TESIS

DETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER

Disusun oleh:

Nama : **Ibnu Mas'ud**
NIM : **19.77.1198**
Konsentrasi : **Informatics Technopreneurship**

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022

TESIS

DETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER

DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACK DETECTION ON COMPUTER NETWORKS

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister

Disusun oleh:

Nama : **Ibnu Mas'ud**
NIM : **19.77.1198**
Konsentrasi : **Informatics Technopreneurship**

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022

HALAMAN PENGESAHAN

DETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER

DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACK DETECTION ON COMPUTER NETWORKS

Dipersiapkan dan Disusun oleh

Ibnu Mas'ud

19.77.1198

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 02 Februari 2022

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 02 Februari 2022
Rektor

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

DETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN NETWORK

DISTRIBUTED DENIAL OF SERVICE (DDOS) ATTACK DETECTION ON COMPUTER NETWORKS

Dipersiapkan dan Disusun oleh

Ibnu Mas'ud

19.77.1198

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 02 Februari 2022

Pembimbing Utama

Prof. Dr. Kusrini, M.Kom.
NIK. 190302106

Anggota Tim Penguji

Dr. Arief Setyanto, S.Si., M.T.
NIK. 190302036

Alva Hendi Muhammad, S.T., M.Eng., Ph.D.
NIK. 190302493

Pembimbing Pendamping

Agung Budi Prasetio, S.T., M.Eng.
NIK. 190302347

Prof. Dr. Kusrini, M.Kom.
NIK. 190302106

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 02 Februari 2022
Direktur Program Pascasarjana

Prof. Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Ibnu Mas'ud
NIM : 19.77.1198
Konsentrasi : Informatics Technopreneurship

Menyatakan bahwa Tesis dengan judul berikut:
DETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER
Dosen Pembimbing Utama : Prof. Dr. Kusrini, M.Kom.
Dosen Pembimbing Pendamping : Agung Budi Prasetio, ST., M.Eng.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 02 Februari 2022
Yang Menyatakan,

Ibnu Mas'ud

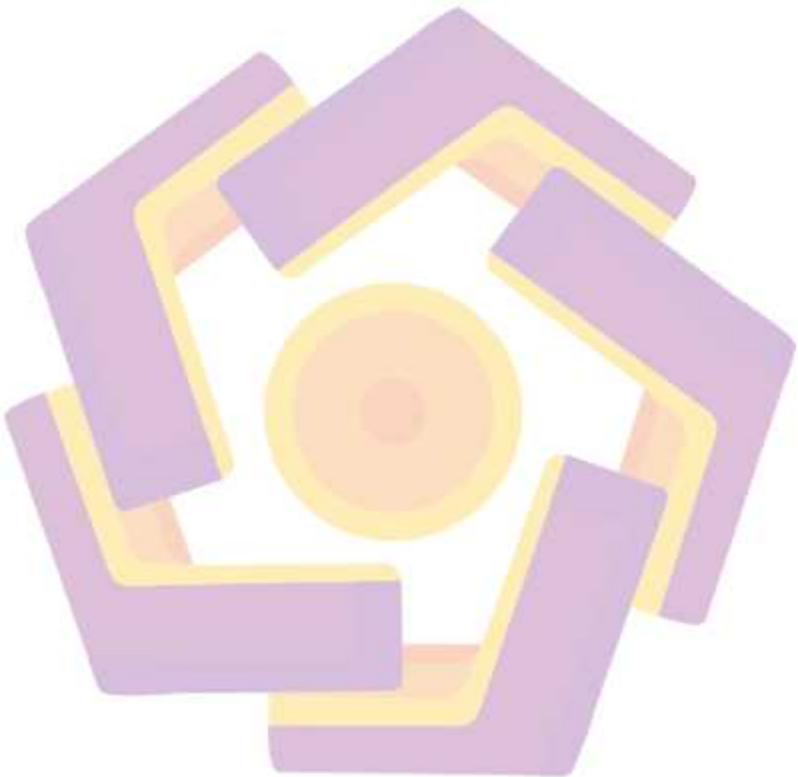
HALAMAN PERSEMBAHAN

Saya persembahkan untuk kedua orang tua saya yang sudah menyertai perjalanan thesis ini dengan doa sehingga dapat selesai dengan baik. Istriku yang telah menyemangati setiap saat, kakak-kakakku, serta rekan-rekan dari Universitas AMIKOM Yogyakarta, terima kasih semuanya.

HALAMAN MOTTO

"Ilmu pengetahuan itu bukanlah yang dihafal, melainkan yang memberi manfaat."

Imam Syafi'i



KATA PENGANTAR

Puji syukur kehadirat Allah SWT karena atas rahmat-Nya penulis dapat menyelesaikan tesis ini dengan baik. Sholawat serta salam penulis sampaikan kepada nabi besar Muhammad SAW. Alhamdulillah penulis dapat menyelesaikan tesis dengan judul "Deteksi Serangan *Distributed Denial of Service* Pada Jaringan Komputer".

Penulis menyadari bahwa penyusunan tesis ini masih jauh dari kata sempurna. Keterbatasan pengetahuan dan kemampuan yang dimiliki serta berbagai hambatan dalam proses pembuatan tesis ini. Namun berkat usaha, do'a, bimbingan, dan kerja keras dari berbagai pihak, penulis dapat menyelesaikan tesis ini dengan baik. Oleh karena itu pada kesempatan ini izinkan penulis mengucapkan terima kasih kepada:

1. Bapak Prof. M. Suyanto, M.M. selaku Rektor Universitas Amikom Yogyakarta;
2. Ibu Prof. Dr. Kusrini, M.Kom. dan Bapak Agung Budi Prasetio, S.T., M.Eng.. selaku dosen pembimbing yang telah membimbing penulis dan memberi masukan dalam penyusunan tesis ini.
3. Tim dosen penguji, Dr. Arief Setyanto, S.Si., M.T. dan Bapak Alva Hendi Muhammad, S.T., M.Eng., Ph.D. telah memberikan masukan untuk kemajuan tesis ini dan memberikan arahan untuk penelitian selanjutnya.

4. Bapak dan Ibu dosen jurusan Magister Teknik Informatika Universitas AMIKOM Yogyakarta yang telah memberikan ilmu dan pengalaman yang sangat bermanfaat.
5. Terima kasih juga kepada kedua orang tua, kakak-kakakku, istriku dan kawan-kawan yang senantiasa memberikan masukan, semangat, serta motivasi.
6. Terima kasih juga kepada Institut Teknologi Tangerang Selatan yang telah mendukung riset penulis. Semoga Institute Teknologi Tangerang Selatan semakin barokah dan sejahtera.
7. Terima kasih juga kepada Dinas Komunikasi dan Informatika Kota Tangerang Selatan yang telah mendukung riset penulis. Semoga Dinas Komunikasi dan Informatika Kota Tangerang Selatan semakin barokah dan sejahtera.

Yogyakarta, 02 Februari 2022

Penulis

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR ISTILAH

DDoS – Distributed Denial of Service

IDS – Instrution Detection System

BSSN – Badan Siber dan Sandi Negara

HTTP – Hyper Text Transfer Protocol

SDMD – Sistem deteksi dan mitigasi serangan DDoS

SVM – Support Vector Machine

INTISARI

Distributed Denial of Service atau yang lebih dikenal dengan DDoS merupakan percobaan serangan dari beberapa sistem komputer yang menargetkan sebuah server sehingga jumlah trafik menjadi terlalu tinggi sehingga server tidak dapat menangani permintaan tersebut. DDoS biasanya dilakukan dengan menggunakan beberapa sistem komputer yang dijadikan sebagai sumber serangan. Sehingga mereka menyerang satu server melalui beberapa komputer sehingga jumlah trafiknya juga bisa lebih tinggi.

Serangan DDoS seperti kemacetan lalu lintas yang mencegah pengemudi mencapai tujuan yang diinginkan tepat waktu. Menurut data, 33% bisnis di dunia telah menjadi korban serangan DDoS. DDoS sulit dilacak. Beberapa jenis serangan DDoS bisa sangat kuat dan bahkan mencapai kecepatan 1,35 Tbps. Selain itu, serangan DDoS dapat menyebabkan kerugian sebesar $40.000 per jam jika terjadi.

Pada penelitian sebelumnya Muhammad Aziz, Rusydi Umar, Faizin Ridho (2019) berdasarkan hasil analisis yang dilakukan bahwa informasi serangan yang telah terdeteksi oleh IDS berdasarkan signature perlu ditinjau akurasinya menggunakan klasifikasi dengan perhitungan statistik. . Berdasarkan analisis dan pengujian yang dilakukan dengan metode jaringan syaraf tiruan, didapatkan akurasi sebesar 95,2381%. Metode neural network dapat diterapkan di bidang forensik jaringan dalam menentukan hasil yang akurat dan membantu memperkuat bukti di persidangan. Model Naïve Bayes tampil relatif buruk secara keseluruhan dan menghasilkan skor akurasi terendah dari penelitian ini (45%) ketika dilatih dengan dataset CICDDoS2019. Untuk model yang sama, presisi adalah 66% dan recall adalah 54%, yang berarti bahwa hampir separuh waktu, model gagal untuk mengidentifikasi ancaman.

Kata kunci: IDS, DDoS, Naïve Bayes, Klasifikasi

ABSTRACT

Distributed Denial of Service or better known as DDoS is an attempted attack from several computer systems that target a server so that the amount of traffic becomes too high so that the server cannot handle the request. DDoS is usually done by using several computer systems that are used as sources of attacks. So they attack one server through several computers so that the amount of traffic can also be higher.

A DDoS attack is like a traffic jam that prevents a driver from reaching their desired destination on time. According to data, 33% of businesses in the world have fallen victim to DDoS attacks. DDoS is hard to trace. Some types of DDoS attacks can be very powerful and even reach speeds of 1.35 Tbps. Additionally, DDoS attacks can cause losses of $ 40,000 per hour if they occur.

In the previous research of Muhammad Aziz, Rusydi Umar, Faizin Ridho (2019) based on the results of the analysis carried out that the attack information that has been detected by the IDS based on signatures needs to be reviewed for accuracy using classification with statistical calculations. Based on the analysis and testing carried out with the artificial neural network method, it was found that the accuracy was 95.2381%. The neural network method can be applied in the field of network forensics in determining accurate results and helping to strengthen evidence at trial. The Naïve Bayes model performed relatively poor overall and produced the lowest accuracy score of this study (45%) when trained with the CICDDoS2019 dataset. For the same model, precision was 66% and recall was 54%, meaning that almost half the time, the model misses to identify threats.

Keyword: IDS, DDoS, Naïve Bayes, Classification

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Distributed Denial of Service atau lebih dikenal dengan nama DDoS adalah sebuah percobaan penyerangan dari beberapa sistem komputer yang menargetkan sebuah server agar jumlah traffic menjadi terlalu tinggi sampai server tidak bisa menghandle requestnya.

DDoS biasa dilakukan dengan menggunakan beberapa sistem komputer yang digunakan sebagai sumber serangan. Jadi mereka melakukan serangan ke satu server melalui beberapa komputer agar jumlah traffic juga bisa lebih tinggi. Serangan DDoS bisa dibilang seperti kemacetan lalu lintas yang menghalangi pengemudi untuk mencapai tujuan yang diinginkan dengan tepat waktu. Menurut data, 33% bisnis di dunia sudah menjadi korban serangan DDoS. DDoS memang susah dilacak. Beberapa jenis serangan DDoS memang bisa menjadi sangat kuat dan bahkan mencapai kecepatan 1.35 Tbps. Selain itu, DDoS attacks bisa menyebabkan kerugian sebesar $40,000 per jamnya jika terjadi.

Direktur Deteksi Ancaman Badan Siber dan Sandi Negara (BSSN), Dr. Sulistyo mengatakan lembaganya mendeteksi 325 juta serangan siber yang menyasar Indonesia sejak Januari hingga Oktober 2020. Dari jumlah itu, serangan paling dominan adalah jenis jebakan tautan phishing, *Distributed Denial of Services* (DDoS), serta ransomware.

Hal tersebut disampaikannya dalam webinar BSSN-Huawei Cyber Scout Hunt *"Cyber Attack"*, Senin (26 Oktober 2020). Pertama, terkait tautan phising. Menurut Sulis, serangan itu dikirimkan menggunakan email dan SMS dengan menyertakan tautan link yang tidak jelas isinya dan berpotensi menjadi kampanye penyebaran malware atau perangkat lunak berbahaya. "Ketika itu diklik yang terjadi itu membuka celah pintu ke dalam gadget kita, untuk berkomunikasi dengan penyerang, itu pertama," ujarnya. Kedua, dari 325 juta serangan itu berkaitan dengan DDoS. Serangan DDoS ini, kata Sulistyo, upaya penyerang membanjiri infrastruktur atau server target. Dampak dari serangan DDoS, kata Sulistyo, adalah lambatnya akses karena bandwidth-nya habis, serta server yang akan diakses pun juga lama sekali. Sehingga, situs yang menjadi target tidak bisa diakses karena serangan itu. "Karena website terkena DDoS maka otomatis yang ingin mendapatkan itu menjadi tidak bisa akses," ujarnya. Ketiga, ransomware. Serangan jenis ini, kata Sulistyo, cukup masif. Ransomware ini adalah turunan dari malware, yang dapat menyandera data yang ada di perangkat.

Pada penelitian sebelumnya Muhammad Aziz, Rusydi Umar, Faizin Ridho (2019) berdasarkan hasil analisis yang dilakukan bahwa informasi serangan yang telah dideteksi oleh IDS yang berbasis signatur perlu ditinjau kembali akurasinya menggunakan klasifikasi dengan perhitungan statistik. Berdasarkan analisis dan pengujian yang dilakukan dengan metode jaringan saraf tiruan, ditemukan hasil akurasi sebesar 95.2381%. Metode jaringan saraf tiruan dapat diterapkan dibidang forensik jaringan dalam menentukan hasil yang akurat dan membantu memperkuat bukti pada persidangan.

Pada penelitian sebelumnya Jodi Chris Jordan Sihombing, Dany Primanita Kartikasari, Adhitya Bhawiyuga (2019) berdasarkan pengujian yang telah dilakukan, kinerja SDMD dalam melakukan deteksi serangan DDoS sangat baik. Akurasi yang didapatkan dalam melakukan deteksi serangan DDoS adalah 96,08%, 95,66%, dan 98,76% untuk masing-masing serangan *syn flooding, udp flooding, icmp flooding*. Sistem juga dapat menanggulangi dan meminimalisir dampak dari serangan DDoS. Hal tersebut dapat dibuktikan dari jumlah paket serangan yang masuk ke *victim host* menurun ketika SDMD diaktifkan.

Pada penelitian sebelumnya Nadila Sugianti, Yayang Galuh, Salma Fatia, Khadijah Fahmi Hayati Holle (2020) berdasarkan pembahasan yang telah dijelaskan dan hasil pengujian yang telah dilakukan mengenai masalah mendeteksi serangan DDOS berbasis HTTP berdasarkan jumlah user, jumlah paket, jumlah paket/user dan panjang data yang ditangkap maka metode logika fuzzy menggunakan metode sugeno mampu digunakan sebagai pendeteksi dalam menentukan serangan DDOS berbasis HTTP dengan tingkat keakuratan mencapai 90%.

