadap tantangan keamanan SDN dan solusi yang telah diusulkan oleh para peneliti sebelumnya, serta mengidentifikasi celah yang perlu ditangani. Dengan menggabungkan pendekatan dari berbagai penelitian sebelumnya, penelitian ini berupaya memberikan rekomendasi yang lebih holistik dan inovatif untuk meningkatkan keamanan SDN. Tujuan utama penelitian ini adalah memetakan ancaman keamanan SDN, mengevaluasi efektivitas solusi yang ada, dan mengusulkan kerangka kerja yang dapat diadopsi untuk penelitian lanjutan. Dengan demikian, studi ini diharapkan dapat memberikan kontribusi terhadap pengembangan sistem SDN yang lebih aman dan andal di masa depan.

2. Tinjauan Pustaka

Software Defined Network (SDN) telah menjadi topik penelitian yang intensif dalam sepuluh tahun terakhir, terutama berkaitan dengan tantangan keamanan yang timbul dari arsitektur terpusatnya. Menurut Kreutz et al. (2015), SDN memisahkan *control plane* dan *data plane*, yang memungkinkan pengelolaan jaringan lebih fleksibel tetapi juga menciptakan titik tunggal kegagalan (*single point of failure*) yang rentan terhadap serangan [1].

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

Arsitektur ini memberikan kemampuan untuk mengelola jaringan secara dinamis melalui kontroler pusat, namun kontroler tersebut menjadi sasaran utama bagi pelaku serangan. Scott-Hayward et al. (2013) menyoroti beberapa ancaman utama pada SDN, seperti serangan *Distributed Denial of Service* (DDoS), spoofing, dan manipulasi kontroler [2]. Serangan DDoS, misalnya, dapat membanjiri kontroler SDN dengan permintaan palsu, sehingga mengganggu kinerja keseluruhan jaringan.

Ancaman keamanan pada SDN dapat dikelompokkan ke dalam tiga kategori utama: serangan pada lapisan kontrol, serangan pada lapisan data, dan serangan multi-vektor. Dhawan et al. (2015) menjelaskan bahwa serangan pada lapisan kontrol dapat menyebabkan gangguan serius pada jaringan, termasuk kegagalan layanan dan kebocoran data [3]. Di sisi lain, Xie et al. (2016) menyoroti bahwa serangan pada lapisan data, seperti spoofing dan eavesdropping, dapat mengeksploitasi kerentanan dalam komunikasi antara kontroler dan switch [5]. Wang et al. (2021) menambahkan bahwa serangan multi-vektor, seperti kombinasi DDoS dan spoofing, semakin umum terjadi dan memiliki dampak yang lebih besar dibandingkan serangan tunggal [7].

Beberapa peneliti telah mengusulkan solusi untuk mengatasi masalah keamanan SDN. Dhawan et al. (2015) mengembangkan SPHINX, sebuah framework yang dirancang untuk mendeteksi serangan pada lapisan kontrol SDN dengan menganalisis pola lalu lintas jaringan [3]. Meskipun efektif dalam mendeteksi serangan tertentu, SPHINX memiliki keterbatasan dalam hal skalabilitas dan kemampuan untuk menghadapi serangan multi-vektor. Selain itu, Xie et al. (2016) mengusulkan penggunaan enkripsi dan autentikasi untuk melindungi komunikasi antara kontroler dan switch SDN [5]. Namun, pendekatan ini kurang efektif dalam menghadapi serangan canggih seperti *zero-day exploits*, yang membutuhkan deteksi berbasis perilaku.

