TUGAS AKHIR - KI141502

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

AHMAD ISMAIL HARRY WICAKSONO
NRP 05111440000032

Dosen Pembimbing I
Tohari Ahmad, S.Kom., MIT., Ph.D.

Dosen Pembimbing II
-

Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

TUGAS AKHIR – KI141502

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

AHMAD ISMAIL HARRY WICAKSONO
NRP 05111440000032

Dosen Pembimbing I
Tohari Ahmad, S.Kom., MIT., Ph.D.

Dosen Pembimbing II
-

Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

(Halaman ini sengaja dikosongkan)

UNDERGRADUATE THESES – KI141502

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

AHMAD ISMAIL HARRY WICAKSONO
NRP 05111440000032

First Advisor
Tohari Ahmad, S.Kom., MIT., Ph.D.

Second Advisor
-

Department of Informatics
Faculty of Information and Communication Technology
Sepuluh Nopember Institute of Technology
Surabaya 2018

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

MENDETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Komputasi Berbasis Jaringan
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:
AHMAD ISMAIL HARRY WICAKSONO
NRP: 05111440000032

Disetujui oleh Pembimbing Tugas Akhir:

1. Tohari Ahmad, S.Kom., MIT., Ph.D. .....
   (NIP. 197505252003121002) (Pembimbing 1)

2. -
   (-) .....
   (Pembimbing 2)

SURABAYA
JULI, 2018

[Halaman ini sengaja dikosongkan]

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

Nama Mahasiswa : AHMAD ISMAIL HARRY WICAKSONO
NRP : 05111440000032
Jurusan : Teknik Informatika FTIK-ITS
Dosen Pembimbing 1 : Tohari Ahmad, S.Kom., MIT., Ph.D.
Dosen Pembimbing 2 : -

Abstrak

Serangan DDoS (Distributed Denial of Service) adalah tipe serangan jaringan berskala besar dengan cara terkoordinasi, yang biasanya diluncurkan secara tidak langsung dengan memanfaatkan komputer lain di Internet yang biasa dinamakan botnet. DDoS bekerja dengan mengirimkan sejumlah paket data secara bersamaan untuk membuat resource target sistem terkuras habis untuk merespon paket data ini. Dengan banyaknya paket data yang masuk maka jaringan komputer akan mengalami *overload*.

Untuk mengatasi masalah tersebut, maka diperlukan metode untuk mendeteksi serangan DDoS yang diterima oleh jaringan komputer. Dalam tugas akhir ini, akan dilakukan klasifikasi paket data yang diterima menggunakan beberapa faktor dan parameter yang didapat pada sistem server yang menerima serangan tersebut. Beban setiap parameter akan didapat melalui data pembelajaran. Klasifikasi digunakan untuk mendeteksi apakah sedang terjadi serangan DDoS pada sistem atau tidak. Selanjutnya, jika terdeteksi sebagai serangan DDoS, maka selanjutnya jaringan komputer harus merubah beberapa konfigurasi pada server untuk mengatasi serangan DDoS ini, seperti membatasi paket yang masuk,membatasi koneksi, menutup koneksi yang lambat, dan blacklist IP yang dicurigai sebagai penyerang.

Pada tugas akhir ini didapatkan algoritma yang dapat mendeteksi serangan DdoS berdasarkan klasifikasi paket yang masuk ke jaringan komputer. Klasifikasi dilakukan dengan melakukan analisa terhadap faktor internal maupun eksternal dari komputer tersebut. Sehingga terdapat beberapa parameter untuk menentukan klasifikasi paket tersebut serangan atau tidak.

Kata kunci: DDOS *Attack*, Keamanan Jaringan, Algoritma Klasifikasi, FTP *attack*.

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

Student's Name	**: AHMAD ISMAIL HARRY WICAKSONO**
Student's ID	**: 05111440000032**
Department	**: Teknik Informatika FTIK-ITS**
First Advisor	**: Tohari Ahmad, S.Kom., MIT., Ph.D.**
Second Advisor	**: -**

Abstract

DDoS attacks (Distributed Denial of Service) are types of large-scale network attacks in a coordinated way, usually launched indirectly by exploiting other computers on the Internet commonly called botnets. DDoS works by using packet data simultaneously to make the target system resource drained out to restore this packet data. With the incoming so many data packets then the computer network can be overload.

To overcome these problems, it is necessary method to detect DDoS attacks received by computer network. In this thesis, will be classified data packets received using several factors and parameters obtained on the server system that receives the attack. The load of each parameter will be obtained through the learning data. Classification is used to detect whether or not a DDoS attack exists on the system. Furthermore, if detected as a DDoS attack, then the computer network must change some configuration on the server to overcome this DDoS attack, such as restrict incoming packets, restrict connections, close slow connections, and blacklist IP suspected as the attacker.

In this thesis we get an algorithm that can detect DDoS attacks based on the classification of packets that enter the computer network. Classification is done by analyzing the internal and external factors of the computer. So there are several

parameters to determine the classification of the packet attack or not.

Keywords: **DDOS** ***Attack, Network Security, Classification Algorithm,*** **FTP** ***attack.***

KATA PENGANTAR

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul "**MENDETEKSI SERANGAN DISTRIBUTED DENIAL OF SERVICE (DDOS) PADA JARINGAN KOMPUTER UNTUK HONEYPOT YANG DINAMIS**". Tugas Akhir ini merupakan salah satu syarat dalam menempuh ujian sidang guna memperoleh gelar Sarjana Komputer. Dengan pengerjaan Tugas Akhir ini, penulis bisa belajar banyak untuk memperdalam dan meningkatkan apa yang telah didapatkan penulis selama menempuh perkuliahaan di Teknik Informatika ITS.

Selesainya Tugas Akhir ini tidak terlepas dari bantuan dan dukungan beberapa pihak, sehingga pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Keluarga penulis yang selalu memberikan dukungan doa, moral, dan material yang tak terhingga kepada penulis sehingga penulis dapat menyelesaikan Tugas Akhir ini.
3. Bapak Tohari Ahmad, S.Kom., MIT., Ph.D. selaku dosen pembimbing penulis yang telah membimbing dan memberikan motivasi, nasehat dan bimbingan dalam menyelesaikan tugas akhir ini.
4. Bapak Darlis Herumurti, S.Kom., M.Kom. selaku kepala jurusan Teknik Informatika ITS.
5. Seluruh dosen dan karyawan Teknik Informatika ITS yang telah memberikan ilmu dan pengalaman kepada penulis selama menjalani masa studi di ITS.
6. Pramesti Widyayanti yang telah banyak meluangkan waktu dan membantu penulis selama berkuliah.

7. Teman-teman seperjuangan RMK NCC/KBJ, yang telah menemani dan menyemangati penulis.
8. Teman-teman angkatan 2014, yang sudah mendukung penulis selama perkuliahan.
9. Sahabat penulis yang tidak dapat disebutkan satu per satu yang selalu membantu, menghibur, menjadi tempat bertukar ilmu dan berjuang bersama-sama penulis.

Penulis menyadari bahwa Tugas Akhir ini masih memiliki banyak kekurangan sehingga dengan kerendahan hati penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan ke depan.

Surabaya, Juni 2018

DAFTAR ISI

DAFTAR GAMBAR

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

[Halaman ini sengaja dikosongkan]

BAB I
PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi melaju sangat pesat dan digunakan dalam berbagai aspek kehidupan kita. Hampir seluruh perusahaan menggunakan teknologi informasi untuk bertukar data dan saling bertukar informasi. Namun, kemajuan teknologi informasi ini tidak diikuti dengan perkembangan keamanan pada jaringan komputer. Dalam tahun ini saja, terdapat laporan serangan DDoS pada berbagai wilayah industri. Pada pertengahan April, Melbourne IT, serta dua anak perusahaannya Netregistry dan TPP Wholesale, mengalami serangan DDoS. Serangan tersebut dimulai pada pukul 10:00 waktu setempat, memaksa organisasi tersebut untuk menginformasikan kepada pelanggan bahwa cloud hosting dan mailing platform mereka, dan layanan lainnya saat ini tidak tersedia. Kemudian, pada pertengahan Agustus, Blizzard Entertainment melaporkan flood junk traffic yang menyebabkan masalah bagi pemain Overwatch dan World of Warcraft. Serangan lain terjadi pada bulan Septermber yang menyerang UK Lottery atau situs undian di Inggris. DdoS menyerang website www.national-lottery.co.uk dan aplikasi mobile nya. Sehingga warga tidak bisa bermain undian tanpa mengunjungi retailer partner untuk membeli tiket. [1] [2]

Serangan DDoS (Distributed Denial of Service) adalah tipe serangan jaringan berskala besar dengan cara terkoordinasi, yang biasanya diluncurkan secara tidak langsung dengan memanfaatkan komputer lain di Internet yang biasa dinamakan botnet. DDoS bekerja dengan mengirimkan sejumlah paket data secara bersamaan untuk membuat resource target sistem terkuras habis untuk merespon paket data ini. Dengan banyaknya paket data yang masuk maka jaringan komputer akan mengalami overload.

Untuk mengatasi masalah tersebut, maka diperlukan metode untuk mendeteksi serangan DDoS yang diterima oleh jaringan komputer. Dalam tugas akhir ini, akan dilakukan klasifikasi paket data yang diterima menggunakan beberapa faktor dan parameter yang didapat pada sistem server yang menerima serangan tersebut. Beban setiap parameter akan didapat melalui data pembelajaran. Klasifikasi digunakan untuk mendeteksi apakah sedang terjadi serangan DDoS pada sistem atau tidak. Selanjutnya, jika terdeteksi sebagai serangan DDoS, maka selanjutnya jaringan komputer harus merubah beberapa konfigurasi pada server untuk mengatasi serangan DDoS ini, seperti membatasi paket yang masuk, membatasi koneksi, menutup koneksi yang lambat, dan blacklist IP yang dicurigai sebagai penyerang.