Pada penelitian sebelumnya Kurniabudi, Abdul Harris, Abdul Rahim, (2020) berdasarkan data hasil eksperimen bahwa teknik seleksi fitur *Information Gain* mampu meningkatkan performa metoda klasifikasi khususnya *Random Forest* yang memiliki performa yang lebih baik dibandingkan *Naïve Bayes, Bayes Network, OneR, AdaBoost* dan *Random Tree* dengan tangkat akurasi 99.99% pada pengujian seluruh data training dan 99.95% pada pengujian menggunakan *10-fold cross validation.* Namun disisi lain, *Random Forest* memiliki waktu yang lebih

lama untuk membangun model dan proses *training* jika dibanding *Naïve Bayes, Bayes Network, OneR, AdaBoost* dan *Random Tree.* Pada eksperimen yang dilakukan pada penelitian ini, peneliti menggunakan *Information Gain* sebagai teknik seleksi fitur terhadap dataset CICIDS-2017 dalam mendeteksi serangan DDoS. Untuk penelitian selanjutnya dapat digunakan teknik seleksi fitur lain yang mungkin dapat meningkatkan performa deteksi serangan DDoS. Selain penggunaanteknik klasifikasi yang lain perlu dipertimbangkan dalam penelitian berikutnya, khususnya yang memiliki performa yang lebih baik dengan waktu komputasi yang rendah.

Pada penelitian sebelumnya Arif Wirawan Muhaammad, Cik Feresa Mohd Foozy, Ahmad Azhari (2020) berdasarkan hasil eksperimen bahwa kombinasi tujuh fitur set data kunci terpilih yang digunakan sebagai input pengklasifikasi jaringan saraf tiruan dalam penelitian ini memberikan nilai akurasi tertinggi sebesar 97.76%.

Pada penelitian sebelumnya Lila Dini Utami, Romi Satria Wahono (2015) bahwa Naïve bayes merupakan salah satu pengklasifikasi yang mengklasifikasikan suatu teks, salah satu contoh yakni review restoran. Naïve bayes sangat sederhana dan efisien, juga sangat populer digunakan untuk klasifikasi teks dan memiliki performa yang baik pada banyak domain. Pengolahan data yang dilakukan ada 3 tahap, yakni naïve bayes, naïve bayes dan information gain, dan naïve bayes, information gain, dan adaboost. Dan ternyata, jika hanya naïve bayes saja yang digunakan, akurasi hanya mencapai 70% dan AUC=0,500. Sama halnya jika naïve bayes disertai dengan information gain, akurasi yang dicapaipun hanya 70% dan AUC=0,500, itu membuktikan bahwa information gain tidak mempengaruhi

akurasi terhadap naïve bayes. Akan tetapi, jika naïve bayes dan information gain disertai pula dengan adaboost, akurasi meningkat 29,5% menjadi 99,5% dan AUC=0,995.

Pada penelitian sebelumnya Al Riza Khadafy, Romi Satria Wahono (2015) berdasarkan hasil eksperimen dan evaluasi pada penelitian ini, secara umum dapat disimpulkan bahwa penerapan algoritma pengklasifikasi NB dapat mengurangi data noise pada dataset berukuran besar dan memiliki banyak kelas atau multi kelas sehingga akurasi klasifikasi algoritma DT dapat meningkat. Hasil akurasi yang didapat menunjukkan bahwa metode yang diusulkan DT+NB lebih unggul dari metode DT, dengan nilai akurasi untuk masing-masing dataset uji seperti Breast Cancer 96,59% (meningkat 21,06%), Diabetes 92,32% (meningkat 18,49%), Glass 87,50% (meningkat 20,68%), Iris 97,22% (meningkat 1,22%), Soybean 95,28% (meningkat 3,77%), Vote 98,98% (meningkat 2,66%), Image Segmentation 99,10% (meningkat 3,36%), dan Tic-tac-toe 93,85% (meningkat 9,30%). Perbandingan nilai akurasi dilakukan dengan uji t atau t-Test antara metode DT dengan metode yang diusulkan DT + NB untuk mendapatkan nilai perbedaan akurasi signifikan antara kedua metode tersebut. Dari hasil perbandingan didapatkan nilai P(T<=t) adalah 0,01321, ini menunjukkan bahwa nilai p lebih kecil daripada nilai alpha (0,01321 < 0,05).

Algoritma Naive Bayes merupakan sebuah metode klasifikasi menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes. Algoritma Naive Bayes memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai

Teorema Bayes. Ciri utama dari Naïve Bayes Classifier ini adalah asumsi yg sangat kuat (naïf) akan independensi dari masing-masing kondisi / kejadian. Naive Bayes Classifier bekerja sangat baik dibanding dengan model classifier lainnya, "Naïve Bayes Classifier memiliki tingkat akurasi yang lebih baik dibanding model classifier lainnya (Xhemali, Hinde, & Stone, 2009)". Keuntungan penggunan adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (training data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Karena yang diasumsikan sebagai variabel independent, maka hanya varians dari suatu variabel dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

Banyak penelitian yang telah dilakukan sebelumnya dan algoritma naive bayes merupakan model yang terbaik dibandingkan model lainnya seperti: logistic regression, neural network, random forest, decission tree, support vector machine dan k-nearest neigbor. Naive bayes merupakan algoritma klasifikasi yang sederhana dan mudah diimplementasikan sehingga algoritma ini sangat efektif apabila diuji dengan dataset yang tepat, terutama bila naïve bayes dengan seleksi fitur, maka naive bayes dapat mengurangi redudant pada data (Witten, Frank, & Hall, 2011). Algoritma naive bayes termasuk dalam supervised learning dan salah satu algoritma pembelajaran tercepat yang dapat menangani sejumlah fitur atau kelas (Lee, 2015).

Dari latar belakang diatas, maka penelitian ini mampu mengurangi kejahatan di dunia maya dengan menghasilkan persentase akurasi deteksi yang lebih baik dan mampu meningkatkan performa sistem deteksi serangan Distributed

Denial of Service dengan mengimplementasikan algoritma Naïve Bayes. Algoritma Naïve Bayes merupakan salah satu metode machine learning yang menggunakan perhitungan probabilitas. Algoritma ini memanfaatkan metode probabilitas dan statistik untuk memprediksi probabilitas di masa depan berdasarkan pengalaman dimasa sebelumnya.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, dapat dirumuskan masalahnya sebagai berikut:

a. Berapa tingkat akurasi hasil deteksi serangan DDoS jenis Syn Flood menggunakan model Gaussian dan Bernoulli algoritma naïve bayes?

1.3. Batasan Masalah

Batasan masalah pada penelitian ini antara lain:

a. Hanya difokuskan pada serangan DDoS jenis Syn Flood, tidak untuk jenis serangan lain
b. Model yang digunakan Gaussian dan Bernoulli Naïve Bayes
c. Seleksi Fitur yang digunakan Pearson Correlation
d. Protocol yang digunakan TCP dan UDP

1.4. Tujuan Penelitian

Adapun tujuan yang hendak dicapai adalah sebagai berikut

a. Untuk mengetahui tingkat akurasi hasil deteksi serangan DDoS jenis Syn Flood menggunakan model Gaussian dan Bernoulli algoritma naïve bayes

1.5. Manfaat Penelitian

Adapun manfaat yang didapat dari penelitian ini diantaranya:

a. Dapat menghasilkan tingkat akurasi hasil deteksi serangan DDoS jenis Syn Flood menggunakan model Gaussian dan Bernoulli algoritma naïve bayes

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Beberapa penelitian terdahulu yang dijadikan acuan dan tinjauan pustaka pada penelitian ini diantaranya:

Muhammad Aziz, Rusydi Umar, Faizin Ridho (2019) berdasarkan hasil analisis yang dilakukan disimpulkan bahwa informasi serangan yang telah dideteksi oleh IDS yang berbasis signatur perlu ditinjau kembali akurasinya menggunakan klasifikasi dengan perhitungan statistik. Berdasarkan analisis dan pengujian yang dilakukan dengan metode jaringan saraf tiruan, ditemukan hasil akurasi sebesar 95.2381%. Metode jaringan saraf tiruan dapat diterapkan dibidang forensik jaringan dalam menentukan hasil yang akurat dan membantu memperkuat bukti pada persidangan.

Jodi Chris Jordan Sihombing, Dany Primanita Kartikasari, Adhitya Bhawiyuga (2019) berdasarkan pengujian yang telah dilakukan, kinerja SDMD dalam melakukan deteksi serangan DDoS sangat baik. Akurasi yang didapatkan dalam melakukan deteksi serangan DDoS adalah 96,08%, 95,66%, dan 98,76% untuk masing-masing serangan *syn flooding, udp flooding, icmp flooding*. Sistem juga dapat menanggulangi dan meminimalisir dampak dari serangan DDoS. Hal tersebut dapat dibuktikan dari jumlah paket serangan yang masuk ke *victim host* menurun ketika SDMD diaktifkan.

Nadila Sugianti, Yayang Galuh, Salma Fatia, Khadijah Fahmi Hayati Holle (2020) Berdasarkan pembahasan yang telah dijelaskan dan hasil pengujian yang telah dilakukan mengenai masalah mendeteksi serangan DDOS berbasis HTTP berdasarkan jumlah user, jumlah paket, jumlah paket/user dan panjang data yang ditangkap maka metode logika fuzzy menggunakan metode sugeno mampu digunakan sebagai pendeteksi dalam menentukan serangan DDOS berbasis HTTP dengan tingkat keakuratan mencapai 90%.

Kurniabudi, Abdul Harris, Abdul Rahim, (2020) berdasarkan data hasil eksperimen maka dapat disimpulkan bahwa teknik seleksi fitur *Information Gain* mampu meningkatkan performa metoda klasifikasi khususnya *Random Forest* yang memiliki performa yang lebih baik dibandingkan *Naïve Bayes, Bayes Network, OneR, AdaBoost* dan *Random Tree* dengan tangkat akurasi 99.99% pada pengujian seluruh data training dan 99.95% pada pengujian menggunakan *10-fold cross validation*. Namun disisi lain, *Random Forest* memiliki waktu yang lebih lama untuk membangun model dan proses *training* jika dibanding *Naïve Bayes, Bayes Network, OneR, AdaBoost* dan *Random Tree*. Pada eksperimen yang dilakukan pada penelitian ini, peneliti menggunakan *Information Gain* sebagai teknik seleksi fitur terhadap dataset CICIDS-2017 dalam mendeteksi serangan DDoS. Untuk penelitian selanjutnya dapat digunakan teknik seleksi fitur lain yang mungkin dapat meningkatkan performa deteksi serangan DDoS. Selain penggunaanteknik klasifikasi yang lain perlu dipertimbangkan dalam penelitian berikutnya, khususnya yang memiliki performa yang lebih baik dengan waktu komputasi yang rendah.

Arif Wirawan Muhaammad, Cik Feresa Mohd Foozy, Ahmad Azhari (2020) berdasarkan hasil eksperimen maka dapat disimpulkan bahwa kombinasi tujuh fitur set data kunci terpilih yang digunakan sebagai input pengklasifikasi jaringan saraf tiruan dalam penelitian ini memberikan nilai akurasi tertinggi sebesar 97.76%.

Lila Dini Utami, Romi Satria Wahono (2015) bahwa Naïve bayes merupakan salah satu pengklasifikasi yang mengklasifikasikan suatu teks, salah satu contoh yakni review restoran. Naïve bayes sangat sederhana dan efisien, juga sangat populer digunakan untuk klasifikasi teks dan memiliki performa yang baik pada banyak domain. Pengolahan data yang dilakukan ada 3 tahap, yakni naïve bayes, naïve bayes dan information gain, dan naïve bayes, information gain, dan adaboost. Dan ternyata, jika hanya naïve bayes saja yang digunakan, akurasi hanya mencapai 70% dan AUC=0,500. Sama halnya jika naïve bayes disertai dengan information gain, akurasi yang dicapaipun hanya 70% dan AUC=0,500, itu membuktikan bahwa information gain tidak mempengaruhi akurasi terhadap naïve bayes. Akan tetapi, jika naïve bayes dan information gain disertai pula dengan adaboost, akurasi meningkat 29,5% menjadi 99,5% dan AUC=0,995.

Al Riza Khadafy, Romi Satria Wahono (2015) berdasarkan hasil eksperimen dan evaluasi pada penelitian ini, secara umum dapat disimpulkan bahwa penerapan algoritma pengklasifikasi NB dapat mengurangi data noise pada dataset berukuran besar dan memiliki banyak kelas atau multi kelas sehingga akurasi klasifikasi algoritma DT dapat meningkat. Hasil akurasi yang didapat menunjukkan bahwa metode yang diusulkan DT+NB lebih unggul dari metode DT, dengan nilai akurasi untuk masing-masing dataset uji seperti Breast Cancer 96,59%

(meningkat 21,06%), Diabetes 92,32% (meningkat 18,49%), Glass 87,50% (meningkat 20,68%), Iris 97,22% (meningkat 1,22%), Soybean 95,28% (meningkat 3,77%), Vote 98,98% (meningkat 2,66%), Image Segmentation 99,10% (meningkat 3,36%), dan Tic-tac-toe 93,85% (meningkat 9,30%). Perbandingan nilai akurasi dilakukan dengan uji t atau t-Test antara metode DT dengan metode yang diusulkan DT + NB untuk mendapatkan nilai perbedaan akurasi signifikan antara kedua metode tersebut. Dari hasil perbandingan didapatkan nilai P(T<=t) adalah 0,01321, ini menunjukkan bahwa nilai p lebih kecil daripada nilai alpha (0,01321 < 0,05). Dengan demikian dapat disimpulkan bahwa ada perbedaan akurasi yang signifikan antara metode DT dengan DT + NB.

2.2. Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Implemetasi Jaringan Saraf Tiruan Untuk Mendeteksi Serangan DDoS Pada Forensik Jaringan	Muhammad Aziz, Rusydi Umar, Faizin Ridho Jurnal 2019	1. Mencirikan karakteristik aktivitas jaringan dalam satu rentang waktu serta memudahkan proses pelatihan dan pengujian klasifikasi data dengan neural network 2. Memberikan akurasi tertinggi dalam mengenali trafik normal dan serangan	Berdasarkan analisis dan pengujian yang dilakukan dengan metode jaringan saraf tiruan, ditemukan hasil akurasi sebesar 95.2381%. Metode jaringan saraf tiruan dapat diterapkan dibidang forensik jaringan dalam menentukan hasil yang akurat dan membantu memperkuat bukti pada persidangan.	Berdasarkan hasil analisis yang dilakukan bahwa informasi serangan yang telah dideteksi oleh IDS yang berbasis signatur perlu ditinjau kembali akurasinya menggunakan klasifikasi dengan perhitungan statistik.	Metode jaringan saraf tiruan dalam tingkat akurasi dalam perhitungan statistik berdasarkan signatur masih belum maksimal
2	Implementasi Sistem Deteksi dan Mitigasi Serangan *Distributed Denial of Service* (DDoS) menggunakan SVM *Classifier* pada	Jodi Chris Jordan Sihombing, Dany Primanita Kartikasari,	1. Untuk mengukur waktu yang dibutuhkan oleh sistem dalam melakukan deteksi dan	1. Sistem deteksi dan mitigasi serangan DDoS (SDMD) menggunakan SVM *classifier* dapat diterapkan pada	Sistem dapat menanggulangi dan meminimalisir dampak dari serangan DDoS. Hal tersebut dapat dibuktikan dari jumlah paket	Metode SVM hanya menanggulangi dan meminimalisir dampak dari serangan DDoS

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Arsitektur *Software-Defined Network* (SDN)	Adhitya Bhawiyuga Jurnal 2019	mitigasi atau menanggulangi berbagai jenis serangan DDoS. 2. Untuk mengukur akurasi sistem dalam melakukan deteksi serangan DDoS 3. Untuk mengetahui kinerja dari modul mitigasi dalam melakukan penganggulangan apabila terjadi serangan	arsitektur SDN. SDMD melakukan deteksi serangan DDoS menggunakan *machine learning* berbasis algoritme *Support Vector Machine* (SVM). Deteksi serangan DDoS dilakukan dengan mengklasifikasikan *traffic* normal dan *traffic* serangan DDoS. Data dari *traffic* jaringan dikumpulkan dari informasi *flow entries* yang terdapat pada *flow table Openflow switch*. Beberapa fitur yang digunakan untuk merepresentasikan *traffic* jaringan yaitu standar deviasi	serangan yang masuk ke *victim host* menurun ketika SDMD diaktifkan.	

