REPOSITOR, Vol. 5, No. 1, Februari 2023, Pp. 603-608
ISSN : 2714-7975
E-ISSN : 2716-1382

Low Rate DDOS Attack Detection Using KNN On SD-IOT

Achmad Irfani Nur Iman[*1], Fauzi Dwi Sumadi[2], Zamah Sari[3]
[1,2,3]Universitas Muhammadiyah Malang
achmadiman90@gmail.com[*1], fauzisumadi@umm.ac.id[2], zamahsari@umm.ac.id[3]

Abstrak

Perangkat Internet Of Things (IoT) sudah sangat berkembang dan dapat di temui dalam kehidupan sehari-hari seperti jam tangan, lampu pintar dan lain sebagainya. Untuk saat ini perangkat IoT sudah mencapai 24 miliar yang terkoneksi ke internet dan jumlahnya akan terus bertambah. Banyaknya perangkat-perangkat IoT yang terkoneksi ke internet maka banyak celah keamanan yang bisa dimanfaatkan oleh orang yang tidak bertanggung jawab untuk melakukan serangan yang berdampak luas pada jaringan. Salah satu serangan yang dapat dilakukan yaitu Low Rate Attack. Untuk menyelesaikan permasalahan tersebut banyak peneliti menciptakan paradigma baru dalam jaringan yaitu memanfaatkan kelebihan-kelebihan Software Defined Network (SDN) untuk diterapkan pada jaringan IoT. Pada penelitian ini mengusulkan metode klasifikasi deteksi low rate attack menggunakan machine learning dengan menggunakan algoritma K-Nearest Neighbors (KNN). Penelitian ini juga mengusulkan skema fitur yang baru untuk dataset dengan memanfaatkan fitur port statistic yang ada pada environment SDN. Hasil penelitian menunjukkan untuk model klasifikasi KNN yang diterapkan mendapatkan hasil yang baik yaitu 92% pada saat evaluasi model yang diterapkan pada environment SD-IoT. Disamping itu, predict loss terendah 1,6% dan predict loss tertinggi 99%, hal tersebut sangat bisa dipengaruhi oleh resource hardware yang digunakan karena sistem deteksi membutuhkan resouce hardware yang tinggi.

Kata Kunci: *IoT, Deteksi, SD-IoT, KNN, Port Statistic*

Abstract

Internet of Things (IoT) devices are highly developed and can be found in everyday life such as watches, smart lights and so on. For now, there are 24 billion IoT devices connected to the internet and the number will continue to grow. The number of IoT devices connected to the internet means there are many security holes that can be exploited by irresponsible people to carry out attacks that have a wide impact on the network. One of the attacks that can be done is Low Rate Attack. To solve these problems, many researchers have created a new paradigm in networking, which is to take advantage of the advantages of Software Defined Network (SDN) to be applied to IoT networks. This study proposes a classification method for detecting low rate attacks using machine learning using the K-Nearest Neighbors (KNN) algorithm. This study also proposes a new feature scheme for the dataset by utilizing the port statistics feature in the SDN environment. The results showed that the KNN classification model applied got good results, namely 92% when evaluating the model applied to the SD-IoT environment. On the other hand, the lowest packet loss is 1.6% and the highest packet loss is 99%, this can be greatly influenced by the hardware resources used because the detection system requires high hardware resources.

Keywords: *IoT, Detection, SD-IoT, KNN, Port Statistic*

1. Pendahuluan

Perangkat *Internet of Things* (IoT) saat ini sudah sangat berkembang dan dapat ditemui didalam kehidupan sehari-sehari seperti jam tangan pintar, lampu pintar dsb. IoT juga banyak melibatkan perangkat-perangkat dengan jumlah yang sangat besar dan saling berhubungan satu sama lainnya seperti fasilitas umum, peralatan rumah tangga, dan perangkat lain yang membutuhkan jaringan [1][2].