1.2 Rumusan Masalah

Tugas akhir ini mengangkat beberapa rumusan masalah sebagai berikut:

1. Bagaimana cara mendeteksi serangan DDoS pada suatu jaringan komputer?
2. Bagaimana perbandingan performa algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer dengan algoritma klasifikasi KNN, SVM, dan ANN?

1.3 Batasan Permasalahan

Permasalahan yang dibahas pada tugas akhir ini memiliki batasan sebagai berikut:

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan antara lain:

1. Algoritma yang digunakan untuk mengklasifikasi adalah algoritma dengan rumus pembobotan yang dianalisa dengan parameter yang didapat.
2. Sistem Operasi yang digunakan adalah Ubuntu 16.04.
3. Tipe serangan DDoS yang digunakan berupa FTP Attack dengan tools THC-Hydra.

4. Mesin maksimal yang digunakan untuk melakukan serangan DDoS berjumlah 2. Total jumlah maksimal paket yang dikirimkan adalah 1000.
5. Arsitektur jaringan komputer menggunakan FTP server dengan tools vsftpd.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah sebagai berikut:

1. Mendeteksi serangan DDoS yang terjadi pada jaringan komputer.
2. Menganalisa perbandingan performa algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer dengan algoritma klasifikasi KNN, SVM, dan ANN.

1.5 Manfaat

Manfaat dari hasil pembuatan tugas akhir ini adalah sebagai berikut:

1. Mempermudah prediksi serangan DDoS
2. Mendapatkan algoritma yang memiliki performa yang baik dalam mendeteksi DDoS.
3. Membuat suatu jaringan komputer yang dapat merespon serangan DDoS dan menanggulanginya
4. Menerapkan ilmu yang dipelajari selama kuliah di Teknik Informatika ITS agar dapat berguna bagi orang banyak.

1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagi berikut:

1.6.1 Penyusunan Proposal Tugas Akhir

Tahap awal tugas akhir ini adalah menyusun proposal tugas akhir. Pada proposal, berisi tentang deskripsi pendahuluan dari

tugas akhir yang akan dibuat. Proposal juga berisi tentang garis besar tugas akhir yang akan dikerjakan sehingga memberikan gambaran untuk dapat mengerjakan tugas akhir sesuai dengan *timeline* yang dibuat. Gagasan untuk mengatasi serangan DdoS dengan Algoritma yang digunakan untuk mengklasifikasi adalah algoritma dengan rumus pembobotan yang dianalisa dengan parameter yang didapat.

1.6.2 Studi Literatur

Pada tahap ini dilakukan untuk mencari informasi dan studi literatur apa saja yang dapat dijadikan sebagai referensi untuk membantu pengerjaan tugas akhir ini. Tahap ini merupakan tahap untuk memahami semua metode yang akan dikerjakan, sehingga memberi gambaran selama pengerjaan tugas akhir. Informasi didapatkan dari buku dan literatur yang berhubungan dengan metode yang digunakan. Informasi yang dicari adalah keamanan jaringan, DDoS *Attack*, Hoenypot, THC Hydra untuk *tools* penyerang, dan bahasa *python* untuk pemrograman.

1.6.3 Implementasi Sistem

Implementasi merupakan tahap untuk mengimplementasikan metode-metode yang sudah diajukan pada proposal Tugas Akhir. Untuk membangun algoritma yang telah dirancang sebelumnya, implementasi dilakukan dengan menggunakan *THC Hydra* sebagai *Attacker* yang akan mengirimkan paket secara *brute-force*, bahasa *python* sebagai bahasa pemrograman, dan *Numpy, SciPy* dan *Sklearn* sebagai *library* untuk mengimplementasikan algoritma klasifikasi.

1.6.4 Pengujian dan Evaluasi

Pada tahap ini algoritma yang telah disusun diuji coba dengan melakukan klasifikasi terhadap paket yang masuk dalam jumlah tertentu. Kemudian dari skenario tersebut akan didapatkan

data akurasi dan ketepatan algoritma dalam melakukan klasifikasi terhadap paket yang masuk. Dari data tersebut, dilakukan analisa untuk membandingkan performa algoritma yang diusulkan dengan metode klasifikasi yang lain seperti KNN, SVM, dan ANN.

1.6.5 Penyusunan Buku

Pada tahap ini disusun buku sebagai dokumentasi dari pelaksanaan tugas akhir yang mencangkup seluruh konsep, teori, implementasi, serta hasil yang telah dikerjakan.

1.7 Sistematika Penulisan Laporan

Sistematika penulisan laporan tugas akhir adalah sebagai berikut:

1. Bab I. Pendahuluan
   Bab ini berisikan penjelasan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi, dan sistematika penulisan dari pembuatan tugas akhir.
2. Bab II. Tinjauan Pustaka
   Bab ini berisi kajian teori dari metode dan algoritma yang digunakan dalam penyusunan tugas akhir ini. Secara garis besar, bab ini berisi tentang Keamanan Jaringan, Serangan DDoS, THC Hydra, *Python*, *SciPy*, dan *Sklearn*.
3. Bab III. Perancangan Perangkat Lunak
   Bab ini berisi pembahasan mengenai perancangan dari metode yang di usulkan untuk mendeteksi serangan DDoS pada FTP Server dengan menganalisa paket yang masuk apakah mempengaruhi *resource* atau tidak.
4. Bab IV. Implementasi
   Bab ini menjelaskan implementasi metode yang di usulkan untuk mendeteksi serangan DDoS pada FTP Server dengan menganalisa paket yang masuk apakah mempengaruhi *resource* atau tidak.
5. Bab V. Pengujian dan Evaluasi

Bab ini berisikan hasil uji coba dari implementasi metode yang di usulkan untuk mendeteksi serangan DDoS pada FTP Server dengan menganalisa paket yang masuk apakah mempengaruhi *resource* atau tidak. Pengujian dilakukan dengan klasifikasi terhadap paket yang masuk dalam jumlah tertentu. Kemudian dari skenario tersebut akan didapatkan data akurasi dan ketepatan algoritma dalam melakukan klasifikasi terhadap paket yang masuk. Dari data tersebut, dilakukan analisa untuk membandingkan performa algoritma yang diusulkan dengan metode klasifikasi yang lain seperti KNN, SVM, dan ANN.

6. Bab VI. Kesimpulan dan Saran
   Bab ini merupakan bab yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan, masalah-masalah yang dialami pada proses pengerjaan tugas akhir, dan saran untuk pengembangan solusi ke depannya.
7. Daftar Pustaka
   Bab ini berisi daftar pustaka yang dijadikan literatur dalam tugas akhir.
8. Lampiran
   Dalam lampiran terdapat kode program secara keseluruhan.

BAB II
TINJAUAN PUSTAKA

Bab ini berisi pembahasan mengenai teori-teori dasar yang digunakan dalam tugas akhir. Teori-teori tersebut diantaranya adalah *Distributed Denial of Service*, dan beberapa teori lain yang mendukung pembuatan tugas akhir. Penjelesan ini bertujuan untuk memberikan gambaran secara umum terhadap program yang dibuat dan berguna sebagai penunjang dalam pengembangan riset yang berkaitan.

2.1 Keamanan Jaringan

Keamanan jaringan adalah kegiatan yang dirancang untuk melindungi kegunaan dan integritas jaringan dan data Anda. Ini mencakup teknologi perangkat keras dan perangkat lunak. Keamanan jaringan yang efektif mengelola akses ke jaringan dan menargetkan berbagai ancaman dan menghentikan mereka ketika memasuki atau menyebar di jaringan Anda. [3]

Suatu jaringan bisa dikatakan aman apabila mengandung 3 faktor yang bisa dikenal dengan CIA (*Confidentiality, Integrity, Availability*). Aspek *confidentiality* atau kerahasiaan adalah aspek dalam keamanan jaringan yang membatasi akses terhadap informasi, dimana hanya orang-orang yang telah mendapatkan izin yang bisa mengakses informasi tertentu. Kemudian aspek *integrity* atau keutuhan, merujuk kepada tingkat kepercayaan terhadap suatu informasi, kepecayaan dalam hal ini mencakup akurasi dan konsistensi terhadap informasi yang ada. Oleh karena itu perlu adanya proteksi terhadap suatu informasi dari modifikasi oleh pihak-pihak yang tidak diizinkan. Aspek yang terakhir adalah aspek *availability*. Konsep availability dari suatu informasi berarti bahwa informasi tersebut selalu tersedia ketika dibutuhkan bagi orang-orang yang memiliki izin terhadap informasi tersebut. Sehingga ketika dibutuhkan oleh user, data/informasi dapat dengan cepat diakses dan digunakan. Konsep CIA bisa dilihat pada gambar 2.1

Gambar 2.1 Konsep CIA [4]

Salah satu serangan terhadap *availability* suatu informasi yang paling dikenal adalah Distributed Denial of Service (DDoS). Tujuan utama dari DDOS attack adalah untuk memenuhi resourse yang disediakan untuk user, sehingga user tidak bisa mengakses informasi yang seharusnya bisa didapatkan.

Agar 3 aspek keamanan jaringan ini terjaga, diperlukan sebuah algoritma klasifikasi untuk mengklasifikasi paket yang masuk pada jaringan komputer. Pada Tugas Akhir ini, penulis menggunakan algortima klasifikasi berdasarkan faktor internal dan eksternal untuk mendeteksi paket yang masuk, sehingga dapat mendeteksi serangan yang terjadi pada jaringan komputer. Implementasi serangan dilakukan menggunakan THC Hydra dan dilakukan pengujian performa dengan membandingkan dengan algoritma lain seperti KNN, SVM dan ANN dari segi akurasi dan ketepatan.