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
				paket, standar deviasi *flow byte*, jumlah IP *source* per interval, jumlah *flow entries* per interval, dan rasio *pair flow entries*. 2. Mekanisme mitigasi serangan DDoS dilakukan dengan menambahkan *flow rule* pada *switch* untuk menyaring paket yang menuju ke *victim host*. Setelah *flow rule* tersebut ditambahkan pada *flow table switch*, *switch* akan melakukan *drop* pada setiap paket yang berasal dari IP *source* penyerang, tetapi setiap paket yang berasal dari IP		

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
				source legitimate host akan diteruskan. 3. Berdasarkan pengujian yang telah dilakukan, kinerja SDMD dalam melakukan deteksi serangan DDoS sangat baik. Akurasi yang didapatkan dalam melakukan deteksi serangan DDoS adalah 96,08%, 95,66%, dan 98,76% untuk masing-masing serangan *syn flooding, udp flooding, icmp flooding*.		
3	Deteksi Serangan Distributed Denial of Services (DDOS)	Nadila Sugianti, Yayang	Membangun aplikasi yang mampu mendeteksi serangan	Pada hasil pengujian, logika fuzzy menggunakan metode	Untuk mendeteksi suatu website terkena serangan atau tidak dapat	Metode fuzzy sugeno tidak dapat dideteksi secara akurat karena pada data acuan yang

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
	Berbasis HTTP Menggunakan Metode Fuzzy Sugeno	Galuh, Salma Fatia, Khadijah Fahmi Hayati Holle Jurnal 2020	DDOS berbasis HTTP menggunakan metode fuzzy sugeno untuk pendekatan sistematis	sugeno mampu digunakan sebagai pendeteksi dalam menentukan serangan DDOS berbasis HTTP dengan tingkat keakuratan mencapai 90%.	memasukkan nilai pada kolom input yang terdapat pada Gambar 5 sesuai dengan data yang sudah di dapatkan.	dipakai memiliki konstanta yang telah ditentukan sebagai pembeda antara data normal, DDOS ringan, dan DDOS berat
4	Seleksi Fitur dengan *Information Gain* untuk meningkatkan deteksi serangan DDoS menggunakan *Random Forest*	Kurniabudi, Abdul Harris, Abdul Rahim, Jurnal, 2020	Meningkatkan performa *Random Forest* dalam mendeteksi serangan DDoS dengan seleksi fitur menggunakan teknik *Information Gain*	Berdasarkan hasil eksperimen diperoleh bahwa teknik yang diusulkan mampu meningkatkan akurasi deteksi DDoS hingga 99.99% dengan tingkat alarm palsu 0.001.	Untuk penelitian selanjutnya dapat digunakan teknik seleksi fitur lain yang mungkin dapat meningkatkan performa deteksi serangan DDoS. Selain penggunaanteknik klasifikasi yang lain perlu dipertimbangkan dalam penelitian berikutnya, khususnya yang memiliki performa yang lebih baik dengan waktu komputasi yang rendah.	Metode *Random Forest* memiliki waktu yang lebih lama untuk membangun model dan proses *training* jika dibanding *Naïve Bayes, Bayes Network, OneR, AdaBoost* dan *Random Tree.*

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	Machine Learning-Based Distributed Denial of Service Attack Detection on Intrusion Detection System Regarding to Feature Selection	Arif Wirawan Muhaammad, Cik Feresa Mohd Foozy, Ahmad Azhari, Jurnal, 2020	Untuk mendeteksi serangan DDoS dengan menggunakan teknik pembelajaran mesin untuk meningkatkan pengembangan kebijakan IDS	Kombinasi tujuh fitur set data kunci terpilih yang digunakan sebagai input pengklasifikasi jaringan saraf tiruan dalam penelitian ini memberikan nilai akurasi tertinggi sebesar 97.76%.	1. Kapan IDS mendeteksi serangan yang dimulai dengan protokol SYN 2. Defisit protokol TCP / IP yang memudahkan penyerang untuk memulai serangan DDoS	Metode jaringan saraf tiruan dengan seleksi fitur membutuhkan waktu yang lama dalam menentukan tingkat akurasi
6	Analisis Serangan DDoS Menggunakan Metode Jaringan Saraf Tiruan	M. Alfine Ridho, Molavi Arman Jurnal 2020	Untuk mengolah data traffic yang didapat dan diolah menggunakan metode fixed moving window	Berdasarkan hasil pengujian pada pengenalan serangan DDoS menggunakan metode Jaringan Saraf Tiruan (JST) yang telah dilakukan maka dapat ditarik kesimpulan, pada pengujian deteksi serangan DDoS menggunakan metode fixed moving window dan Jaringan Saraf Tiruan didapatkan	Jeda waktu untuk log traffic di percepat	Metode jaringan saraf tiruan dengan seleksi fitur membutuhkan waktu yang lama dalam menentukan tingkat akurasi

Tabel 2.1. Matriks literatur review dan posisi penelitian
Deteksi Serangan Distributed Denial of Service (DDoS) Pada Jaringan Komputer (lanjutan)

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
				accuracy sebesar 95% serta penggunaan feature extraction fixed moving window yang dikombinasikan dengan Jaringan Saraf Tiruan mampu mengenali data normal maupun data DDoS dengan baik.		

2.3. Landasan Teori

2.3.1 Denial of Service (DoS)

Serangan *Denial of Service* (DoS) merupakan jenis serangan terhadap sistem dalam jaringan internet dengan cara menghabiskan *resource* yang dimiliki oleh suatu sistem sehingga tidak dapat menjalankan fungsinya dengan benar dan secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan sistem yang diserang tersebut. Serangan DoS memanfaatkan kelemahan sistem pada keterbatasan sumber daya, baik *bandwidth*, kemampuan menyimpan memori, *server* dan kelemahan lainnya. Kebanyakan DoS menyerang bisnis kecil hingga menengah yang tidak memiliki sumber daya yang besar. Pada dasarnya tujuan penyerang hanya untuk membuat sistem lumpuh, tapi tak jarang juga ada yang kemudian meminta biaya tebusan untuk menghentikan serangan.

Dalam serangan DoS penyerang menggunakan satu komputer dan satu koneksi *internet* saja ketika meluncurkan serangan. Untuk melancarkan serangan yang berskala lebih besar, penyerang bisa menggunakan banyak komputer dan banyak koneksi *internet* yang dikontrol secara bersamaan dengan menggunakan *botnet*. *Botnet* merupakan sejumlah komputer yang terinfeksi *malware* tanpa disadari oleh penggunanya. Serangan DoS secara bersama-sama tersebut disebut *Distributed Denial of Service* (DDoS).

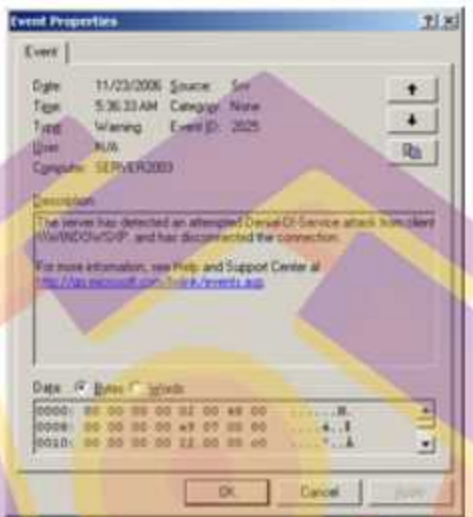
Dalam sebuah serangan *Denial of Service*, si penyerang akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut:

1. Membanjiri lalu lintas jaringan dengan jumlah data sehingga lalu lintas jaringan yang datang dari pengguna yang terdaftar dijadikan tidak dapat masuk ke dalam sistem jaringan. Teknik ini dinamakan sebagai *traffic flooding*.
2. Membanjiri jaringan dengan jumlah request terhadap sebuah layanan jaringan yang disedakan oleh sebuah host sehingga request yang datang dari pengguna terdaftar tidak dapat dilayani oleh layanan tersebut. Teknik ini dinamakan sebagai *request flooding*.
3. Mengganggu komunikasi selang sebuah host dan kliennya yang terdaftar dengan menggunakan jumlah cara, termasuk dengan mengubah informasi konfigurasi sistem atau bahkan perusakan fisik terhadap komponen dan server.

Bentuk serangan Denial of Service awal adalah serangan SYN Flooding Attack, yang pertama kali muncul pada tahun 1996 dan mengeksploitasi terhadap kelemahan yang terdapat di dalam protokol Transmission Control Protocol (TCP). Serangan-serangan lainnya hasilnya dikembangkan untuk mengeksploitasi kelemahan yang terdapat di dalam sistem operasi, layanan jaringan atau aplikasi untuk menjadikan sistem, layanan jaringan, atau aplikasi tersebut tidak dapat melayani pengguna, atau bahkan mengalami crash. Beberapa tool yang digunakan untuk memperagakan serangan DoS pun jumlah dikembangkan sesudah itu (bahkan beberapa tool dapat diperoleh secara bebas), termasuk di selangnya Bonk, LAND, Smurf, Snork, WinNuke, dan Teardrop.

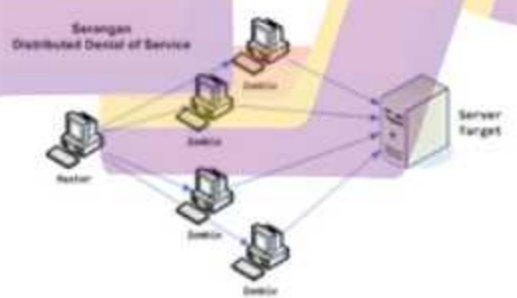
Meskipun demikian, serangan terhadap TCP merupakan serangan DoS yang sering diperagakan. Hal ini disebabkan karena jenis serangan lainnya (seperti halnya memenuhi ruangan hard disk dalam sistem, mengunci salah seorang akun

pengguna yang valid, atau memodifikasi tabel routing dalam sebuah router) membutuhkan penetrasi jaringan terlebih dulu, yang kemungkinan penetrasinya kecil, lebih-lebih jika sistem jaringan tersebut telah diperkuat.



Gambar 2.1 Percobaan serangan Denial of Service yang diperagakan terhadap sebuah host dengan sistem operasi Windows Server 2003 Service Pack 2.

Penolakan Layanan secara Terdistribusi (*DDos*)



Gambar 2.2 Cara kerja serangan Distributed Denial of Service sederhana

Penolakan Layanan secara Terdistribusi (*Distributed Denial of Service (DDos)*) adalah salah satu jenis serangan *Denial of Service* yang menggunakan jumlah host penyerang (baik itu menggunakan komputer yang didedikasikan untuk memperagakan penyerangan atau komputer yang "dipaksa" dijadikan *zombie*) untuk menyerang satu buah host target dalam sebuah jaringan.

Serangan Denial of Service klasik bersifat "satu lawan satu", sehingga dibutuhkan sebuah *host* yang kuat (baik itu dari daya pemrosesan atau sistem operasinya) demi membanjiri lalu lintas host target sehingga mencegah klien yang valid untuk mengakses layanan jaringan pada server yang dijadikan target serangan. Serangan DDoS ini menggunakan teknik yang bertambah canggih dibandingkan dengan serangan Denial of Service yang klasik, yakni dengan meningkatkan serangan beberapa kali dengan menggunakan beberapa buah komputer sekaligus, sehingga dapat mengakibatkan server atau semuanya segmen jaringan dapat dijadikan "*tidak berfaedah sama sekali*" untuk klien.

Serangan DDoS pertama kali muncul pada tahun 1999, tiga tahun sesudah serangan Denial of Service yang klasik muncul, dengan menggunakan serangan SYN Flooding, yang mengakibatkan beberapa server web di Internet mengalami "downtime". Pada awal Februari 2000, sebuah serangan yang luhur diperagakan sehingga beberapa situs web terkenal seperti Amazon, CNN, eBay, dan Yahoo! mengalami "downtime" selama beberapa jam. Serangan yang bertambah baru lagi pernah dilancarkan pada bulan Oktober 2002 ketika 9 dari 13 root DNS Server diserang dengan menggunakan DDoS yang sangat luhur yang dinamakan dengan "Ping Flood". Pada puncak serangan, beberapa server tersebut

pada tiap detiknya mendapatkan bertambah dari 150.000 *request* paket Internet Control Message Protocol (ICMP). Untungnya, karena serangan hanya diperagakan selama setengah jam saja, lalu lintas Internet pun tidak terlalu terpengaruh dengan serangan tersebut (setidaknya tidak semuanya mengalami kerusakan).

Tidak seperti dampaknya yang dijadikan sebuah kerumitan yang sangat tinggi (bagi para administrator jaringan dan server yang memperagakan perbaikan server dampak dari serangan), teori dan praktik untuk memperagakan serangan DDoS justru sederhana, yakni sebagai berikut:

1. Menjalankan tool (biasanya berupa program (perangkat lunak) kecil) yang secara otomatis akan memindai jaringan untuk menemukan host-host yang rentan (*vulnerable*) yang terkoneksi ke Internet. Sesudah host yang rentan ditemukan, tool tersebut dapat menginstalasikan salah satu jenis dari Trojan Horse yang dinamakan sebagai DDoS Trojan, yang akan mengakibatkan host tersebut dijadikan *zombie* yang dapat dikontrol secara jarak jauh (bahasa Inggris: *remote*) oleh sebuah komputer master yang digunakan oleh si penyerang asli untuk melancarkan serangan. Beberapa tool {*software*} yang digunakan untuk memperagakan serangan serperti ini adalah TFN, TFN2K, Trinoo, dan Stacheldraht, yang dapat diunduh secara lepas sama sekali di Internet.
2. Ketika si penyerang merasa telah mendapatkan jumlah host yang cukup (sebagai zombie) untuk memperagakan penyerangan, penyerang akan menggunakan komputer master untuk memberikan sinyal penyerangan terhadap jaringan target atau host target. Serangan ini umumnya diperagakan

dengan menggunakan beberapa bentuk SYN Flood atau skema serangan DoS yang sederhana, tapi karena diperagakan oleh jumlah host zombie, maka jumlah lalu lintas jaringan yang diciptakan oleh mereka adalah sangat luhur, sehingga "memakan habis" semua sumber daya Transmission Control Protocol yang terdapat di dalam komputer atau jaringan target dan dapat mengakibatkan host atau jaringan tersebut mengalami "downtime".

