ANALISIS PERMASALAHAN JARINGAN KOMPUTER LAN MENGGUNAKAN APLIKASI *WIRESHARK*

SKRIPSI

Oleh :
Kelvianto
150210011

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
2018/2019

ANALISIS PERMASALAHAN JARINGAN KOMPUTER LAN MENGGUNAKAN APLIKASI *WIRESHARK*

SKRIPSI

Untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana

Oleh :
Kelvianto
150210011

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
2018/2019

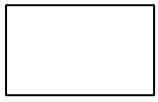
PERNYATAAN

Dengan ini saya menyatakan bahwa :

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (sarjana, dan/atau magister), baik di Universitas Putera Batam maupun di perguruan tinggi lain.
2. Skripsi ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi.

Batam, 05 September 2019

Yang membuat pernyataan,



Kelvianto
150210011

ANALISIS PERMASALAHAN JARINGAN KOMPUTER LAN MENGGUNAKAN APLIKASI *WIRESHARK*

Oleh:
Kelvianto
150210011

SKRIPSI

Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana

Telah disetujui oleh Pembimbing pada tanggal
seperti tertera di bawah ini

Batam, 05 September 2019

Andi Maslan, S.T., M.SI.
Pembimbing

ABSTRAK

Jaringan komputer pada era sekarang sangat bermanfaat dan banyak dipakai untuk mempercepat serta memperlancar arus data atau informasi dimana manfaat yang didapatkan yaitu dari segi waktu, pengiriman data, menggali informasi, efektifitas, serta efisiensi. Tetapi pada jaringan komputer tidak menutup kemungkinan terjadinya serangan atau ancaman salah satunya yaitu serangan *Denial-of-Service* (DoS). Serangan DoS dapat terjadi pada jaringan komputer *Local Are Network* (LAN) apabila pemantauan jaringan yang masih minim. Tujuan penelitian yang hendak dicapai yaitu menangkap setiap paket data yang ada dalam jaringan dan dan dapat mendeteksi serangan DoS pada jaringan komputer. Solusi atau cara untuk mendeteksi serangan DoS pada suatu jaringan adalah dengan menggunakan aplikasi *wireshark* dimana sistem kerja *wireshark* yaitu menangkap setiap aktivitas atau kegiatan yang dilakukan dalam suatu jaringan. Dari sekumpulan data yang telah direkam oleh *wireshark*, ada beberapa data yang termasuk normal dan ada beberapa data yang termasuk serangan DoS. Sehingga pada penelitian ini untuk mengklasifikasikan data yang normal dan DoS yaitu dengan menggunakan algoritma *Naive Bayes* yang dijalankan dengan menggunakan aplikasi *Waikato Environment for Knowledge Analysis* (WEKA). Hasil pengujian pengklasifikasian menggunakan algoritma *Naive Bayes* dengan bantuan aplikasi WEKA yang dilakukan sebanyak 3 kali pengujian pada penelitian ini didapatkan nilai akurasi yang tinggi dimana pada pengujian ke-1 sebesar 99,9935%, pada pengujian ke-2 sebesar 99,9962%, dan pada pengujian ke-3 sebesar 99,9932%, sehingga persentase rata-rata akurasi sebesar 99,9943%. Dapat terdeteksi serangan DoS jenis *UDP Flooding* pada jaringan yang dianalisa dengan sistem *wireshark* dimana diketahui IP penyerang 169.254.3.22 mengirim jumlah paket data yang banyak ke *port* UDP IP server 169.254.74.248.

Kata Kunci : *Wireshark, Denial-of-Service* (DoS), Algoritma *Naive Bayes*, *Waikato Environment for Knowledge Analysis* (WEKA)

ABSTRACT

Computer networks in this era are very useful and widely used to accelerate and facilitate the flow of data or information where the benefits obtained are in terms of time, data transmission, information gathering, effectiveness, and efficiency. But on a computer network does not close the possibility of attacks or threats, one of them is a Denial-of-Service (DoS) attack. DoS attack can happen on Local Are Network (LAN) computer networks if network monitoring is still minimal. Research objectives to be achieved is to capture every packet data in the network and be able to detect DoS attacks on computer networks.The solution to detect DoS attack on a network is to use a wireshark application where the wireshark work system is capturing every activity in a network. From a set of data that has been recorded by a wireshark, there are some data that is normal and there are some data is DoS attack. So in this study to classify normal data and DoS is using the Naive Bayes algorithm which is run by using the Waikato Environment for Knowledge Analysis (WEKA) application. The results of the classification test using the Naive Bayes algorithm with the help of WEKA application conducted 3 times of testing in this study obtained a high accuracy value where in the first test was 99.9935%, in the 2nd test was 99.9962%, and in 3rd testing 99.9932%, so thatThe test results in this study obtained an average percentage of accuracy is 99.9943%. UDP Flooding in DoS type attacks can be detected on the network that is analyzed with wireshark system where the attacker IP is known 169.254.3.22 sending a large number of packet data to the UDP port IP server 169.254.73.248.

Keywords : *Wireshark, Denial-of-Service* (*DoS*), *Naive Bayes Algorithm, Waikato Environment for Knowledge Analysis* (*WEKA*)

KATA PENGANTAR

Penulis menghaturkan puji syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan proposal yang merupakan salah satu persyaratan untuk melakukan skripsi (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

Penulis menyadari bahwa proposal ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati.

Dengan segala keterbatasan, penulis menyadari pula bahwa proposal ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada :

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Bapak **Andi Maslan, S.T., M.SI.**
3. Bapak **Andi Maslan, S.T., M.SI** selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan selalu mencurahkan hidayah serta taufikNya, Amin.

Batam, 05 September 2019

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR RUMUS

BAB I
PENDAHULUAN

1.1 Latar Belakang Penelitian

Jaringan komputer digunakan untuk mempercepat serta memperlancar arus data pada tempat tertentu di era sekarang seperti internal perusahaan, perusahaan yang memiliki cabang di kota lain, dan lain-lain. Jaringan komputer telah dikenal oleh masyarakat dikarenakan memiliki banyak manfaat baik dari segi waktu pengiriman data, biaya pengiriman, efektifitas, serta efisiensi dalam berbisnis. Tetapi dalam pengiriman data pada jaringan komputer ini tidak menutup kemungkinan dari berbagai serangan. Jaringan komputer dapat mengecek sistem komputer agar terhindar dari serangan-serangan komputer yaitu dengan *troubleshooting*. Jaringan komputer memiliki beberapa jenis berdasarkan jangkauan yaitu LAN (*Local Area Network*), MAN (*Metropolitan Area Network*), dan WAN (*Wide Area Network*). Pada permasalahan ini hanya dibahas jaringan komputer dengan jangkauan LAN. LAN biasanya dipakai oleh perusahaan atau organisasi yang masih kecil atau belum besar. Namun pada jangkauan LAN juga kemungkinan data akan di curi oleh orang luar.

Jenis-jenis serangan yang ada pada jaringan komputer tidaklah sedikit, ada jenis serangan DoS, *sniffing*, *time bomb*, dan masih ada banyak jenis serangan. Jenis-jenis serangan pada jaringan komputer sangat berbahaya, dimana jenis serangannya ada yang merusak data, mencuri data, memperlambat sistem kerja

komputer, mengisi banyak data yang tidak penting kedalam sistem komputer tanpa sepengetahuan pengguna, dan lain-lain.

