Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI:10.25126/JTIIK.938188
Vol.11, No.3, Juni 2024, hlm. 461-474 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

DETEKSI DINI GANGGUAN JARINGAN *DISTRIBUTED DENIAL OF SERVICE* (*DDOS*) MENGGUNAKAN METODE SHANNON *ENTROPY* PADA SOFTWARE DEFINED NETWORK *(SDN)*

Achmad Solichin*[1], Ludi Nugroho[2]

[1,2] Universitas Budi Luhur, Jakarta
Email:[1]achmad.solichin@budiluhur.ac.id, [2]ludinugroho@gmail.com
[*]Penulis Korespondensi

(Naskah masuk: 20 Desember 2023, diterima untuk diterbitkan: 06 Mei 2024)

Abstrak

Software Defined Networking (SDN) adalah arsitektur jaringan baru yang memisahkan antara *control* dan *data plane*. Aspek keamanan utama dalam *control plane* salah satunya adalah serangan *DoS* dan *DDoS*. Serangan *DDoS* mengakibatkan terjadinya penurunan performa jaringan yang berjalan sangat lambat. Serangan *DDoS* dilakukan dengan menyusupi dan membanjiri *bandwidth* ke sumber daya target, sehingga dapat menyebabkan penolakan layanan bagi pengguna yang mengaksesnya. Tak hanya itu, serangan *DDoS* menyebabkan penurunan sumber daya jaringan seperti kapasitas memory dan CPU. Akibatnya kerusakan signifikan pada sistem yang menjadi korban serangan dapat mengalami kerugian, baik secara finansial, reputasi bahkan kehilangan pelanggan yang membutuhkan layanan tersebut. Mencegah serangan *DDoS* diperlukan suatu tindakan pencegahan yaitu dengan deteksi dini serangan *DDoS* untuk mengurangi dampak serangan dan memulihkan sistem dengan lebih cepat. Deteksi dini yang disebabkan oleh *DDoS* pada jaringan *SDN* dilakukan melalui pendekatan metrik *entropy* berbasis teori informasi. Penelitian ini memfokuskan pendeteksian dini pada serangan *DDoS* di dalam lingkungan *SDN* melalui metode Shannon *Entropy* dengan mendeteksi lalu lintas atau trafik normal dan *DDoS*. Penelitian ini menggunakan *dataset* publik dari *InSDN* yang diterbitkan pada tahun 2020 untuk menentukan nilai ambang batas lalu lintas normal dan *DDoS*. Hasilnya, penelitian ini berhasil mendeteksi dini lalu lintas normal dan lalu lintas serangan *DDoS* dengan nilai *entropy* sesuai ambang batas, dengan nilai akurasi 100%, presisi 100% dan recall 100% yang dihitung menggunakan *confusion matrix*. Deteksi dini menampilkan akurasi dan performa yang dapat berkontribusi banyak dalam menunjang tingkat keamanan melalui pencegahan tahap awal, sehingga hasilnya dapat meningkatkan keamanan dan efektifitas pada lingkungan *SDN*.

Kata kunci: deteksi dini, *Software Defined Networking*, teori informasi, *distributed denial of service*, *dataset*, lalu lintas.

EARLY DETECTION OF DISTRIBUTED DENIAL OF SERVICE (DDOS) NETWORK INTERFERENCE USING SHANNON ENTROPY METHOD IN SOFTWARE DEFINED NETWORK (SDN)

Abstract

Software Defined Networking (SDN) is a new network architecture that separates the control and data planes. One of the main security aspects in the control plane is DoS and DDoS attacks. DDoS attacks result in a decrease in network performance which is very slow. DDoS attacks are carried out by compromising and flooding the bandwidth to the target resource, which can cause a denial of service for users who access it. Not only that, DDoS attacks cause a decrease in network resources such as memory and CPU capacity. As a result, significant damage to the system, victims of attacks can suffer losses, both financially, reputationally and even lose customers who need the service. Preventing DDoS attacks requires preventive measures, namely early detection of DDoS attacks to reduce the impact of attacks and restore the system more quickly. Early detection caused by DDoS on SDN networks is carried out using an information theory-based entropy metric approach. This research focuses on early detection of DDoS attacks in the SDN environment using the Shannon Entropy method by detecting normal and DDoS traffic. This research uses a public dataset from InSDN published in 2020 to determine the threshold values for normal traffic and DDoS. As a result, this research succeeded in early detection of normal traffic and DDoS attack traffic with entropy values according to the threshold, with values of 100% accuracy, 100% precision and 100% recall calculated using a confusion matrix. Early detection displays accuracy and performance that can

contribute greatly to supporting security levels through early stage prevention, so that the results can increase security and effectiveness in the SDN environment.

Keywords: early *detection, Software Defined Networking, information theory, distributed denial of service, dataset, traffic.*

1. PENDAHULUAN

Software Defined Networking (SDN) adalah arsitektur jaringan baru yang memisahkan antara *control* dan *data plane*. *SDN* dikembangkan untuk mengurangi kompleksitas jaringan dan mengelola seluruh jaringan secara terpusat melalui *control plane*. *Control plane* dalam *SDN* lebih mudah untuk dikelola dibandingkan dengan arsitektur jaringan secara konvensional, yang dimana *control plane* dan *data plane* terdapat di setiap perangkat jaringan seperti *router* maupun *switch*. Komunikasi antara *control plane* dan *data plane* dihubungkan dengan *application programming Interfaces (API)*, sehingga *control plane* dapat melakukan kontrol langsung pada keseluruhan perangkat *data plane*. Melalui *control plane* terpusat, *SDN* dapat mencegah pelanggaran keamanan, akan tetapi dapat juga membawa ancaman dan kerentanan baru, dan *control plane* dapat menjadi satu pusat titik kegagalan dan menjadi target serangan yang membawa *malicious activities* di dalam jaringan *SDN* yang dapat menyebabkan *control plane* menjadi tidak dapat diakses. Aspek keamanan utama dalam *control plane* salah satunya adalah serangan DoS dan *DDoS* (Singh and Behal, 2020).

Serangan *distributed denial of service* (*DDoS*) dilakukan dengan menyusupi dan membanjiri bandwidth ke sumber daya target seperti server, situs web dan sumber daya lainnya, sehingga dapat menyebabkan penolakan layanan bagi pengguna yang mengaksesnya. Akibatnya kerusakan signifikan pada sistem yang menjadi korban serangan dapat mengalami kerugian, baik secara finansial, reputasi bahkan dapat kehilangan pelanggan yang membutuhkan layanan tersebut. Dengan deteksi dini serangan *DDoS*, dapat dilakukan suatu tindakan pencegahan untuk mengurangi dampak serangan dan memulihkan sistem dengan lebih cepat, sebelum menyebabkan kerugian menjadi lebih parah. Misalnya dengan mengalihkan lalu lintas yang tidak perlu atau memblokir lalu lintas yang terindikasi mencurigakan. Serangan *DDoS* biasanya ditandai oleh beberapa perilaku seperti performa jaringan yang berjalan sangat lambat dalam membuka file atau mengakses sebuah situs, kenaikan lalu lintas jaringan yang sangat mendadak, jumlah lalu lintas yang mencurigakan yang berasal dari satu alamat IP atau rentang IP Address, hingga terbatasnya sumber daya jaringan termasuk kapasitas memori dan CPU, sehingga mengganggu layanan yang tersedia oleh sistem.

Control plane dapat dicegah dari serangan *distributed denial of service* (*DDoS*) melalui deteksi dini pada tahap awal. Perangkat lunak pengontrol dapat dijalankan di laptop atau server yang kuat, dan istilah "dini" tergantung pada toleransi perangkat dan properti lalu lintas. Namun, jika deteksi terjadi pada beberapa ratus paket pertama, mitigasi dapat diterapkan sebelum pengontrol benar-benar dibanjiri dengan sejumlah besar paket berbahaya (Mousavi and St-Hilaire, 2015).