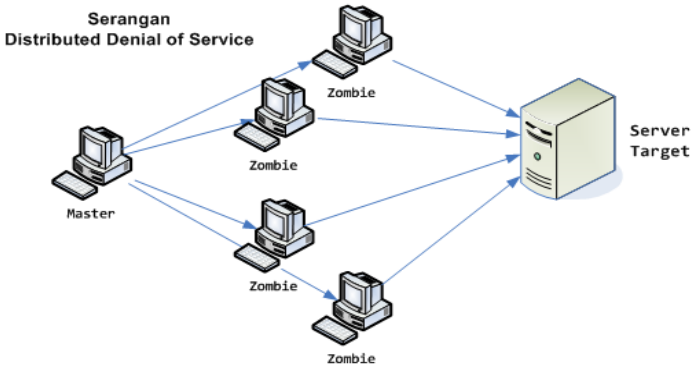
2.2 DDOS (Distributed Denial Of Service)

Serangan Distributed Denial of Service (DDoS) adalah serangan untuk mencegah pengguna untuk menggunakan segala sumber daya yang berasal dari komputer korban. Serangan DDoS

adalah serangan berskala besar dengan cara terkoodinasi, yang biasanya dilakukan secara tidak langsung dengan bantuan komputer lain di internet. Ada beberapa jenis serangan DDoS. Serangan DDoS terbagi menjadi dua kelas utama: (1) serangan untuk menghabiskan bandwidth dan (2) serangan untuk menghabiskan sumber daya. Jika terjadi serangan bandwidth, jaringan korban akan dibanjiri traffic yang akan mencegah traffic yang legal untuk mencapai komputer korban. Dalam kasus serangan pada sumber daya, serangan tersebut ditargetkan untuk menghabiskan sumber daya komputer korban dengan mengirimkan banyak paket data. [5]

Serangan DDoS pada layer aplikasi dilakukan terutama untuk tujuan yang spesifik, termasuk mengganggu transaksi dan akses ke database. Ini membutuhkan lebih sedikit sumber daya dan sering disertai serangan ke layer jaringan. [6]

Ilustrasi serangan *Distributed Denial of Service* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Ilustrasi serangan *Distributed Denial of Service (DDOS)* [7]

2.3 THC Hydra

THC Hydra adalah sebuah *tools* untuk melumpuhkan dan membuka login yang mendukung banyak protokol untuk menyerang. Pada THC Hydra Modul baru mudah ditambahkan, disamping itu, sangat fleksibel dan sangat cepat. Hydra juga mendukung untuk HTTP, POP3, IMAP dan SMTP, beberapa mekanisme login seperti plain dan MD5 digest dll. Hydra telah diuji untuk dikompilasi di Linux, Windows / Cygwin, Solaris 11, FreeBSD 8.1, OpenBSD, OSX, QNX / Blackberry, dan tersedia di bawah GPLv3 dengan perluasan lisensi OpenSSL khusus. [8]

Pada Tugas Akhir ini, penulis menggunakan THC Hydra sebagai *tools* untuk melakukan serangan DdoS yang akan mengeksploitasi FTP server. Sehingga *resource* target seperti *CPU Usage* atau *memory usage* akan ikut berpengaruh. Logo *THC-Hydra* ditunjukkan pada gambar 2.3.

Gambar 2.3 Logo Hydra [9]

2.4 *Python*

Python adalah bahasa pemrograman yang memungkinkan Anda bekerja dengan cepat dan mengintegrasikan sistem secara lebih efektif. Python dikembangkan di bawah lisensi open source yang disetujui OSI, membuatnya dapat digunakan dan didistribusikan secara bebas, bahkan untuk penggunaan komersial. Lisensi Python dikelola oleh Python Software Foundation. [10]

Pada Tugas Akhir ini, bahasa pemrograman *python* akan digunakan sebagai bahasa pemrograman dalam melakukan pengambilan data, dan juga untuk mengklasifikasi data. Logo *Python* ditunjukkan pada gambar 2.4.

Gambar 2.4 Logo *Python* [11]

2.5 Scipy

Scipy adalah *library* dari bahasa pemrograman *python* yang digunakan untuk matematika, sains, dan *engineering. Scipy* digunakan oleh penulis untuk menghitung parameter yang didapatkan dari pengambilan data, untuk selanjutnya diolah agar bisa didapatkan klasifikasi data yang *valid*. *Scipy* memiliki beberapa *package* utama, diantaranya *Numpy, pandas, Sympy,* dan lain sebagainya. [12]

Pada Tugas Akhir ini, *Scipy* digunakan sebagai *library* dalam menghitung parameter yang didapatkan untuk selanjutnya dilakukan klasifikasi dari data yang telah dihitung. Selain menghitung dengan menggunakan *Numpy*, *Scipy* juga berguna untuk melakukan analisis terhadap data yang didapat

menggunakan *pandas*. Logo dari *Scipy* ditunjukkan pada gambar 2.5.

Gambar 2.5 Logo *Scipy* [13]

2.6 Sklearn

Sklearn adalah *library* dalam *pyhton* yang digunakan untuk melakukan *data mining* dan analisis data. *Sklearn* dibangun pada *Numpy*, *Scipy*, dan matplotlib. *Sklearn* dapat digunakan secara bebas dan juga bisa digunakan untuk komersial dengan lisensi BSD. [14]

Pada Tugas Akhir ini, *Sklearn* digunakan untuk melakukan klasifikasi pembanding algoritma menggunakan algoritma KNN, SVM dan ANN. Selain itu *Sklearn* juga digunakan untuk analisis data dari segi akurasi dan presisi. Logo *Sklearn* bisa dilihat pada gambar 2.6.

Gambar 2.6 Logo *Sklearn* [15]

2.7 *Numpy*

NumPy atau *Numeric Python* merupakan pustaka perangkat lunak bahasa pemrograman *Python* bersifat open source yang digunakan untuk pengolahan ilmiah matematika. Selain digunakan untuk hal ilmiah, *NumPy* juga bisa digunakan untuk container multidimensi yang efisien untuk data generik. Tipe data arbitrary (tipe data yang dapat disesuaikan dengan keinginan) dapat didefinisikan yang memungkinkan *NumPy* secara lancar dan cepat mengintegrasikan dengan berbagai tipe basis data. [16] Logo Numpy bisa dilihat pada gambar 2.7.

Gambar 2.7 Logo *Numpy* [17]

[Halaman ini sengaja dikosongkan]

BAB III
PERANCANGAN

Bab ini membahas mengenai perancangan implemetasi sistem yang dibuat pada Tugas Akhir. Bagian yang akan dijelaskan pada bab ini berawal dari deskripsi umum, perancangan skenario, hingga alur dan implementasinya.

3.1 Deskripi Umum

Pada tugas akhir ini, simulasi serangan DDoS akan dilakukan pada kluster komputer yang diarahkan pada salah satu komputer yang ada di jaringan komputer. DdoS akan disimulasikan dengan sistem yang terjumlah 19 komputer penyerang dengan target salah satu komputer yang ada di jaringan tersebut. Penyerang akan langusng menyerang salah satu komputer sehingga komputer tersebut *overload*.

Terdapat 19 penyerang DDoS yang menyerang secara bersamaan menggunakan beberapa macam serangan DDoS seperti ICMP Echo Flood atau TCP Flood. Serangan DDoS dilakukan dengan mengambil beberapa parameter dari faktor internal maupun eksternal komputer seperti CPU usage, Memory usage, banyaknya penyerang, total paket yang masuk, paket yang dikirim tiap penyerang, juga banyak proses yang berhenti untuk membuat keputusan dalam mendeteksi serangan DDoS. Parameter tersebut akan dianalisa nilai bobotnya dalam mendeteksi serangan DDoS.

Jika terdeteksi bahwa paket yang akan datang tergolong sebagai serangan DDoS, maka sistem akan mengubah konfigurasi komputer dengan cara mengubah komputer yang terkena serangan ke dalam bentuk honeypot yang sudah terkonfigurasi untuk menangkap paket yang dianggap sebagai serangan DDoS didalam sistem

3.2 Perancangan Algoritma Klasifikasi

Dari beberapa parameter tersebut, data tersebut akan di normalisasi dengan (3.1) kemudian diambil data belajar dan data latihan terlebih dahulu.

$$x_{baru} = \left(\left(\frac{x_i - x_{minlama}}{x_{maxlama} - x_{minlama}}\right) \times (x_{maxbaru} - x_{minbaru})\right) + x_{minbaru} \quad (3.1)$$

Dimana x_{baru} adalah data yang telah di normalisasi, xi adalah data yang di dapat. $X_{maxlama}$ dan $x_{minlama}$ adalah nilai maksimal dan minimal pada data yang didapat. Juga $x_{maxbaru}$ dan $x_{minbaru}$ adalah nilai maksimal dan minimal yang kita tentukan untuk normalisasi.

Setelah di normalisasi data tersebut akan diolah sehingga menghasilkan rating yang di spesifikasi pada (3.2).