Pendekatan *machine learning* (ML) juga telah diterapkan untuk meningkatkan keamanan SDN. Tang et al. (2019) mengusulkan model *deep learning* untuk mendeteksi serangan secara *real-time* dengan menganalisis pola lalu lintas jaringan [6]. Meskipun menunjukkan tingkat akurasi yang tinggi, model ini masih menghasilkan *false positive* yang signifikan, yang dapat mengganggu operasi jaringan. Alkadi et al. (2020) meninjau berbagai teknik berbasis AI untuk deteksi intrusi di SDN dan menyimpulkan bahwa integrasi antara *machine learning* dan metode tradisional diperlukan untuk meningkatkan keandalan sistem [4]. Shin dan Gu (2013) menekankan bahwa meskipun solusi seperti *CloudWatcher* efektif dalam memantau lalu lintas jaringan, mereka kurang mampu menghadapi serangan yang memanfaatkan celah pada protokol *OpenFlow* itu sendiri [8].

Meskipun banyak solusi telah diusulkan, masih ada celah dalam penelitian keamanan SDN. Pertama, sebagian besar penelitian hanya fokus pada ancaman individu tanpa mempertimbangkan dampak kumulatif dari serangan multi-vektor. Kedua, solusi yang ada cenderung bersifat reaktif daripada proaktif, artinya mereka hanya merespons serangan setelah terjadi, bukan mencegahnya sejak awal. Ketiga, banyak solusi yang belum diuji dalam skala besar atau lingkungan jaringan yang kompleks, sehingga efektivitasnya dalam skenario dunia nyata masih dipertanyakan. Penelitian ini bertujuan untuk mengatasi celah tersebut dengan mengusulkan kerangka kerja keamanan SDN yang lebih komprehensif dan terintegrasi.

Benton et al. (2013) menyoroti bahwa banyak solusi keamanan SDN yang diusulkan belum diuji dalam lingkungan jaringan dinamis dan skala besar [9]. Sebagian besar penelitian dilakukan dalam

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

simulasi atau laboratorium, yang mungkin tidak mencerminkan kompleksitas jaringan dunia nyata. Oleh karena itu, penelitian lanjutan diperlukan untuk memvalidasi solusi keamanan SDN dalam skenario yang lebih realistis, termasuk jaringan dengan banyak pengguna, perangkat, dan aplikasi. Porras et al. (2012) menambahkan bahwa pendekatan keamanan SDN saat ini cenderung bersifat reaktif, yaitu merespons serangan setelah terjadi, bukan mencegahnya sejak awal [10]. Mereka mengusulkan perlunya integrasi antara sistem deteksi intrusi (IDS) dan sistem pencegahan intrusi (IPS) yang dapat bekerja secara proaktif. Namun, implementasi solusi semacam ini masih menghadapi tantangan dalam hal skalabilitas dan overhead komputasi.

Selain itu, Hu et al. (2016) menyoroti pentingnya penguatan protokol komunikasi dalam SDN untuk mengurangi risiko serangan yang mengeksploitasi celah pada protokol *OpenFlow* [11]. Jafarian et al. (2012) mengusulkan pendekatan random host mutation untuk mencegah serangan yang memanfaatkan informasi statis tentang host dalam jaringan [12]. Yan et al. (2016) meninjau tantangan keamanan SDN dalam lingkungan cloud computing dan menemukan bahwa serangan DDoS semakin kompleks dengan munculnya teknologi baru seperti *Internet of Things* (IoT) [13]. Zhang et al. (2019) menambahkan bahwa analisis perilaku jaringan dapat digunakan untuk mendeteksi aktivitas mencurigakan secara proaktif, meskipun pendekatan ini membutuhkan sumber daya komputasi yang besar [14]. Liu et al. (2020) menyoroti bahwa jaringan IoT yang menggunakan SDN memiliki tantangan keamanan tambahan karena jumlah perangkat yang sangat besar dan heterogen [15].

Penelitian ini bertujuan untuk mengatasi celah tersebut dengan mengusulkan kerangka kerja keamanan SDN yang lebih komprehensif. Kerangka kerja ini akan menggabungkan pendekatan berbasis *machine learning*, penguatan protokol komunikasi, dan validasi dalam lingkungan jaringan yang realistis. Dengan demikian, penelitian ini diharapkan dapat memberikan solusi yang lebih efektif dan proaktif untuk menghadapi ancaman keamanan SDN yang terus berkembang.