Berita yang dilansir oleh detikINET pada tahun 2017 yang memberitakan bahwa Indonesia dibombardir 205 juta serangan *cyber* yang dicatat selama 11 bulan dari Januari sampai November oleh *Indonesia Security Incident Response Team on Internet Infrastructure / Coordinator Center* (Id-SIRTII/CC). Peneliti mengakses *website* berita tersebut pada tanggal 27 maret 2019. Serangan *cyber* yang paling banyak terdeteksi yaitu jenis-jenis *malware*, dirincikan aktivitas *malware* yang terdeteksi yaitu sekitar 37,72% berkaitan dengan serangan DoS (*Denial-of-Service Attack*), sekitar 20,93% berupa *exploit*, 18% aktivitas *trojan*, 15% tercatat sebagai *bad unknown*, dan sisanya berupa *adware*, *shell code*, *cnc*, *misc attack*, *network scan*, dan *web application attack*. Kesimpulan yang bisa didapatkan pada berita ini bahwa serangan yang paling banyak dilakukan ialah serangan DoS dimana serangan ini bertujuan untuk memperburuk komputer atau *server* yang diserang.

Untuk mengatasi masalah-masalah tersebut, maka peneliti akan menggunakan aplikasi *wireshark*. Aplikasi tersebut dapat menangkap setiap data yang mengalir pada jaringan baik data normal maupun data serangan. Agar dapat mendeteksi serangan pada data-data yang telah di *capture* oleh *wireshark*, maka digunakan algoritma *naive bayes* untuk mengklasifikasikan data mana yang termasuk normal dan data mana yang termasuk serangan dengan bantuan aplikasi WEKA (*Waikato Environment for Knowledge Academy*) dimana aplikasi ini telah memiliki algoritma *naive bayes* sehingga dapat menjalankan proses

pengklasifikasian. Pengumpulan data yaitu dengan cara mengamati paket data yang di capture kemudian dilakukan beberapa kali pengujian pengklasifikasian. Pengujiannya dilakukan dengan memanfaatkan beberapa komputer yang terhubung dengan jaringan lokal.

Sarana serta referensi yang memadai diperlukan untuk mendapatkan hasil yang maksimal. Kasus diatas dapat diangkut ke dalam laporan ini sebagai pertimbangan dan dipandang penting dengan mengambil judul **"Analisis Permasalahan Jaringan Komputer LAN Menggunakan Aplikasi *Wireshark*"**.

1.2 Identifikasi Masalah

Terdapat beberapa identifikasi masalah berdasarkan latar belakang diatas, yaitu:

1. Ada jenis-jenis serangan yang akan masuk kedalam jaringan komputer.
2. Perlu adanya aplikasi atau bantuan dalam hal mendeteksi serangan.
3. Perlu adanya aplikasi *monitoring* jaringan dalam memantau komunikasi data.

1.3 Pembatasan Masalah / Lingkup

Pembahasan masalah yang dibahas agar tidak menyimpang dari pokok permasalahan dan terarah dengan baik, maka permasalahan dibatasi sebagai berikut:

1. Tempat penelitian berupa non-instansi atau di rumah.

2. Jenis serangan yang di analisis yaitu serangan DoS jenis UDP *Flooding*.
3. Aplikasi untuk memasukkan serangan DoS yaitu LOIC (*Low Orbit Ion Cannon*).
4. Algoritma mendeteksi serangan DoS yaitu Algoritma *Naive Bayes*.
5. Menggunakan aplikasi WEKA (*Waikato Environment for Knowledge Academy*) untuk menjalankan algoritma *Naive Bayes*.

1.4 Perumusan Masalah

Perumusan masalah ditentukan berdasarkan identifikasi masalah, sehingga dirumuskan sebagai berikut:

1. Bagaimana cara menganalisa paket data pada jaringan komputer LAN?
2. Bagaimana cara mendeteksi jenis serangan DoS pada jaringan komputer?

1.5 Tujuan Penelitian

Tujuan yang hendak dicapai dalam penelitian ini ialah:

1. Menangkap setiap paket data yang ada dalam jaringan menggunakan aplikasi *wireshark*.
2. Menggunakan algoritma *naive bayes* untuk mendeteksi serangan DoS pada jaringan komputer dengan bantuan aplikasi WEKA untuk menjalankan proses pengklasifikasian untuk mendapatkan hasil akurasi.

1.6 Manfaat Penelitian

1.6.1 Manfaat Teoritis

Hasil penelitian ini dapat dijadikan referensi dalam menganalisa permasalahan jaringan komputer LAN.

1.6.2 Manfaat Praktis

Manfaat praktis pada penelitian ini dapat memberikan manfaat kepada beberapa pihak, yaitu:

1. Mahasiswa, dapat menjadi sumber atau acuan jika ingin menganalisa permasalahan jaringan komputer LAN.
2. Kampus, dapat meningkatkan akreditas serta nama baik pihak kampus.
3. Masyarakat, dapat pemahaman mengenai pentingnya keamanan data pada jaringan komputer.

BAB II
KAJIAN PUSTAKA

2.1 Teori Dasar

Pada teori dasar, hal-hal umum mengenai dasar jaringan akan dijelaskan untuk materi dan bahan penelitian. Berikut adalah konsep dan penjelasan tentang jaringan komputer, standar jaringan komputer, jenis jaringan komputer, model *OSI layer*, dan *wireshark*.

2.1.1 Jaringan Komputer

Berdasarkan kutipan dari (Maslan, 2012) bahwa jaringan komputer adalah sebuah kumpulan dari komputer, printer, dan peralatan lainnya yang terhubung dalam satu kesatuan dan membentuk suatu sistem tertentu. Informasi bergerak melalui kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar informasi (data), mencetak data pada printer yang sama dan dapat secara simultan menggunakan program aplikasi yang sama.

Berdasarkan kutipan dari (Maslan, 2012), manfaat-manfaat jaringan komputer yaitu:

1. Saling berbagi *file*.
2. Dapat menukar data, baik berupa video, audio, dan gambar.
3. Bisa menggunakan printer secara bersamaan.
4. Menghemat dari segi biaya.

5. Meningkatnya efisiensi kerja.
6. Memelihara *file* menjadi lebih mudah.
7. Meningkatkan kinerja sistem.

2.1.2 Standar Jaringan Komputer

Beberapa badan dunia yang melakukan standarisasi jaringan komputer yaitu ISO (*International Organization for Standardization*), ITU (*International Telecommunication Union*), ANSI (*AmSasaan National Standard Institute*), NCITS (*National Committee for Information Technology Standardization*), IEEE (*Institute of Electrical and Electronics Engineers*) dan lain-lain. Beberapa badan pekerja yang dibentuk oleh IEEE yang banyak membuat standarisasi peralatan telekomunikasi seperti yang tertera pada tabel berikut (Maslan, 2012) :

Tabel 2. 1 Badan Pekerja di IEEE

Working Group	Bentuk Kegiatan
IEEE802.1	Standarisasi *interface* lapisan atas HILI (*High Level Interface*) dan *Data Link* termasuk MAC (*Medium Access Control*) dan LLC (*Logical Link Control*).
IEEE802.2	Standarisasi lapisan LLC.
IEEE802.3	Standarisasi lapisan MAC untuk CSMA/CD (10Base5, 10Base2, 10BaseT, dll.)
IEEE802.4	Standarisasi lapisan MAC untuk Token Bus.
IEEE802.5	Standarisasi lapisan MAC untuk Token Ring.
IEEE802.6	Standarisasi lapisan MAC untuk MAN-DQDB.