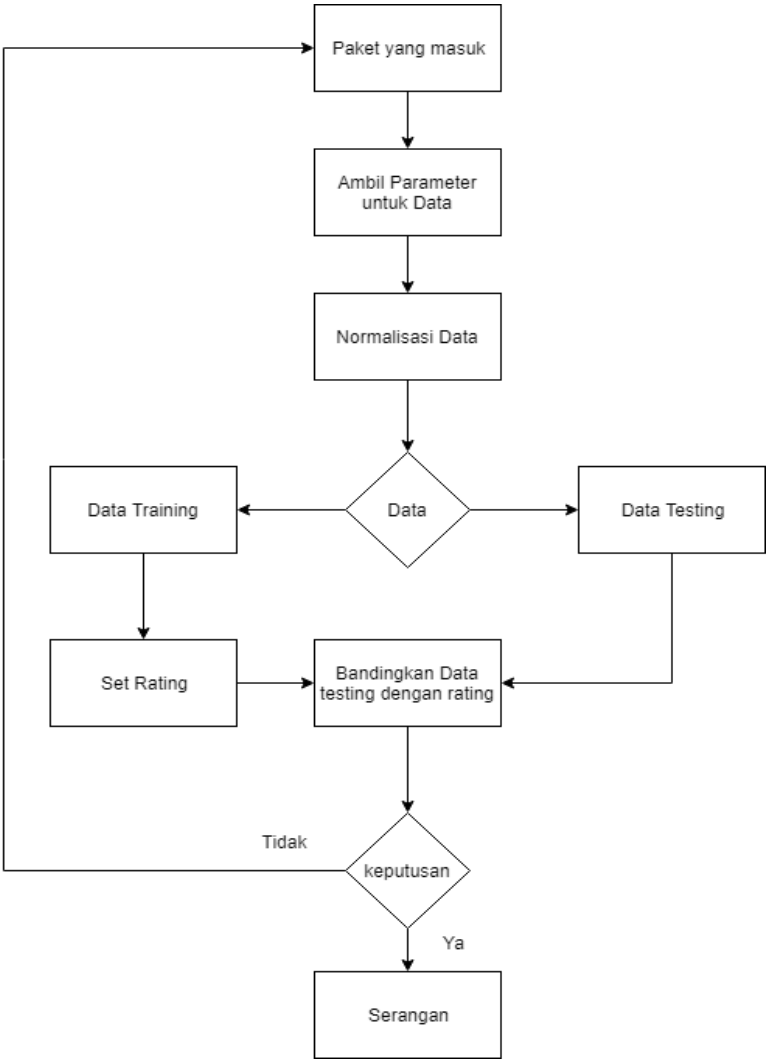
$$R = \sum x_{baru} \times B_i$$
$$status \begin{cases} Harmless, if\ R < Tresshold \\ Harmful, if\ R \geq Tresshold \end{cases} \quad (3.2)$$

Dimana R adalah rating, x_{baru} adalah nilai parameter setelah di normalisasi, dan Bi adalah bobot tiap parameter dimana jika di total, nilai B akan berjumlah 1. Nilai B_i akan dihitung pada (3.3).

$$B_i = \frac{\overline{x_{baru}}}{\sum \overline{x_{baru}}} \quad (3.3)$$

Dimana $\overline{x_{baru}}$ adalah nilai rata rata dari list data tiap parameter yang di dapat dan $\sum \overline{x_{baru}}$ adalah total dari nilai rata rata tiap list.

Jika rating paket tersebut < *tresshold*, maka paket tersebut bukan merupakan serangan DDoS. Namun, jika paket tersebut memiliki rating =< *tresshold*, maka paket tersebut adalah serangan DDoS. Analisa tersebut akan dilanjutkan dengan mengklasifikasikan serangan DDoS dengan data testing. Alur deteksi DDoS diilustrasikan pada gambar 3.1 Jika paket terdeteksi sebagai DDoS maka konfigurasi server akan berubah menjadi honeypot. Namun jika tidak, maka akan melanjutkan mendeteksi paket yang masuk.



Gambar 3.1 *Flowchart* alur serangan

BAB IV
IMPLEMENTASI

Bab ini membahas implementasi dari perancangan metode yang telah dijelaskan pada bab sebelumnya. Kode program diimplementasikan menggunakan bahasa pemrograman Python.

4.1 Lingkungan Implementasi

Lingkungan implementasi komputer target untuk media pengambilan data parameter dari serangan DdoS pada Tabel 4.1 Lingkungan Implementasi Pengambilan Data. Selain itu juga untuk mengimplementasikan algoritma KNN, SVM, ANN, dan algoritma klasifikasi berdasarkan parameter internal dan eksternal komputer pada Tabel 4.1 Lingkungan Implementasi .

Tabel 4.1 Lingkungan Implementasi Pengambilan Data

Komponen	**Spesifikasi**
CPU	Intel(R) Core(TM) i5-7200U CPU @ 2.50 GHz x 4
Sistem Operasi	Elementary OS Loki 64-bit
Memori	7.8 GiB
Penyimpanan	50 GB

Tabel 4.2 Lingkungan Implementasi Perangkat Lunak

Perangkat	**Jenis Perangkat**	**Spesifikasi**
Perangkat Keras	**Prosesor**	Intel® Core™ i5-7200U CPU @ 2.50 GHz x 4
	Memori	8 GB 2400 MHz DDR4
	Penyimpanan	1 TB Hard Drive
Perangkat Lunak	**Sistem Operasi**	Windows 10 Enterprise 64-bit
	Perangkat Pengembang	PyCharm Community 2018.1.4

4.2 Implementasi Pengambilan Data

Pada implementasi pengambilan data, terdapat 4 tahap, yaitu pengambilan data untuk komputer 1 *core*, 2 *core*, 3 *core*, dan 4 *core*.

Pada pengambilan data, kita persiapkan komputer target terlebih dahulu, dalam kasus ini akan digunakan *vagrant* sebagai komputer target. Lakukan konfigurasi pada *Vagrantfile* agar komputer memiliki 1 *core* seperti pada gambar 4.1. dimana v.cpus = 1 berarti vagrant dengan 1 core. Untuk vagrant dengan 2 core dan 3 core bisa dilakukan dengan mengganti v.cpus = 2 atau v.cpus = 3. Untuk 4 core langsung dengan komputer itu sendiri. Kemudian install ftp server seperti pada gambar 4.2. Dan juga *install htop* untuk mengamati internal komputer. Instalasi htop bisa dilihat pada gambar 4.3.

Vagrantfile

```
19  # `vagrant box outdated`. This is not recommended.
20  # config.vm.box_check_update = false
21
22  # Create a forwarded port mapping which allows access to a specific port
23  # within the machine from a port on the host machine. In the example below,
24  # accessing "localhost:8080" will access port 80 on the guest machine.
25  # config.vm.network "forwarded_port", guest: 80, host: 8080
26
27  # Create a private network, which allows host-only access to the machine
28  # using a specific IP.
29  # config.vm.network "private_network", ip: "192.168.33.10"
30
31  # Create a public network, which generally matched to bridged network.
32  # Bridged networks make the machine appear as another physical device on
33  # your network.
34  config.vm.network "public_network"
35
36  # Share an additional folder to the guest VM. The first argument is
37  # the path on the host to the actual folder. The second argument is
38  # the path on the guest to mount the folder. And the optional third
39  # argument is a set of non-required options.
40  # config.vm.synced_folder "../data", "/vagrant_data"
41
42  # Provider-specific configuration so you can fine-tune various
43  # backing providers for Vagrant. These expose provider-specific options.
44  # Example for VirtualBox:
45  #
46  config.vm.provider "virtualbox" do |v|
47  #   # Display the VirtualBox GUI when booting the machine
48  #   vb.gui = true
49    v.cpus = 1
50  #   # Customize the amount of memory on the VM:
51    v.memory = "1024"
52  end
```

Gambar 4.1 Konfigurasi *vagrant*

```
Sudo apt-get Update
Sudo apt-get install Vsftpd
```

Gambar 4.2 Instalasi *Ftp Server*

```
Sudo apt-get Update
Sudo apt-get install htop
```

Gambar 4.3 Instalasi *htop*

Kemudian, siapkan komputer serangan. Dalam kasus ini akan digunakan *virtualbox* dengan *operating system Ubuntu* Server 16.04 64-bit. Siapkan 19 *virtualbox* sebagai komputer penyerang dan *intsall hydra* pada setiap *virtualbox*. Dengan command pada gambar 4.4.

```
Sudo apt-get Update
Sudo apt-get install Hydra
```

Gambar 4.4 Instalasi *Hydra*

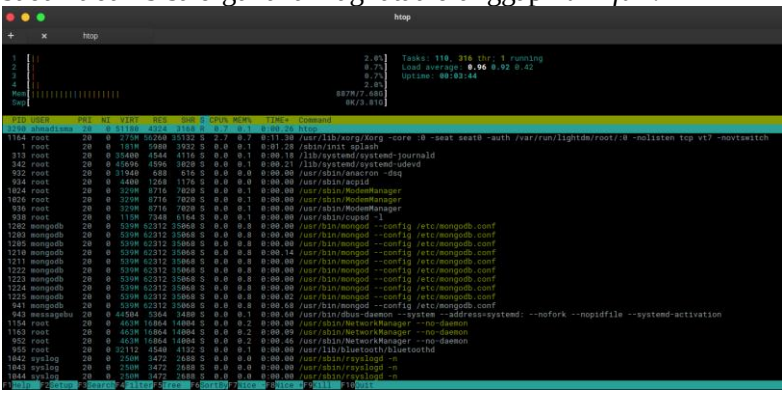
Setelah itu gunakan *Hydra* sebagai *tools* penyerangan pada komputer target. Pada perintah hydra, kita bisa mendapatkan beberapa parameter. Yaitu, jumlah serangan, dan *thread* serangan. Ketik perintah pada komputer serangan seperti pada gambar 4.5. -l adalah username dari login, yaitu root. -P adalah jumlah paket yang akan diserang yaitu dengan menggunakan *file* dalam ekstensi .txt. dalam kasus ini digunakan 200, 400, 600, 800, dan 1000 jumlah data lalu, masukkan alamat ip ftp server yang akan diserang. Dan -t adalah thread serangannya yang digunakan ialah 16, 28, 40, 52, dan 64.

```
Hydra -l root -P worst-password.txt ftp://192.168.1.14
-t 64
```

Gambar 4.5 Melakukan Serangan dengan *Hydra*

Setelah serangan selesai. Amati keadaan internal komputer yaitu *CPU usage*, *memory usage* dan *dead process*. Itu semua dapat diamati dengan bantuan htop. Yang terlihat pada gambar 4.6.