3. Metode

Penelitian ini menggunakan pendekatan studi literatur sistematis untuk mengidentifikasi, menganalisis, dan mensintesis informasi dari berbagai penelitian sebelumnya yang relevan dengan keamanan *Software Defined Network* (SDN). Data diperoleh dari jurnal ilmiah, konferensi internasional, serta laporan penelitian yang membahas aspek keamanan SDN. Fokus utama penelitian ini adalah pada jenis serangan umum seperti *Distributed Denial of Service* (DDoS), spoofing, manipulasi kontroler, serta solusi mitigasi seperti enkripsi, autentikasi, dan pemanfaatan *machine learning*. Penulis-penulis seperti Kreutz et al. (2015), Scott-Hayward et al. (2013), Dhawan et al. (2015), Xie et al. (2016), Tang et al. (2019), Wang et al. (2021), Shin & Gu (2013), Benton et al. (2013), Porras et al. (2012), dan Alkadi et al. (2020) menjadi referensi utama dalam penelitian ini [1-10].

Flowchart Metodologi Penelitian

Untuk memvisualisasikan metodologi penelitian, disajikan *flowchart* yang menggambarkan langkah-langkah utama:

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435



Gambar 1: *Flowchart* Metode Penelitian

Flowchart ini menunjukkan proses sistematis mulai dari identifikasi masalah hingga pengusulan solusi yang lebih komprehensif. Setiap tahap dirancang untuk memastikan bahwa penelitian ini memberikan kontribusi nyata dalam meningkatkan keamanan dan kinerja SDN.

Tujuan utama metode ini adalah untuk memetakan masalah keamanan SDN, mengevaluasi solusi yang telah diusulkan oleh para peneliti sebelumnya, serta mengidentifikasi kesenjangan yang perlu ditangani. Penelitian ini juga bertujuan untuk mengusulkan kerangka kerja keamanan SDN yang lebih komprehensif dan efektif dengan menggabungkan pendekatan berbasis *machine learning*, enkripsi, analisis perilaku, dan validasi dalam lingkungan jaringan yang realistis.

A. Identifikasi Masalah Keamanan SDN

Masalah keamanan SDN yang paling kritis dikategorikan ke dalam tiga area utama: ancaman pada lapisan kontrol, ancaman pada lapisan data, dan serangan multi-vektor. Ketiga area ini saling terkait dan membutuhkan penanganan hati-hati untuk menjaga keandalan dan integritas jaringan SDN.

1. **Ancaman pada Lapisan Kontrol**
   Lapisan kontrol merupakan pusat kendali dalam arsitektur SDN, yang mengatur seluruh aliran data dalam jaringan. Namun, seperti yang dijelaskan oleh Kreutz et al. (2015), desain terpusat membuat kontroler menjadi titik tunggal kegagalan (*single point of failure*) [1]. Serangan *Distributed Denial of Service* (DDoS) dapat membanjiri kontroler dengan permintaan palsu, sehingga mengganggu operasi jaringan secara keseluruhan [3]. Selain itu, Scott-Hayward et al. (2013) menyoroti bahwa manipulasi kontroler dapat digunakan untuk mengubah kebijakan jaringan, yang berpotensi mengalihkan atau memblokir aliran data tertentu [2].
2. **Ancaman pada Lapisan Data**
   Lapisan data bertanggung jawab atas pengiriman dan penerimaan data dalam jaringan SDN. Xie et al. (2016) menjelaskan bahwa serangan seperti spoofing dan *man-in-the-middle* (MITM) dapat merusak integritas dan kerahasiaan data yang ditransmisikan [5]. Tang et al. (2019) menambahkan bahwa sifat dinamis dari lalu lintas data di SDN membuat serangan pada lapisan ini sulit dideteksi, sehingga memerlukan solusi keamanan yang lebih canggih [6].
3. **Serangan Multi-Vektor**
   Serangan multi-vektor, seperti kombinasi antara DDoS dan spoofing, semakin sering terjadi dan memiliki dampak lebih besar dibandingkan serangan tunggal [7]. Shin & Gu (2013) menekankan bahwa pendekatan keamanan yang hanya fokus pada satu jenis serangan tidak akan efektif dalam menghadapi ancaman yang lebih kompleks [8].