Untuk dapat menyelesaikan permasalahan-permasalahan yang ada pada jaringan IoT, beberapa peneliti menciptakan paradigma baru dalam jaringan yang memisahkan antara pusat kontrol dengan pengiriman data yang disebut SDN *(Software Define Network)*. SDN mampu memfasilitasi managemen data seperti data *processing, storage, transmission*, dan *acquition*

[3][4]. IoT juga memanfaatkan SDN untuk memiliki kontrol yang terpusat, abstraksi dari perangkat jaringan dan fleksibel. SD-IoT *controller* mudah untuk diprogram sesuai kebutuhan administrator jaringan dan memiliki sensor virtual yang dipisahkan dari *controller* yang mampu diatur dan dikonfigurasi secara mandiri *(autonomously)* serta sangat fleksibel [2].

Implementasi dari kontrol sistem yang terpusat menjadikan isu keamanan yang dapat dieksploitasi oleh pihak yang tidak bertanggung jawab untuk melakukan serangan yang berdampak luas pada topologi jaringan yang terhubung langsung dengan *controller*. Contoh serangan yang dapat dilakukan secara *local* ataupun global ialah DDoS *(Distribued Denial Of Service)*. DDoS diklasifikasikan menjadi dua jenis serangan yaitu HRDDoS *(High Rate Distributed Denial Of Service)* dan LRDDoS *(Low Rate Distributed Denial Of Service)*. Ada dua tipe serangan LRDDoS yaitu shrew attack dan *reduction of quality* (RoQ) [5][6]. Jenis serangan tersebut dapat memberikan dampak pada links yang terhubung langsung ke *controller* yang menyebabkan *controller* menjadi tidak stabil [5][6].

Pada penelitian sebelumnya, pada penelitian [2] menggunakan SD-IoT *Framework* untuk mendeteksi dan memitigasi serangan HRDDOS dengan menggunakan algoritma yang telah diusulkannya yang berpacu pada kesamaan nilai kosinus dalam mengukur kesamaan antara vektor yang sehubungan dengan tingkat *packet* yang masuk melalui *port input*. Algoritma yang diusulkan dibandingkan dengan dua algoritma yang lain dan memiliki hasil yang baik dalam memitigasi serangan. Pada penelitian [7][8], menggunakan pendekatan dengan skema NIDS *(Network Security and Intrusion Detection System)* yang berbasis *machine learning* dan menggunakan algoritma *random forest* untuk mendeteksi serta memitigasi serangan LRDDOS. Dengan menggunakan dataset serangan CICIDS2017 dan NSL-KDD sebagai sumber untuk menentukan metrik teori informasi memiliki hasil yang lebih baik untuk mendeteksi serangan *low rate* yaitu 96.8% sampai 99.1% untuk *flooding*. Selanjutnya pada penelitian [9] juga menggunakan pendekatan *machine learning* dan menggunakan fitur *statefull* dan *stateless* dari paket OpenFlow untuk mendeteksi serangannya. Dataset yang dikumpulkan dari *controller* dan *switch* masing-masing memiliki 204,888 dan 48,509 data, yang berisi trafik yang normal seperti MQTT, HTTP, HTTPS, pesan PING dll, Peneliti menggunakan algoritma *Support Vector Machine* (SVM) dengan RBF kernel, KNN, *Multinomial Naïve Bayes* (NB) dan *Random Forest* (RF) menggunakan *gini impurity*. Algorima SVM dan KNN memiliki *recall rate* yang terbaik yaitu 0,95.

Berdasarkan pada penelitian sebelumnya, dalam penelitian kali ini penulis mengusulkan metode deteksi serangan dengan *machine learning* dan menggunakan algoritma *K-Nearest Neighbor* pada jaringan SD-IoT serta dataset yang berbeda pada penelitian sebelumnya karena menggunakan struktur data dari *port statistic* yang dimiliki oleh OpenFlow. Diharapkan metode tersebut dapat meningkatkan akurasi dan presisi pada penelitian sebelumnya.

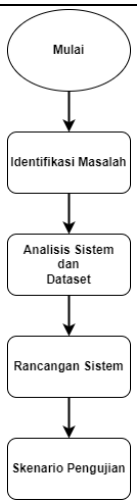
2. Metode Penelitian

2.1 Studi Literatur

Pada tahapan ini penulis akan melakukan pengumpulan semua informasi terkait *low rate attack* dan metode klasifikasi menggunakan *machine learning* khususnya dengan menggunakan algortima KNN. Agar menambah wawasan dan pemahaman tentang permasalahan yang diangkat berdasarkan referensi yang diantaranya jurnal, buku, internet serta sumber pustaka dan dokumentasi lainnya.