Hampir semua platform komputer dapat dibajak sebagai sebuah *zombie* untuk memperagakan serangan seperti ini. Sistem-sistem populer, semacam Solaris, Linux, Microsoft Windows dan beberapa varian UNIX dapat dijadikan zombie, jika memang sistem tersebut atau aplikasi yang berlanjut di atasnya memiliki kelemahan yang dieksploitasi oleh penyerang.

Beberapa contoh Serangan DoS lainnya adalah:

1. Serangan Buffer Overflow, mengirimkan data yang melebihi kapasitas sistem, misalnya paket ICMP yang berukuran sangat luhur.
2. Serangan SYN, mengirimkan data TCP SYN dengan alamat palsu.
3. Serangan Teardrop, mengirimkan paket IP dengan nilai *offset*yang membingungkan.
4. Serangan Smurf, mengirimkan paket ICMP bervolume luhur dengan alamat *host* lain.
5. ICMP Flooding

2.3.2 Distributed Denial of Service (DDoS)

Serangan denial-of-service (DDoS) terdistribusi adalah upaya jahat untuk mengganggu lalu lintas normal server, layanan, atau jaringan yang ditargetkan dengan membanjiri target atau infrastruktur di sekitarnya dengan membanjiri lalu lintas Internet.

Serangan DDoS mencapai efektivitas dengan memanfaatkan beberapa sistem komputer yang dikompromikan sebagai sumber lalu lintas serangan. Mesin yang dieksploitasi dapat mencakup komputer dan sumber daya jaringan lainnya. Dari tingkat tinggi, serangan DDoS seperti kemacetan lalu lintas yang tidak terduga yang menyumbat jalan raya, mencegah lalu lintas reguler tiba di tujuannya.

Serangan DDoS dilakukan dengan jaringan mesin yang terhubung ke Internet. Jaringan ini terdiri dari komputer dan perangkat lain yang telah terinfeksi malware, memungkinkan mereka untuk dikontrol dari jarak jauh oleh penyerang. Perangkat individu ini disebut sebagai bot (atau zombie), dan sekelompok bot disebut botnet.

Setelah botnet dibuat, penyerang dapat mengarahkan serangan dengan mengirimkan instruksi jarak jauh ke setiap bot. Ketika server atau jaringan korban ditargetkan oleh botnet, setiap bot mengirimkan permintaan ke alamat IP target, yang berpotensi menyebabkan server atau jaringan menjadi kewalahan, sehingga mengakibatkan penolakan layanan ke lalu lintas normal. Karena setiap bot adalah perangkat Internet yang sah, memisahkan lalu lintas serangan dari lalu lintas normal bisa jadi sulit.

Berbagai jenis serangan DDoS menargetkan berbagai komponen koneksi jaringan. Untuk memahami bagaimana berbagai serangan DDoS bekerja, perlu diketahui bagaimana koneksi jaringan dibuat. Sambungan jaringan di Internet terdiri dari banyak komponen atau "lapisan" yang berbeda. Ibarat membangun rumah dari awal, setiap lapisan dalam model memiliki tujuan yang berbeda. Model OSI, yang ditunjukkan di bawah, adalah kerangka kerja konseptual yang digunakan untuk menggambarkan konektivitas jaringan dalam 7 lapisan berbeda.

Tabel 2.2. OSI Model

Layer		Data
7	Application	Data
6	Presetation	Data
5	Session	Data
4	Transport	Segments
3	Network	Packets
2	Datalink	Frames
1	Physical	Bits

Meskipun hampir semua serangan DDoS melibatkan banyak sekali perangkat atau jaringan target dengan lalu lintas, serangan dapat dibagi menjadi tiga kategori. Seorang penyerang dapat menggunakan satu atau lebih vektor serangan yang berbeda, atau vektor serangan siklus sebagai tanggapan terhadap tindakan balasan yang diambil oleh target.

Naïve Bayes Classifier merupakan sebuah metoda klasifikasi yang berakar pada teorema Bayes. Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dr Naïve Bayes

Classifier ini adalah asumsi yg sangat kuat (naïf) akan independensi dari masing-masing kondisi atau kejadian.

Menurut Olson Delen (2008) menjelaskan Naïve Bayes unt setiap kelas keputusan, menghitung probabilitas dg syarat bahwa kelas keputusan adalah benar, mengingat vektor informasi obyek. Algoritma ini mengasumsikan bahwa atribut obyek adalah independen. Probabilitas yang terlibat dalam memproduksi perkiraan akhir dihitung sebagai jumlah frekuensi dr " master " tabel keputusan.

Konsep sederhana DDoS attack adalah membanjiri lalu lintas jaringan dengan banyak data. Konsep Denial of Service bisa dibagi menjadi 3 tipe penggunaan, yakni sebagai berikut :

1. Request flooding merupakan teknik yang digunakan dengan membanjiri jaringan menggunakan banyak request. Akibatnya, pengguna lain yang terdaftar tidak dapat dilayani.
2. Traffic flooding merupaka teknik yang digunakan dengan membanjiri lalu lintas jaringan dengan banyak data. Akibatnya, pengguna lain tidak bisa dilayani.
3. Mengubah sistem konfigurasi atau bahkan merusak komponen dan server juga termasuk tipe denial of service, tetapi cara ini tidak banyak digunakan karena cukup sulit untuk dilakukan.

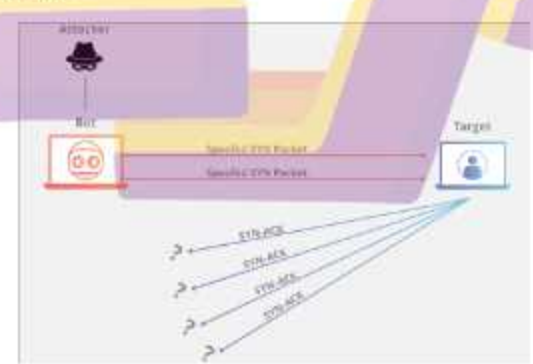
Sedangkan jika kategorikan berdasarkan layer OSI, ada serangan pada layer aplikasi, protokol, dan volumetrik.

1. DDoS Layer Aplikasi

Gambar 2.3. http flood ddos attack

Kategori penyerangan ini adalah mengambil semua sumber daya dari target. Target dari serangan adalah later dimana halaman website dieksekusi pada server dan mengirimkan respon ke HTTP request. Sangat ringan jika hanya melayani satu request. Sedangkan akan menjadi masalah jika melayani banyak request secara bersamaan apalagi jika menjalankan query database juga.

2. DDoS Protokol



Gambar 2.4. syn flood ddos attack

Serangan ini mengekploitasi TCP dengan cara mengirimkan paket SYN dengan spoof alamat IP dalam jumlah yang besar. Setiap koneksi yang masuk akan ditanggapi oleh server yang menunggu proses koneksi berjalan, namun tidak pernah terjadi. Hal ini akan mengakibatkan proses yang terus berjalan pada server yang dapat menyebabkan overload.

3. Volumetrik DDoS

Gambar 2.5. ntp amplification botnet ddos attack

Tujuan dari serangan DDoS ini adalah menghabiskan semua bandwidth yang tersedia antara target dengan jaringan internet. Caranya adalah dengan membuat lalu lintas yang sangat padat, seperti penggunaan botnet.

Tidak semua jenis yang dapat saya bahas, masih ada banyak tipe serangan DDoS yang tercatat selama ini, seperti serangan Memcached DDoS, NTP Apmlification, DNS Flood, UDP Flood, dan masih banyak lainya.

2.3.3 Algoritma Naïve Bayes

Naive Bayes Classifier bekerja sangat baik dibanding dengan model classifier lainnya. Hal ini dibuktikan oleh Xhemali , Hinde Stone dalam jurnalnya "Naïve Bayes vs. Decision Trees vs. Neural Networks in the Classification of

Training Web Pages" mengatakan bahwa "Naïve Bayes Classifier memiliki tingkat akurasi yg lebih baik dibandingmodel classifier lainnya". (Binus, 2019)

Keuntungan penggunan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (training data) yg kecil unt menentukan estimasi parameter yg diperlukan dalam proses pengklasifikasian. Karena yg diasumsikan sebagai variable independent, maka hanya varians dr suatu variable dalam sebuah kelas yg dibutuhkan unt menentukan klasifikasi, bukan keseluruhan dr matriks kovarians.

Kegunaan Naïve Bayes:

1. Mengklasifikasikan dokumen teks seperti teks berita ataupun teks akademis
2. Sebagai metode machine learning yang menggunakan probabilitas
3. Untuk membuat diagnosis medis secara otomatis
4. Mendeteksi atau menyaring spam

Kelebihan Naïve Bayes:

1. Bisa dipakai untuk data kuantitatif maupun kualitatif
2. Tidak memerlukan jumlah data yang banyak
3. Tidak perlu melakukan data training yang banyak
4. Jika ada nilai yang hilang, maka bisa diabaikan dalam perhitungan.
5. Perhitungannya cepat dan efisien
6. Mudah dipahami
7. Mudah dibuat

8. Pengklasifikasian dokumen bisa dipersonalisasi, disesuaikan dengan kebutuhan setiap orang
9. Jika digunakan dalaam bahasa pemrograman, *code*-nya sederhana
10. Bisa digunakan untuk klasifikasi masalah biner ataupun *multiclass*

Kekurangan Naïve Bayes:

1. Apabila probabilitas kondisionalnya bernilai nol, maka probabilitas prediksi juga akan bernilai nol
2. Asumsi bahwa masing-masing variabel independen membuat berkurangnya akurasi, karena biasanya ada korelasi antara variabel yang satu dengan variabel yang lain
3. Keakuratannya tidak bisa diukur menggunakan satu probabilitas saja. Butuh bukti-bukti lain untuk membuktikannya.
4. Untuk membuat keputusan, diperlukan pengetahuan awal atau pengetahuan mengenai masa sebelumnya. Keberhasilannya sangat bergantung pada pengetahuan awal tersebut Banyak celah yang bisa mengurangi efektivitasnya dirancang untuk mendeteksi kata-kata saja, tidak bisa berupa gambar

Persamaan dari teorema Bayesdapat dilihat di bawah ini :

$$P(H|X) = \frac{P(X|H).P(H)}{P(H)}.$$

Dimana :

X : data dengan classyang belum diketahui

H : hipotesis data menggunakan suatu class spesifik

P(H|X) : probabilitas hipotesis H berdasar kondisi X (parteriori probabilitas)

P(H) : probabilitas hipotesis H (prior probabilitas)

P(X|H) : probabilitas X bedasarkan kondisi pada hipotesis H

P(X) : probabilitas H

Untuk menjelaskan metode Naive Bayes, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang di analisis tersebut. Karena itu, metode Naive Bayesdi atas disesuaikan sebagai berikut (Saleh, 2015)

$$P(C|F1 \ldots . Fn) = \frac{P(C)P(F1 \ldots Fn|C)}{P(F1 \ldots Fn)}.$$

Di mana Variabel C mempresentasikan kelas, sementara variabel F1...Fn mempresentasikan karakteristik petunjuk yang dibutuhkan untuk menentukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (Posterior) adalah peluang munculnya kelas C (sebelum masuknya sampel tersebut, seringkali disebut prior), dikali dengan peluang kemunculan karakteristik – karakteristik sampel pada kelas C (disebut likelihood), dibagi dengan peluang kemunculan karakteristik – karakteristik secara global (disebut juga evidence). Karena itu, rumus di atas dapat pula ditulis secara sederhana sebagai berikut (Saleh, 2015):

$$posterior = \frac{prior\ x\ likelihood}{evidence}$$

Nilai Evidence selalu tetap untuk setiap kelas pada satu sampel. Nilai dari Posterior tersebut nantinya akan dibandingkan dengan nilai – nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan.

Cara kerja pengklasifikasi Naive Bayes:

Mari kita memahami cara kerja Naive Bayes melalui contoh. Diberikan contoh kondisi cuaca dan bermain olahraga. Perlu menghitung probabilitas bermain olahraga. Sekarang, perlu mengklasifikasikan apakah pemain akan bermain atau tidak, berdasarkan kondisi cuaca.

Pendekatan Pertama (Dalam hal fitur tunggal)

Klasifikasi Naive Bayes menghitung probabilitas suatu peristiwa dalam langkah-langkah berikut:

Langkah 1: Hitung probabilitas sebelumnya untuk label kelas yang diberikan

Langkah 2: Temukan probabilitas Peluang dengan setiap atribut untuk setiap kelas

Langkah 3: Masukkan nilai ini dalam Formula Bayes dan hitung probabilitas posterior.

Langkah 4: Lihat kelas mana yang memiliki probabilitas lebih tinggi, mengingat input milik kelas probabilitas lebih tinggi.

Untuk menyederhanakan perhitungan probabilitas sebelum dan posterior, Anda dapat menggunakan tabel dua frekuensi dan kemungkinan. Kedua tabel ini akan membantu Anda menghitung probabilitas sebelum dan sesudah. Tabel Frekuensi berisi kemunculan label untuk semua fitur. Ada dua tabel kemungkinan.

Kemungkinan Tabel 1 menunjukkan probabilitas label sebelumnya dan Kemungkinan Tabel 2 menunjukkan probabilitas posterior.

Sekarang anggaplah Anda ingin menghitung probabilitas bermain ketika cuaca Overcast/mendung.

Kemungkinan bermain:

P (Yes | Overcast) = P (Overcast | Yes) P (Yes) / P (Overcast) ..... (1)

1. Hitung Kemungkinan Sebelumnya:

   P (Overcast) = 4/14 = 0.29

   P (Yes) = 9/14 = 0.64

2. Hitung Kemungkinan Posterior:

   P (Overcast | Yes) = 4/9 = 0,44

3. Masukkan probabilitas Prior dan Posterior dalam persamaan (1)

   P (Yes | Overcast) = 0.44 * 0.64 / 0.29 = 0.98 (Lebih Tinggi)

Demikian pula, Anda dapat menghitung probabilitas tidak bermain:

Kemungkinan tidak bermain:

P (No | Overcast) = P (Overcast | No) P (No) / P (Overcast) ..... (2)

1. Hitung Kemungkinan Sebelumnya:

   P (Overcast) = 4/14 = 0.29

P (No) = 5/14 = 0,36

2. Hitung Kemungkinan Posterior:

   P (Overcast | No) = 0/9 = 0

3. Masukkan probabilitas Prior dan Posterior dalam persamaan (2)

   P (No | Overcast) = 0 * 0,36 / 0,29 = 0

Probabilitas kelas 'Yes' lebih tinggi. Jadi Anda dapat menentukan di sini apakah cuaca mendung/overcast membuat pemain akan berolahraga.

Pendekatan Kedua (Dalam hal banyak fitur)

Sekarang anggaplah Anda ingin menghitung probabilitas bermain ketika cuaca Overcast (mendung), dan suhunya Mild (ringan).