Sumber : (Maslan, 2012)

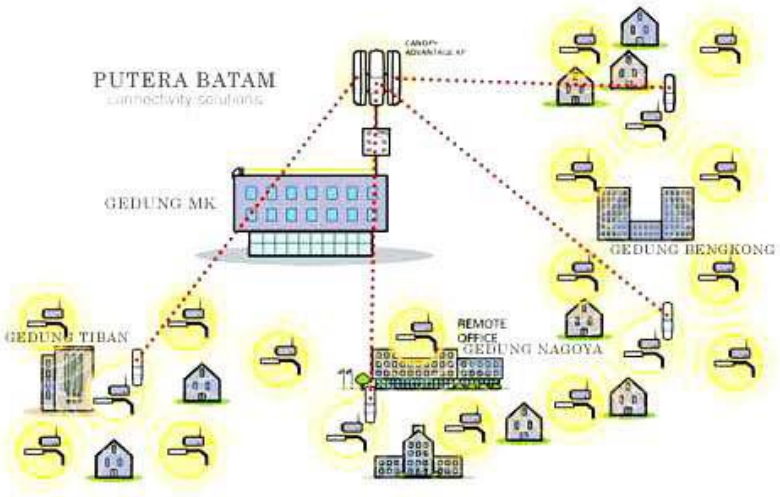
2.1.3 Jenis Jaringan Komputer

Jenis-jenis jaringan komputer berdasarkan jangkauan ada 4, yaitu (Maslan, 2012) :

1. **LAN (*Local Area Network*)**, merupakan jaringan dalam lingkup gedung atau kampus yang berukuran hanya beberapa kilometer. LAN biasanya digunakan sebagai penghubung antar komputer dan *workstation* dalam kantor atau perusahaan untuk pemakaian bersama dan saling berbagi informasi.

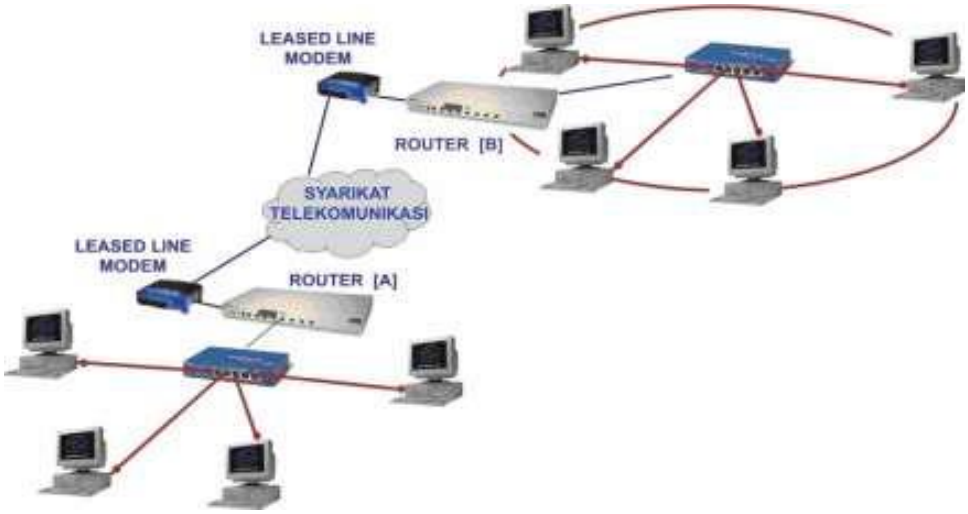
Gambar 2. 1 LAN (*Local Area Network*)

2. **MAN *(Metropolitan Area Network)***, merupakan versi yang berukuran lebih besar daripada LAN, teknologi yang digunakan masih sama dengan LAN. MAN mencakup antar perusahaan yang letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi atau umum. MAN mampu menunjang data dan suara bahkan dapat digunakan untuk aplikasi TV kabel.



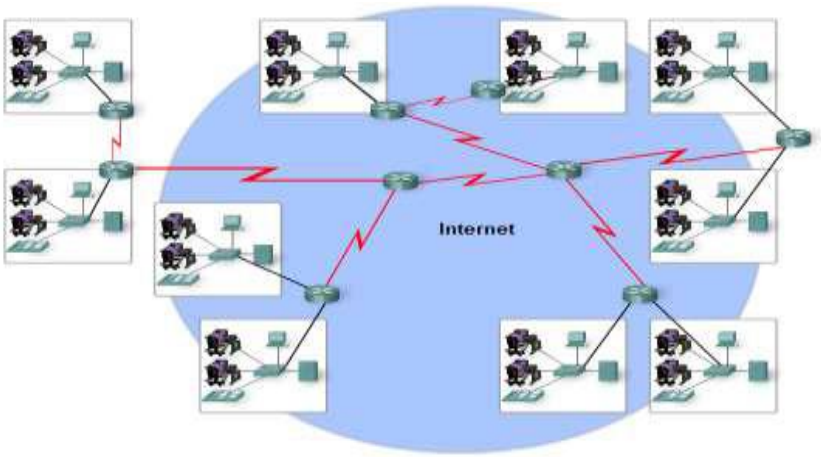
Gambar 2. 2 MAN (*Metropolitan Area Network*)

3. **WAN *(Wide Area Network)***, jangkauannya mencakup daerah geografis yang luas seringkali mencakup negara bahkan benua. Teknologi yang digunakan kurang lebih sama dengan LAN.



Gambar 2. 3 WAN (*Wide Area Network*)

4. **INTERNET *(Interconnected Network)***, jangkauannya mencakup seluruh dunia yang merupakan gabungan dari LAN, MAN, dan WAN yang ada.



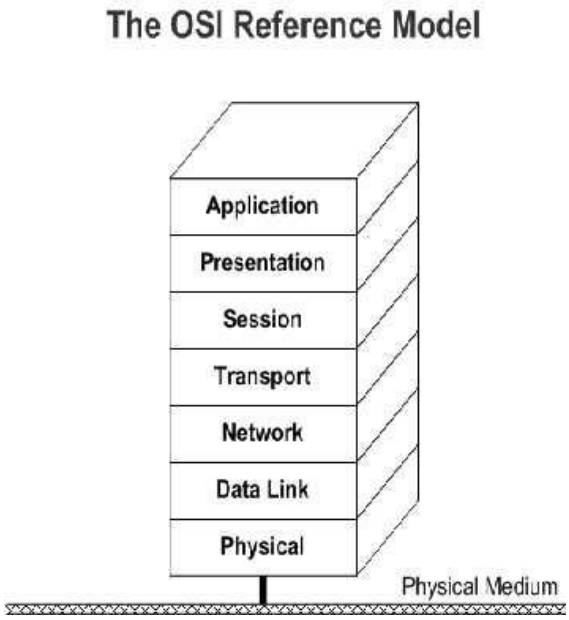
Gambar 2. 4 Internet

2.1.4 Model *OSI Layer*

Model interkoneksi sistem terbuka, yang lebih dikenal sebagai model OSI, yang artinya peta jaringan yang pada awalnya dikembangkan sebagai standar universal untuk membuat jaringan. Tetapi alih-alih melayani sebagai model dengan protokol yang disetujui yang akan digunakan di seluruh dunia, model OSI telah menjadi alat pengajaran yang menunjukkan bagaimana tugas yang berbeda dalam suatu jaringan arus ditangani untuk mempromosikan data terbebas dari kesalahan transmisi. Interkoneksi Sistem Terbuka model (OSI Model) yaitu model konseptual yang mencirikan dan menstandardisasi fungsi internal sebuah sistem komunikasi dengan mempartisipasi ke lapisan abstraksi. (Chinmay, Vibhu; Garg, 2015)

Model OSI bukan berupa protokol, tetapi merupakan referensi model, atau struktur abstrak yang menggambarkan fungsi dan interaksi berbagai data protokol

komunikasi. Ini memberikan konseptual struktur yang membantu mendiskusikan dan membandingkan jaringan fungsi, seperti bantuan sistem klasifikasi lainnya. (Chinmay, Vibhu; Garg, 2015)



Gambar 2. 5 Model *OSI Layer*

Tujuh lapis model OSI adalah :