Terlihat cpu usage, jika 1 core akan tetap ditulis seperti itu, namun untuk 2,3, dan 4 core data yang diambil dari *cpu usage* adalah rata-rata. Untuk *memory usage*, data yang diambil ialah berapa persen memory yang digunakan. Sedangkan untuk *dead process*, data yang diambil ialah jumlah *dead process* selama serangan berlangsung. Status *dead process* bisa terlihat dari kolom S yang menunjukkan "D". Setelah data diambil, lakukan reboot untuk menghilangkan paket yang menumpuk setelah serangan karena akan berpengaruh pada pengambilan data selanjutnya. Untuk penentuan status apakah paket yang diterima komputer target serangan atau bukan, dilakukan 2 hal, yaitu jika komputer *hang* atau *freeze* maka dianggap *harmfull* karena berarti komputer target sudah tidak bisa digunakan lagi atau *resource* yang ada sudah benar benar penuh. Atau juga bisa dilakukan dengan melakukan *ping* dari komputer lain ke komputer target. Apabila komputer target tidak bisa di *ping* maka berarti komputer target sudah tidak bisa digunakan lagi atau dianggap *harmfull*.



Gambar 4.6 htop

4.3 Implementasi Algoritma KNN, SVM, dan ANN

Setelah mendapatkan data. Data yang digunakan ialah data yang bisa diketahui oleh internal komputer yaitu *CPU core*, *CPU Usage*, *Memory Usage*, dan *Dead Process*. Kemudian bagi data menjadi data *training*, dan data *test*, dalam kasus ini digunakan 20% data *training* dan 80% data *test*. Lalu olah data dengan mengimplementasikan 3 algoritma sebagai pembanding. Dengan menggunakan KNN, SVM, dan ANN.

4.3.1 Implementasi KNN

Dengan menggunakan *library sklearn* dari *python*, kita dapat mengimplementasikan KNN seperti pada gambar 4.7. dimana *n_neighbors* menyatakan neighbournya adalah 3.

```
neigh = KNeighborsClassifier(n_neighbors=3)
neigh.fit(x_train, y_train)
```

Gambar 4.7 Implementasi KNN

Dimana *x_train* adalah data *training*, dan *y_train* adalah label dari tiap baris data tersebut.

4.3.2 Implementasi SVM

Dengan menggunakan *library sklearn* dari *python*, kita dapat mengimplementasikan KNN seperti pada gambar 4.8.

```
svm = svm.SVC()
svm.fit(x_train, y_train)
```

Gambar 4.8 Implementasi SVM

Dimana *x_train* adalah data *training*, dan *y_train* adalah label dari tiap baris data tersebut.

4.3.3 Implementasi ANN

Dengan menggunakan *library sklearn* dari *python*, kita dapat mengimplementasikan ANN seperti pada gambar 4.9.

```
ann = MLPClassifier()
ann.fit(x_train, y_train)
```

Gambar 4.9 Implementasi ANN

Dimana *x_train* adalah data *training*, dan *y_train* adalah label dari tiap baris data tersebut.

4.4 Implementasi Algoritma Klasifikasi berdasarkan Parameter Internal dan Eksternal Komputer

Algoritma ini terdiri dari 3 tahap. Yang pertama adalah normalisasi. Pada tahap ini semua nilai pada data training akan dinormalisasi terlebih dahulu dengan rumus yang ada pada rumus (3.1). Untuk mengimplementasikannya digunakan *library numpy*. Perhitungan normalisasi ditunjukkan pada gambar 4.10. normalisasi dilakukan dengan nilai minimum adalah 0 dan nilai maksimal adalah 1.

```
def normalisasi(list):
    x_baru =[]
    x_maxlama = (max(list))
    x_minlama = (min(list))
    x_maxbaru = 1
    x_minbaru = 0
    for i in range (len(list)):
        xi = (list[i])
        x =( ((xi - x_minlama)/(x_maxlama - x_minlama))
* (x_maxbaru - x_minbaru) ) + x_minbaru
        x_baru.append(x)
```

Gambar 4.10 Normalisasi

Dimana inputnya adalah list tiap kolom parameter yaitu kolom *cpu core*, kolom *cpu usage*, kolom *memory usage*, dan kolom *dead process*.

Untuk tahap kedua, dilakukan pembobotan tiap paramter. Pembobotan dilakukan dengan rumus (3.3). perhitungan bobot ditunjukkan pada gambar 4.11.

```
def bobot(x,list):
    Bi = mean_x(x) / sum (list)
    return Bi
```

Gambar 4.11 Pembagian Bobot

Maka didapatkan bobot setiap parameter yang ada. Ditunjukkan pada tabel 4.3.

Tabel 4.3 Hasil Bobot Setiap Parameter

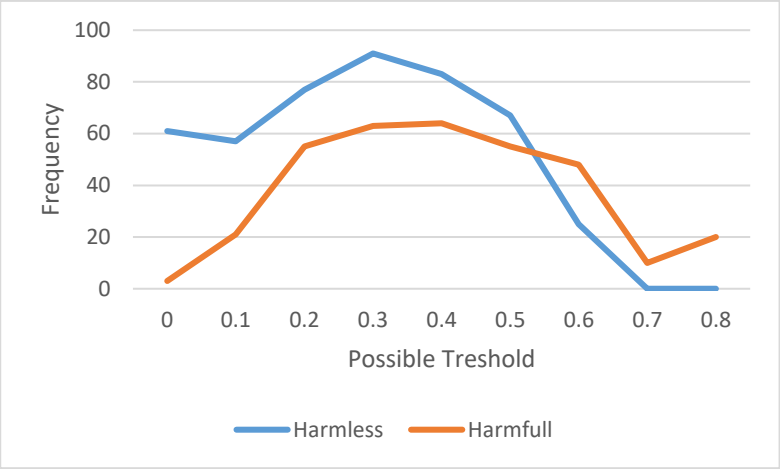
Bobot *CPU Core*	0.417449452382
Bobot *CPU Usage*	0.10905078305
Bobot *Memory Usage*	0.128361885806
Bobot *Dead Process*	0.345137878762

Setelah mendapatkan bobot dari setiap parameter, lalu dilakukan tahap ketiga, yaitu perhitungan seperti pada rumus (3.2) yang ditunjukkan pada gambar 4.12.

```
for i in range (len(list)):
        rating.append( (bobotA*normalisasiA[i]) +
(bobotB*normalisasiB[i]) + (bobotC*normalisasiC[i]) +
(bobotD*normalisasiD[i])  )
```

Gambar 4.12 Algoritma yang diusulkan

Kemudian hasil dari rating tiap baris dikelompokkan berdasarkan label dari tiap baris apakah data tersebut *harmless* atau *harmfull* yang kemudian data tersebut akan digunakan untuk menentukan *tresshold*.



Gambar 4.13 Hasil Data Training

Terlihat dari grafik pada gambar 4.13 terjadi overlap data antara rating 0,6 sampai rating 0,8. Di titik 0,6 dapat dilihat data *harmless* mulai menurun. Sehingga dapat di tentukan nilai dari *tresshold* adalah 0,6. Kemudian nilai dari *tresshold* ini yang menentukan apakah paket yang diterima merupakan serangan atau bukan.

BAB V
HASIL UJI COBA DAN EVALUASI

Bab ini membahas mengenai uji coba dan evaluasi dari skenario yang telah dilakukan.

5.1 Lingkungan Uji Coba

Uji coba dilakukan pada perangkat target dengan spesifikasi seperti yang tertera pada Tabel 5.1 dan perangkat penyerang dengan spesifikasi seperti yang tertera pada Tabel 5.2.

Tabel 5.1 Spesifikasi Perangkat target yang Digunakan

Komponen	**Spesifikasi**
CPU	Intel(R) Core(TM) i5-7200U CPU @ 2.50 GHz x 4
Sistem Operasi	Elementary OS Loki 64-bit
Memori	7.8 GiB
Penyimpanan	50 GB

Tabel 5.2 Spesifikasi Perangkat penyerang yang Digunakan

Komponen	**Spesifikasi**
CPU	Intel(R) Core(TM) i3-3200U CPU @ 1.80 GHz x 4
Sistem Operasi	Ubuntu server 16.04 64-bit
Memori	1 GiB
Penyimpanan	10 GB

Pengujian dilakukan dengan menjalankan skenario yang disimulasikan dari Perangkat penyerang ke perangkat target. Dari simulasi tersebut dihasilkan sebuah dataset dengan beberapa parameter yang kemudian akan dibagi menjadi *data training* dan *data testing.* Setelah itu dilakukan Algoritma Klasifikasi yang

mempertimbangkan faktor internal dan eksternal komputer dan Algoritma pembanding yaitu KNN, SVM, dan ANN. Kemudian akan dilihat tingkat akurasi dan presisinya.

5.2 Skenario Uji Coba

Persiapkan komputer untuk melakukan skenario uji coba. Yaitu, komputer untuk komputer penyerang dan komputer untuk komputer target. Komputer penyerang masing masing di*install THC-Hydra* untuk melakukan skenario penyerangan. Sedangkan komputer target ajab di*install htop* untuk melakukan pengamatan pada serangan. Keterangan Gambar Ilustrasi bisa dilihat pada Tabel 5.3

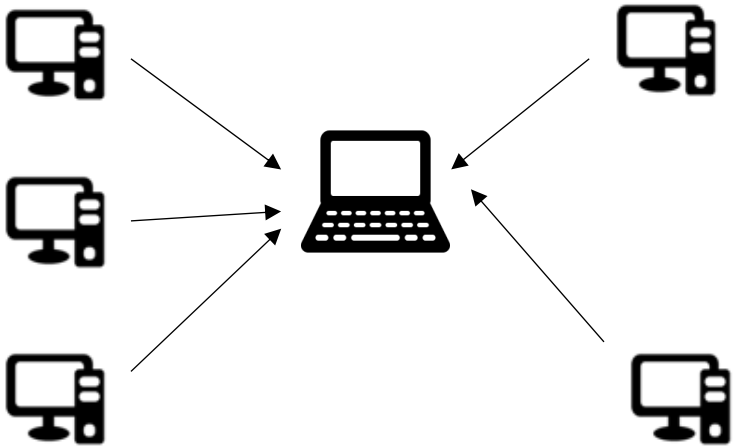
Tabel 5.3 Keterangan Gambar Ilustrasi

Gambar	Keterangan
	Komputer Target
	Komputer Penyerang

5.2.1 Skenario 5 Penyerang

Terdapat 5 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.1

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.