Kemungkinan bermain:

P (Play = Yes | Weather = Overcast, Temp = Mild) = P (Weather = Overcast, Temp = Mild | Play = Yes) P (Play = Yes) ..... (1)

P (Weather = Overcast, Temp = Mild | Play = Yes) = P (Overcast | Yes) P (Mild | Yes) ..... (2)

1. Hitung Probabilitas Sebelumnya:

   P (Yes) = 9/14 = 0.64

2. Hitung Probabilitas Posterior:

   P (Overcast | Yes) = 4/9 = 0,44 P (Mild | Yes) = 4/9 = 0,44

3. Masukkan probabilitas Posterior dalam persamaan (2)

   P (Weather = Overcast, Temp = Mild | Play = Yes) = 0,44 * 0,44 = 0,1936 (Lebih Tinggi)

4. Masukkan probabilitas Prior dan Posterior dalam persamaan (1)

P (Play = Yes | Weather = Overcast, Temp = Mild) = 0.1936 * 0.64 = 0.124

Demikian pula, dapat menghitung probabilitas tidak bermain:

Kemungkinan tidak bermain:

P (Play = No | Weather = Overcast, Temp = Mild) = P (Weather = Overcast, Temp = Mild | Play = No) P (Play = No) ..... (3)

P (Weather = Overcast, Temp = Mild | Play = No) = P (Weather = Overcast | Play = No) P (Temp = Mild | Play = No) ..... (4)

1. Hitung Kemungkinan Sebelumnya:

   P (Tidak) = 5/14 = 0,36

2. Hitung Probabilitas Posterior:

   P (Weather = Overcast | Play = No) = 0/9 = 0 P (Temp = Mild | Play = No) = 2/5 = 0,4

3. Masukkan probabilitas posterior dalam persamaan (4)

   P (Weather = Overcast, Temp = Mild | Play = No) = 0 * 0,4 = 0

4. Masukkan probabilitas sebelum dan belakang dalam persamaan (3)

   P (Play = No | Weather = Overcast, Temp = Mild) = 0 * 0,36 = 0

Probabilitas kelas 'Yes' lebih tinggi. Jadi dapat mengatakan di sini bahwa jika cuaca mendung maka pemain akan bermain olahraga.

Jenis pengklasifikasi Naive Bayes :

1. Naive Bayes Multinomial

Biasanya digunakan ketika fitur mewakili frekuensi beberapa peristiwa. Model tersebut merupakan varian dari Naive Bayes yang banyak digunakan pada Natural Language Processing (NLP). Ini menggunakan teorema Bayes untuk

memprediksi tag teks seperti sepotong email atau artikel surat kabar. Dan untuk setiap tag dalam sampel yang diberikan, ia menghitung probabilitas dan mengeluarkan tag dengan probabilitas tertinggi.

2. Bernoulli Naive Bayes

Model ini digunakan untuk **data diskrit** dan bekerja pada distribusi Bernoulli. Bernoulli Naive Bayes memiliki karakteristik utama yang hanya menerima fitur biner, seperti benar atau salah, ya atau tidak, sukses atau gagal, 0 atau 1. Jadi, setiap kali nilai fitur biner, kami menggunakan pengklasifikasi Bernoulli Naive Bayes.

Sekarang untuk nilai biner seperti p dan q, mari kita pertimbangkan 'p' sebagai probabilitas sukses dan 'q' sebagai probabilitas gagal yang juga sama dengan (1-p), kemudian untuk variabel acak 'X' di Bernoulli distribusi:

$$p(x) = P(X = x) = \begin{cases} p & x = 1 \\ 1 - p & x = 0 \end{cases}$$

Aturan keputusan untuk Bernoulli Naive Bayes adalah:

$$P(x_i \mid y) = P(i|y)x_i + (1 - P(i|y))(1 - x_i)$$

di mana, i adalah peristiwa, dan x_i harus biner (0 atau 1).

3. Gaussian Naive Bayes

Ketika berhadapan dengan data kontinu , model ini umum digunakan, seharusnya nilai-nilai kontinu yang terkait dengan setiap kelas didistribusikan sesuai dengan distribusi normal (atau distribusi Gaussian).

Aturan keputusan untuk Gaussian Naive Bayes adalah:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Dimana μ adalah mean dan σ adalah standar deviasi bahwa kita harus memperkirakan dari data.

2.3.4 Feature Selection

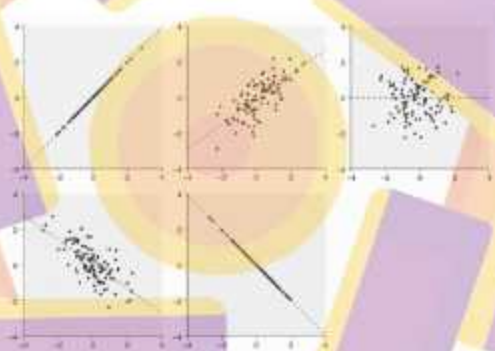
Feature selection atau seleksi fitur adalah proses pemilihan fitur-fitur yang paling berkontribusi pada variabel prediksi atau output baik secara otomatis atau pun manual. Memiliki fitur yang tidak relevan dalam data dapat mengurangi keakuratan model dan membuat model belajar berdasarkan fitur yang tidak relevan.

Seleksi fitur, sebagai strategi preprocessing data, telah terbukti efektif dan efisien dalam menyiapkan data untuk berbagai masalah data mining dan machine learning (Li, J. et. al, 2017). Tujuan pemilihan fitur meliputi: membangun model yang lebih sederhana dan lebih mudah dipahami, meningkatkan kinerja penambangan data, dan menyiapkan data yang bersih dan mudah dipahami.

2.3.5. Pearson Correlation

Pearson correlation adalah salah satu metode pengukuran korelasi antar dua variabel kontinyu yang dicetuskan oleh Karl Pearson dalam disiplin ilmu matematika statistika (Courtney, Matthew Gordon Ray, 2018).

Pearson corellation adalah ukuran hubungan linier antara dua variabel, misal X dan Y, memberikan nilai antara +1.0 dan -1.0, di mana 1.0 adalah korelasi positif sempurna, 0.0 (nol) tidak ada korelasi, dan -1.0 adalah korelasi negatif sempurna. Contoh distribusi data yang mungkin terkait dengan lima korelasi Pearson diilustrasikan pada Gambar 2.6.



Gambar 2.6. Contoh lima koefisien korelasi product-moment Pearson (sumber: The SAGE Encyclopedia of Educational Research, Measurement, and Evaluation)

BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

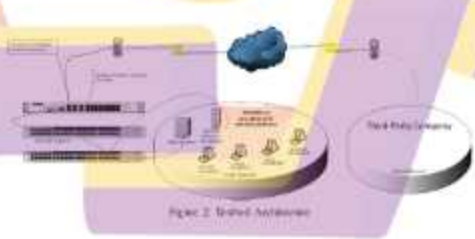
Jenis penelitian ini menggunakan eksperimental yaitu dengan meneliti Instruction Detection System Distributed Denial of Service dengan menggunakan algoritma Naïve Bayes Model GaussianNB dan BernoulliNB pada jaringan komputer. Sifat penelitian ini dilakukan secara deskriptif, untuk mempermudah dalam peneitian dibuatkan beberapa parameter pengukuran yang ditetapkan untuk mendapatkan hasil yang optimal. Pada penelitian ini menggunakan pendekatan kualitatif pada Instruction Detection System Distributed Denial of Service dengan menggunakan algoritma Naïve Bayes Model GaussianNB dan BernoulliNB pada jaringan komputer.

3.2. Metode Pengumpulan Data

Data yang digunakan yaitu set data CICDDoS2019 yang dikeluarkan oleh Canada Institute for Cybersecurity. Data set ini sejak awal mulai menarik para peneliti untuk melakukan analisis serta pengembangan model dan algoritma baru (Panigrahi & Borah, 2018). Set data CICDDoS2019 berisi kelas normal dan serangan terbaru dalam format PCAP. Selain itu tersedia juga data dengan format CSV yang sudah di ekstrak menggunakan CICFlowMeter dengan tujuan pembelajaran machine learning atau deep learning yang dapat dengan bebas digunakan oleh peneliti. Dalam penelitian ini, pertama-tama peneliti Canada

Institute for Cybersecurity meninjau kumpulan data yang ada secara komprehensif dan mengusulkan taksonomi baru untuk serangan DDoS. Kedua, peneliti Canada Institute for Cybersecurity menghasilkan kumpulan data baru, yaitu CICDDoS2019, yang memperbaiki semua kekurangan saat ini. Ketiga, menggunakan set data yang dihasilkan, peneliti Canada Institute for Cybersecurity mengusulkan deteksi baru dan pendekatan klasifikasi keluarga berdasarkan sekumpulan fitur aliran jaringan. Terakhir, peneliti Canada Institute for Cybersecurity menyediakan rangkaian fitur terpenting untuk mendeteksi berbagai jenis serangan DDoS dengan bobotnya yang sesuai.

CICDDoS2019 berisi serangan DDoS umum yang normal dan paling mutakhir, yang menyerupai data dunia nyata (PCAP) yang sebenarnya. Ini juga mencakup hasil analisis lalu lintas jaringan menggunakan CICFlowMeter-V3 dengan aliran berlabel berdasarkan stempel waktu, sumber, dan IP tujuan, port sumber dan tujuan, protokol dan serangan (file CSV).



Gambar 3.1. Arsitektur dan Topologi Jaringan

Menghasilkan lalu lintas latar belakang yang realistis adalah prioritas utama peneliti Canada Institute for Cybersecurity dalam membangun kumpulan data ini. peneliti Canada Institute for Cybersecurity telah menggunakan sistem Profil B yang

peneliti Canada Institute for Cybersecurity usulkan (Sharafaldin, dkk. 2016) untuk membuat profil perilaku abstrak interaksi manusia dan menghasilkan lalu lintas latar belakang normal naturalistik di tempat pengujian yang diusulkan (Gambar 2). Untuk kumpulan data ini, peneliti Canada Institute for Cybersecurity membangun perilaku abstrak dari 25 pengguna berdasarkan protokol HTTP, HTTPS, FTP, SSH, dan email.

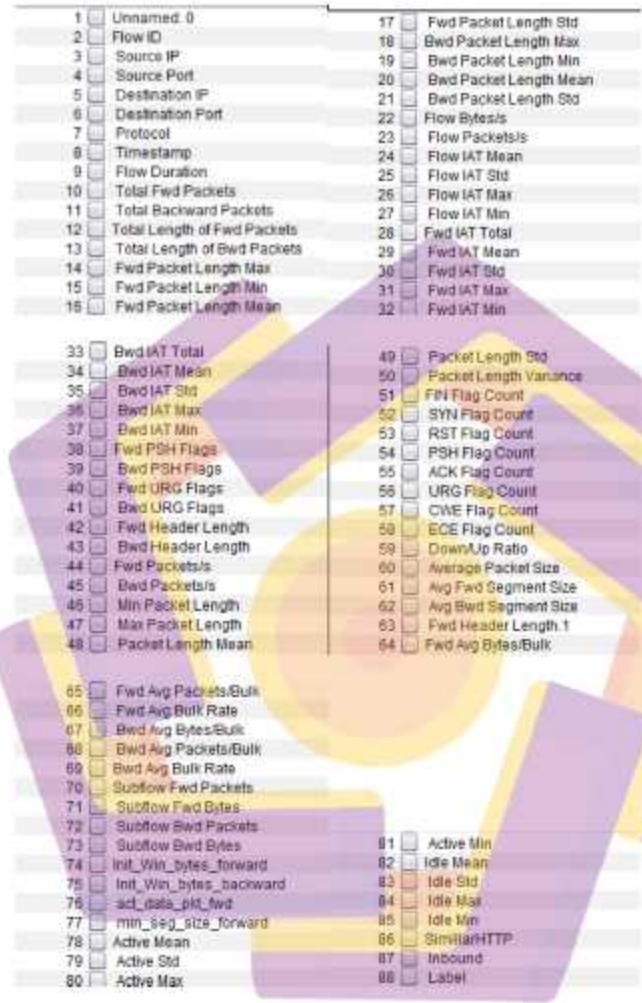
Dalam kumpulan data ini, peneliti Canada Institute for Cybersecurity memiliki berbagai serangan DDoS reflektif modern seperti PortMap, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, SYN, NTP, DNS, dan SNMP. Jangka waktu penangkapan untuk hari pelatihan tanggal 12 Januari dimulai pukul 10.30 dan berakhir pukul 17.15, dan untuk hari uji coba pada tanggal 11 Maret dimulai pukul 09.40 dan berakhir pukul 17.35. Serangan kemudian dilakukan selama periode ini. Seperti yang ditunjukkan Tabel III, peneliti Canada Institute for Cybersecurity mengeksekusi 12 serangan DDoS termasuk NTP, DNS, LDAP, MSSQL, NetBIOS, SNMP, SSDP, UDP, UDP-Lag, WebDDoS, SYN dan TFTP pada hari pelatihan dan 7 serangan termasuk PortScan, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag dan SYN pada hari pengujian. Volume lalu lintas untuk WebDDoS sangat rendah dan PortScan baru saja dijalankan pada hari pengujian dan tidak akan diketahui untuk mengevaluasi model yang diusulkan.