Deteksi dini perlu dilakukan pada tahap awal sebelum pengontrol bermasalah. Deteksi tersebut dilakukan dengan mendeteksi lalu lintas atau trafik normal dan *DDoS*. Jika lalu lintas transmisi lebih rendah dari ambang batas, itu berarti lingkungan jaringan dalam keadaan normal, tidak ada beban berat atau serangan *DDoS* terjadi. Sedangkan jika lalu lintas lebih tinggi dari ambang batas, itu mungkin mewakili kelainan di lingkungan. Beban yang berlebihan dapat berarti bahwa lingkungan jaringan sedang diserang oleh *DDoS* (Hong, Lee and Lee, 2019). Untuk penetapan ambang batas dapat dilakukan dengan menjalankan *dataset* untuk mendapatkan nilai ambang batas yang sesuai dan merupakan kelipatan dari standar deviasi dari nilai *entropy* (Mousavi and St-Hilaire, 2015).

Agar deteksi dini dapat dikatakan berjalan dengan baik maka kriteria yang dibutuhkan adalah metode yang cepat dan efektif, yang dapat berjalan bersama *controller*. Sehingga secara bersamaan dapat mencegah adanya proses penggunaan *power* atau *resources*, terutama saat puncak dari serangan *DDoS* terjadi (Mousavi and St-Hilaire, 2015). Selain itu kriteria lain dari deteksi dini adalah metode keakuratannya dalam melakukan deteksi dini di tahap awal (Sahoo *et al*., 2018).

Jadi, mencegah *control plane* dari serangan *DDoS* merupakan suatu hal yang penting saat ini, karena setiap layer pada *control plane* dapat menjadi target serangan dengan karakter serangan yang berbeda (Elsayed, Le-Khac and Jurcut, 2020). Serangan *DoS* dan *DDoS* adalah ancaman jaringan umum di *SDN* dan arsitektur jaringan tradisional (Naous *et al.*, 2008). Penyerang *DDoS* mengirim paket SYN untuk menghabiskan sumber daya korban seperti SYN, UDP, ICMP, dan serangan LAND (Tandon, 2020). Berdasarkan *Netscout's 14th Annual Worldwide Infrastructure Security Report* dari tahun 2008 – 2018, serangan *DDoS* terbesar dilaporkan oleh responden survei adalah 841 Gbps pada tahun 2018. Sementara yang lain melaporkan serangan *DDoS* sebesar 450 Gbps, 394 Gbps, dan 300 Gbps (Netscout Systems, 2019). Serangan *DDoS* telah di klasifikasi oleh Arbor Networks seperti serangan *DDoS* Volumetrik, serangan protokol, serangan *DDoS*

tingkat rendah, dan serangan *DDoS* berbasis kerentanan (Tandon, 2020).

Deteksi dini gangguan jaringan yang disebabkan oleh *DDoS* pada jaringan *SDN* dilakukan melalui pendekatan metrik entropi dan divergensi berbasis teori informasi yang banyak digunakan untuk deteksi serangan *DDoS*. Entropi mewakili keacakan dalam fitur jaringan, sedangkan metrik divergensi mewakili kesamaan dua distribusi probabilitas. Konsep pengukuran ketidakpastian awalnya diciptakan oleh Claude Shannon pada tahun 1948 (Shannon, 1948; Singh and Behal, 2020). Dengan menggunakan ukuran entropi, dapat dilihat bagaimana perilaku jaringan saat ini menyimpang dari perilaku jaringan normal, yang berujung pada deteksi serangan *DDoS*. Banyak rekan peneliti memberikan pendekatan pertahanan *DDoS* menggunakan metrik entropi (Singh and Behal, 2020).

Pengklasifikasi berbasis teori informasi ini dapat dilatih untuk menentukan perilaku abnormal lalu lintas jaringan dengan lebih akurat. Beberapa *classifier* yang umum digunakan berdasarkan teori informasi adalah Shannon *Entropy* (Giotis *et al.*, 2014; Wang, Jia and Ju, 2015; Mousavi and St-Hilaire, 2015; Tsai *et al*., 2017; Boite *et al.*, 2017; Sahoo, Tiwary and Sahoo, 2018; Ahalawat *et al.*, 2019; Bawany and Shamsi, 2019; Cui *et al.*, 2019; Hong, Lee and Lee, 2019; Sun *et al.*, 2019; Maddu and Rao, 2024), *ø-Entropy* (Li and Wu, 2020), Generalized *Entropy* (Sahoo *et al.*, 2018; Sahoo, Tiwary and Sahoo, 2018), Generalized Information Distance (GID) (Sahoo, Tiwary and Sahoo, 2018), KL Divergence (Sahoo, Tiwary and Sahoo, 2018), Conditional *Entropy* (Xuanyuan, Ramsurrun and Seeam, 2019) dan Joint *Entropy* (Kalkan *et al.*, 2018). Solusi berdasarkan metrik teori informasi seperti diatas, menggunakan nilai ambang batas yang telah ditentukan sebelumnya (bergantung pada perilaku jaringan dasar) untuk deteksi anomali. Karena jaringan berbasis *SDN* belum digunakan secara publik, maka untuk menentukan perilaku dasar yang benar dari jaringan berbasis *SDN* merupakan tantangan di depan komunitas penelitian (Singh and Behal, 2020).

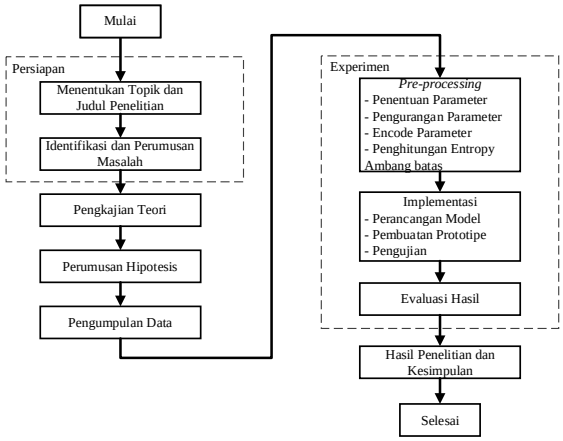
Adapun beberapa penelitian mengenai serangan *DDoS* pada jaringan *SDN* dengan menggunakan metode *machine learning* seperti penelitian berjudul Detection and defense of *DDoS* attack–based on deep learning in OpenFlow-based *SDN* (Li *et al.*, 2018), TDDAD: Time-based detection and defense scheme against *DDoS* attack on *SDN* controller (Cui *et al.*, 2018), Design of Ensemble Learning Methods for DDoS Detection in SDN Environment (Deepa, Sudar and Deepalakshmi, 2019), Efficient distributed denial-of-service attack defense in *SDN*-based cloud (Phan and Park, 2019), Machine Learning Approach Equipped with Neighbourhood Component Analysis for *DDoS* Attack Detection in Software-Defined Networking (Tonkal *et al.*, 2021), Low Rate DDoS Detection Using Weighted Federated Learning in SDN Control Plane in IoT Network (Ali *et al.*, 2023).

Penelitian ini memfokuskan pendeteksian dini pada serangan *DDoS* di dalam lingkungan *SDN* melalui metode Shannon *Entropy*. Alasan utama peneliti menggunakan metode Shannon *Entropy* untuk deteksi serangan *DDoS* adalah kemampuannya untuk mengukur keacakan dalam paket yang masuk ke jaringan. Semakin tinggi keacakan, semakin tinggi entropi, dan sebaliknya (Mousavi and St-Hilaire, 2015; Omar, Ho and Urbina, 2019) Peneliti menggunakan public *dataset* dari *InSDN* yang diterbitkan pada tahun 2020 untuk menentukan nilai ambang batas. Dalam menentukan nilai ambang batas, peneliti menentukan nilai *entropy* terendah dan tinggi, kemudian membandingkan nilai *entropy* tersebut sebagai pembanding (Mousavi and St-Hilaire, 2015). Walaupun ada beberapa public *dataset* yang telah ada sebelumnya seperti KDD'99 dan NSL-KDD, *dataset* tersebut telah usang dan memiliki ketidakcocokan pada lingkungan *SDN*, sehingga peneliti memilih *InSDN* (Elsayed, Le-Khac and Jurcut, 2020).