1. Fisik (*Physical Layer*)

Lapisan Fisik adalah kabel, serat, kartu yang sebenarnya, switch, dan mekanik serta peralatan listrik lainnya yang membentuk jaringan. Lapisan ini mengubah data digital menjadi sinyal yang dapat dikirim menyusuri kabel untuk mengirim data. Sinyal-sinyal ini sering berupa listrik, tetapi seperti dalam kasus serat optik bisa juga menjadi sinyal non-listrik seperti optik atau jenis-jenis pulsa lain yang dapat dikodekan secara digital. (Chinmay, Vibhu; Garg, 2015)

2. Hubungan data (*Data Link Layer*)

Data Link Layer adalah tempat informasi dikonversi ke dalam "paket" koheren dan *frame* diteruskan ke lapisan yang lebih tinggi. Pada dasarnya, lapisan *data link* membongkar data mentah yang berasal dari lapisan fisik dan menerjemahkan informasi dari lapisan atas menjadi data mentah yang akan dikirim melalui *physical layer*. Tautan data layer juga bertanggung jawab untuk menangkap serta kompensasi untuk menjalankan kesalahan yang terjadi di lapisan fisik. (Chinmay, Vibhu; Garg, 2015)

3. Jaringan (*Network Layer*)

Lapisan jaringan adalah tempat untuk mengatur data masuk dan data keluar. Lapisan jaringan di mana router bekerja untuk memastikan bahwa data benar-benar diatasi kembali sebelum meneruskan ke perjalanan paket berikutnya. (Chinmay, Vibhu; Garg, 2015)

4. Transportasi (*Transport Layer*)

Lapisan transportasi bertanggung jawab untuk *streaming* data di seluruh jaringan. Pada level ini, datanya tidak memikirkan dalam hal paket individu tetapi lebih dalam hal percakapan. Untuk menyempurnakannya, protokol ini didefinisikan sebagai aturan komunikasi yang dipakai. Protokol menyaksikan transmisi lengkap yang memiliki banyak paket kemudian memeriksa kesalahan dalam percakapan, mengakui transmisi yang sukses dan meminta pengiriman ulang jika kesalahan terdeteksi. (Chinmay, Vibhu; Garg, 2015)

5. Sesi (*Session Layer*)

Lapisan sesi adalah tempat dimana koneksi dibuat, dirawat dan diakhiri. Ini biasanya mengacu pada permintaan aplikasi untuk data melalui jaringan. Sedangkan lapisan *transport* menangani aliran aktual data, sedangkan lapisan sesi bertindak sebagai penyiar, memastikan bahwa program dan aplikasi yang diminta dan mengirim data permintaan yang sedang diisi. Pada istilah teknis, lapisan sesi mensinkronkan data transmisi. (Chinmay, Vibhu; Garg, 2015)

6. Presentasi (*Presentation Layer*)

Lapisan presentasi adalah tempat data diterima kemudian diubah menjadi format aplikasi yang dituju untuk bisa dipahamin. Pekerjaan yang paling bagus dilakukan pada lapisan ini yaitu sebagai penerjemah. Contoh, data sering dienkripsi pada lapisan presentasi sebelum diteruskan ke lapisan lain. Ketika data diterima, akan didekripsi dan diteruskan ke aplikasi yang ditujukan untuk masuk format yang diharapkan. (Chinmay, Vibhu; Garg, 2015)

7. Aplikasi (*Application Layer*)

Lapisan aplikasi mengkoordinasi akses jaringan untuk perangkat lunak yang dijalankan pada komputer atau alat tertentu. Protokol pada lapisan aplikasi menangani permintaan aplikasi perangkat lunak yang berbeda yang membuat ke jaringan. Jika pada *browser* ingin unduh sebuah gambar, klien email ingin memeriksa server dan pada program *file-sharing* yang ingin mengunggah sebuah film, protokol di lapisan aplikasi akan mengatur dan melaksanakan permintaan ini. (Chinmay, Vibhu; Garg, 2015)

2.2 Teori Khusus

Hal-hal yang akan dimuat pada teori khusus yaitu menjelaskan alat, media, atau variabel yang dipakai dan mendukung materi penelitian ini. Dengan dijelaskannya materi pada alat, media, atau variabel yang dipakai tentu dapat menjelaskan serta menjawab latar belakang penelitian ini.

2.2.1 *Wireshark*

Wireshark menurut (Diansyah, 2015) seperti media atau alat yang dapat digunakan oleh *user*, apakah untuk kebaikan ataupun kejahatan. Hal tersebut dikarenakan *wireshark* dapat dipakai untuk menggali informasi yang sensitif yang berkeliaran pada jaringan, misalnya *password*, *cookies*, dan lain-lain.

Wireshark Network Protocol Analyzer berguna sebagai pengamatan pada *frame* Ethernet. Program dipasang di komputer yang menjalankan Windows XP. Program ini bisa dipakai sebagai *sniffer*, yang artinya alat untuk menangkap komunikasi jaringan dalam waktu nyata untuk melihat hasil dalam *offline mode* dan menilai perilaku *frame* yang diambil sebagai *file* (dengan ekstensi. topi). Untuk pengamatan *frame* Ethernet yang benar melewati *port* switch, switch Ethernet membutuhkan harus dikonfigurasikan sehingga paket dikirim melalui *port* kemudian menghubungkan sakelar ke perangkat ‘‘*sniffing*” (PC) (Kowalik, Rasolomampionona, & Januszewski, 2017).

2.2.2 *Denial-of-Service* (DoS)

Segala peristiwa yang mengurangi, mengganggu atau sepenuhnya menghilangkan komunikasi jaringan dikategorikan sebagai serangan *Denial-of-Service* (DoS). Untuk jaringan informasi apa pun, 'ketersediaan perangkat' merupakan faktor yang paling penting, dan serangan DoS menargetkan 'ketersediaan jaringan' dengan mencegah komunikasi antar perangkat jaringan dari pengaksesan layanan yang disediakan. Dengan demikian, serangan DoS dianggap sebagai masalah keamanan yang penting. Serangan-serangan ini dapat diinisialisasi dari tempat-tempat terpencil dengan perintah belaka, dikombinasikan dengan alat canggih; penyerang bahkan bisa melakukan serangan *Distributed Denial-of-Service* (DDoS), yang efisien dalam menjatuhkan jaringan yang besar. Agak sulit untuk menemukan serangan DoS sebelum layanan belum tersedia. (Kasinathan, Pastrone, Spirito, & Vinkovits, 2013)

Berbeda dengan serangan DDoS, serangan *Distributed Denial-of-Service* (DDoS) merupakan salah satu jenis serangan DoS yang menggunakan banyak *host* sebagai penyerang. Penerapan DDoS yaitu dengan memakai komputer "*zombie*" untuk membuat sejumlah paket data, sehingga serangan menjadi terkoordinasi dan karena berasal dari beberapa komputer *zombie* pada waktu yang sama, bahkan serangan ini dapat menghancurkan. (Al-munawar & Sediyono, 2017)

Serangan DoS dibagi menjadi 3 kategori utama (Al-munawar & Sediyono, 2017):

1. Serangan Berbasis Volume

Serangan ini termasuk *ICMP Flood, UDP Flood* dan serangan paket palsu lainnya. Tujuan utama dari penyerang adalah untuk mengkonsumsi *bandwidth* dari situs target. Besarnya serangan diukur dalam *bits per second* (Bps).

2. Serangan Berbasis Protokol

Serangan ini termasuk *SYN Flood,* serangan paket terfragmentasi, *ping of death*, *smurf attack*, dsb. Tujuan utama dari penyerang adalah untuk mengkonsumsi sumber daya *server* sebenarnya, seperti *firewall*. Besarnya serangan diukur dalam paket per detik.