Gambar 5.1 Ilustrasi Penyerangan 5 Penyerang

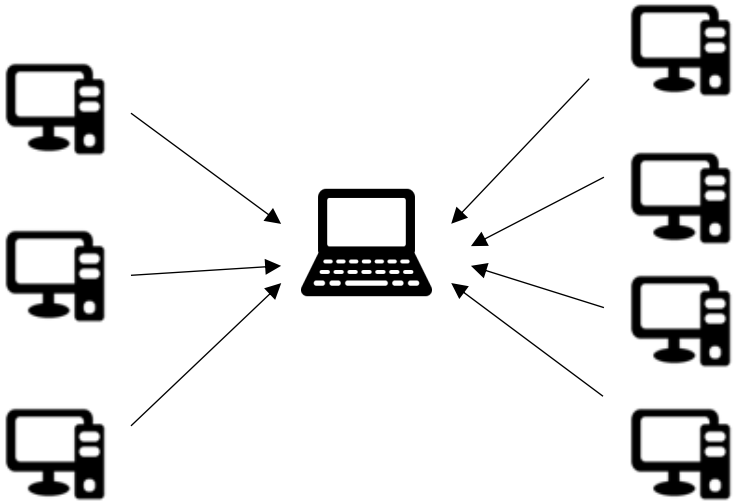
Tabel 5.4 Hasil Uji Coba 5 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.84	0.83	0.015
SVM	0.90	0.87	0.119
ANN	0.83	0.82	0.543
Metode yang dibuat	0.65	0.60	0.011

Dari Tabel 5.4 Hasil Uji Coba didapatkan bahwa algoritma SVM memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

5.2.2 Skenario 7 Penyerang

Terdapat 7 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.2



Gambar 5.2 Ilustrasi Penyerangan 7 Penyerang

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.

Tabel 5.5 Hasil Uji Coba 7 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.92	0.92	0.021
SVM	0.88	0.87	0.108
ANN	0.88	0.88	0.540
Metode yang dibuat	0.69	0.67	0.012

Dari Tabel 5.5 Hasil Uji Coba didapatkan bahwa algoritma KNN memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

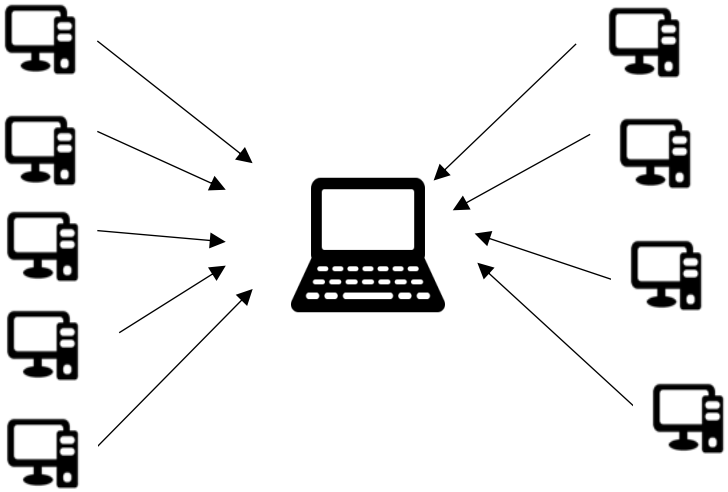
5.2.3 Skenario 9 Penyerang

Terdapat 9 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.3

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.

Tabel 5.6 Hasil Uji Coba 9 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.88	0.88	0.017
SVM	0.80	0.82	0.103
ANN	0.82	0.82	0.403
Metode yang dibuat	0.63	0.58	0.012



Gambar 5.3 Ilustrasi Penyerangan 9 Penyerang

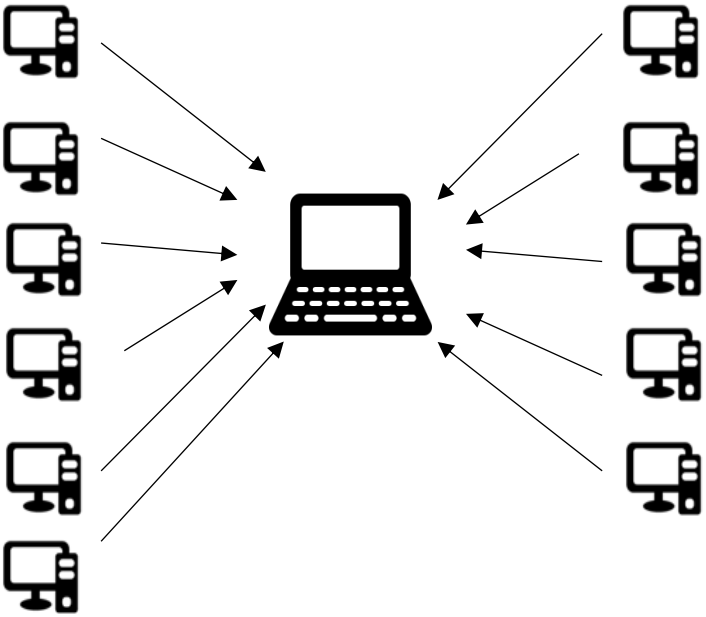
Dari Tabel 5.6 Hasil Uji Coba didapatkan bahwa algoritma KNN memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

5.2.4 Skenario 11 Penyerang

Terdapat 11 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.4

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan

dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.



Gambar 5.4 Ilustrasi Penyerangan 11 Penyerang

Tabel 5.7 Hasil Uji Coba 11 Penyerang

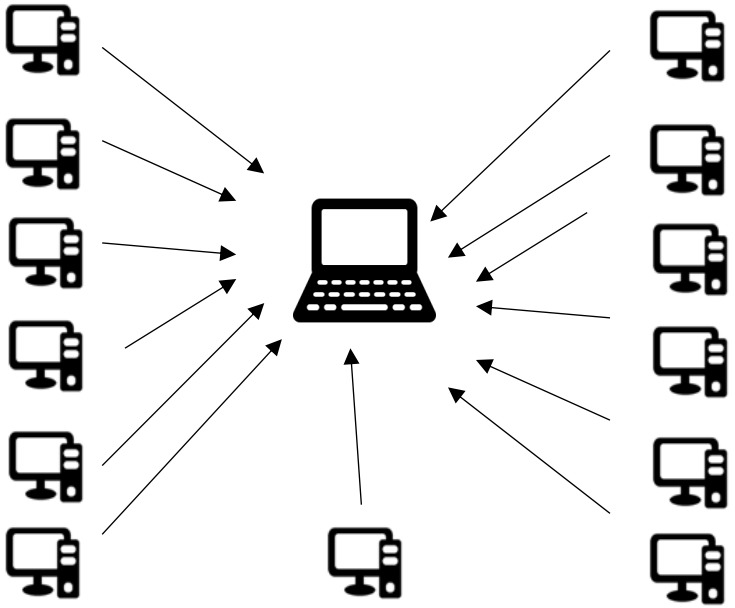
Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.92	0.91	0.020
SVM	0.87	0.87	0.105
ANN	0.88	0.88	0.635
Metode yang dibuat	0.65	0.58	0.012

Dari Tabel 5.7 Hasil Uji Coba didapatkan bahwa algoritma KNN memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki

tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

5.2.5 Skenario 13 Penyerang

Terdapat 13 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.5



Gambar 5.5 Ilustrasi Penyerangan 13 Penyerang

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan

dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.

Tabel 5.8 Hasil Uji Coba 13 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.91	0.91	0.020
SVM	0.86	0.86	0.103
ANN	0.88	0.88	0.536
Propose Method	0.70	0.59	0.012

Dari Tabel 5.8 Hasil Uji Coba didapatkan bahwa algoritma KNN memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

5.2.6 Skenario 15 Penyerang

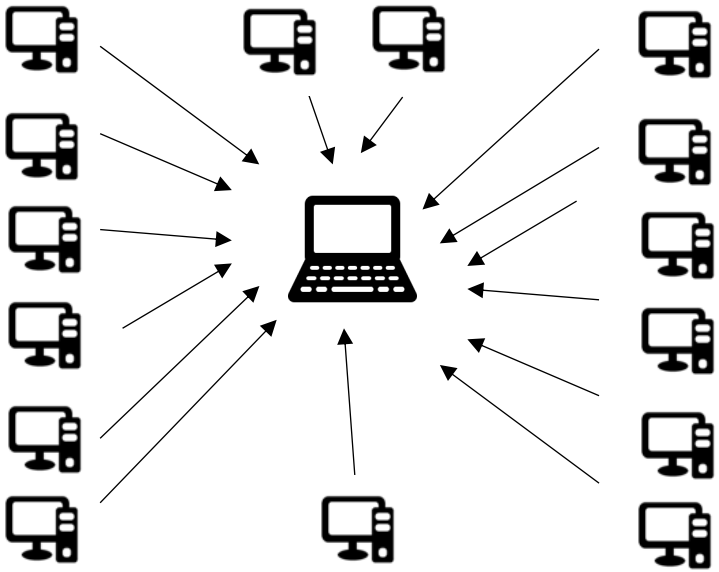
Terdapat 15 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.6

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.