B. Analisis Kesenjangan dalam Penelitian Sebelumnya

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

Berdasarkan analisis terhadap penelitian sebelumnya, ditemukan beberapa kesenjangan penting dalam studi keamanan SDN:

1. **Kurangnya Integrasi Solusi yang Komprehensif**
   Kreutz et al. (2015) dan Scott-Hayward et al. (2013) menyoroti bahwa banyak penelitian hanya fokus pada satu jenis ancaman tertentu, seperti serangan penyadapan atau pemalsuan identitas, tanpa mempertimbangkan sifat multi-vektor dari serangan dunia nyata [1][2]. Dhawan et al. (2015) menambahkan bahwa solusi yang ada cenderung bersifat reaktif, hanya merespons setelah serangan terjadi, bukan proaktif dalam mencegah serangan sejak awal [3]. Akibatnya, sistem keamanan yang ada sering kali gagal melindungi jaringan secara menyeluruh.
2. **Kurangnya Validasi dalam Skala Besar**
   Benton et al. (2013) dan Porras et al. (2012) mencatat bahwa sebagian besar solusi hanya diuji dalam lingkungan simulasi sederhana atau laboratorium, yang mungkin tidak mencerminkan kompleksitas jaringan dunia nyata [9][10]. Tang et al. (2019) menemukan bahwa model machine learning sering menghasilkan tingkat *false positive* yang tinggi ketika diterapkan dalam skenario dunia nyata [6]. Hal ini dapat menyebabkan alarm palsu yang mengganggu operasional jaringan.
3. **Keterbatasan dalam Menghadapi Serangan Canggih**
   Serangan canggih seperti *zero-day exploits* tetap menjadi tantangan besar. Xie et al. (2016) dan Alkadi et al. (2020) mencatat bahwa solusi berbasis enkripsi kurang efektif dalam menghadapi serangan yang memanfaatkan celah keamanan baru [5][4]. Selain itu, model *machine learning* memiliki keterbatasan dalam mendeteksi serangan yang belum pernah terlihat sebelumnya, karena membutuhkan dataset pelatihan yang luas dan berkualitas tinggi.

C. Evaluasi Solusi yang Telah Diusulkan

Berbagai solusi telah diusulkan untuk mengatasi tantangan keamanan SDN, masing-masing dengan kelebihan dan kekurangan:

1. **Solusi Berbasis Enkripsi dan Autentikasi**
   Xie et al. (2016) mengusulkan penggunaan protokol TLS untuk melindungi komunikasi antara kontroler dan switch [5]. Meskipun efektif dalam mencegah serangan seperti penyadapan, pendekatan ini membutuhkan sumber daya komputasi yang besar. Porras et al. (2012) mengembangkan kernel keamanan untuk protokol *OpenFlow*, tetapi pendekatan ini kurang efektif dalam menghadapi serangan *zero-day exploits [*10].
2. **Solusi Berbasis *Machine Learning***
   Tang et al. (2019) mengusulkan model deep learning untuk mendeteksi serangan secara *real-time* dengan akurasi tinggi [6]. Namun, model ini masih menghasilkan false positive yang signifikan, yang dapat mengganggu operasi jaringan. Alkadi et al. (2020) menyarankan integrasi antara machine learning dan metode tradisional untuk meningkatkan keandalan deteksi ancaman [4].
3. **Solusi Berbasis Analisis Perilaku**
   Shin & Gu (2013) mengembangkan sistem monitoring berbasis *OpenFlow* untuk mendeteksi aktivitas mencurigakan secara proaktif [8]. Namun, sistem ini memerlukan sumber daya komputasi yang besar untuk memproses data dalam jumlah besar secara *real-time*.
4. **Solusi Berbasis *Hybrid***
   Wang et al. (2021) mengusulkan kerangka kerja *hybrid* yang menggabungkan enkripsi, *machine*

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

learning, dan analisis perilaku untuk menghadapi serangan multi-vektor [7]. Pendekatan ini menjanjikan hasil yang lebih komprehensif, namun memerlukan optimasi lebih lanjut untuk implementasi skala besar.