3. Serangan Berbasis Aplikasi

Serangan ini termasuk serangan seperti *low-and-slow rate attack*, *GET/POST flood,* serangan yang menargetkan kerentanan Apache, Windows atau open BSD dan banyak lagi. Berisi *request-request* yang tampaknya "sah", tujuan dari serangan ini adalah untuk membuat *web server crash*, dan besarnya serangan ini diukur dalam *request* per detik.

Ada beberapa jenis serangan DoS dimana semua serangan tersebut merupakan serangan yang bahaya, jenis-jenis serangan DoS menurut beberapa pakar, yaitu:

1. SYN *Flood Attack*

SYN *flood attack* memanfaatkan kerentanan jabat tangan tiga arah TCP, sehingga server harus berisi struktur data yang besar untuk paket SYN yang masuk tanpa memandang keasliannya. Selama SYN *flood attack*, paket SYN dikirim oleh penyerang dengan alamat IP sumber yang tidak dikenal. Jabat tangan tiga arah terjadi ketika server menyimpan informasi permintaan dari klien ke dalam tumpukan memori dan kemudian menunggu konfirmasi dari klien. Mengingat bahwa alamat IP sumber dalam SYN *flood attack* tidak diketahui, paket konfirmasi untuk permintaan yang dibuat oleh SYN *flood attack* tidak diterima. Setiap koneksi setengah terbuka mengakumulasi di tumpukan memori sampai habis, karena itu tumpukan memori menjadi penuh. Akibatnya, tidak ada permintaan yang dapat diproses, dan layanan sistem dinonaktifkan. Dengan demikian, SYN *flood attack* dianggap sebagai salah satu metode *flood* yang paling kuat. (Alomari, Manickam, Gupta, Karuppayah, & Alfaris, 2012)

2. ICMP (*Internet Control Message Protocol*) *Flood Attack*

ICMP didasarkan pada protokol IP yang dapat mendiagnosis status jaringan. ICMP *flood attack* adalah serangan *bandwidth* yang menggunakan paket ICMP yang dapat diarahkan ke mesin individual atau ke seluruh jaringan. Ketika sebuah paket dikirim dari mesin ke alamat *broadcast* IP di jaringan lokal, semua mesin di jaringan menerima paket tersebut. Ketika sebuah paket dikirim dari mesin ke alamat *broadcast* IP di luar jaringan lokal, paket tersebut dikirim ke semua mesin di jaringan target. (Alomari et al., 2012)

3. HTTP *Flood Attack*

Serangan yang membombardir server Web dengan permintaan HTTP disebut HTTP *flood attack*. Untuk mengirim permintaan HTTP, koneksi TCP yang valid yang memerlukan alamat IP asli harus dibuat. Penyerang mengirim permintaan HTTP melalui alamat IP bot dan kemudian merumuskan permintaan HTTP dengan cara yang berbeda untuk memaksimalkan kekuatan serangan serta menghindari deteksi. Misalnya seorang penyerang dapat memanipulasi *Botnet* untuk mengirim permintaan HTTP untuk mengunduh file besar dari target. File tersebut kemudian dibaca oleh target dari *hard disk*, disimpan dalam memori, dan akhirnya dimuat ke dalam paket, yang dikirim kembali ke *Botnet*. Oleh karena itu, permintaan HTTP sederhana dapat secara signifikan mengonsumsi sumber daya dalam CPU, memori, perangkat *input* / *output*, dan tautan Internet luar. (Alomari et al., 2012)

4. *Smurf Attack*

Serangan ini bergantung pada jaringan yang salah terkonfigurasi perangkat yang memungkinkan pesan dikirim ke semua kendaraan host di jaringan tertentu melalui alamat *broadcast* jaringan. Jaringan tersebut kemudian berfungsi sebagai penguat *smurf*. Dalam serangan seperti ini, pelaku akan mengirim pesan IP yang banyak ke alamat respon yang dipalsukan tampak seperti alamat kendaraan korban. (Verma, Hasbullah, & Kumar, 2012)

5. Ping Flood

Serangan ini didasarkan pada pengiriman korban yang membawa sejumlah pesan *ping* yang banyak. Ini sangat mudah diluncurkan, syarat utamanya adalah akses ke *bandwidth* yang lebih besar daripada kendaraan korban. (Verma et al., 2012)

6. UDP Flood

Contoh UDP *flood* meliputi "serangan rapuh". Dalam serangan rapuh seorang penyerang mengirimkan paket yang banyak ke UDP *echo traffic* ke alamat IP *broadcast*, semuanya memiliki alamat sumber yang palsu. Ini adalah penulisan ulang sederhana dari kode serangan *smurf*. (Verma et al., 2012)

UDP *flooding* adalah jenis serangan *flooding* dimana banyak Datagram UDP dihasilkan oleh *malicious system* (bot). Datagram UDP ini membanjiri seluruh jaringan, dan ketika mencapai pada suatu sistem menyebabkan kemacetan. Datagram UDP ini mudah membanjiri jaringan tanpa batasan apa pun karena setiap sistem deteksi bekerja berdasarkan dua kriteria yaitu deteksi berbasis tanda dan deteksi berbasis fitur. Meskipun kedua kriteria tersebut merupakan datagram UDP, maka gerak-gerik serangan tampak aman dan dengan mudah melewati sistem deteksi. (Bijalwan, Wazid, Pilli, & Joshi, 2015)

UDP (*User Datagram Protocol*) *flooding* adalah jenis serangan yang memanfaatkan protokol UDP dengan mengurangi sambungan (*connectionless*) untuk menyerang target. UDP adalah jenis serangan yang cukup mudah dilakukan dibandingkan dengan jenis serangan lain. UDP *flooding* dapat menyebabkan

komputer server yang menjadi target mengalami *hang* akibat besarnya paket data yang diterima komputer server tersebut. (Aprilianto, Fadila, & Muslim, 2017)

Ada beberapa aplikasi atau *tools* serangan DoS yang biasanya digunakan oleh penyerang untuk mengeksekusi serangan tersebut, aplikasi atau *tools* serangan DoS menurut beberapa pakar, yaitu:

1. *Black Energy*

Black Energy adalah bot DoS berbasis web yang digunakan oleh peretas Rusia yang tidak dikenal. *Black Energy* dengan mudah mengontrol bot berbasis web minimal melalui sintaks dan struktur, sehingga menghasilkan berbagai serangan. Satu atau lebih peretas Rusia rupanya mengembangkan alat ini. Sementara itu, sebagian besar sistem *Command & Control Black Energy* dapat terlihat di Malaysia dan di Rusia, dengan situs Rusia menjadi target utama. Salah satu fitur utama yang *Black Energy* bot promosikan di forum adalah kemampuan untuk menargetkan lebih dari satu alamat IP setiap nama host. Alat ini terus digunakan secara luas untuk menolak layanan dari situs Web komersial. (Alomari et al., 2012)

2. *Low Orbit Ion Cannon* (LOIC)

LOIC adalah alat serangan DoS berbasis Botnet yang melepaskan *flood* di server. *Flood* ini tampaknya dihasilkan dari volume lalu lintas HTTP yang besar. Namun, alat ini baru digunakan oleh grup anonim untuk memfasilitasi lalu lintas berbahaya melalui *Zeus Botnet*, yang merupakan program *malware* tingkat lanjut yang tidak dapat dengan mudah dihapus. Kelompok peretas melakukan serangan

terbesar pada 2012 terhadap situs-situs terkenal, seperti Departemen Kehakiman (DOJ) dan Biro Investigasi Federal (FBI). (Alomari et al., 2012)

3. HULK *Script*

Untuk mengimplementasikan serangan DoS dari satu mesin, skrip dapat dibuat dalam berbagai bahasa pemrograman. Aplikasi Phyton sebagai aplikasi dimana skrip memanggil layanan permintaan-respons beberapa kali dalam jumlah waktu tertentu. Skrip HULK awalnya dikembangkan sebagai bukti konsep untuk menggambarkan betapa mudahnya untuk menjatuhkan server *web*. Skrip HULK bekerja dengan membuka permintaan HTTP GET *flood* ke target secara berlebihan. Skrip HULK unik karena setiap permintaan memiliki *header* acak dan nilai parameter URL memintas mesin *caching* server. Serangan ini mengeksekusi serangan ke server dan secara terus menerus dengan membuat permintaan-respon sampai mesin utama menghentikannya. (Mahjabin, 2018)

4. XML-RPC *Script*

Serangan XML-RPC DoS lebih rumit dan lebih berbahaya untuk server *web* dan *cloud*. XML-RPC adalah cara sederhana dan portabel untuk melakukan panggilan prosedur jarak jauh melalui HTTP. Serangan ini dapat digunakan pada aplikasi Perl, Java, Python, C, C ++, PHP, dan banyak bahasa pemrograman lainnya. XML-RPC menggunakan daftar server dari mesin yang tersedia yang sedang dibuat beberapa permintaan ke server. XML-RPC adalah serangan DoS murni karena mengirim permintaan ke *host* untuk menyerang domain tertentu.