Tabel 5.9 Hasil Uji Coba 15 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.90	0.89	0.015
SVM	0.88	0.88	0.106
ANN	0.91	0.90	0.395
Metode yang dibuat	0.68	0.63	0.012

Dari Tabel 5.9 Hasil Uji Coba didapatkan bahwa algoritma ANN memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

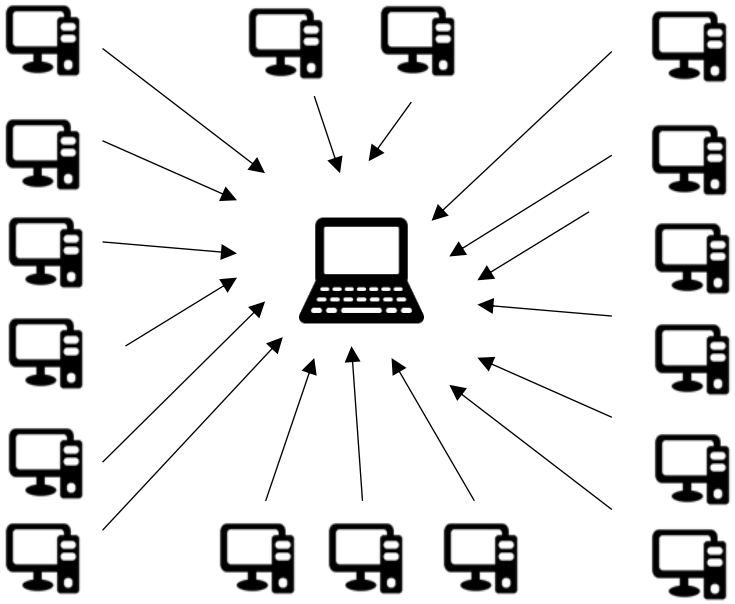


Gambar 5.6 Ilustrasi Penyerangan 15 Penyerang

5.2.7 Skenario 17 Penyerang

Terdapat 17 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.7

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.



Gambar 5.7 Ilustrasi Penyerangan 17 Penyerang

Tabel 5.10 Hasil Uji Coba 17 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.82	0.82	0.020
SVM	0.88	0.88	0.101
ANN	0.85	0.85	0.594
Metode yang dibuat	0.58	0.58	0.008

Dari Tabel 5.10 Hasil Uji Coba didapatkan bahwa algoritma SVM memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. Metode yang diusulkan menempati urutan pertama dan algoritma ANN masih menempati urutan terakhir.

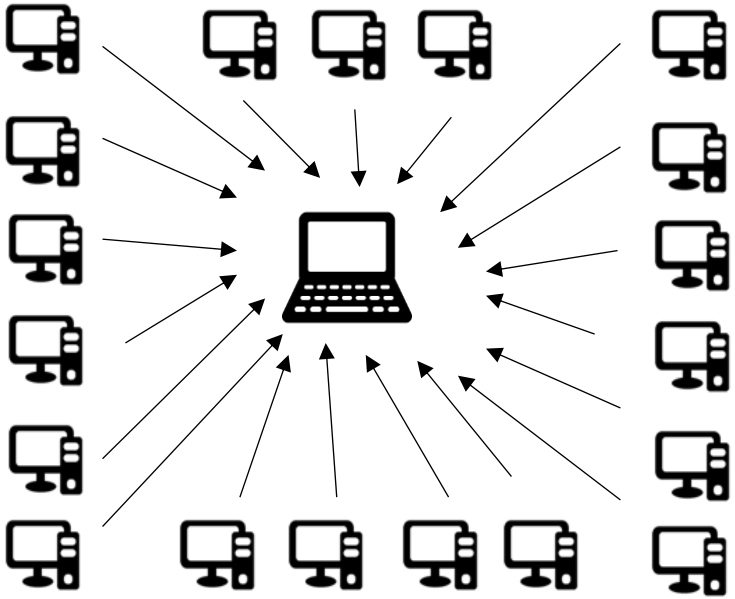
5.2.8 Skenario 19 Penyerang

Terdapat 17 komputer penyerang dan 1 komputer target. Komputer penyerang akan menyerang komputer target secara bersamaan dengan *tools THC-Hydra*. Ilustrasi penyerangan dapat terlihat pada gambar 5.8

Setelah penyerangan dilakukan, kita mendapatkan sejumlah data yang nantinya akan dilakukan pengujian terhadap algroitma yang diusulkan. Kemudian algoritma tersebut akan dibandingkan dengan beberapa algoritma klasifikasi yang sudah ada lalu kita analisis dari segi akurasi dan presisinya.

Tabel 5.11 Hasil Uji Coba 19 Penyerang

Algoritma	Akurasi	Presisi	Waktu (s)
KNN	0.84	0.84	0.014
SVM	0.87	0.86	0.104
ANN	0.82	0.82	0.505
Metode yang dibuat	0.58	0.55	0.016



Gambar 5.8 Ilustrasi Penyerangan 19 Penyerang

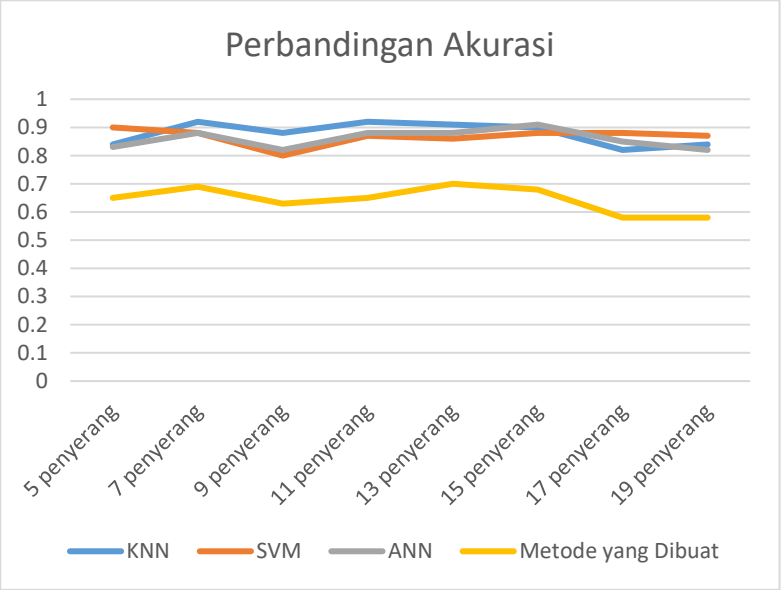
Dari Tabel 5.11 Hasil Uji Coba didapatkan bahwa algoritma SVM memiliki akurasi dan presisi yang paling tinggi nilainya. Sedangkan untuk algoritma yang di usulkan memiliki tingkat akurasi dan presisi yang paling rendah. Namun dari segi kecepatan. KNN menempati urutan pertama, sedangkan metode yang diusulkan menempati urutan kedua dan algoritma ANN masih menempati urutan terakhir.

5.3 Evaluasi

Berdasarkan skenario uji coba pada skenario 5 penyerang hingga skenario 19 penyerang didapatkan beberapa evaluasi dari sisi akurasi, presisi, dan kecepatan metode.

5.3.1 Akurasi

Dari sisi akurasi metode akan dilakukan evaluasi dari skenario 5 penyerang hingga skenario 19 penyerang. Dari beberapa skenario tersebut didapatkan *track record* akurasi seperti pada gambar 5.9

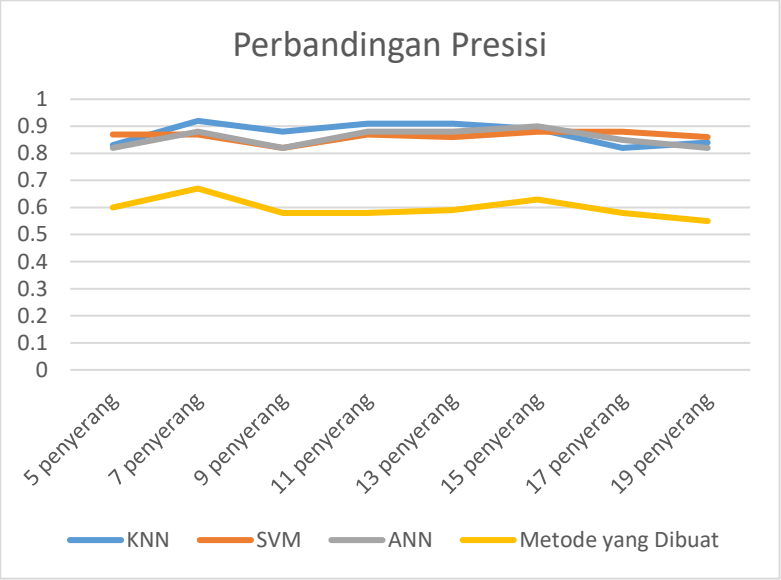


Gambar 5.9 Perbandingan Akurasi

Terlihat dari grafik pada gambar 5.9 metode yang dibuat memiliki akurasi paling rendah dibandingkan dengan metode KNN, SVM, dan ANN. Sedangkan untuk akurasi tertinggi metode KNN, SVM, dan ANN menduduki akurasi tertinggi secara bergantian. Metode KNN menduduki tempat tertinggi pada skenario 7, 9, 11, dan 13 penyerang. Metode SVM menduduki tingkat akurasi tertinggi pada skenario 5, 17, dan 19 penyerang. Sedangkan untuk metode ANN menduduki tingkat akurasi tertinggi pada skenario 15 penyerang.