Pada eksperimen ini peneliti menggunakan topologi *SDN* yang disimulasikan dalam *Mininet* emulator (Lantz and O'Connor, 2015), untuk mendeteksi dini serangan *DDoS*. Parameter yang digunakan sebagai kriteria pendeteksian dini adalah *Source IP, Destination IP, Source Port, Destination Port, dan protocol*. Parameter tersebut dapat dapat digunakan sebagai dasar untuk menilai apakah lingkungan sistem secara keseluruhan tunduk pada serangan *DDoS* (Hong, Lee and Lee, 2019). Penerapan Shannon *Entropy* pada jaringan *SDN* bertujuan untuk meningkatkan akurasi dalam mendeteksi normal dan *DDoS* trafik. Penelitian ini dapat mendeteksi dini potensi serangan normal dan *DDoS* trafik dari serangan *DDoS* pada jaringan *SDN*, dan berkontribusi banyak dalam menunjang tingkat keamanan. Sehingga dalam hasilnya dapat menampilkan performa dari deteksi dini serangan dalam upaya meningkatkan keamanan dan efektifitas pada lingkungan *SDN*.

2. METODE PENELITIAN

Penelitian yang dilakukan ini termasuk ke dalam metode kuantitatif. Penelitian metode kuantitatif yaitu dengan melakukan perhitungan secara matematis terhadap populasi data atau sampel tertentu. Adapun sifat penelitian berupa penelitian secara eksperimental. Penelitian ini melakukan eksperimen melalui pengujian klasifikasi dalam deteksi dini normal dan *DDoS* trafik dari serangan *distributed denial of service (DDoS).* Adapun langkah-langkah yang dilakukan dalam penelitian ini akan ditunjukkan oleh gambar 1.



Gambar 1. Langkah-Langkah Penelitian

2.1. Menentukan Topik dan Judul Penelitian

Pada tahap ini adalah awal permulaan dari penelitian dimana peneliti menentukan topik masalah dan judul penelitian dengan membandingkan pada penelitian sebelumnya yang menggunakan metode *machine learning.* Berikut beberapa perbandingan serangan *DDoS* pada penelitian sebelumnya pada tabel 1.

Tabel 1. Tinjauan Studi

Penulis	Judul	Permasalahan	Metode
(Li et al., 2018)	Detection and defense of DDoS attack–based on deep learning in OpenFlow-based SDN	Tingkat misdiagnosis yang masih tinggi dan tingkat penilaian yang tidak tepat dalam pemrosesan lalu lintas jaringan dan deteksi serangan DDoS baru	CNN RNN LTSM
(Cui et al., 2018)	TDDAD: Time-based detection and defense scheme against DDoS attack on SDN controller	SDN dapat dengan mudah terganggu oleh serangan DDoS baru yang memicu pesan Paket Masuk yang sangat besar. Karena solusi yang ada berfokus pada pemeriksaan status jaringan saat ini dengan fitur konten untuk mendeteksi serangan, solusi tersebut mungkin dapat menyesatkan	BPNN
(Deepa, Sudar and Deepalak shmi, 2019)	Design of Ensemble Learning Methods for DDoS Detection in SDN Environment	Kesulitan dalam menganalisis jumlah paket yang besar yang mempengaruhi akurasi dalam pendeteksian dan waktu respon	KNN
(Phan and Park, 2019)	Efficient distributed denial-of-service attack defense in SDN-based cloud	Vulnerability pada kontroler SDN yang disebabkan oleh serangan DDoS mengakibatkan kontroler dan switch openflow berhenti bekerja dan kehabisan sumber daya	HIPF
(Myint Oo et al., 2019)	Advanced Support Vector Machine-(ASVM-) based detection for Distributed Denial of Service (DDoS) attack on Software Defined Networking (SDN)	Sulitnya menerapkan klasifikasi multikelas pada SVM. Dan lamanya waktu pelatihan dan pengujian yang diperlukan untuk algoritma SVM	Advanced Support Vector Machine (ASVM)
(Xuanyuan, Ramsurrun and Seeam, 2019)	Detection and mitigation of DDoS attacks using conditional *entropy* in software-defined networking	Kesulitan dalam mencapai tingkat deteksi tinggi tanpa melibatkan false negative dengan metode *entropy* yang ada	Conditional *Entropy*
(Novaes et al., 2020)	Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment	Kesulitan dalam melakukan deteksi anomaly dengan algoritma machine learning karena limitasi atribut dalam proses training	LSTM
(Tonkal et al., 2021)	Machine Learning Approach Equipped with Neighbourhood Component Analysis for DDoS Attack Detection in Software-Defined Networking	Kesulitan dalam menentukan trafik normal dan trafik serangan	Neighbourhood Component Analysis (NCA)
(Rafiee and Shirmarz, 2022)	Self-Organization Map (SOM) Algorithm for DDoS Attack Detection in	Kesulitan dalam deteksi flow DDoS pada kontroller SDN	Self-Organization Map (SOM)

Penulis	Judul	Permasalahan	Metode
	Distributed Software Defined Network (D-SDN)		
(Ali et al., 2023)	Low Rate DDoS Detection Using Weighted Federated Learning in SDN Control Plane in IoT Network	Kesulitan dalam mendeteksi dan membedakan low rate DDoS sebagai malicious traffic atau trafik yang sah	Weighted Federated Learning (WFL)

2.2. Identifikasi dan Perumusan Masalah

Pada tahap ini dilakukan identifikasi dan perumusan masalah yaitu adanya penurunan performa jaringan yang berjalan sangat lambat dalam membuka *file* atau mengakses sebuah situs *controller* jaringan *SDN*, peningkatan lalu lintas jaringan yang sangat mendadak hingga penurunan sumber daya jaringan. Berdasarkan identifikasi diatas maka dirumuskan permasalahan dalam penelitian ini adalah bagaimana penggunaan deteksi dini potensi normal dan *DDoS* trafik dari serangan *distributed denial of service* (*DDoS*) pada jaringan *SDN* untuk meningkatkan akurasi dan performa dengan menggunakan metode Shannon *Entropy*? dan berapa nilai ambang batas yang paling optimal pada model deteksi dini gangguan jaringan *distributed denial of service* (*DDoS*) dengan metode Shannon *Entropy* pada jaringan *SDN*?

2.3. Pengkajian Teori

Setelah mengetahui permasalahan yang ada, langkah selanjutnya adalah dengan mengumpulkan informasi, konsep, dan teori yang akan dijadikan landasan teori yang berasal dari buku, jurnal ilmiah dan sumber lain yang berkaitan dengan penelitian ini.

2.4. Perumusan Hipotesis

Hipotesis merupakan jawaban sementara dari sebuah penelitian, oleh karena itu suatu hipotesis hendaknya didasarkan pada teori atau asumsi. Hipotesis dari penelitian ini yaitu diduga tingkat akurasi dan performa yang baik pada metode Shannon *Entropy* dapat mendeteksi dini potensi normal dan *DDoS* trafik dari serangan *distributed denial of service* (*DDoS*) pada jaringan *SDN*, dan diduga nilai ambang batas yang optimal dapat dijadikan model untuk deteksi dini gangguan jaringan *distributed denial of service* (*DDoS*) dengan metode Shannon *Entropy* pada jaringan *SDN*.