2.2 Alur Penelitian

Pada tahapan ini penulis memaparkan alur dari penelitian yang akan dilakukan seperti identifikasi masalah, analisis sistem dan dataset yang digunakan, rancangan sistem dan skenario pengujian.



Gambar 1 Alur penelitian

2.3 Identifikasi Masalah

Berdasarkan latar belakang penelitian yang sudah dijelaskan diatas, penelitian ini bereksperimen atau melakukan percobaan untuk mendeteksi serangan jaringan khususnya *low rate attack* dengan metode *machine learning* dan menggunakan dataset yang memanfaatkan *flow statistic* yang ada pada *environment* SDN. Dalam penelitian ini penulis akan mengimplementasikan guna untuk mengidentifikasi dan menganalisis serangan *low rate* jenis UDP pada jaringan SD-IoT dengan menggunakan metode klasifikasi *machine learning* dan menggunakan algoritma KNN. Penelitian ini juga mengusulkan skema fitur baru untuk dataset dengan memanfaatkan fitur *port statistic* yang ada pada *environment* SDN.

2.4 Analisis Sistem dan Dataset

Pada tahap ini menjelaskan tentang kebutuhan sumber daya perangkat dan perangkat keras yang digunakan, serta kebutuhan data yang akan digunakan sebagai dataset dengan memanfaatkan fitur *port statistic* untuk mengekstraksi *header packet information* IPv4 dan UDP.

2.5 Rancangan Sistem

Pada tahap ini menjelaskan mengenai langkah-langkah serangan dan deteksi serangan *low rate* berdasarkan dari model KNN. Adapun tahapannya meliputi: perancangan topologi yang diemulasikan menggunakan mininet, membangun model klasifikasi deteksi, simulasi dan pengujian sistem.

2.6 Skenario Pengujian

Pada tahap ini dilakukan skenario pengujian yang memiliki tujuan mengetahui performa deteksi dari model klasifikasi KNN yang sudah dibuat dan diimplementasikan pada *controller* SD-IoT. Model klasifikasi KNN merupakan inti dari deteksi serangan *low rate*, terdapat empat paramater hasil yang digunakan sebagai acuan, yang diantaranya : *accuracy, precision, recall* dan *F1-score*. Model klasifikasi akan diletakkan pada *controller* SDN, sehingga model tersebut akan melakukan klasifikasi pada saat paket-paket uji yang masuk ke dalam *controller*. Ada satu parameter yang menjadi tambahan yaitu *predict loss* yang digunakan untuk bahan evaluasi karena pada saat melalukan pengiriman paket uji memiliki kemungkinan ada paket yang tidak terdeteksi oleh model klasifikasi.

2.7 Evaluasi

Metode evaluasi adalah faktor kunci untuk mengetahui performa dari klasifikasi sebuah model. Dibutuhkan analisa model yang tepat dengan parameter hasil keakuratan klasifikasi pada dataset, jumlah prediksi secara akurat pada semua jumlah prediksi untuk kelas tertentu *(precission)*, jumlah prediksi pada semua jumlah data yang tersedia untuk kelas tertentu pada dataset *(recall)* dan *F1-Score*. Persamaan 1 berikut evaluasi performa dari model.

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{precission} = \frac{TP}{FP + TP} \quad (2)$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{F1} - \text{Score} = \frac{2 \text{ x precision x recall}}{\text{precision x recall}} \text{ x } 100 \quad (4)$$

3. Hasil dan Pembahasan

Pada tahap ini merupakan hasil dari penelitian yang telah dilakukan oleh penulis berdasarkan parameter uji yang sudah ditentukan.