XML-RPC dapat menjadi jauh lebih efisien dalam hal *flooding*, membuat server tidak tersedia dan memecah server. (Mahjabin, 2018)

2.2.3 *Waikato Environment for Knowledge Analysis* (WEKA)

WEKA adalah sebuah paket *tools machine learning* praktis. WEKA merupakan singkatan dari Waikato *Environment for Knowledge Analysis*, yang dibuat di Universitas Waikato, New Zealand untuk penelitian, pendidikan dan berbagai aplikasi. WEKA mampu menyelesaikan masalah-masalah *data mining* di dunia nyata, khususnya klasifikasi yang mendasari pendekatan *machine learning*. Perangkat lunak ini ditulis dalam hirarki *class* Java dengan metode berorientasi objek dan dapat berjalan hampir di semua *platform*. (Purnamasari, Henharta, Sasmita, Ihsani, & Wicaksana, 2013)

WEKA mudah digunakan dan diterapkan pada beberapa tingkatan yang berbeda. Tersedia implementasi algoritma pembelajaran *state of the art* yang dapat diterapkan pada dataset dari *command line*. WEKA mengandung *tools* untuk *preprocessing* data, klasifikasi, regresi, *clustering*, aturan asosiasi, dan visualisasi. Pengguna dapat melakukan *preprocess* pada data, memasukkannya dalam sebuah skema pembelajaran, dan menganalisis *classifier* yang dihasilkan dan performanya, semua itu tanpa menulis kode program sama sekali. Contoh penggunaan WEKA adalah dengan menerapkan sebuah metode pembelajaran ke dataset dan menganalisis hasilnya untuk memperoleh informasi tentang data, atau menerapkan beberapa metode dan membandingkan performanya untuk dipilih. (Purnamasari et al., 2013)

2.2.4 Pengklasifikasian Data

Klasifikasi adalah suatu bentuk analisis data yang mengekstraksi model yang menggambarkan kelas data penting. Model semacam ini yang disebut *classifier*, memprediksi label kelas kategorikal (diskrit, tidak berurutan). Banyak metode klasifikasi yang telah diusulkan oleh para peneliti dalam pembelajaran mesin, pengenalan pola, dan statistik. Cara kerja klasifikasi data adalah proses dua langkah yang terdiri dari langkah pembelajaran (di mana model klasifikasi dibangun) dan langkah klasifikasi (di mana model ini digunakan untuk memprediksi label kelas untuk data yang diberikan). (Han, Kamber, & Pei, 2012)

Metode klasifikasi menurut (Han et al., 2012) yang diusulkan oleh para peneliti untuk memudahkan dalam hal pengklasifikasian, bebeerapa metode klasifikasi yaitu:

1. Metode Klasifikasi *Naive Bayes*

Klasifikasi *Bayesian* adalah pengklasifikasian statistik. Klasifikasi ini dapat memprediksi probabilitas keanggotaan kelas seperti probabilitas yang dimiliki *tuple* tertentu ke kelas tertentu. Klasifikasi *Bayesian* didasarkan pada teorema *Bayes*. Klasifikasi *Bayesian* juga menunjukkan akurasi dan kecepatan tinggi ketika diterapkan ke *database* yang besar.

Studi pembanding algoritma klasifikasi telah menemukan *classifier Bayesian* sederhana yang dikenal sebagai *Naive Bayesian Classifier* dimana dapat dibandingkan dalam kinerja dengan pohon keputusan dan saraf yang dipilih

klasifikasi jaringan. *Naive Bayesian Classifier* mengasumsikan bahwa pengaruh nilai atribut pada kelas tertentu tidak tergantung pada nilai atribut lainnya. Asumsi ini disebut kondisi kelas kemerdekaan. Klasifikasi ini dibuat untuk menyederhanakan perhitungan yang terlibat sehingga dalam hal ini dianggap *naive*.

2. Metode Klasifikasi *Backpropagation*

Backpropagation adalah algoritma pembelajaran jaringan saraf atau yang biasa disebut *neural network*. Bidang jaringan saraf awalnya diciptakan oleh psikolog dan neurobiologis yang berusaha mengembangkan dan menguji analog neuron komputasi. Secara kasar, jaringan saraf adalah sekumpulan unit *input* / *output* yang terhubung dimana setiap koneksi memiliki bobot yang terkait dengannya. Selama fase pembelajaran, jaringan saraf ini belajar dengan menyesuaikan bobot sehingga dapat memprediksi label kelas yang benar dari *input tuple*. Jaringan saraf melibatkan waktu pelatihan yang lama, membutuhkan sejumlah parameter yang biasanya baik untuk ditentukan secara empiris seperti topologi jaringan atau "struktur."

3. Metode Klasifikasi SVM (*Support Vector Machine*)

Klasifikasi *Support Vector Machine* (SVM), yaitu metode untuk klasifikasi data linier dan nonlinier. Singkatnya, SVM adalah algoritma yang bekerja mengikuti arus. Metode ini menggunakan pemetaan nonlinear untuk mengubah data pelatihan asli ke dimensi yang lebih tinggi. Dalam dimensi baru ini, metode ini mencari linear optimal yang memisahkan *hyperplane* (“batas keputusan" yang

memisahkan tupel dari satu kelas dari yang lain). Dengan pemetaan nonlinear yang sesuai untuk dimensi yang cukup tinggi, data dari dua kelas selalu dapat dipisahkan oleh *hyperplane*. SVM menemukan *hyperplane* ini yang menggunakan vektor dukungan (tupel pelatihan "penting") dan *margin* (ditentukan oleh vektor dukungan).

4. Metode Klasifikasi *K-Nearest-Neighbor*

Metode klasifikasi *K-Nearest-Neighbor* pertama kali dijelaskan pada awal 1950-an. Metodenya memiliki karya yang padat ketika diberi set pelatihan yang besar dan tidak mendapatkan popularitas sampai tahun 1960-an ketika peningkatan daya komputasi tersedia. Sejak itu telah banyak digunakan di bidang pengenalan pola. Klasifikasi *K-Nearest-Neighbor* didasarkan pada pembelajaran dengan analogi, yaitu dengan membandingkan tupel tes yang diberikan dengan tupel pelatihan yang serupa dengannya.