2.5. Metode Pengumpulan Data

Metode pengumpulan data yang akan digunakan pada penelitian ini adalah capture paket normal dan *DDoS* trafik menggunakan *Scapy tools*. Dan dengan menggunakan *dataset* public *InSDN* yang diterbitkan pada tahun 2020 pada link https://aseados.ucd.ie/*datasets*/*SDN*/ sebagai penentuan nilai ambang batas. *InSDN* terdiri dari 83 atribut dan 136.743 baris yang berisi benign dan variasi serangan pada setiap kategori yang ada di dalam lingkungan *SDN*, atribut tersebut diantaranya *Flow-id*, *Src-IP*, *Src-Port*, *Dst-IP*, *Dst-Port*, *Protocol*-Type dan lain-lainnya.

2.6. *Pre-processing*

Pada tahap ini *Pre-processing* dibagi kedalam 4 bagian:

1. Penentuan Parameter

   Menentukan parameter yang akan dijadikan acuan sebagai kriteria deteksi *DDoS* pada *SDN* yaitu *Source IP*, *Destination IP*, *Source Port*, *Destination Port*, dan *protocol.*
2. Pengurangan Parameter

   Dataset InSDN memiliki 83 parameter, sehingga perlu dikurangi menjadi beberapa parameter penting.
3. Encode Parameter

   Dataset yang sudah dilakukan pengurangan parameter dan filter label dilakukan encoding atau mengubah nilai parameter menjadi nilai numerik.
4. Perhitungan *Entropy* Ambang Batas

   Dataset yang sudah diubah menjadi nilai numerik kemudian dikalkulasi perhitungan nilai *entropy*, sebagai penentuan nilai ambang batas.

2.7. Implementasi

Pada tahap ini implementasi menggunakan *OpenFlow environment* menggunakan simulasi dari *Mininet*, kemudian mengimplementasikan *POX* sebagai kontroler *SDN*. *Pox controller* memberikan kemudahan dalam implementasi dan efisiensi dalam menjalankan program *Python.* Berikut tahap-tahap dalam implementasi:

1. Membuat perancangan model algoritma deteksi trafik normal dan *DDoS* menggunakan metode Shannon *Entropy*.
2. Membuat prototipe topologi *SDN* pada Mininet yang terdiri dari Pox *control*ler, Router, OpenFlow Switch dan Hosts.
3. Membuat deteksi trafik normal dan *DDoS* pada Pox *control*ler yang sudah ditentukan nilai ambang batas dengan metode Shannon *Entropy* pada pre-processing sebelumnya.
4. Pengujian deteksi dini normal dan *DDoS* trafik serangan *DDoS* menggunakan Scapy tools.

2.8. Evaluasi Hasil

Pada proses evaluasi hasil dilakukan dengan cara menghitung nilai akurasi, presisi dan *recall* dengan menggunakan *confusion matrix* pada tabel 2.

Tabel 2. *Confusion Matrix*

Kelas	Terklarifikasi Positif	Terklarifikasi Negatif
Positif	TP (True Positive)	FN (False Negative)
Negatif	FP (False Positive)	TN (True Negative)

Berdasarkan nilai *True Negative* (TN), *False Positive* (FP), *False Negative* (FN), dan *True Positive* (TP) dapat diperoleh nilai akurasi, presisi dan *recall*. Nilai akurasi menggambarkan seberapa akurat sistem dapat mengklasifikasikan data secara benar. Dengan kata lain, nilai akurasi merupakan perbandingan antara data yang terklasifikasi benar dengan keseluruhan data. Nilai akurasi dapat diperoleh dengan Persamaan 1. Nilai presisi menggambarkan jumlah data kategori positif yang diklasifikasikan secara benar dibagi dengan total data yang diklasifikasi positif. Presisi dapat diperoleh dengan Persamaan 2. Sementara itu, *recall* menunjukkan berapa persen data kategori positif yang terklasifikasikan dengan benar oleh sistem. Nilai *recall* diperoleh dengan Persamaan 3 (Solichin, 2017).

$$Akurasi = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \; x \; 100\% \quad (1)$$

$$Presisi = \frac{\text{TP}}{\text{FP} + \text{TP}} \; x \; 100\% \quad (2)$$

$$Recall = \frac{\text{TP}}{\text{FN} + \text{TP}} \; x \; 100\% \quad (3)$$

Keterangan:
True Positive (TP) jumlah data positif yang terklarifikasi benar oleh sistem.
True Negative (TN) jumlah data negatif yang terklarifikasi benar oleh sistem.
False Positive (FP) jumlah data positif namun terklarifikasi salah oleh sistem.
False Negative (FN) jumlah data negatif namun terklarifikasi salah oleh sistem.

2.9. Hasil Penelitian dan Kesimpulan

Pada tahap ini diperoleh hasil dari Deteksi Dini Gangguan Jaringan *Distributed denial of service* (*DDoS*) Menggunakan Metode Shannon *Entropy* pada *Software Defined Network (SDN)*, kemudian dibuat kesimpulan dari hasil penelitian.

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data

Data merupakan sebuah hal penting dalam pelaksanaan penelitian, data yang didapatkan pada penelitian ini adalah data yang bersumber dari *dataset InSDN* (Elsayed, Le-Khac and Jurcut, 2020) dan capture paket dari packet generation di dalam jaringan *SDN*. *Dataset* tersebut memiliki 83 parameter dan sudah memiliki label. Berikut detil label *dataset* yang ditujukan dalam Tabel 3.

Tabel 3. Label *Dataset*

Dataset	BFA	*DDoS*	DoS	Probe	U2R
Total	295	73529	1145	61757	17
Total Keseluruhan			136743		

Dari label *dataset* yang ditujukan pada Tabel 3, penelitian ini memfokuskan pada label *DDoS* dan *DoS* sebagai acuan yang digunakan dalam menentukan nilai ambang batas. Seperti yang ditujukan pada Tabel 4.

Tabel 4. Label *Dataset* Setelah di Filter

Dataset	*DDoS*	DoS
Total	73529	1145
Total Keseluruhan	74674	

Kemudian setelah dilakukan filter, selanjutnya adalah mengurangi parameter dari total 83 parameter menjadi 6 parameter. Dari ke 6 parameter tersebut, dilakukanlah proses *encode* nilai parameter menjadi nilai numerik. Data yang diproses dalam penelitian ini adalah data yang memiliki parameter seperti *Source IP*, *Destination IP*, *Source Port*, *Destination Port*, dan *protocol*.

Pada pengumpulan data menggunakan *capture* paket dari *packet generation*, penelitian ini menggunakan *Scapy tools*. *Scapy* merupakan aplikasi *Python* yang digunakan untuk menghasilkan paket jaringan dengan membuat paket UDP dan memalsukan sumber *IP address*. *Scapy tools* dapat digunakan untuk pengujian deteksi dini normal dan *DDoS* trafik. Trafik *DDoS* yang menuju satu host memiliki rate yang lebih tinggi dibandingkan dengan trafik normal.

3.2. *Pre-processing*

3.2.1. Penentuan Parameter

Pada tahap pre-processing, dilakukan persiapan *dataset InSDN* yang kemudian diload *dataset* menggunakan *pandas library* pada program *python*. *Dataset* yang diload memiliki 83 atribut. Dan memiliki 136.743 baris yang berisi benign dan variasi serangan yang sudah terlabel, seperti *BFA*, *DDoS*, *DoS*, *Probe* dan *U2R* seperti ditujukan pada Tabel 2. Setelah dilakukan proses load *dataset*, selanjutnya dilakukan penentuan parameter dari total 83 parameter menjadi 6 parameter, yaitu *Src IP*, *Src Port*, *Dst IP*, *Dst Port*, *Protocol*, dan Label. Menurut (Hong, Lee and Lee, 2019) dasar penghitungan entropi dapat berupa *Src IP*, *Src Port*, *Dst IP*, *Dst Port*, *Protocol*, dan sebagainya.