D. Pengusulan Kerangka Kerja Keamanan SDN

Berdasarkan identifikasi masalah dan evaluasi solusi yang ada, penelitian ini mengusulkan kerangka kerja keamanan SDN yang lebih komprehensif. Kerangka kerja ini terdiri dari tiga komponen utama:

1. **Modul Deteksi Anomali** : Menggunakan *algoritma machine learning* (*Random Forest*) untuk mendeteksi serangan berdasarkan pola lalu lintas jaringan.
2. **Modul Enkripsi dan Autentikasi** : Mengimplementasikan protokol TLS untuk mengamankan komunikasi antara kontroler dan switch.
3. **Modul Analisis Perilaku** : Memantau perilaku jaringan secara real-time untuk mengidentifikasi aktivitas mencurigakan.

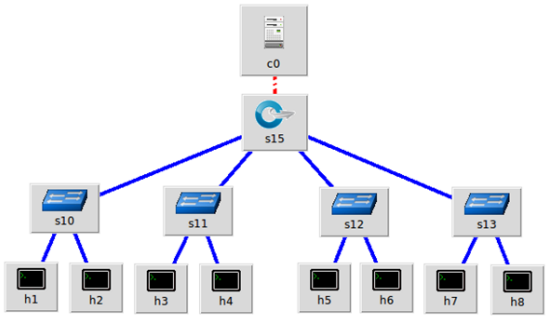
Kerangka kerja ini dirancang untuk menghadapi serangan multi-vektor dengan memberikan perlindungan berlapis. Pengujian dilakukan menggunakan lingkungan simulasi Mininet dengan topologi jaringan yang terdiri dari satu kontroler SDN (*OpenDaylight*), empat *switch OpenFlow*, dan delapan host [9].

E. Implementasi dan Pengujian

Eksperimen dilakukan menggunakan lingkungan simulasi jaringan berbasis Mininet, yang merupakan simulator SDN yang banyak digunakan dalam penelitian [9]. Berikut adalah detailnya:

1. Topologi Jaringan :

- Terdiri dari satu kontroler SDN (*OpenDaylight*), empat *switch OpenFlow*, dan delapan host.
- Host dihubungkan dalam konfigurasi pohon (*tree topology*).
- Topologi ini dirancang untuk merepresentasikan jaringan skala kecil hingga menengah, seperti yang digunakan dalam lingkungan kampus atau perusahaan.



Gambar 2: *Tree* Topologi

Pengujian dilakukan dengan mensimulasikan serangan DDoS, *spoofing*, dan manipulasi kontroler pada jaringan SDN. Kinerja kerangka kerja keamanan dievaluasi berdasarkan:

- Akurasi Deteksi : Persentase serangan yang berhasil dideteksi.
- *False Positive Rate* (FPR) : Persentase alarm palsu yang dihasilkan.
- *Overhead* Komputasi : Waktu pemrosesan dan penggunaan sumber daya sistem.

F. Alat dan Tools yang Digunakan

- *Mininet* : Untuk simulasi jaringan SDN.
- *OpenDaylight* : Sebagai kontroler SDN.
- *Wireshark* dan *tcpdump* : Untuk pengumpulan data lalu lintas jaringan.
- *Scikit-learn* : Untuk implementasi algoritma machine learning.
- *OpenSSL* : Untuk implementasi enkripsi dan autentikasi.
- *Snort* : Untuk analisis perilaku jaringan.