3.1 Model Klasifikasi KNN

Pembuatan model klasifikasi KNN berdasarkan data *train* dan data *test* dilakukan perulangan sebanyak dua puluh kali untuk mendapatkan nilai k yang terbaik, hyperparameter dan sebagainya seperti *accuracy, precission, recall* dan *F1-score*. Nilai k yang terbaik terdapat pada perulangan ke dua puluh dengan hasil 0.97. Setelah nilai k sudah ditentukan maka model disimpan dalam bentuk knn.sav.

```
----------------------------------------
20
Accuracy: 0.9743961594239136
Precision: 0.9747833433410058
Recall: 0.9743975895969093
F1-Score: 0.9743909940744702
----------------------------------------
```

Gambar 2 Hasil Perulangan untuk Nilai K

3.2 Dataset

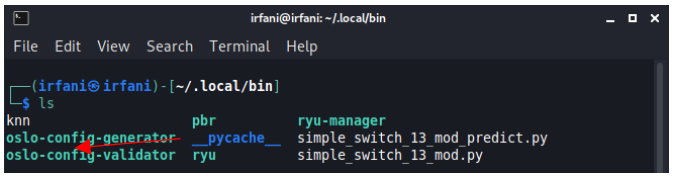
Hasil dari ektraksi file *.pcap menjadi dataset dengan memanfaatkan struktur data dari *port statistic* yang dimiliki oleh OpenFlow memiliki jumlah 160.007 untuk data *train* dan 39.995 untuk data *test*. *File* tersebut dikirimkan oleh xterm h1 ke *controller* yang menjalankan aplikasi ryu. Berikut Tabel 1 dibawah ini merupakan sample dataset yang sudah diekstraksi.

Tabel 1. Sample Dataset

……..	Src_ip	Dst_ip	Src_port	Dst_port	……..	Label
……..	247.168.216.203	10.0.0.6	65467	5683	……..	LDDoS_UDP
……..	23.3.237.197	10.0.0.6	60284	5683	……..	LDDoS_UDP
……..	10.0.0.1	10.0.0.6	54977	5683	……..	NORMAL_UDP
……..	10.0.0.1	10.0.0.6	54977	5683	……..	NORMAL_UDP
……..	162.178.178.87	10.0.0.6	64260	5683	……..	LDDoS_UDP
……..	212.33.250.92	10.0.0.6	51007	5683	……..	LDDoS_UDP
……..	55.34.5.41	10.0.0.6	51101	5683	……..	LDDoS_UDP

3.3 Proses Deteksi

Pada tahapan ini, hasil yang di dapatkan ketika proses deteksi sudah dilakukan yaitu *file* yang bernama knn. *File* tersebut nantinya akan di *generate* ke dalam *file* csv untuk dilakukan komparasi antara *file* knn.csv dan data *test* yang nantinya akan memiliki *output* indexreal.npy indexsims.npy.



Gambar 3. Hasil dari Deteksi

3.4 Kalkulasi dan Evaluasi

Pada tahapan kalkulasi, penulis memanfaatkan numpy untuk *load data* agar saat dilakukan kalkulasi memiliki waktu yang singkat. Hasil dari kalkulasi dapat dilihat pada Tabel 2 dibawah ini.

Tabel 2. Hasil Evaluasi

Parameter (%)	KNN				
	10 pps	20 pps	50 pps	100 pps	200 pps
Accuracy	50.51	92.71	13.21	58.47	62.28
Precission	50.51	92.71	13.21	58.47	62.28
Recall	50.51	92.71	13.21	58.47	62.28
F1-Score	50.51	92.71	13.21	58.47	62.28
Predict Loss	1.682	82.67	82.48	99.27	99.27

Terlihat pada tabel diatas akurasi yang tertinggi terdapat terdapat pada serangan dengan *packet rate* 20 pps yaitu 92.71. Selanjutnya, *predict loss* yang terkecil terdapat pada serangan dengan *packet rate* 10 pps yaitu 1.682. Jadi, hasil pengujian ini memperlihatkan bahwa semakin tinggi *packet rate* yang masuk ke dalam *controller* maka akan semakin besar juga *predict loss* yang tidak diklasifikasi oleh model deteksi *low rate*, hal tersebut berkaitan erat dengan perangkat keras untuk mengolah data yang dilakukan oleh *controller*.