3.2.2. Pengurangan Parameter

Setelah dilakukan penentuan parameter menjadi 6 parameter, yaitu *Src IP*, *Src Port*, *Dst IP*, *Dst Port*, *Protocol*, dan Label. Tahap selanjutnya adalah pengurangan parameter dari total 83 parameter menjadi 6 parameter dengan menggunakan *pandas*

library pada program *python* yang sama. Berikut hasil pengurangan parameter yang disajikan dengan 5 data awal dan 5 data akhir, yang ditujukan pada Tabel 5.

Tabel 5. Pengurangan Parameter *Dataset InSDN* (Sampel)

No	Src IP	Src Port	Dst IP	Dst Port	Protocol	Label
0	192.168.3.130	38694	200.175.2.130	4444	6	U2R
1	192.168.3.130	38693	200.175.2.130	4444	6	U2R
2	200.175.2.130	33747	192.168.3.130	3632	6	U2R
…	...	...	...	...	...	...
136740	200.175.2.130	44791	192.168.3.130	139	6	U2R
136741	192.168.3.130	41966	200.175.2.130	4444	6	U2R
136742	192.168.3.130	41967	200.175.2.130	4444	6	U2R

Kemudian setelah parameter dikurangi menjadi 6 parameter, *dataset* dilakukan filter label, dari yang sebelumnya memiliki 5 label menjadi 2 label yaitu *DDoS* dan *DoS*. Pada program *python*, parameter yang sudah ditentukan selanjutnya dipilih parameter "Label" yang berisi nilai "*DDoS*" dan "*DoS*" kemudian dilakukan filter. Adapun kriteria pemilihan label *DDoS* dan *DoS* adalah sebagai berikut:

1. Membatasi analisis hanya pada label *DDoS* dan DoS, dan mendapatkan informasi tentang tingkat kompleksitas atau ketidakpastian dalam distribusi serangan pada *dataset*.
2. Fokus pada ancaman *DDoS* dan DoS untuk menganalis serangan yang bertujuan mengancam ketersediaan sistem.
3. Mengidentifikasi pola dan variasi serangan *DDoS* dan DoS pada *dataset*.
4. Sebagai acuan untuk perhitungan nilai ambang batas yang bersumber dari *dataset*.

Berikut hasil filter label yang ditujukan pada Tabel 6.

Tabel 6. Filter Label *Dataset InSDN* (Sampel)

No	Src IP	Src Port	Dst IP	Dst Port	Protocol	Label
298	6.234.132.122	0	192.168.3.130	0	0	*DDoS*
299	143.97.107.22	0	192.168.3.130	0	0	*DDoS*
...	...	...	...	...	...	...
74970	200.175.2.130	57914	192.168.3.130	80	6	DoS
74971	200.175.2.130	57912	192.168.3.130	80	6	DoS

3.2.3. *Encode* Parameter

Setelah tahap pengurangan parameter dan filter label, tahap selanjutnya adalah melakukan proses *encoding* nilai parameter menjadi nilai numerik, agar nantinya dapat diproses perhitungan nilai *entropy* sebagai penentuan ambang batas. Proses encoding yang digunakan adalah dengan metode label *encoding*. Dimana metode ini mengasosiasikan setiap nilai non-numerik dengan nilai numerik unik, yaitu dengan menetapkan angka 0, 1, 2, dst., secara berurutan pada setiap nilai yang unik yang terdapat pada setiap parameter. Berikut hasil encode nilai parameter yang ditujukan pada Tabel 7 yang diurutkan berdasarkan *Src IP*.

Tabel 7. *Encoded Dataset InSDN* (Sampel)

No	Src IP	Src Port	...	Dst Port_encoded	Protocol_encoded
70132	1.1.128.28	0	...	0	0
61496	1.1.148.200	0	...	0	0
...	...	...	...	...	...
61304	99.97.150.2	0	...	0	0
58003	99.99.5.7	0	...	0	0

3.2.4. Perhitungan *Entropy* Ambang Batas

Tahap akhir dari *pre-processing* adalah perhitungan nilai *entropy* untuk menentukan nilai ambang batas. Penetapan nilai ambang batas dengan perhitungan nilai *entropy* yang diambil dari probabilitas yang muncul dari atribut *dataset* yang sudah ditentukan sebelumnya dan diproses *encode*. Perhitungan *entropy* dilakukan pada atribut *Src IP_encoded, Src Port_encoded, Dst IP_encoded, Dst Port_encoded*, dan *Protocol_encoded* yang telah diubah menjadi nilai numerik menggunakan metode label encoding, langkah-langkah perhitungan dijelaskan sebagai berikut:

1. Hitung jumlah kemunculan setiap nilai atribut untuk masing-masing atribut dalam *dataset*. Hal ini akan memberikan gambaran tentang seberapa sering setiap nilai muncul dalam *dataset*.
2. Dengan menggunakan jumlah kemunculan, probabilitas ini akan memberikan informasi tentang seberapa sering nilai tertentu muncul secara relatif terhadap total sampel.
3. Menggunakan rumus umum dari Shannon *entropy* $H(x) = -\sum_1^n p_i \, log_2 \, p_i$ untuk menghitung entropi dari setiap nilai atribut. Dalam rumus ini, probabilitas kemunculan nilai atribut menggunakan rumus $P_i = \frac{f_i}{f_t}$. Proses ini dilakukan untuk setiap nilai atribut dalam setiap atribut yang terlibat.

Dari langkah-langkah diatas menghasilkan nilai *entropy* yang ditujukan pada Tabel 8.

Tabel 8. Nilai *Entropy* Parameter

Src IP_encoded	Src Port_encoded	Dst IP_encoded	Dst Port_encoded	Protocol_encoded
16,0441	0,3501	0,0325	0,1934	0,1533

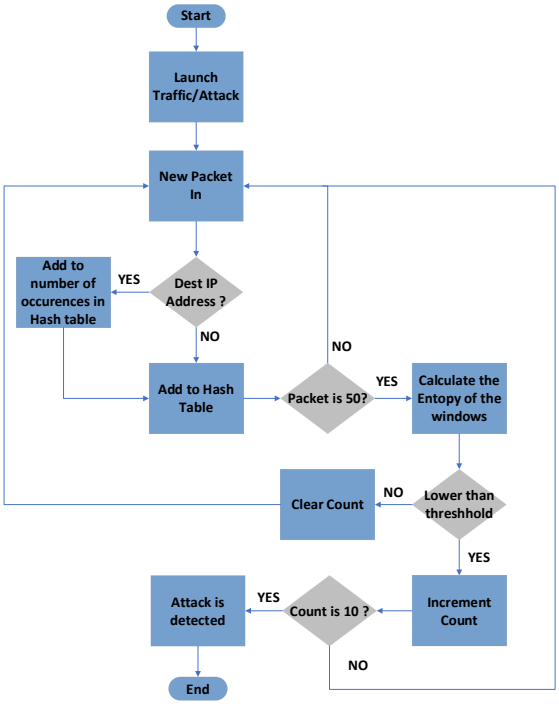
3.3. Perancangan Model

Pada perancangan model deteksi dini gangguan jaringan *distributed denial of service (DDoS)* menggunakan metode Shannon *entropy* pada *software defined network (SDN)* dilakukan pendekatan simulasi trafik yang terdiri dari trafik normal dan *DDoS*. Trafik tersebut berisikan paket *window*, setiap paket *window* yang masuk akan diuraikan untuk menentukan alamat IP tujuan, kemudian jika paket sudah mencapai sebanyak 50 paket *window* maka nilai *entropy* akan dihitung dan dibandingkan dengan nilai ambang batas. Menurut (Mousavi and St-Hilaire, 2015; Omar, Ho and Urbina, 2019; Rajan and Aravindhar, 2023) alasan menggunakan 50 paket *window* adalah sebagai berikut:

1. Terbatasnya jumlah koneksi baru yang masuk ke setiap host di jaringan. Di *SDN*, setelah koneksi dibuat, paket tidak akan melewati pengontrol kecuali ada permintaan baru.
2. Bahwa jumlah switch dan host yang terbatas dapat dihubungkan ke setiap pengontrol.
3. Jumlah komputasi yang dilakukan untuk setiap window. Dari daftar 50 paket window, nilai dapat dihitung lebih cepat dari 500 paket dan serangan di window 50 paket terdeteksi lebih awal.
4. Penggunaan CPU lebih rendah dibandingkan dengan menggunakan lebih dari 50 paket window.