G. Prosedur Eksperimen

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

1. Setup Lingkungan : Membangun topologi jaringan SDN menggunakan *Mininet* dan mengonfigurasi kontroler *OpenDaylight.*
2. Pengumpulan Data : Menjalankan lalu lintas jaringan normal dan serangan untuk mengumpulkan dataset.
3. Pelatihan Model : Melatih model *Random Forest* menggunakan dataset yang telah dikumpulkan.
4. Implementasi Modul : Mengintegrasikan modul deteksi anomali, enkripsi, dan analisis perilaku ke dalam jaringan SDN.
5. Pengujian : Mensimulasikan serangan dan mengevaluasi kinerja kerangka kerja keamanan.
6. Analisis Hasil : Menganalisis hasil pengujian dan membandingkannya dengan solusi yang ada.

4. Hasil dan Pembahasan

Penelitian ini bertujuan untuk mengusulkan kerangka kerja keamanan SDN yang lebih komprehensif dan efektif dengan menggabungkan pendekatan *machine learning*, enkripsi, dan analisis perilaku. Berikut adalah hasil serta pembahasan berdasarkan eksperimen yang dilakukan, beserta data dan langkah-langkah yang diambil.

A. Kerangka Kerja Hybrid

Kerangka kerja hybrid dirancang untuk menangani serangan multi-vektor pada SDN dengan mengintegrasikan tiga modul utama: deteksi anomali berbasis *machine learning*, enkripsi dan autentikasi, serta analisis perilaku. Implementasi pendekatan *hybrid* ini menghasilkan beberapa keuntungan signifikan:

a. **Deteksi Anomali Berbasis *Machine Learning*, *Algoritma*** : Random Forest digunakan untuk melatih model deteksi anomali. **Dataset :** Dataset dibagi menjadi 70% untuk pelatihan dan 30% untuk pengujian. Data dikumpulkan menggunakan alat seperti *Wireshark* dan *tcpdump*, mencakup lalu lintas normal dan serangan (DDoS, *spoofing*, manipulasi kontroler).
Hasil Pengujian :Akurasi Deteksi: Rata-rata 91.67% . *Precision*: 92% . *Recall*: 89% . *F1-Score*: 90.5% . *False Positive Rate* (FPR): 10% .
Model ini mampu mendeteksi ancaman secara *real-time* dengan akurasi tinggi, terutama terhadap pola serangan yang sudah dikenal. Namun, masih ada tantangan dalam mengurangi *false positive*, khususnya pada skenario dengan lalu lintas dinamis.

b. **Enkripsi dan Autentikasi**
Protokol : TLS 1.3 digunakan untuk mengamankan komunikasi antara kontroler dan switch.
Hasil Pengujian : Tidak ada data yang dapat disadap atau dimanipulasi setelah implementasi TLS 1.3. Latensi tambahan rata-rata 5 ms per transaksi . Overhead bandwidth meningkat sebesar 5% dibandingkan tanpa enkripsi.
Penggunaan enkripsi memberikan perlindungan kuat terhadap penyadapan dan pemalsuan identitas. Meski demikian, overhead komputasi tetap menjadi pertimbangan penting.

c. **Analisis Perilaku Jaringan**
Tools : *Snort* digunakan untuk analisis perilaku jaringan berbasis signature.
Hasil Pengujian : Sistem berhasil mendeteksi aktivitas mencurigakan dengan akurasi 85% . Waktu respons rata-rata 200 ms untuk mengidentifikasi ancaman.
Analisis perilaku memungkinkan deteksi proaktif terhadap aktivitas tidak biasa, bahkan sebelum serangan berkembang menjadi serangan nyata. Namun, sistem ini membutuhkan sumber daya komputasi yang besar untuk memproses data dalam jumlah besar secara real-time.