4. Kesimpulan

Berdasarkan dari seluruh tahapan implementasi dan hasil pengujian yang telah dilakukan pada sistem deteksi serangan *low rate* menggunakan metode klasifikasi KNN pada SD-IoT, maka kesimpulan dari penelitian ini secara keseluruhan pada model klasifikasi KNN yang diterapkan memiliki hasil yang kurang baik pada evaluasi model yang diterapkan pada *environment* SD-IoT. Hasil penelitian terlihat perbedaanya pada *packet rate* 10 pps memiliki *predict loss* yang sangat kecil yaitu 1.682 serta 100 pps dan 200 pps memiliki *predict loss* yang sangat besar yaitu 99.27. Hal tersebut bisa dipengaruhi oleh perangkat keras yang digunakan terutama CPU, semakin tinggi perangkat keras yang digunakan maka akan meningkatkan performa untuk mendeteksi serangan.

5. Saran

Dari penelitian yang sudah dilakukan saran untuk penelitian berikutnya dalam bidang yang sama diharapkan adanya dataset yang dikumpulkan dari jaringan yang nyata dengan memperhatikan fitur-fitur yang mendukung pada lingkungan SD-IoT. Kebutuhan perangkat keras yang tinggi khususnya CPU untuk *controller* karena hal tersebut sangat penting, semakin tinggi perangkat keras yang digunakan maka tingkat efisiensi deteksi serangan akan meningkat. Dan terakhir diharapkan pada penelitian berikutnya bisa mengimplementasikan dengan algoritma *machine learning* yang berbeda serta pada perangkat jaringan yang nyata.

Referensi

[1] P. Kaliyar, W. Ben Jaballah, M. Conti, and C. Lal, “LiDL: Localization with early detection of sybil and wormhole attacks in IoT Networks,” *Comput. Secur.*, vol. 94, 2020, doi: 10.1016/j.cose.2020.101849.

[2] D. Yin, L. Zhang, and K. Yang, “A DDoS Attack Detection and Mitigation with Software-Defined Internet of Things Framework,” *IEEE Access*, vol. 6, no. Mcc, pp. 24694–24705, 2018, doi: 10.1109/ACCESS.2018.2831284.

[3] T. M. C. Nguyen, D. B. Hoang, and T. Dat Dang, “Toward a programmable software-defined IoT architecture for sensor service provision on demand,” *2017 27th Int. Telecommun. Networks Appl. Conf. ITNAC 2017*, vol. 2017-Janua, pp. 1–6, 2017, doi: 10.1109/ATNAC.2017.8215419.

[4] T. M. C. Nguyen, D. B. Hoang, and T. Dat Dang, “A software-defined model for IoT clusters: Enabling applications on demand,” *Int. Conf. Inf. Netw.*, vol. 2018-Janua, pp. 776–781, 2018, doi: 10.1109/ICOIN.2018.8343223.

[5] P. Bhale, S. Biswas, and S. Nandi, “LORD: LOw Rate DDoS Attack Detection and Mitigation Using Lightweight Distributed Packet Inspection Agent in IoT Ecosystem,” *Int. Symp. Adv. Networks Telecommun. Syst. ANTS*, vol. 2019-Decem, pp. 2–7, 2019, doi:

10.1109/ANTS47819.2019.9118052.

[6] F. D. Setiawan Sumadi and C. S. Kusuma Aditya, “Comparative Analysis of DDoS Detection Techniques Based on Machine Learning in OpenFlow Network,” *2020 3rd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2020*, no. 143, pp. 152–157, 2020, doi: 10.1109/ISRITI51436.2020.9315510.

[7] F. S. Dantas Silva, E. Silva, E. P. Neto, M. Lemos, A. J. Venancio Neto, and F. Esposito, “A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios,” *Sensors (Switzerland)*, vol. 20, no. 11, pp. 1–28, 2020, doi: 10.3390/s20113078.

[8] P. Krishnan, S. Duttagupta, and K. Achuthan, “VARMAN: Multi-plane security framework for software defined networks,” *Comput. Commun.*, vol. 148, no. July, pp. 215–239, 2019, doi: 10.1016/j.comcom.2019.09.014.

[9] H. Cheng, J. Liu, T. Xu, B. Ren, J. Mao, and W. Zhang, “Machine learning based low-rate DDoS attack detection for SDN enabled IoT networks,” *Int. J. Sens. Networks*, vol. 34, no. 1, pp. 56–69, 2020, doi: 10.1504/ijsnet.2020.109720.