Nilai ambang batas sudah ditentukan berdasarkan perhitungan nilai *entropy dataset InSDN* pada tahap *pre-processing*. Jika nilai *entropy* di atas nilai ambang batas maka trafik tersebut adalah trafik normal. Dan jika nilai *entropy* di bawah nilai ambang batas dan berlangsung selama setidaknya hingga lima (5) periode *entropy* secara berturut-turut, maka hal ini dianggap sebagai adanya serangan *DDoS*. Deteksi dalam lima periode *entropy* tersebut mengindikasikan adanya 250 paket *window* yang merupakan intrusi, sehingga dapat memberikan peringatan dini terhadap serangan dalam jaringan. Deteksi dini dalam 5 periode berturut-turut juga memiliki *false positive* terendah. Keuntungan lain adalah mencegah kemungkinan *switch* yang rusak dan memotong beberapa *host* dan mengurangi paket baru yang masuk ke dalam kontroler dan dapat membantu administrator jaringan untuk melakukan aksi pencegahan (Mousavi and St-Hilaire, 2015).

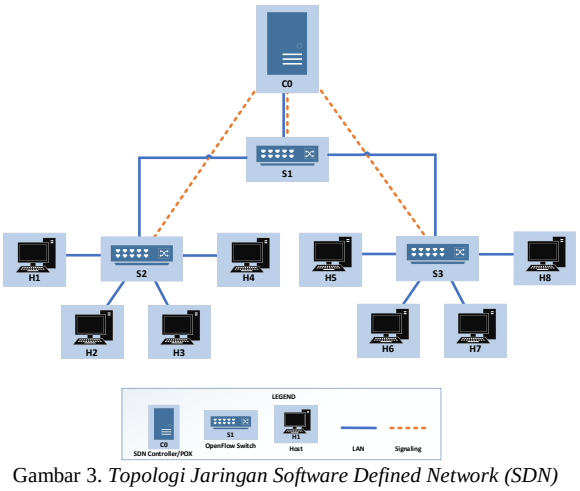
Pada penelitian ini siklus *entropy* yang digunakan adalah dengan 10 periode *entropy* dengan total 500 paket *window*, dan dengan topology jaringan yang lebih sederhana yaitu dengan 1 kontroler, 3 *switch* dan 8 *host*. Alasan peneliti menggunakan 10 periode adalah dengan adanya topology lebih sederhana, tidak membuat beban *CPU* menjadi *overload* dan dapat memaksimalkan akurasi penggunaan perhitungan nilai *entropy* dalam deteksi dini serangan *DDoS* jaringan *SDN*. Berikut adalah *flowchart* deteksi *DDoS* yang dapat digunakan untuk menentukan trafik normal dan *DDoS*, seperti ditujukan pada gambar 2.



Gambar 2. *Flowchart* Deteksi *DDoS*

3.4. Perancangan Sistem

Agar sistem dapat berjalan dengan baik dibutuhkan juga sebuah topologi jaringan *SDN* yang dijalankan di dalam emulator Mininet. Berikut topologi yang digunakan pada gambar 3.



Gambar 3. *Topologi Jaringan Software Defined Network (SDN)*

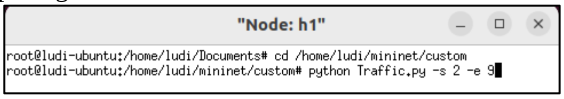
3.5. Pengujian Sistem

Pada tahap pengujian adalah proses menjalankan simulasi trafik normal dan *DDoS* yang kemudian diujikan kepada sistem untuk melakukan

deteksi dini ganguan jaringan *distributed denial of service (DDoS)* menggunakan metode Shannon *entropy*. Berikut beberapa penjelasan mengenai tahap pengujian sebagai berikut:

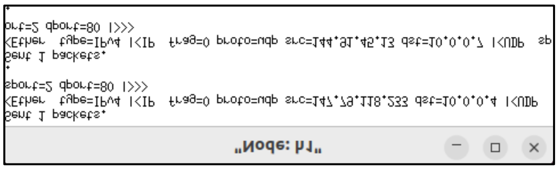
3.5.1. *Launch Traffic*

Tahap *launch traffic* dilakukan setelah pemilihan *interface* yang akan di *capture* pada *wireshark*, selanjutnya adalah menjalankan *command "python Traffic.py -s 2 -e 9"* pada *xterm* terminal di node H1 atau host1 untuk menjalankan trafik, seperti pada gambar 4.



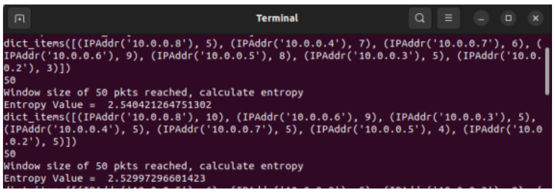
Gambar 4. Tampilan *Launch Traffic*

Setelah *command "python Traffic.py -s 2 -e 9"* dijalankan, node H1 akan mengirim paket secara acak ke semua node yang ada di dalam jaringan *SDN*. Paket dikirim sebanyak 2000 paket dengan *source ip address* dibuat secara acak, seperti pada gambar 5.



Gambar 5. Tampilan Proses *Launch Traffic*

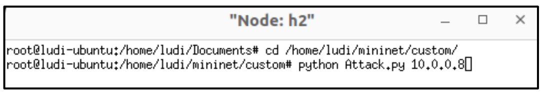
Selanjutnya, saat paket dikirimkan oleh node H1, *Pox controller* melakukan kalkulasi nilai *entropy* pada setiap 50 paket yang masuk, seperti pada gambar 6.



Gambar 6. Tampilan Proses *Calculate Entropy Launch Traffic*

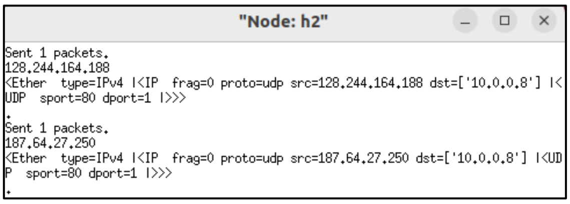
3.5.2. *Launch Attack*

Setelah *launch traffic* selesai, tahap selanjutnya adalah melakukan pemilihan *interface* yang akan di *capture* pada *wireshark*, kemudian menjalankan *command "python Attack.py 10.0.0.8"* pada *xterm* terminal di node H2 atau host2 untuk menjalankan *attack* atau serangan *DDoS* ke node H8 dengan *ip address* 10.0.0.8, seperti pada gambar 7.



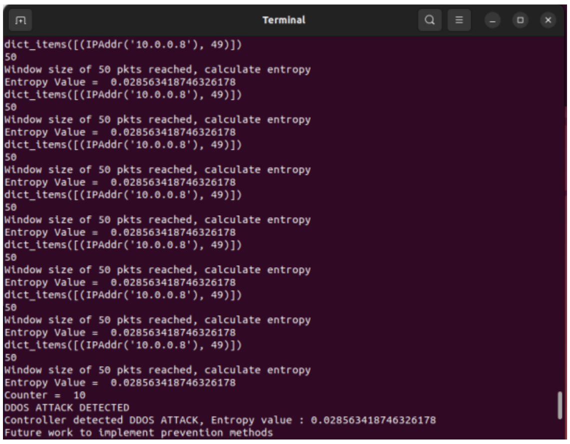
Gambar 7. Tampilan *Launch Attack*

Setelah *command "python Attack.py 10.0.0.8"* dijalankan, node H2 akan mengirim paket secara terus menerus ke node H8. Paket dikirim sebanyak 500 paket dengan *source ip address* dibuat secara acak untuk memalsukan *source ip address* aslinya, seperti pada gambar 8.