2.3 *Tools*

Peralatan yang digunakan untuk merancang jaringan komputer pada penelitian ini yaitu:

1. 3 Laptop
2. *Switch* TL-SF1005D
3. Kabel UTP (*Unshielded Twisted Pair*)

2.4 Penelitian Terdahulu

Hasil-hasil penelitian terdahulu yang dilakukan oleh peneliti lain, dapat dijadikan acuan atau referensi buat peneliti untuk mengembangkan dan memperdalam penelitian yang sedang dilakukan. Beberapa penelitian terdahulu yang berupa jurnal yang dijadikan referensi dalam penelitian ini adalah:

Penelitian terdahulu pertama adalah penelitian yang berjudul "Analisa Pencegahan Aktivitas Ilegal Didalam Jaringan Menggunakan *Wireshark*" pada tahun 2015 (ISSN 2337-3601) yang ditulis oleh Tengku Mohd Diansyah. Hasil penelitian yang dilakukan dapat disimpulkan bahwa dari data yang didapatkan mengenai protokol jaringan hasil dari penyaringan paket data menggunakan *wireshark* yaitu pada *wireshark* untuk menyaring paket. Caranya mudah, pada *wireshark* hanya memilih filter paket pada kolom filter, sehingga *administrator* jaringan dapat menganalisa paket jaringan yang sedang berjalan.

Penelitian terdahulu kedua adalah penelitian yang berjudul "Karakteristik Konsumsi Daya Komputer dengan Perubahan Tingkat Serangan *Distributed Denial of Service* (DDoS)" pada tahun 2017 (ISSN 2540-7589) yang ditulis oleh Nawirah Al-Munawar dan Agung Sediyono. Hasil penelitian yang dilakukan dapat disimpulkan bahwa dari hasil percobaan DDoS yang dilakukan menggunakan LOIC pada penelitian ini, semakin besar *thread* maka *request* yang dikirimkan komputer penyerang semakin melambat dan komputer penyerang dapat mengalami *hang*.

Penelitian terdahulu ketiga adalah penelitian yang berjudul "*Forensics of Random-UDP Flooding Attacks*" pada tahun 2015 (DOI 10.4304/jnw.10.5.287-293) yang ditulis oleh Anchit Bijalwan, Mohammad Wazid, Emmanuel S. Pilli, dan R. C. Joshi. Hasil penelitian yang dilakukan dapat disimpulkan bahwa serangan *flooding* dapat dengan mudah menyebabkan serangan DoS. Serangan *flooding Random*-UDP adalah sebuah jenis serangan yang berbeda dimana penyerang mengirim beberapa datagram UDP dengan ukuran yang berbeda setiap waktu. Sistem dibanjiri oleh beberapa datagram UDP, yang menyebabkan penolakan layanan ke sistem dan sumber daya. Dengan bantuan teknik yang diusulkan, dapat dengan mudah mengidentifikasi sumber serangan tersebut. Pekerjaan ini dapat diperluas hingga ke rancangan kerangka kerja berbasis pembelajaran untuk forensik *flooding* lainnya yang dihasilkan oleh serangan *zero day*.

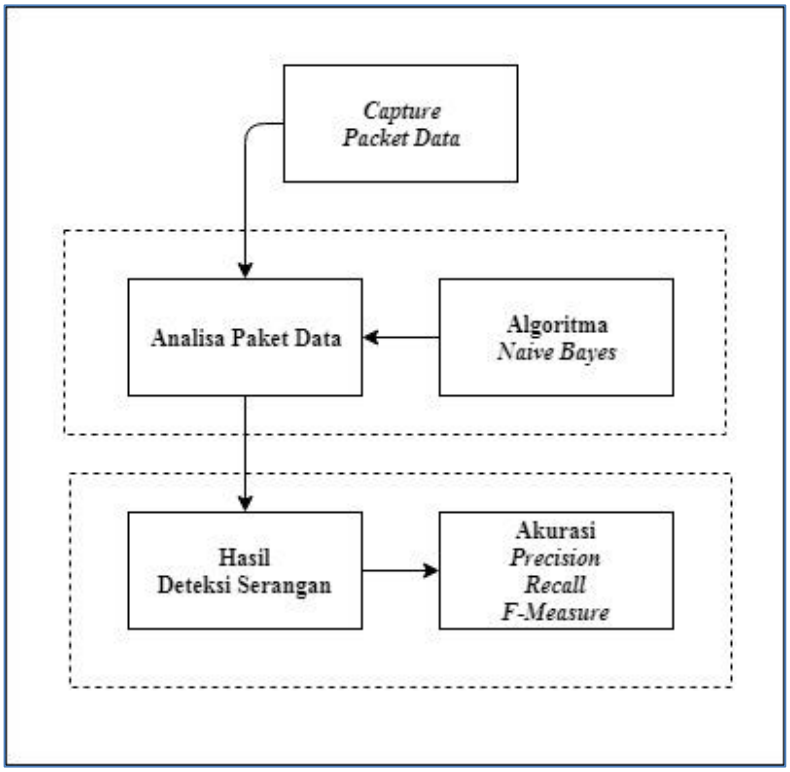
Penelitian terdahulu keempat adalah penelitian yang berjudul "*An Efficient Defense Method against UDP Spoofed Flooding Traffic of Denial of Service (DoS) Attacks in VANET*" pada tahun 2012 (ISBN 978-1-4673-4529-3) yang ditulis oleh Karan Verma, Halabi Hasbullah, dan Ashok Kumar. Hasil penelitian yang dilakukan dapat disimpulkan bahwa untuk bertahan melawan *spoofed flooding attack*, mengusulkan metode yang efisien yang bisa mendeteksi dua jenis *spoofing* yaitu *random spoofing* dan *subnet spoofing*. Ketika serangan malicious terdeteksi, metode ini dapat mengklasifikasikan lebih jauh menjadi *random spoofing*, *subnet spoofing*, atau diperbaiki tipe *spoofing* dengan menganalisis tabel *hash* untuk IP sumber karakteristik.

Simulasi eksperimen menunjukkan bahwa metode yang diusulkan menghasilkan deteksi dan hasil klasifikasi dengan biaya komputasi yang rendah.

Penelitian terdahulu kelima adalah penelitian yang berjudul "Sistem Pencegahan UDP DNS Flood dengan *Filter Firewall* Pada Router Mikrotik" pada tahun 2017 (ISSN 2356-2579) yang ditulis oleh Doni Aprilianto, Triyana Fadila, dan Much Aziz Muslim. Hasil penelitian yang dilakukan dapat disimpulkan bahwa penerapan *filter firewall* pada router mikrotik dapat mengurangi jumlah paket data UDP yang dikirimkan oleh *attacker* melalui port DNS sebanyak 60% dari jumlah paket yang masuk jika tanpa *firewall*. Sehingga router dapat berjalan secara normal kembali.

2.5 Kerangka Pemikiran

Kerangka pemikiran merupakan langkah-langkah yang telah dirumuskan untuk mendapatkan kesimpulan dari masalah yang diteliti. Berikut gambaran sekilas skema kerangka pemikiran.



Gambar 2. 6 Kerangka Pemikiran

Berdasarkan skema kerangka pemikiran diatas, tahapan yang akan dilakukan meliputi langkah-langkah berikut :

1. *Capture Packet Data*, ialah proses menangkap informasi yang mengalir dalam jaringan komputer.
2. Analisa Paket Data, merupakan proses pendeteksian data normal dan data serangan dengan menggunakan aplikasi WEKA.
3. Algoritma *Naive Bayes*, yaitu rumus pengklasifikasian data untuk mengklasifikasi data mana yang termasuk normal dan data mana yang termasuk serangan.