Tabel 3.1. Simulasi Serangan

Days	Attacks	Attack Time
First Day	PortMap NetBIOS LDAP MSSQL UDP UDP-Lag SYN	9:43 - 9:51 10:00 - 10:09 10:21 - 10:30 10:33 - 10:42 10:53 - 11:03 11:14 - 11:24 11:28 - 17:35
Second Day	NTP DNS LDAP MSSQL NetBIOS SNMP SSDP UDP UDP-Lag WebDDoS SYN TFTP	10:35 - 10:45 10:52 - 11:05 11:22 - 11:32 11:36 - 11:45 11:50 - 12:00 12:12 - 12:23 12:27 - 12:37 12:45 - 13:09 13:11 - 13:15 13:18 - 13:29 13:29 - 13:34 13:35 - 17:15

3.3. Metode Analisis Data

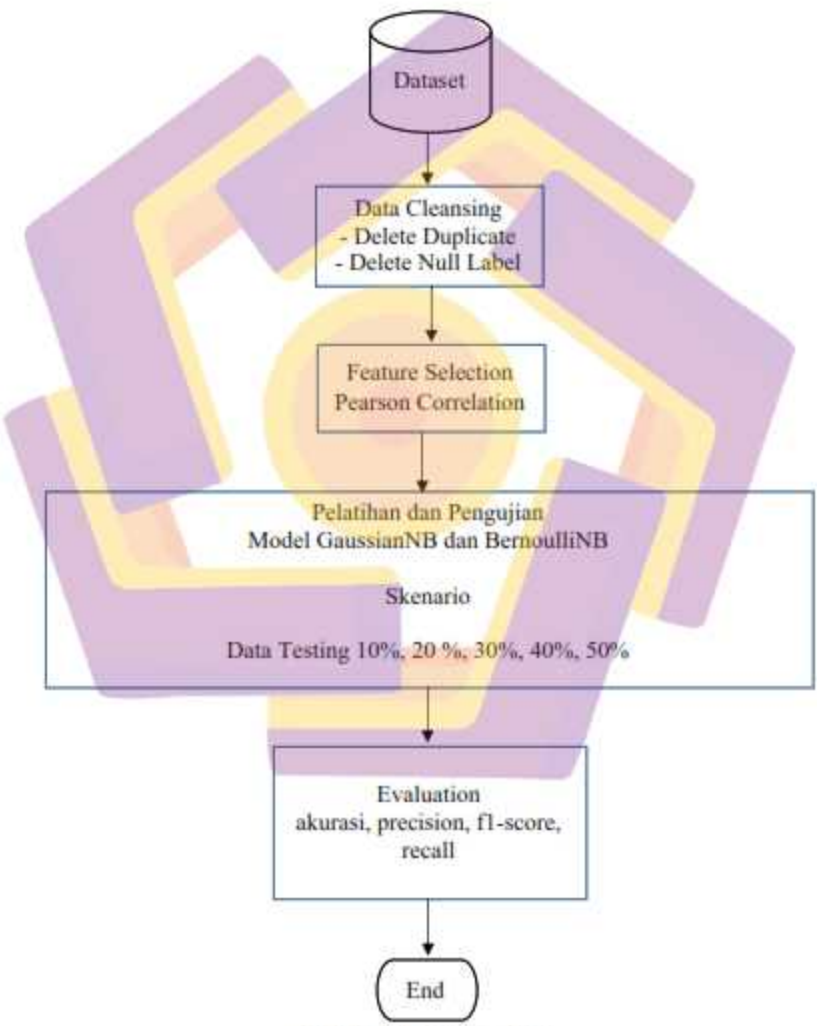
Proses analisis data dimulai dari dataset berhasil dikumpulkan, analisis yang akan dilakukan dalam penelitian ada dua analisis utama, analisis pertama adalah analisis fitur apa saja yang berpengaruh terhadap klasifikasi sebuah dataset dapat diklasifikasikan sebagai informasi serangan DDoS atau tidak, maka dari dataset yang sudah dikumpulkan akan dilakukan ekstraksi fitur dengan fitur-fitur sebagai berikut:



Untuk analisis yang kedua yaitu mengetahui algoritma mana yang terbaik, maka akan dilakukan penghitungan akurasi pada masing-masing model. Penghitungan akurasi akan dilakukan menggunakan dataset yang sudah dibuat dengan cara membagi menjadi data training dan data test. Validasi keakuratan pemodelan akan dilakukan pengukuran precision dan recall.

3.4. Alur Penelitian

Bagian ini berisi diagram alur langkah penelitian secara lengkap dan terinci termasuk di dalamnya tercermin algoritma, rute, pemodelan-pemodelan, desain, yang terkait dengan aspek perancangan sistem.



Gambar 3.4 Diagram Alur

Penjelasan mengenai alur penelitian adalah sebagai berikut:

1. Dataset

Persiapan data merupakan tahap penting yang bertujuan untuk memanipulasi data menjadi format yang dapat dimengerti oleh sistem. Kebanyakan data yang diunakan untuk dianalisis merupakan data yang tidak lengkap dan tidak konsisten. Oleh karena itu perlu dilakukan praproses data yang bertujuan untuk meningkatkan akurasi dan efisiensi dari hasil yang didapatkan. Praproses data seangatlah penting dan vital dalam menganalisis lalu lintas jaringan karena polanya memiliki tipe data dan dimensi berbeda-beda.

2. Pembersihan Data

Tahap praproses yang pertama dilakukan yaitu pembersihan data. Pembersihan data perlu dilakukan karena sering didapati data yang hilang atau rusak. Penting untuk memahami sumber data yang hilan atau rusak tersebut. Bisa jadi kesalahan pada saat transfer data, kesalahan yang disebabkan oleh sistem ataupun disebabkan oleh permasalahan lainnya. Jika hal tersebut ada pada data yang digunakan, maka akan menghambar proses analisis yang dilakukan. Oleh sebab itu data tersebut perlu dihilangkan dengan cara mengganti data tersebut dengan nilai konstan atau dengan nilai tengah atau rata-rata dari data yang sekolong dengan data tersebut.

3. Feature Selection

Feature corellation menjadi hal yang cukup penting dalam proses *feature selection,* pengecekan korelasi antar fitur dapat membantu memahami keterhubungan antara fitur seperti wawasan apakah terdapat fitur yang bergantung

terhadap fitur yang lainnya, apakah ada indikasi hubungan sebab akibat dari keterhubungan fitur-fitur tersebut.

4. Pelatihan dan Pengujian

Setelah didapatkan model yang telah di-training sesuai dengan fitur yang terpilih sesuai metode feature selection, langkah selanjutnya adalah melakukan testing dengan skenario membagi data testing menjadi 10%, 20%, 30%, 40%, 50%.

5. Evaluasi

Dari hasil testing tersebut akan dilakukan evaluasi berdasarkan akurasi, precision, f1-score, recall.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. Analisis Pengumpulan Data

Pengumpulan data merupakan salah satu bagian penting dalam penelitian ini, data yang akan digunakan untuk penelitian ini adalah data yang berasal dari CICDDoS2019 berisi serangan DDoS normal dan terbaru, menyerupai data nyata (PCAP). Ini juga mencakup analisis analisis lalu lintas jaringan menggunakan CICFlowMeter-V32 (A. H. Lashkari, Y. Zang, G. Owhuo, M. S. I. Mamun, and G. D. Gil, "CICFlowMeter," Github. 2017) dan arus berlabel. Sistem B-Profile (unb.ca, 2019) digunakan untuk profil perilaku abstrak interaksi manusia dan menghasilkan lalu lintas latar belakang normal naturalistik. Untuk dataset ini, perilaku abstrak dari 25 pengguna dibangun berdasarkan protokol HTTP, HTTPS, FTP, SSH, dan email (unb.ca, 2019). Dataset mencakup berbagai serangan DDoS reflektif modern seperti Port Map, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, SYN, NTP, DNS, dan SNMP. Periode penangkapan untuk hari pelatihan pada tanggal 12 Januari dimulai pada pukul 10:30 dan berakhir pada pukul 17:15, dan untuk hari pengujian pada tanggal 11 Maret dimulai pada pukul 09:40 dan berakhir pada pukul 17:35. Serangan kemudian dilakukan selama periode ini.

Berikut arsitektur dan topologi serangan:

Gambar 4.1. Arsitektur dan Topologi Jaringan

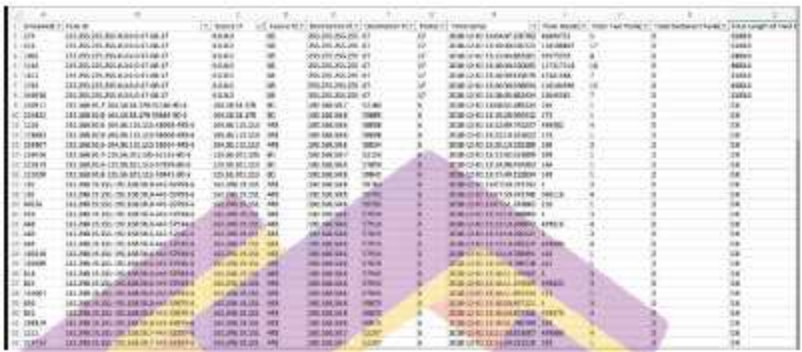
4.1.1. Dataset

Persiapan data merupakan tahap penting yang bertujuan untuk memanipulasi data menjadi format yang dapat dimengerti oleh sistem. Kebanyakan data yang diunakan untuk dianalisis merupakan data yang tidak lengkap dan tidak konsisten. Oleh karena itu perlu dilakukan praproses data yang bertujuan untuk meningkatkan akurasi dan efisiensi dari hasil yang didapatkan. Praproses data seangatlah penting dan vital dalam menganalisis lalu lintas jaringan karena polanya memiliki tipe data dan dimensi berbeda-beda.

Gambar 4.2. Data raw dari CICDDoS2019 hasil konversi ke bentuk csv

Proses ekstraksi menghasilkan data dengan format csv yang berisi fitur-fitur. Tetapi dari hasil ekstraksi yang dilakukan, fitur label tidak dapat di

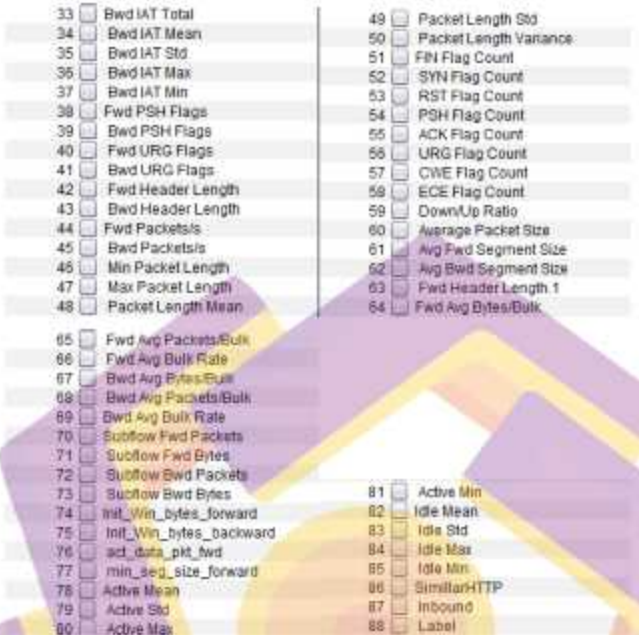
klasifikasikan secara otomatis, mana yang termasuk kelas normal ataupun kelas serangan.



Gambar 4.3. Data raw dari hasil konversi bentuk csv tanpa koma

Set data ini menyediakan data lengkap berisi jenis serangan pada jaringan dan juga data terpisah setiap serangan. Data yang digunakan pada penelitian ini yaitu data yang khusus berisi kelas normal dan kelas serangan DDoS saja. Adapun fitur-fitur yang ada pada data ini yaitu sebanyak 88 fitur seperti dibawah ini:

1	Unnamed: 0	17	Fwd Packet Length Std
2	Flow ID	18	Bwd Packet Length Max
3	Source IP	19	Bwd Packet Length Min
4	Source Port	20	Bwd Packet Length Mean
5	Destination IP	21	Bwd Packet Length Std
6	Destination Port	22	Flow Bytes/s
7	Protocol	23	Flow Packets/s
8	Timestamp	24	Flow IAT Mean
9	Flow Duration	25	Flow IAT Std
10	Total Fwd Packets	26	Flow IAT Max
11	Total Backward Packets	27	Flow IAT Min
12	Total Length of Fwd Packets	28	Fwd IAT Total
13	Total Length of Bwd Packets	29	Fwd IAT Mean
14	Fwd Packet Length Max	30	Fwd IAT Std
15	Fwd Packet Length Min	31	Fwd IAT Max
16	Fwd Packet Length Mean	32	Fwd IAT Min



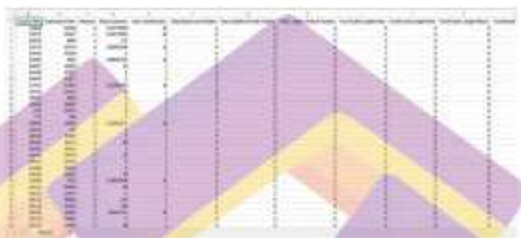
Gambar 4.4. Seluruh Fitur Dataset

4.1.2. Pembersihan Data

Tahap praproses yang pertama dilakukan yaitu pembersihan data. Pembersihan data perlu dilakukan karena sering didapati data yang hilang atau rusak. Penting untuk memahami sumber data yang hilan atau rusak tersebut. Bisa jadi kesalahan pada saat transfer data, kesalahan yang disebabkan oleh sistem ataupun disebabkan oleh permasalahan lainnya. Jika hal tersebut ada pada data yang digunakan, maka akan menghambar proses analisis yang dilakukan. Oleh sebab itu data tersebut perlu dihilangkan dengan cara mengganti data tersebut dengan nilai konstan atau dengan nilai tengah atau rata-rata dari data yang sekolong dengan data tersebut.

data.fillna (method −'ffill', inplace − True)

beberapa hal yang dilakukan diantaranya adalah menghapus duplikasi data dan fitur yang tidak berpengaruh dari hasil pengamatan dibuang dahulu seperti Unnamed, Flow ID, Source IP, Destination IP, Timestamp.



Gambar 4.5. Hasil Pembersihan

Fungsi *head()* berfungsi untuk menampilkan lima baris awal dataset. Tujuannya untuk menunjukan bahwa dataset yang telah diambil sudah terbaca. Gambar 4.6 dibawah ini menunjukan sebagian dataset menggunakan fungsi *head()*.

Gambar 4.6. Sebagian isi Dataset

Fungsi *str.replace()* yaitu berfungsi untuk mengganti nama fitur jika ada spasi dan diganti dengan garis bawah (_) . berikut tampilan penggantian nama fitur

Gambar 4.7. Penggantian nama fitur

4.1.3. Feature Selection Metode Pearson Correlation

Feature corellation menjadi hal yang cukup penting dalam proses *feature selection,* pengecekan korelasi antar fitur dapat membantu memahami keterhubungan antara fitur seperti wawasan apakah terdapat fitur yang bergantung terhadap fitur yang lainnya, apakah ada indikasi hubungan sebab akibat dari keterhubungan fitur-fitur tersebut.

Pada pearson Correlation ini akan menghapus nilai korelasi dengan nilai lebih besar dari 0.50, dan menghasilkan 32 fitur, seperti gambar dibawah ini :

Gambar 4.8 Hasil pemilihan fitur

Berikut fitur terpilih dengan metode Pearson Correlation:

1. source_port
2. destination_port
3. protocol
4. flow_duration
5. total_backward_packets
6. total_length_of_fwd_packets
7. flow_iat_min
8. fwd_iat_min
9. bwd_iat_total
10. bwd_iat_min
11. fwd_psh_flags
12. bwd_psh_flags

13. fwd_urg_flags
14. bwd_urg_flags
15. bwd_packets/s
16. fin_flag_count
17. syn_flag_count
18. psh_flag_count
19. ece_flag_count
20. fwd_avg_bytes/bulk
21. fwd_avg_packets/bulk
22. fwd_avg_bulk_rate
23. bwd_avg_bytes/bulk
24. bwd_avg_packets/bulk
25. bwd_avg_bulk_rate
26. init_win_bytes_forward
27. init_win_bytes_backward
28. min_seg_size_forward
29. active_mean
30. active_min
31. simillarhttp
32. label

4.2. Normalisasi Atribut Label

Untuk normalisasi atribut label menggunakan *LabelEncoder()* dari scikit-learn. Atribut label yang dinormalisasikan dalam hal ini yaitu pada fitur label. Angka 0 merepresentasikan kelas Syn, sedangkan angka 1 merepresentasikan kelas BENIGN, untuk lebih jelasnya seperti pada Gambar 4.9.