Gambar 8. Tampilan Proses *Launch Attack*

Selanjutnya, saat paket dikirimkan oleh node H2, *Pox controller* melakukan kalkulasi nilai *entropy* pada setiap 50 paket yang masuk, seperti pada gambar 9.



Gambar 9. Tampilan Proses *Calculate Entropy Launch Attack*

3.5.3. Hasil Pengujian Trafik Normal

Setelah melakukan pengujian trafik normal diatas, maka didapatkan hasil dengan sebanyak 40 pengujian. Dari setiap 50 paket yang masuk, sistem memulai menghitung nilai *entropy*, kemudian dikomparasi dengan nilai ambang batas. Nilai ambang batas yang digunakan adalah nilai ambang batas yang ditentukan oleh kalkulasi nilai *entropy* pada *pre-processing.* Pada penelitian ini ambang batas yang digunakan adalah ambang batas dari parameter *Dst IP_encoded* yaitu 0,0325 atau dibulatkan menjadi 0,033. Jika nilai *entropy* lebih besar dari nilai *entropy* ambang batas, maka trafik dinyatakan sebagai trafik normal yang ditujukan pada Tabel 9.

Tabel 9. Hasil Pengujian Trafik Normal

No	Paket	Nilai *Entropy*	Ambang Batas	Hasil (*Entropy* > Ambang Batas)
1	50	2,540421264751302	0,033	Normal
2	50	2,52997296601423	0,033	Normal
...	...	...	...	...
39	50	2,467853327389322	0,033	Normal
40	50	2,470687516390848	0,033	Normal

3.5.4. Hasil Pengujian Trafik *DDoS*

Selanjutnya setelah melakukan pengujian trafik *DDoS* dengan 2 (dua) serangan, maka didapatkan hasil dengan sebanyak 80 pengujian dengan masing-masing trafik sebanyak 40 pengujian. Dari setiap 50 paket yang masuk, sistem memulai menghitung nilai *entropy*, kemudian di komparasi dengan nilai ambang batas yang didapat dari parameter *Dst IP_encoded* yaitu 0,0325 atau dibulatkan menjadi 0,033. Jika nilai *entropy* lebih kecil dari nilai *entropy* ambang batas, sistem akan melakukan iterasi sebanyak 10 kali, dan jika dalam 10 kali iterasi tersebut nilai *entropy* masih dibawah ambang batas, maka trafik dinyatakan sebagai trafik *DDoS* atau terdeteksi serangan *DDoS* yang ditujukan pada Tabel 10.

Tabel 10. Hasil Pengujian Trafik *DDoS*

No	Paket	Nilai *Entropy*	Ambang Batas	Iterasi	Hasil (*Entropy* < Ambang Batas)
1	50	0,028563418746326178	0,033	10	*DDoS* Detected
2	50	0,028563418746326178	0,033	10	*DDoS* Detected
...	...	...	...	...	...
7	50	0,028563418746326178	0,033	10	*DDoS* Detected
8	50	0,028563418746326178	0,033	10	*DDoS* Detected

3.6. Evaluasi Hasil

Hasil dari pengujian yang telah dilakukan pada penelitian ini dengan menggunakan metode shannon *entropy* terhadap simulasi trafik normal dan *DDoS*, selanjutnya adalah dengan mengevaluasi hasil yang telah didapatkan dari pengujian dengan menggunakan *confusion matrix* yang ditujukan pada Tabel 11.

Tabel 11. Nilai *Entropy* Parameter

Prediksi Aktual	Trafik Normal	Trafik *DDoS*
Trafik Normal	40	0
Trafik *DDoS*	0	8

Sehingga dari data tabel di atas dapat dihitung nilai akurasi, presisi, dan *recall* nya sebagai berikut:

$$Akurasi = \frac{40+8}{40+8+0+0}\ x\ 100\% = 100\% \quad (1)$$

$$Presisi = \frac{40}{0+40}\ x\ 100\% = 100\% \quad (2)$$

$$Recall = \frac{40}{0+40}\ x\ 100\% = 100\% \quad (3)$$

Dari hasil evaluasi terhadap data uji didapat bahwa metode shannon *entropy* dapat melakukan pendeteksian dini dengan tingkat akurasi, presisi dan *recall* sebesar 100%.

3.7. PEMBAHASAN

Hasil penelitian diatas menunjukkan bahwa metode Shannon *Entropy* yang diterapkan untuk mendeteksi trafik normal dan *DDoS* memberikan hasil yang sangat memuaskan. Evaluasi model dengan menggunakan *confusion matrix* pada Tabel 10 menunjukkan kinerja yang sangat tinggi, dengan nilai presisi dan recall mencapai 100%. Penting untuk diperhatikan bahwa akurasi 100% menunjukkan bahwa model berhasil mendeteksi semua jenis lalu lintas yang diuji, baik reguler maupun *DDoS*. Selain itu, nilai akurasinya adalah 100% yang berarti setiap prediksi kelas trafik normal dari model, semuanya sebenarnya trafik normal.

Selain itu, nilai pemulihan yang mencapai 100% menunjukkan kemampuan model dalam mendeteksi dan memulihkan semua kejadian lalu lintas umum yang nyata. Hal ini sangat penting dalam konteks deteksi *DDoS*, yang mana akurasi dan presisi tinggi merupakan faktor penting. Secara keseluruhan, hasil penelitian ini memberikan keyakinan bahwa metode Shannon *Entropy* merupakan cara yang efektif untuk mendeteksi serangan *DDoS* pada tahap awal dan memberikan kinerja yang optimal dalam hal presisi, akurasi, dan *recall*. Oleh karena itu, model ini dapat meningkatkan keamanan jaringan secara signifikan, memungkinkan deteksi cepat terhadap potensi ancaman, dan mengoptimalkan respons terhadap serangan *DDoS*.

4. KESIMPULAN

Metode Shannon *Entropy* berhasil melakukan deteksi *DDoS* dari paket yang masuk kedalam jaringan *SDN*. Dari pengujian yang dilakukan dari paket normal sebanyak 40 pengujian diketahui nilai *entropy* diatas ambang batas, sehingga trafik tersebut dikatakan sebagai trafik normal. Sedangkan dari pengujian yang dilakukan dari paket *DDoS* sebanyak masing-masing 40 pengujian, terdeteksi nilai *entropy* dibawah ambang batas, sehingga trafik tersebut dikatakan sebagai trafik yang terdeteksi *DDoS*.

Hasil evaluasi terhadap akurasi dan performa metode Shannon *Entropy* dalam deteksi dini potensi serangan *DDoS* pada jaringan *SDN* menggunakan confusion matrix menghasilkan nilai akurasi 100%, presisi 100% dan *recall* 100%.

DAFTAR PUSTAKA

AHALAWAT, A. ET AL. 2019 '*Entropy* Based DDoS Detection and Mitigation in OpenFlow Enabled SDN', Proceedings - International Conference on Vision Towards Emerging Trends in Communication and Networking, ViTECoN 2019, pp. 1–5. Available at: https://doi.org/10.1109/ViTECoN.2019.889

9721.

ALI, M.N. ET AL. 2023 'Low Rate DDoS Detection Using Weighted Federated Learning in SDN Control Plane in IoT Network', Applied Sciences (Switzerland), 13(3). Available at: https://doi.org/10.3390/app13031431.