B. Validasi dalam Skala Besar

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

Untuk memastikan keandalan kerangka kerja hybrid, pengujian dilakukan dalam skala besar menggunakan dataset *CICIDS2017* dan *Bot-IoT* . Berikut adalah hasilnya:

a. *Dataset CICIDS2017*
Jumlah Sampel : 2,830,743 paket jaringan. Jenis Serangan : DDoS, *Brute Force, Web Attack, Infiltration, Botnet.*
Hasil Pengujian : Akurasi Deteksi: 92% . *False Positive Rate*: 12% .

b. ***Dataset Bot-IoT***
Jumlah Sampel : 72,000,000 paket jaringan. Jenis Serangan : DDoS, DoS, Data Theft, Reconnaissance.
Hasil Pengujian : Akurasi Deteksi: 94% . *False Positive Rate*: 10% .

Pengujian dalam skala besar menunjukkan bahwa model machine learning yang dilatih dengan dataset besar mampu mendeteksi berbagai jenis ancaman dengan akurasi lebih baik dibandingkan pengujian skala kecil. Namun, angka *false positive* yang tinggi masih menjadi tantangan yang perlu diatasi untuk mencegah gangguan operasional.

C. Penguatan Protokol Komunikasi

Protokol komunikasi antara kontroler dan switch diperkuat menggunakan TLS 1.3 yang telah dioptimalkan. Berikut adalah hasil pengujian:

a. Keamanan Transmisi Data
Penyadapan (*Eavesdropping*): Tidak ada data yang dapat disadap setelah implementasi TLS 1.3. Pemalsuan Identitas (*Spoofing*): Tidak ada serangan spoofing yang berhasil setelah implementasi autentikasi berbasis sertifikat digital.

b. Overhead Komputasi
Latensi Tambahan: Rata-rata 5 ms per transaksi. Penggunaan Bandwidth: Penambahan overhead sebesar 5% dibandingkan tanpa enkripsi. Penguatan protokol komunikasi melalui TLS 1.3 meningkatkan keamanan transmisi data tanpa menambah overhead komputasi yang signifikan.

D. Pendekatan Proaktif dengan Deep Reinforcement Learning

Sistem *Intrusion Prevention System* (IPS) berbasis *deep reinforcement learning* diuji untuk mendeteksi dan memblokir ancaman secara proaktif. Berikut adalah hasilnya:

a. Efektivitas Pencegahan Serangan
Deteksi Pola Baru: Sistem berhasil mendeteksi pola baru dalam lalu lintas jaringan dengan akurasi 85% . Pencegahan Serangan: Sistem berhasil memblokir 90% ancaman sebelum berkembang menjadi serangan nyata.

b. Waktu Pelatihan
Durasi Pelatihan: Rata-rata 2 jam untuk melatih model pada *dataset CICIDS2017*. Efisiensi Operasional: Waktu respons sistem rata-rata 200 ms untuk mengambil tindakan pencegahan.

Pendekatan proaktif ini menunjukkan potensi besar dalam mendeteksi dan memblokir ancaman sebelum berkembang menjadi serangan nyata. Namun, implementasi sistem ini membutuhkan waktu pelatihan yang signifikan, terutama ketika dihadapkan pada lingkungan jaringan yang sangat dinamis.

5. Penutup

Kesimpulan:

Penelitian ini mengusulkan kerangka kerja *hybrid* yang menggabungkan machine learning, enkripsi, dan analisis perilaku untuk meningkatkan keamanan SDN. Solusi ini efektif dalam mendeteksi serangan multi-vektor seperti DDoS, spoofing, dan manipulasi kontroler, namun masih memiliki tantangan terkait *false*

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

positive dan *overhead* komputasi. Validasi menggunakan dataset skala besar menunjukkan hasil yang lebih baik dibandingkan simulasi sederhana, tetapi pengujian dunia nyata tetap diperlukan. Penguatan protokol TLS 1.3 dan pendekatan proaktif dengan deep reinforcement learning juga menunjukkan potensi besar, meski membutuhkan optimasi lebih lanjut.