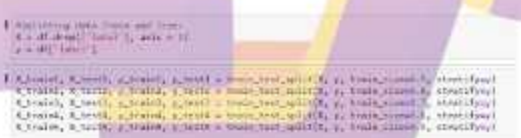
```
print("Sebelum Labelencoding: ", df_labels.unique())
print("Setelah Labelencoding: ", df['label'].unique())

Sebelum Labelencoding:  ['Syn' 'BENIGN']
Setelah Labelencoding:  [0 1]
```

Gambar 4.9. Label Encoding

4.3. Splitting Data Training dan Data Testing

Sebelum membagi data training dan data testing, dilakukan penghapusan fitur label untuk variabel X. sedangkan variabel y hanya menggunakan fitur label. Parameter *Stratify* juga digunakan untuk pembagian data sesuai dengan porsi antara label 0 dan 1 sesuai dengan ukuran data train. Pembagian data training dan data testing dapat dilihat pada Gambar 4.10 dibawah ini :



Gambar 4.10. Splitting Data Training dan Data Testing

Target analisis dapat dilihat pada Gambar 4.11 dan Gambar 4.12, dimana kelas 0 sebanyak 268224 dan kelas 1 sebanyak 318

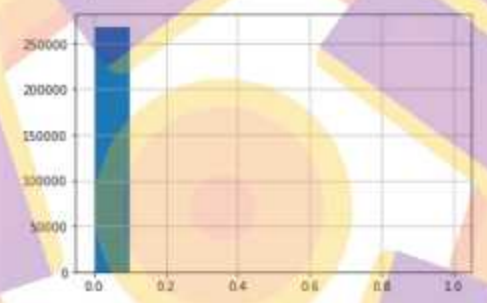
```
#Target Analysis:
y.value_counts()
```

```
0    268224
1       318
Name: label, dtype: int64
```

Gambar 4.11. Target Analisis

```
y.hist()
```

```
<AxesSubplot:>
```



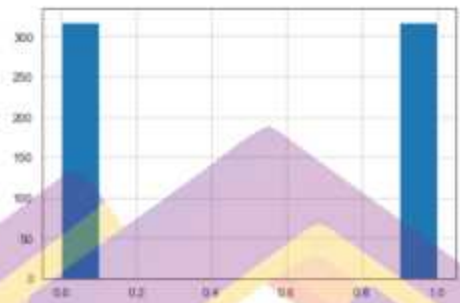
Gambar 4.12. Jumlah Data tiap kelas

4.4. Implementasi Seleksi Fitur

Resampling adalah proses sampling ulang terhadap sebuah data. Resampling bertujuan utnuk mengatasi ketidakseimbangan kelas pada data (*class imbalance)*, dimana dataset pada setiap kategori tidak seimbang, atau lebih dominan terhadap salah satu kategori. Pada penelitian ini menggunakan fungsi undersampling dengan mengurangi sampling data pada kelas 0 diseimbangkan dengan kelas 1, seperti pada Gambar 4.13 Undersampling Data

```
Y_bal.hist()
```

```
<AxesSubplot:>
```



Gambar 4.13. Undersampling Data

4.5. Pengujian dan Evaluasi Klasifikasi Menggunakan Model Gaussian NB

Pada skenario ini akan diujikan klasifikasi menggunakan model GaussianNB. Pembagian porsi data training dan data testing dibagi menjadi 5 skenario yaitu porsi data training 90%, data training 80%, data training 70%, data training 60%, data training 50%.

4.5.1. Data Training 90% dan Data Testing 10%

```
report Gaussian 90:
              precision    recall  f1-score   support

           0       0.84      1.00      0.91        32
           1       1.00      0.81      0.90        32

    accuracy                           0.91        64
   macro avg       0.92      0.91      0.91        64
weighted avg       0.92      0.91      0.91        64

conf_m_1:
[[32  0]
 [ 6 26]]

score_1: 0.90625
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 90,62%, *precission* sebesar 92%, *recall* sebesar 91% dan *f1-score* sebesar 91%.

4.5.2. Data Training 80% dan Data Testing 20%

```
report Gaussian 80:
              precision    recall  f1-score   support

           0       0.81      1.00      0.90        64
           1       1.00      0.77      0.87        64

    accuracy                           0.88       128
   macro avg       0.91      0.88      0.88       128
weighted avg       0.91      0.88      0.88       128

conf_m_2:
[[64  0]
 [15 49]]

score_2: 0.8828125
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 88,25%, *precission* sebesar 91%, *recall* sebesar 88% dan *f1-score* sebesar 88%.

4.5.3. Data Training 70% dan Data Testing 30%

```
report Gaussian 70:
              precision    recall  f1-score   support

           0       0.78      1.00      0.88        82
           1       1.00      0.79      0.88       109

    accuracy                           0.88       191
   macro avg       0.89      0.89      0.88       191
weighted avg       0.91      0.88      0.88       191

conf_m_3:
[[82  0]
 [23 86]]

score_3: 0.8795811518324608
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 87,95%, *precission* sebesar 89%, *recall* sebesar 89% dan *f1-score* sebesar 88%.

4.5.4. Data Training 60% dan Data Testing 40%

```
report Gaussian 60:
              precision    recall  f1-score   support

           0       0.76      0.98      0.86       126
           1       0.98      0.70      0.81       129

    accuracy                           0.84       255
   macro avg       0.87      0.84      0.84       255
weighted avg       0.87      0.84      0.84       255

conf_m_4:
[[124   2]
 [ 39  90]]

score_4: 0.8392156862745098
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 83,92%, *precission* sebesar 87%, *recall* sebesar 84% dan *f1-score* sebesar 84%.

4.5.5. Data Training 50% dan Data Testing 50%

```
report Gaussian 50:
              precision    recall  f1-score   support

           0       0.80      0.99      0.88       159
           1       0.98      0.75      0.85       159

    accuracy                           0.87       318
   macro avg       0.89      0.87      0.87       318
weighted avg       0.89      0.87      0.87       318

conf_m_5:
[[157   2]
 [ 39 120]]

score_5: 0.8710691823899371
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 87,10%, *precission* sebesar 89%, *recall* sebesar 87% dan *f1-score* sebesar 87%.

Berdasarkan hasil pengujian dan evaluasi algoritma Naïve Bayes dengan menggunakan model GaussianNB dapat dilihat pada Tabel 4.1

Tabel 4.1. Hasil Pengujian Model GaussianNB

Splitting Data	GaussianNB			
	Precission	Recall	F1-Score	Accuracy
90%	92%	91%	91%	90,62%
80%	91%	88%	88%	88,25%
70%	89%	89%	88%	87,95%
60%	87%	84%	84%	83,92%
50%	89%	87%	87%	87,10%

4.6. Model BernoulliNB

Pada skenario ini akan diujikan klasifikasi menggunakan model BernoulliNB. Pembagian porsi data training dan data testing dibagi menjadi 5 skenario yaitu porsi data training 90%, data training 80%, data training 70%, dan data training 60%.

4.6.1. Data Training 90% dan Data Testing 10%

```
report Bernoulli 90:
              precision    recall  f1-score   support

           0       0.90      0.84      0.87        32
           1       0.85      0.91      0.88        32

    accuracy                           0.88        64
   macro avg       0.88      0.88      0.87        64
weighted avg       0.88      0.88      0.87        64

conf_m_1:
[[27  5]
 [ 3 29]]

score_1: 0.875
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 87,50%, *precission* sebesar 88%, *recall* sebesar 88% dan *f1-score* sebesar 87%.

4.6.2. Data Training 80% dan Data Testing 10%

```
report Bernoulli 80:
              precision    recall  f1-score   support

           0       0.87      0.83      0.85        64
           1       0.84      0.88      0.85        64

    accuracy                           0.85       128
   macro avg       0.85      0.85      0.85       128
weighted avg       0.85      0.85      0.85       128

conf_m_2:
[[53 11]
 [ 8 56]]

score_2: 0.8515625
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 85,15%, *precission* sebesar 85%, *recall* sebesar 85% dan *f1-score* sebesar 85%.

4.6.3. Data Training 70% dan Data Testing 30%

```
report Bernoulli 70:
              precision    recall  f1-score   support

           0       0.90      0.88      0.89        82
           1       0.91      0.93      0.92       109

    accuracy                           0.91       191
   macro avg       0.90      0.90      0.90       191
weighted avg       0.91      0.91      0.91       191

conf_m_3:
[[ 72  10]
 [  8 101]]

score_3: 0.9057591623036649
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 90,57%, *precission* sebesar 90%, *recall* sebesar 90% dan *f1-score* sebesar 90%.

4.6.4. Data Training 60% dan Data Testing 40%

```
report Bernoulli 60:
              precision    recall  f1-score   support

           0       0.84      0.82      0.83       126
           1       0.83      0.84      0.84       129

    accuracy                           0.83       255
   macro avg       0.83      0.83      0.83       255
weighted avg       0.83      0.83      0.83       255

conf_m_4:
[[103  23]
 [ 20 109]]

score_4: 0.8313725490196079
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 83,13%, *precission* sebesar 83%, *recall* sebesar 83% dan *f1-score* sebesar 83%.

4.6.5. Data Training 50% dan Data Testing 50%

```
report Bernoulli 50:
              precision    recall  f1-score   support

           0       0.88      0.82      0.85       159
           1       0.83      0.89      0.86       159

    accuracy                           0.85       318
   macro avg       0.85      0.85      0.85       318
weighted avg       0.85      0.85      0.85       318

conf_m_5:
[[130  29]
 [ 18 141]]