5.3.2 Presisi

Dari sisi presisi metode akan dilakukan evaluasi dari skenario 5 penyerang hingga skenario 19 penyerang. Dari beberapa skenario tersebut didapatkan *track record* akurasi seperti pada gambar 5.10

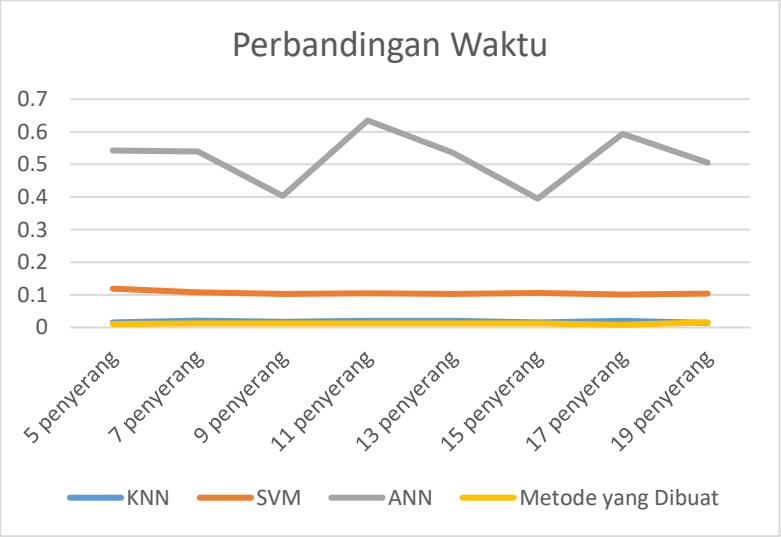


Gambar 5.10 Perbandingan Presisi

Terlihat dari grafik pada gambar 5.10 metode yang dibuat memiliki presisi paling rendah dibandingkan dengan metode KNN, SVM, dan ANN. Sedangkan untuk presisi tertinggi metode KNN, SVM, dan ANN menduduki tingkat presisi tertinggi secara bergantian. Metode KNN menduduki tingkat presisi tertinggi pada skenario 7, 9, 11, dan 13 penyerang. Metode SVM menduduki tingkat akurasi tertinggi pada skenario 5, 17, dan 19 penyerang. Sedangkan untuk metode ANN menduduki tingkat akurasi tertinggi pada skenario 15 penyerang.

5.3.3 Kecepatan

Dari sisi kecepatan metode akan dilakukan evaluasi dari skenario 5 penyerang hingga skenario 19 penyerang. Dari beberapa skenario tersebut didapatkan *track record* akurasi seperti pada gambar 5.11



Gambar 5.11 Perbandingan Waktu

Terlihat dari grafik pada gambar 5.10 metode yang dibuat memiliki kecepatan paling cepat dibandingkan dengan metode KNN, SVM, dan ANN dengan melihat waktu yang paling rendah. Metode yang dibuat menduduki tingkat kecepatan tertinggi pada skenario 5, 7, 9, 11, 13, 15, dan 17 penyerang. Metode KNN menduduki tingkat kecepatan tertinggi pada skenario 19 penyerang. Sedangkan untuk metode ANN menduduki tingkat kecepatan yang paling rendah karena memiliki waktu yang paling tinggi.

BAB VI
KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diperoleh dari Tugas Akhir yang telah dikerjakan dan saran tentang pengembangan dari Tugas Akhir ini yang dapat dilakukan di masa yang akan datang.

6.1 Kesimpulan

Kesimpulan yang diperoleh dari hasil uji coba dan evaluasi dari pengujian Algoritma yang diusulkan dalam Tugas Akhir ini adalah sebagai berikut:

1. Dengan menggunakan algoritma klasifikasi yang mempertimbangkan parameter dari internal dan eksternal komputer bisa mendeteksi adanya paket yang berasal dari serangan.
2. Perbandingan performa algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer dengan algoritma KNN, SVM, dan ANN adalah sebagai berikut :
    a. Untuk skenario 5 penyerang nilai akurasi dan presisi paling tinggi ialah metode SVM dengan nilai akurasi 0,90 dan nilai presisi 0,87. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,011 detik dan menempati urutan pertama.
    b. Untuk skenario 7 penyerang nilai akurasi dan presisi paling tinggi ialah metode KNN dengan nilai akurasi 0,92 dan nilai presisi 0,92. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,012 detik dan menempati urutan pertama.
    c. Untuk skenario 9 penyerang nilai akurasi dan presisi paling tinggi ialah metode SVM dengan nilai akurasi 0,88 dan nilai presisi 0,88. Sedangkan dari segi

kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,012 detik dan menempati urutan pertama.

d. Untuk skenario 11 penyerang nilai akurasi dan presisi paling tinggi ialah metode KNN dengan nilai akurasi 0,92 dan nilai presisi 0,91. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,012 detik dan menempati urutan pertama.
e. Untuk skenario 13 penyerang nilai akurasi dan presisi paling tinggi ialah metode KNN dengan nilai akurasi 0,91 dan nilai presisi 0,91. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,012 detik dan menempati urutan pertama.
f. Untuk skenario 15 penyerang nilai akurasi dan presisi paling tinggi ialah metode ANN dengan nilai akurasi 0,91 dan nilai presisi 0,90. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,012 detik dan menempati urutan pertama.
g. Untuk skenario 17 penyerang nilai akurasi dan presisi paling tinggi ialah metode SVM dengan nilai akurasi 0,88 dan nilai presisi 0,88. Sedangkan dari segi kecepatan algoritma klasifikasi berdasarkan faktor internal dan eksternal komputer memiliki nilai 0,008 detik dan menempati urutan pertama.
h. Untuk skenario 19 penyerang nilai akurasi dan presisi paling tinggi ialah metode SVM dengan nilai akurasi 0,87 dan nilai presisi 0,86. Sedangkan dari segi kecepatan algoritma klasifikasi dengan metode KNN memiliki nilai 0,014 detik dan menempati urutan pertama.

6.2 Saran

Saran yang diberikan dari hasil uji coba dan evaluasi dari pengujian algoritma yang diusulkan dalam Tugas Akhir ini adalah sebagai berikut:

1. Implementasi algoritma bisa dilakukan dengan menambah parameter lain. Seperti ip dari resource paket.
2. Diperlukan pengembangan algoritma untuk meningkatkan kualitas akurasi dan presisi.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

[1] D. Bisson, “5 Notable DDoS Attacks of 2017,” 2017. [Online]. Available: https://www.tripwire.com/state-of-security/featured/5-notable-ddos-attacks-2017/. [Diakses 3 January 2018].

[2] Securelist, “DDoS attacks in Q3 2017,” 2017. [Online]. Available: https://securelist.com/ddos-attacks-in-q3-2017/83041/. [Diakses 3 January 2018].

[3] Cicso, “What Is Network Security,” Cisco, [Online]. Available: https://www.cisco.com/c/en/us/products/security/what-is-network-security.html. [Diakses 3 January 2018].

[4] I. S. Buzz, “CIA Triad and New Emerging Technologies: Big Data and IoT,” [Online]. Available: https://www.informationsecuritybuzz.com/isbuzz-expert-panel/cia-triad-and-new-emerging-technologies-big-data-and-iot/. [Diakses 6 June 2018].

[5] S. Specht, “Taxonomies of Distributed Denial of Service Networks, Attacks, Tools, and Countermeasure,” 2003.

[6] K. J. Higgins, “DDoS Attacks Used 'Headless' Browserin 150-hour Siege,” October 2013. [Online]. Available: https://web.archive.org/web/20140122165039/http://www.darkreading.com/attacks-breaches/ddos-attack-used-headless-browsers-in-15/240162777. [Diakses 3 January 2018].

[7] M. Syukrillah, “Pengertian dan Beberapa Hal tentang DDoS,” [Online]. Available: http://www.pengonaq-media.com/2015/08/pengertian-dan-beberapa-hal-tentang-ddos.html. [Diakses 6 June 2018].

[8] V. Hauser, “Hydra,” THC Hydra, [Online]. Available: https://www.thc.org/thc-hydra/. [Diakses 5 January 2018].

[9] Kitploit, “Kitploit,” [Online]. Available: https://www.kitploit.com/2016/06/thc-hydra-82-network-logon-cracker.html. [Diakses 26 Juny 2018].

[10] Python, “Python,” Python, [Online]. Available: https://www.python.org/. [Diakses 5 January 2018].

[11] Python.org, “python-logo.png,” [Online]. Available: https://Python.org. [Diakses 25 Juni 2018].

[12] “SciPy,” Scientific Computing Tools for Python, [Online]. Available: https://scipy.org/. [Diakses 6 June 2018].

[13] SciPy.org, “scipyshiny_small.png,” [Online]. Available: Scipy.org. [Diakses 25 Juni 2018].

[14] “Scikit-Learn,” Scikit-Learn, [Online]. Available: http://scikit-learn.org/stable/index.html. [Diakses 6 June 2018].

[15] scikit-learn.org, “scikit-learn-logo-notext.png,” [Online]. Available: scikit-learn.org. [Diakses 25 Juni 2018].

[16] Numpy.org, “numpy,” [Online]. Available: http://www.numpy.org/. [Diakses 25 Juni 2018].

[17] Numpy, “Numpy-logo.png,” [Online]. Available: http://www.numpy.org/. [Diakses 26 Juni 2018].

BIODATA PENULIS

Ahmad Ismail Harry Wicaksono lahir di Surabaya pada 19 April 1996. Penulis menempuh pendidikan formal dimulai dari TK Al-Hidayah (2001-2002), SDN Pakis X/538 Surabaya (2002-2008), SMPN Unggulan Amantul Ummah Surabaya (2008-2011), SMAN 1 Jember (2011-2014) dan S1 Teknik Informatika ITS (2014-2018). Bidang studi yang diambil oleh penulis pada saat berkuliah di Teknik Informatika ITS adalah Komputasi Berbasis Jaringan (KBJ). Penulis aktif dalam organisasi seperti Himpunan Mahasiswa Teknik Computer-Informatika (2015) dan Keluarga Muslim Informatika (2015-2016). Penulis juga aktif dalam kegiatan kepanitiaan seperti SCHEMATICS 2015 dan SCHEMATICS 2016 divisi Perlengkapan dan Transportasi (2015) dan divisi *National Logic Competition* (2016). Penulis pernah kerja praktik di PT. Perkebunan Nusantara X periode Juli – Agustus 2017. Penulis dapat dihubungi melalui email x5ahmadismail@gmail.com.