BAWANY, N.Z. AND SHAMSI, J.A. 2019 'SEAL: SDN based secure and agile framework for protecting smart city applications from DDoS attacks', Journal of Network and Computer Applications, 145(April), p. 102381. Available at: https://doi.org/10.1016/j.jnca.2019.06.001.

BOITE, J. ET AL. 2017 'Statesec: Stateful monitoring for DDoS protection in software defined networks', in 2017 IEEE Conference on Network Softwarization (NetSoft). IEEE, pp. 1–9. Available at: https://doi.org/10.1109/NETSOFT.2017.8004113.

CUI, J. ET AL. 2018 TDDAD: Time-based detection and defense scheme against DDoS attack on SDN controller, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer International Publishing. Available at: https://doi.org/10.1007/978-3-319-93638-3_37.

CUI, J. ET AL. 2019 'DDoS detection and defense mechanism based on cognitive-inspired computing in SDN', Future Generation Computer Systems, 97, pp. 275–283. Available at: https://doi.org/10.1016/j.future.2019.02.037.

DEEPA, V., SUDAR, K.M. AND DEEPALAKSHMI, P. 2019 'Design of Ensemble Learning Methods for DDoS Detection in SDN Environment', in Proceedings - International Conference on Vision Towards Emerging Trends in Communication and Networking, ViTECoN 2019. IEEE, pp. 1–6. Available at: https://doi.org/10.1109/ViTECoN.2019.8899682.

ELSAYED, M.S., LE-KHAC, N.A. AND JURCUT, A.D. 2020 'InSDN: A novel SDN intrusion *dataset*', IEEE Access, 8, pp. 165263–165284. Available at: https://doi.org/10.1109/ACCESS.2020.3022633.

GIOTIS, K. ET AL. 2014 'Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments', Computer Networks, 62, pp. 122–136. Available at: https://doi.org/10.1016/j.bjp.2013.10.014.

HONG, G.C., LEE, C.N. AND LEE, M.F. 2019 'Dynamic threshold for DDoS mitigation in SDN environment', 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2019, (November), pp. 1–7. Available at: https://doi.org/10.1109/APSIPAASC47483.2019.9023229.

KALKAN, K. ET AL. 2018 'JESS: Joint *Entropy*-Based DDoS Defense Scheme in SDN', IEEE Journal on Selected Areas in Communications, 36(10), pp. 2358–2372. Available at: https://doi.org/10.1109/JSAC.2018.2869997.

LANTZ, B. AND O'CONNOR, B. 2015 'A Mininet-based Virtual Testbed for Distributed SDN Development', ACM SIGCOMM Computer Communication Review, 45(4), pp. 365–366. Available at: https://doi.org/10.1145/2829988.2790030.

LI, C. ET AL. 2018 'Detection and defense of DDoS attack–based on deep learning in OpenFlow-based SDN', International Journal of Communication Systems, 31(5), p. e3497. Available at: https://doi.org/10.1002/dac.3497.

LI, R. AND WU, B. 2020 'Early detection of DDoS based on \varphi-*entropy* in SDN networks', Proceedings of 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2020, (Itnec), pp. 731–735. Available at: https://doi.org/10.1109/ITNEC48623.2020.9084885.

MADDU, M. AND RAO, Y.N. 2024 'Network intrusion detection and mitigation in SDN using deep learning models', International Journal of Information Security, 23(2), pp. 849–862. Available at: https://doi.org/10.1007/s10207-023-00771-2.

MOUSAVI, S.M. AND ST-HILAIRE, M. 2015 'Early detection of DDoS attacks against SDN controllers', 2015 International Conference on Computing, Networking and Communications, ICNC 2015, pp. 77–81. Available at: https://doi.org/10.1109/ICCNC.2015.7069319.

NAOUS, J. ET AL. 2008 'Implementing an OpenFlow switch on the NetFPGA platform', Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, ANCS '08, pp. 1–9. Available at: https://doi.org/10.1145/1477942.1477944.

NETSCOUT SYSTEMS (2019) 'NETSCOUT's 14th Annual Worldwide Infrastructure Security Report', Retrieved from Netscout, 14(SECR_005_EN-1901–WISR), p. 44. Available at: https://www.netscout.com/sites/default/files/2019-03/SECR_005_EN-1901–WISR.pdf.

OMAR, T., HO, A. AND URBINA, B. 2019 'Detection of DDoS in SDN Environment Using *Entropy*-based Detection', 2019 IEEE International Symposium on Technologies for Homeland Security, HST 2019, pp. 1–6. Available at: https://doi.org/10.1109/HST47167.2019.9032893.

PHAN, T. V. AND PARK, M. 2019 'Efficient distributed denial-of-service attack defense in sdn-based cloud', IEEE Access, 7(c), pp. 18701–18714. Available at: https://doi.org/10.1109/ACCESS.2019.2896783.

RAJAN, D.M. AND ARAVINDHAR, D.D.J. 2023 'Detection and Mitigation of DDOS Attack in SDN Environment Using Hybrid CNN-LSTM', Migration Letters, 20(S13), pp. 407–419. Available at: https://doi.org/10.59670/ml.v20is13.6472.

SAHOO, K.S. ET AL. 2018 'An early detection of low rate DDoS attack to SDN based data center networks using information distance metrics', Future Generation Computer Systems, 89, pp. 685–697. Available at: https://doi.org/10.1016/j.future.2018.07.017.

SAHOO, K.S., TIWARY, M. AND SAHOO, B. 2018 'Detection of high rate DDoS attack from flash events using information metrics in software defined networks', 2018 10th International Conference on Communication Systems and Networks, COMSNETS 2018, 2018-Janua, pp. 421–424. Available at: https://doi.org/10.1109/COMSNETS.2018.8328233.

SHANNON, C.E. 1948 'A Mathematical Theory of Communication', Bell System Technical Journal, 27(3), pp. 379–423. Available at: https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

SINGH, J. AND BEHAL, S. 2020 'Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions', Computer Science Review, 37, p. 100279. Available at: https://doi.org/10.1016/j.cosrev.2020.100279.

SOLICHIN, A. 2017 'Mengukur Kinerja Algoritma Klasifikasi dengan Confusion Matrix', Retrieved from Achmatim [Preprint]. Available at: https://achmatim.net/2017/03/19/mengukur-kinerja-algoritma-klasifikasi-dengan-confusion-matrix/.

SUN, G. ET AL. 2019 'DDoS Attacks and Flash Event Detection Based on Flow Characteristics in SDN', Proceedings of AVSS 2018 - 2018 15th IEEE International Conference on Advanced Video and Signal-Based Surveillance [Preprint]. Available at: https://doi.org/10.1109/AVSS.2018.8639103.

TANDON, R. 2020 'A Survey of Distributed Denial of Service Attacks and Defenses'. Available at: https://doi.org/https://arxiv.org/abs/2008.01345v1.

TONKAL, Ö. ET AL. 2021 'Machine learning approach equipped with neighbourhood component analysis for ddos attack detection in software-defined networking', Electronics (Switzerland), 10(11). Available at: https://doi.org/10.3390/electronics10111227.

TSAI, S.C. ET AL. 2017 'Defending cloud computing environment against the challenge of DDoS attacks based on software defined network', Smart Innovation, Systems and Technologies, 63, pp. 285–292. Available at: https://doi.org/10.1007/978-3-319-50209-0_35.

WANG, R., JIA, Z. AND JU, L. 2015 'An *entropy*-based distributed DDoS detection mechanism in software-defined networking', Proceedings - 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2015, 1, pp. 310–317. Available at: https://doi.org/10.1109/Trustcom.2015.389.

XUANYUAN, M., RAMSURRUN, V. AND SEEAM, A. 2019 'Detection and mitigation of DDoS attacks using conditional *entropy* in software-defined networking', Proceedings of the 11th International Conference on Advanced Computing, ICoAC 2019, pp. 66–71. Available at: https://doi.org/10.1109/ICoAC48765.2019.246818.