score_5: 0.8522012578616353
```

Dari hasil evaluasi yang dilakukan diperoleh *accuracy* sebesar 85,22%, *precission* sebesar 85%, *recall* sebesar 85% dan *f1-score* sebesar 85%.

Berdasarkan hasil pengujian dan evaluasi algoritma Naïve Bayes dengan menggunakan model GaussianNB dapat dilihat pada Tabel 4.2

Tabel 4.2. Hasil Pengujian Model GasussianNB

Splitting Data	BernoulliNB			
	Preccision	Recall	F1-Score	Accuracy
90%	88%	88%	87%	87,50%
80%	85%	85%	85%	85,15%
70%	90%	90%	90%	90,57%
60%	83%	83%	83%	83,13%
50%	85%	85%	85%	85,22%

4.7. Analisis Hasil

Berdasarkan hasil pengujian menggunakan kedua model dan besaran akurasi untuk pemodelan klasifikasi adanya informasi serangan DDoS dari Syn Flood pada jaringan komputer dapat disimpulkan sebagai berikut:

Tabel 4.3. Hasil Analisis

Model NB/ Splitting	GaussianNB			
	Precission	Recall	F1-Score	Accuracy
90%	92%	91%	91%	90,62%

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa algoritma Naïve Bayes mampu mendeteksi serangan DDoS dengan baik. Model GaussianNB diperoleh dengan pembagian data training 90% dan data testing 30%, yaitu accuracy sebesar 90,62%, precission 92%, recall 91% dan f1-score sebesar 91%. Namun pada model BernoulliNB hasil pengujian terbaik pada pembagian data training 70% dan data testing 30%, yaitu accuracy sebesar 90,57%, precission 90%, recall 90% dan f1-score sebesar 90%. Dari Percobaan diatas dapat disimpulkan bahwa pembagian data berpengaruh terhadap nilai akurasi, bagaimana algoritma mempelajari data latih.

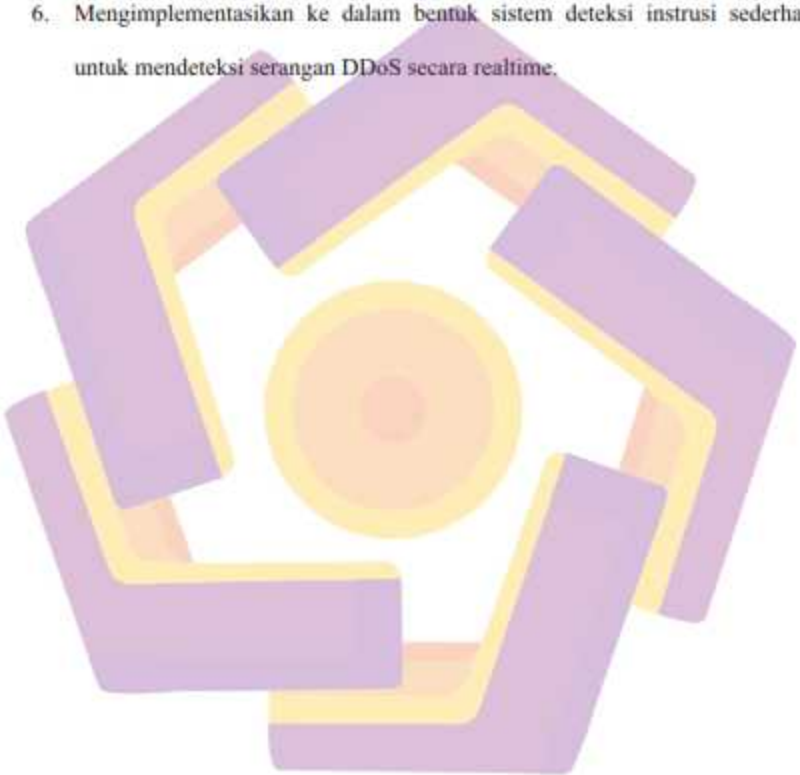
Dari Percobaan kedua model Naïve Bayes yaitu GaussianNB dan BernoulliNB tingkat akurasi hasil deteksi serangan DDoS jenis Syn Flood diperoleh nilai terbaik pada model GaussianNB yaitu accuracy sebesar 90,62%, precission 92%, recall 91% dan f1-score sebesar 91%.

5.2. Saran

Peneliti menyadari bahwa penelitian ini masih belum sempurna. Dengan demikian perlu adanya pengembangan untuk mendapatkan hasil yang lebih baik. Adapun beberapa saran dari peneliti antara lain:

1. Melakukan pembuktian dengan seleksi fitur yang berbeda

2. Melakukan pembuktian dengan pendekatan yang berbeda
3. Mengoptimalkan waktu komputasi dengan memanfaatkan algoritma lain.
4. Melakukan percobaan dengan data set yang berbeda.
5. Memvisualisaikan hasil ke dalam GUI seperti website atau lainnya.
6. Mengimplementasikan ke dalam bentuk sistem deteksi instrusi sederhana untuk mendeteksi serangan DDoS secara realtime.



DAFTAR PUSTAKA

PUSTAKA MAJALAH, JURNAL ILMIAH ATAU PROSIDING

A. a, Alfantookh, "DoS Attacks Intelligent Detection using Neural Networks," J. King Saud Univ. -Comput. Inf. Sci., Vol. 18, no. 2006, pp. 31-51, 2006.

A. Andhare, P. Arvind, and B. Patil, "Denial-of-Service Attack Detection Using GeneticBased Algorithm," vol. 2, no. 2, pp. 94–98, 2012.

A. H. Lashkari, Y. Zang, G. Owhuo, M. S. I. Mamun, and G. D. Gil, "CICFlowMeter," Github. 2017.

A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, no. 1, p. 20, 2019.

A. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "An Evaluation Framework for Intrusion Detection Dataset," in ICISS 2016 - 2016 International Conference on Information Science and Security, 2017, pp. 1–6.

A. Nisioti, A. Mylonas, P. D. Yoo, and V. Katos, "From intrusion detection to attacker attribution: A comprehensive survey of unsupervised methods," IEEE Commun. Surv. Tutorials, vol. 20, no. 4, pp. 3369–3388, 2018.

Alfine Ridho, M, Molavi Arman, "Analisis Serang DDoS Menggunakan Metode Jaringan Saraf Tiruan", Sisfokom, Palembang, vol. 09. No. 03, PP 373-379, 2020.

Alfa Saleh, Implementasi Metode Klasifikasi Naïve Bayes Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga , Yogyakarta, Citec Journal, Vol. 2, No. 3, Mei 2015 – Juli 2015.

Al Riza Khadafy, Romi Satria Wahono, "Penerapan Naive Bayes untuk Mengurangi Data Noise pada Klasifikasi Multi Kelas dengan Decision Tree", Jakarta, Journal of Intelligent Systems, Vol. 1, No. 2, December 2015.

Arif Wirawan Muhammad, Cik Feresa Mohd Foozy, Ahmad Azhari, "Machine Learning-Based Distributed Denial of Service Attack Detection on Intrusion Detection System Regarding to Feature Selection", New South Wales, vol. 4, no. 1, pp. 01-08, 2020.

Aziz Muhammad, Rusydi Umar, Faizin Ridho, "Implementasi Jaringan Saraf Tiruan Untuk Untuk Mendeteksi Serang DDoS Pada Forensik Jaringan", Yogyakarta, Vol. 3, No. 01, 2019.

C. Canongia and R. Mandarino, "Cybersecurity: The new challenge of the information society," in Handbook of Research on Business Social Networking: Organizational, Managerial, and Technological Dimensions, 2011.

C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: Classification and state-of-the-art," Comput. Networks, vol. 44, no. 5, pp. 643–666, 2004.

C. Thomas, V. Sharma, and N. Balakrishnan, "Usefulness of DARPA dataset for intrusion detection system evaluation," in Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2008, 2008.

Chris Jordan Sihombing Jodi, Dany Primanita Kartikasari, Adhitya Bhawiyuga, "Implementasi Sistem Deteksi dan Mitigasi Serangan Distributed Denial of Service (DDoS) menggunakan SVM Classifier pada Arsitektur Software Defined Network (SDN)", Malang, Vol. 3, No. 10, halm. 9608-9613, 2019.

D. M. Farid, N. Harbi, E. Bahri, M. Z. Rahman, and C. M. Rahman, "Attacks classification in adaptive intrusion detection using decision tree," World Acad. Sci. Eng. Technol., pp. 368–372, 2010.

D. M. Powers, "Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation," J. Mach. Learn. Technol., vol. 2, 2007.

Dong Shi, Khushnood Abbas, Raj Jain, "A Survey on Distributed Denial of Service (DDoS) Attact in SDN and Cloud Computing Environments", India, Vol. 7, pp. 80813-80828, 2019.

E. D. Meutia, J. Teknik, E. Universitas, and S. Kuala, "Internet of Things–Keamanan dan Privasi," in *Seminar Nasional dan Expo Teknik Elektro ISSN*, 2015, pp. 2088–9984.

F. Beer, T. Hofer, D. Karimi, and U. Bühler, "A new attack composition for network security," in Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI), 2017.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, and B. Thirion, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.

Fitriyani,Romi Satria Wahono, "Integrasi Bagging dan Greedy Forward Selection pada Prediksi Cacat Software dengan Menggunakan Naïve Bayes", Jakartaa, Journal of Software Engineering, Vol. 1, No. 2, December 2015.

G. C. Kessler and D. E. Levin, Denial-of-Service Attacks, 4th ed. John Wiley & Sons, 2015.

Hall, M.A.: *Correlation-based feature selection for discrete and numeric class machine learning. In Proceedings of the 17th Intl.* Conf. Machine Learning (2000) 359-366.

I. Alsmadi and D. Xu, "Security of Software Defined Network: A Survey, " Comput. Secur., vol. 53, pp. 79-108, 2015.

I. Sofi, A. Mahajan, and V. Mansotra, "Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks," Int. Res. J. Eng. Technol., 2017.

Iman Sharafaldin, Arash Habibi Lashkari, Saqib Hakak, and Ali A. Ghorbani, "Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy", IEEE 53rd International Carnahan Conference on Security Technology, Chennai, India, 2019

J. B. Fraley and J. Cannady, "The promise of machine learning in cybersecurity," in SouthEastCon 2017, 2017, pp. 1–6.

J. Mchugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," ACM Trans. Inf. Syst. Secur., vol. 3, no. 4, pp. 1094–9224, 2000.

J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," Comput. Commun. Rev., vol. 34, no. 2, pp. 39–53, 2004.

J. O. Nehinbe, "A critical evaluation of datasets for investigating IDSs and IPSs researches," in Proceedings of 2011, 10th IEEE International Conference on Cybernetic Intelligent Systems, CIS 2011, 2011, pp. 1–6.

J. O. Nehinbe, "A simple method for improving intrusion detections in corporate networks," in Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, 2010.

J. Heidemann and C. Papadopoulos, "Uses and challenges for network datasets," in Proceedings - Cybersecurity Applications and Technology Conference for Homeland Security, CATCH 2009, 2009.

Kurniabudi, Abdul Harris, Abdul Rahim, "Seleksi Fitur dengan Information Gain untuk Meningkatkan Deteksi Serangan DDoS Menggunakan Random Forest", Jambi, Vol. 19, No. 1, halm. 56-66, 2020

K. Hengst, "DDoS through the Internet of Things," pp. 1-9, 2016.

K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS), National Institute of Standards and Technology. Special Publication (NIST SP), 2007.

Kusrini dan Taufiq Lutfi, Emha. (2009). "*Algoritma Data Mining.*" Yogyakarta: Andi.

L. Dhanabal and S. P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," Int. J. Adv. Res. Comput. Commun. Eng., vol. 4, no. 6, 2015.

L. Rokach, "Ensemble-based classifiers," Artif. Intell. Rev., vol. 33, pp. 1–39, 2010.

Lila Dini Utami, Romi Satria Wahono, "Integrasi Metode Information Gain Untuk Seleksi Fitur dan Adaboost Untuk Mengurangi Bias Pada Analisis Sentimen Review Restoran Menggunakan Algoritma Naïve Bayes", Jakarta, Journal of Intelligent Systems, Vol. 1, No. 2, December 2015.

M. Aamir and S. M. A. Zaidi, "DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation," Int. J. Inf. Secur., pp. 1–25, 2019.

M. Ghorbani, Ali A., Lu, Wei, Tavallaee, Network Intrusion Detection and Prevention. Springer, 2010.

M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," IEEE Commun. Surv. Tutorials, vol. 16, no. 1, pp. 303–336, 2014.

M. Małowidzki, P. Berezi, and M. Mazur, "Network Intrusion Detection: Half a Kingdom for a Good Dataset," in ECCWS 2017 16th European Conference on Cyber Warfare and Security, 2017.

M. Mohri, A. Rostamizadeh, and A. Talwalkar, Foundations of Machine Learning, 2nd ed. London, England: The MIT Press, 2018.

M. Ring, S. Wunderlich, D. Grūdl, D. Landes, and A. Hotho, "Flow-based benchmark data sets for intrusion detection," in European Conference on Information Warfare and Security, ECCWS, 2017.

M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of networkbased intrusion detection data sets," Comput. Secur., vol. 86, pp. 147–167, 2019.

M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "Detailed Analysis of the KDD CUP 99 Data Set," Submitted to Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), 2009.

M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, "DDoS attack detection using machine learning techniques in cloud computing environments," in Proceedings of 2017 International Conference of Cloud Computing Technologies and Applications, CloudTech 2017, 2018.

N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in 2015 Military Communications and Information Systems Conference, MilCIS 2015 - Proceedings, 2015.

N. Sharma, A. Mahajan, and V. Mansotra, "Machine Learning Techniques Used in Detection of DOS Attacks: A Literature Review," Int. J. Adv. Res. Comput. Sci. Softw. Eng., 2016.

O. Yavanoglu and M. Aydos, "A review on cyber security datasets for machine learning algorithms," in Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017, 2017, pp. 2186–2193.

OWASP, "OWASP Top 10 - 2017 - The Ten Most Critical Web Application Security Risks," Top 10 2017, 2017.

P. Twomey, "Cyber Security Threats." The Lowy Institute for International Policy, Sydney, 2010.

R. Bace and P. Mell, "NIST special publication on intrusion detection systems," Special Publication (NIST SP), 2001.

R. Koch, "Towards next-generation intrusion detection," in 2011 3rd International Conference on Cyber Conflict, ICCC 2011 - Proceedings, 2011.

R. Das and T. H. Morris, "Machine learning and cyber security," in 2017 International Conference on Computer, Electrical and Communication Engineering, ICCECE 2017, 2018, pp. 1–7.

R. Sokullu, "GTS Attack : An IEEE 802 . 15 . 4 MAC Layer Attack in Wireless Sensor Networks," Int. J., vol. 2, no. 1, pp. 105–116, 2009.

R. Von Solms and J. Van Niekerk, "From information security to cyber security," Comput. Secur., vol. 38, pp. 97–102, 2013.

R. Wirth, "CRISP-DM : Towards a Standard Process Model for Data Mining," Proc. Fourth Int. Conf. Pract. Appl. Knowl. Discov. Data Min., pp. 29–39, 2000.

S. Dua and X. Du, Data Mining and Machine Learning in Cybersecurity. Boca Raton, Florida: Auerbach Publications, 2016.

S. K. Sahu, S. Sarangi, and S. K. Jena, "A detail analysis on intrusion detection datasets," in Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014, 2014, pp. 1348–1353.

S. Nadila, Y. Galuh, S. Fatia, K. Fahmi Hayati Holle, Deteksi Serangan Distributed Denial of Services (DDOS) Berbasis HTTP Menggunakan Metode Fuzzy Sugeno, JISKa, Vol. 4, No. 3, Pp. 156 – 164, Januari 2020

S. Raza, L. Wallgren, and T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things," Ad Hoc Networks, vol. 11, no. 8, pp. 2661–2674, 2013.

Scikit-learn, "StratifiedKFold," Scikit-learn 0.22.2 Documentation, 2019.

Sukmawati Anggraini Putri, Romi Satria Wahono," Integrasi SMOTE dan Information Gain pada Naive Bayes untuk Prediksi Cacat Software", Jakarta, Journal of Software Engineering, Vol. 1, No. 2, December 2015.

T. H. Morris, Z. Thornton, and I. Turnipseed, "Industrial Control System Simulation and Data Logging for Intrusion Detection System Research," Seventh Annu. Southeast. Cyber Secur. Summit, 2015.

T. Mitchell, Machine Learning. Burr Ridge, IL: McGraw Hill, 1997.

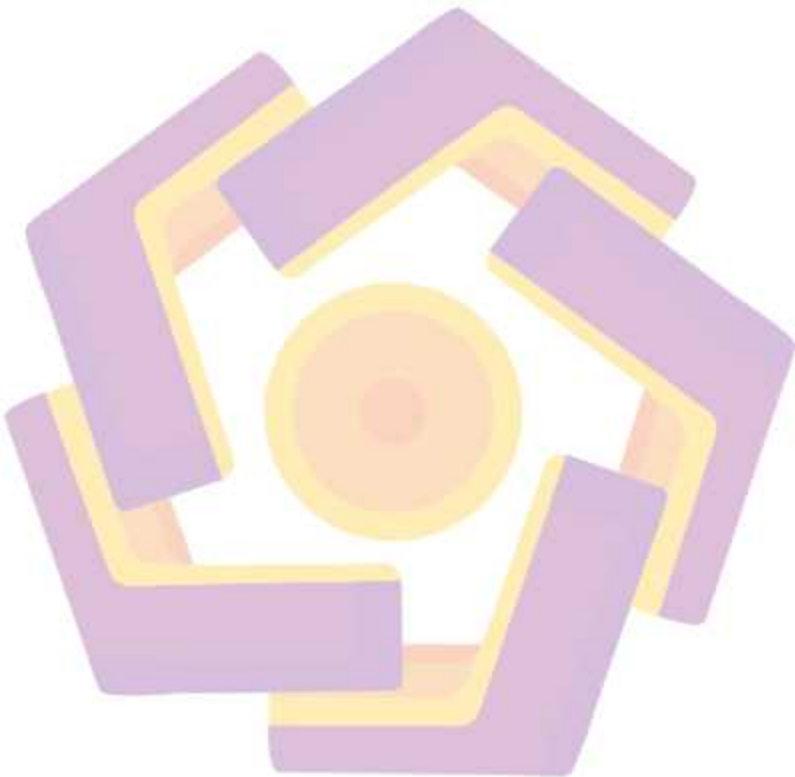
The Cooperative Association for Internet Data Analysis, "CAIDA - The Cooperative Association for Internet Data Analysis," CAIDA. 2010.

University Of California, "KDD-Cup Dataset '99," The UCI KDD Archive, 1999.

University Of California, "KDD-Cup Dataset '98," The UCI KDD Archive, 1998.

V. Hema and C. E. Shyni, "DoS Attack Detection Based on Naive Bayes Classifier," Middle-East J. Sci. Red. Signal Process. Secur., Vol, 23, pp. 398- 405, 2015.

Y. C. Wu, H. R. Tseng, W. Yang, and R. H. Jan, "DDoS detection and traceback with decision tree and grey relational analysis," in 3rd International Conference on Multimedia and Ubiquitous Engineering, MUE 2009, 2009.



PUSTAKA ELEKTRONIK

Haldi Widianto, Mochammad, 23 Desember 2019, Algoritma Naïve Bayes, https://binus.ac.id/bandung/2019/12/algoritma-naive-bayes/

Defense Advanced Research Projects Agency, "1999 DARPA Intrusion Detection Evaluation Dataset," 1999. [Online]. Available: https://www.ll.mit.edu/r-d/datasets/1999- darpa-intrusion-detection-evaluation-dataset.

Gobel, Tenri, 26 Oktober 2020, BSSN-HUAWEI CYBER SCOUT HUNT, https://cyberthreat.id/read/8970/Sulistyo-BSSN-Ungkap-Tiga-Jenis-Serangan-Siber-Dominan-di-Indonesia-Periode-Januari-Oktober-2020

Iman Sharafaldin, Arash Habibi Lashkari, Saqib Hakak, and Ali A. Ghorbani, "Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy", IEEE 53rd International Carnahan Conference on Security Technology, Chennai, India, 2019.

Nexus Guard, "Nexusguard Research Shows DNS Amplification Attacks Grew Nearly 4,800% Year-over-Year; Highlighted by Sharp Increase in TCP SYN Flood," 2019. [Online]. Available: https://www.nexusguard.com/newsroom/press-release/dns-amplificationattacks-rise-twofold-in-q1-0-0.

P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," RFC Editor, 2000. [Online]. Available: https://tools.ietf.org/html/rfc2827.

Pandas.pydaya.org, "Pandas.Dataframe.Fillna," Pandas 1.0.3 Documentation, 2014. [Online]. Available: https://pandas.pydata.org/pandasdocs/stable/reference/api/pandas.DataFrame.fillna.html.

Scikit-learn, "Train_test_split," Scikit-learn 0.22.2 Documentation, 2019. [Online]. Available: https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html.

Scikit-learn, "GaussianNB," scikit-learn.org, 2019. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html

University of New Brunswick, "DDoS Evaluation Dataset (CICDDoS2019)," unb.ca, 2019. [Online]. Available: https://www.unb.ca/cic/datasets/ddos-2019.html.

LAMPIRAN