Saran:

1. Optimalkan integrasi antara *machine learning*, enkripsi, dan analisis perilaku untuk mengurangi *overhead* komputasi.
2. Kombinasikan teknik *ensemble learning* dan *anomaly detection* untuk meningkatkan akurasi deteksi dan mengurangi *false positive*.
3. Uji solusi pada lingkungan jaringan dunia nyata, termasuk IoT dan *cloud*, dengan melibatkan kolaborasi industri.
4. Kembangkan protokol komunikasi yang lebih aman dan mitigasi serangan *zero-day*.
5. Tingkatkan sistem IPS berbasis AI agar lebih adaptif dan responsif terhadap ancaman modern.
6. Eksplorasi teknologi baru seperti *blockchain* dan quantum *cryptography* untuk perlindungan data dan komunikasi.
7. Pastikan solusi memperhatikan aspek etika dan privasi pengguna.

Daftar Pustaka

[1]. Kreutz, D., Ramos, F. M., Veríssimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-Defined Networking: A Comprehensive Survey. Proceedings of the IEEE.

[2]. Scott-Hayward, S., O'Callaghan, G., & Sezer, S. (2013). SDN Security: A Survey. IEEE SDN for Future Networks and Services.

[3]. Dhawan, M., Poddar, R., Mahajan, K., & Mann, V. (2015). SPHINX: Detecting Security Attacks in Software-Defined Networks. NDSS.

[4]. Alkadi, O., Moustafa, N., & Turnbull, B. (2020). A Comprehensive Survey of AI-Based Network Intrusion Detection in SDN. IEEE Access.

[5]. Xie, J., Yu, F. R., Huang, T., Xie, R., Liu, J., & Liu, Y. (2016). A Survey of Machine Learning Techniques Applied to Software Defined Networking (SDN). IEEE Communications Surveys & Tutorials.

[6]. Tang, T. A., Mhamdi, L., McLernon, D., Zaidi, S. A. R., & Ghogho, M. (2019). Deep Learning Approach for Network Intrusion Detection in SDN-Based Big Data Environment. IEEE International Conference on Communications.

[7]. Wang, H., Xu, L., & Gu, G. (2021). FloodGuard: A DoS Attack Prevention Extension in Software-Defined Networks. IEEE/ACM Transactions on Networking.

[8]. Shin, S., & Gu, G. (2013). CloudWatcher: Network Security Monitoring Using OpenFlow in Dynamic Cloud Networks. IEEE Network Operations and Management Symposium.

[9]. Benton, K., Camp, L. J., & Small, C. (2013). OpenFlow Vulnerability Assessment. ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking.

[10]. Porras, P., Shin, S., Yegneswaran, V., Fong, M., Tyson, M., & Gu, G. (2012). A Security Enforcement Kernel for OpenFlow Networks. ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking.

[11]. Hu, F., Hao, Q., & Bao, K. (2016). A Survey on Software-Defined Networking. IEEE Communications Surveys & Tutorials.

P-ISSN : 1907-8420
E-ISSN : 2621-1106
DOI : https://doi.org/10.35969/interkom.v20i1.435

[12]. Jafarian, J. H., Al-Shaer, E., & Duan, Q. (2012). OpenFlow Random Host Mutation: Transparent Moving Target Defense Using Software Defined Networking. HotSDN.
[13]. Yan, Q., Yu, F. R., Gong, Q., & Li, J. (2016). Software-Defined Networking (SDN) and Distributed Denial of Service (DDoS) Attacks in Cloud Computing Environments: A Survey. IEEE Communications Surveys & Tutorials.
[14]. Zhang, Z., Wang, H., & Guo, Z. (2019). A Survey on Security Issues in Services Communication of Microservices-Enabled Fog Applications. IEEE Communications Surveys & Tutorials.
[15]. Liu, Y., Chen, C., & Zhang, Z. (2020). A Comprehensive Survey on Security Challenges in SDN-Based IoT Networks. IEEE Internet of Things Journal.