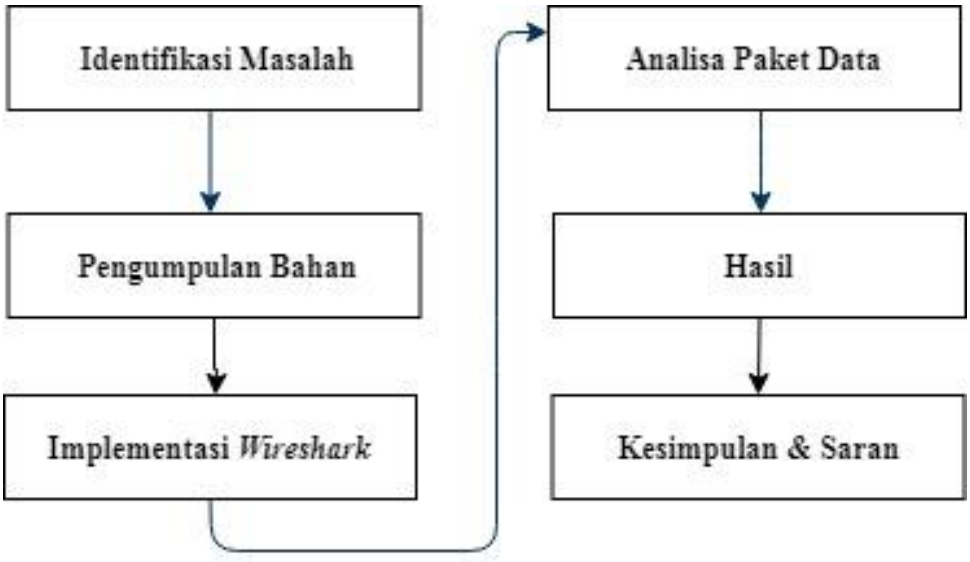
4. Hasil Deteksi Serangan, yaitu hasil deteksi menggunakan algoritma *naive bayes* dengan menggunakan aplikasi WEKA sehingga mendapatkan hasil data yang berupa normal dan data yang berupa serangan.
5. Akurasi, *precision*, *recall*, dan *F-Measure*, ialah hasil metode evaluasi yang diuji untuk mengklasifikasikan data normal dan data serangan, sehingga menghasilkan akurasi, *presicion*, *recall*, dan *F-Measure* guna untuk menganalisa paket data yang telah di *capture*.

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Desain Penelitian adalah gambaran sebuah penelitian yang dilakukan oleh peneliti untuk menyelesaikan masalah yang dirumuskan. Dengan adanya desain penelitian, maka pembaca akan mudah mendapat gambaran yang peneliti akan lakukan dari tahap awal hingga selesai. Desain penelitian analisa permasalahan jaringan komputer LAN menggunakan aplikasi *wireshark* digambarkan dibawah ini.



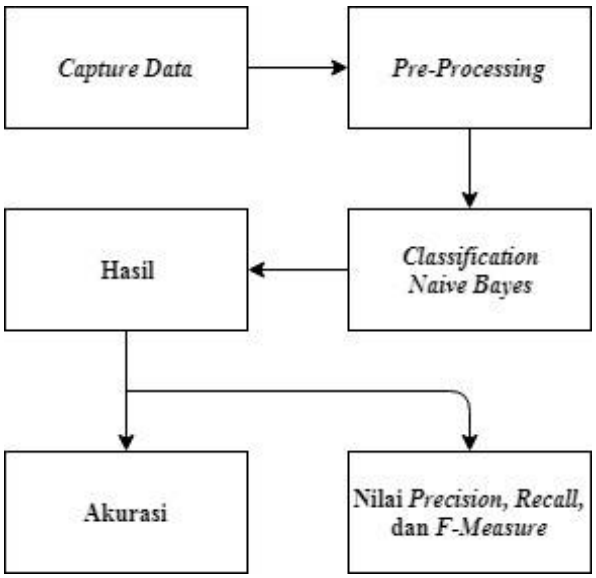
Gambar 3. 1 Desain Penelitian

Berikut adalah pembahasan dari gambar di atas :

1. Identifikasi masalah, merupakan dasar dalam penelitian yang sudah dibahas pada bab 1.
2. Pengumpulan bahan, yang dibutuhkan dalam sebuah penelitian dari segi *hardware* dan *software*.
3. Implementasi *wireshark*, yaitu aplikasi untuk *capture* data dari jaringan komputer yang dianalisa.
4. Analisa paket data, menganalisa data yang telah di *capture* dari aplikasi *wireshark* untuk mengetahui serangan yang datang kemudian diklasifikasikan dengan aplikasi WEKA untuk mengetahui data mana yang termasuk normal dan yang mana termasuk serangan.
5. Hasil, setelah dianalisa paket data maka akan muncul hasil analisa.
6. Pengambilan kesimpulan.

3.2 Arsitektur Pengklasifikasian Data dalam Mendeteksi Serangan

Arsitektur pengklasifikasian data dalam mendeteksi serangan ialah langkah umum dalam pengklasifikasian data untuk mendapatkan hasil yang mudah dibaca. Arsitektur pengklasifikasian data digambarah dibawah ini.



Gambar 3. 2 Arsitektur Pengklasifikasian Data

Berikut adalah pembahasan dari gambar diatas:

1. *Capture Data*, yaitu menangkap setiap gerak-gerik yang dilakukan dalam sebuah jaringan menggunakan aplikasi *wireshark*.
2. *Pre-Processing*, yaitu mengubah hasil *capture* yang dilakukan *wireshark* kedalam format yang mudah dipahami. Pada tahap ini, ada 2 fitur yang diubah menjadi numerik sebelum melakukan *classification*, 2 fitur tersebut yaitu IP dan Protokol
3. *Classification Naive Bayes*, data-data yang sudah dikonversikan menjadi angka numerik dapat dilakukan klasifikasi, pada penelitian ini metode klasifikasi yang dipakai ialah metode *Naive Bayes*.

4. Hasil, setelah melakukan klasifikasi maka akan muncul hasil dalam bentuk *confusion matrix* dan menghasilkan juga nilai akurasi, presisi, *recall*, dan *F-Measure*.
5. Akurasi, yaitu hasil keakuratan yang dilakukan pada penerapan klasifikasi ini dalam bentuk persentase dengan membandingkan nilai prediksi dengan nilai sesungguhnya.
6. Nilai *Precision*, *Recall*, dan *F-Measure*, yaitu hasil yang ditemukan setelah melakukan klasifikasi dimana *precision* merupakan tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem, *recall* ialah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi, sedangkan *F-Measure* merupakan salah satu perhitungan evalusasi dalam informasi temu kembali yang mengkombinasikan *recall* dan *precision*.

3.3 *Feature Extraction*

Feature Extraction yang digunakan pada data penelitian terdiri dari 6 atribut dan 1 kelas atau label, berikut perinciannya:

Tabel 3. 1 Tabel *Feature Extraction*

No	Keterangan	Tipe Data	Atribut / Kelas
1	*Time*	*Time*	Atribut
2	*Source*	*Decimal*	Atribut
3	*Destination*	*Decimal*	Atribut
4	*Protocol*	*Number*	Atribut
5	*Length*	*Decimal*	Atribut
6	Label	*Text*	Kelas

3.4 Metode Klasifikasi *Naive Bayes*

Sesuai dengan teori yang sudah dijelaskan pada bab sebelumnya, maka pada penelitian ini menggunakan metode klasifikasi *Naive Bayes*, dimana rumusnya ialah:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

Rumus 3. 1 Rumus *Naive Bayes*

Keterangan :

X = Data dengan class yang belum diketahui.

C = Kelas

P(C|X) = Probabilitas Hipotesis berdasarkan kondisi (*Posteriori Probability*)

P(C) = Probabilitas Hipotesis (*Prior Probability*)

P(X|C) = Probabilitas berdasarkan kondisi pada hipotesis

P(X) = Probabilitas C

Jika diberikan set data dengan banyak atribut, maka perhitungan atau analisa akan susah untuk dilakukan. Sehingga dibentuk asumsi indepedensi (*naive*) dari kondisi kelas yang independen. Rumusnya yaitu:

$$P(X|C_i) = P(X_1|C_i)x\ P(X_2|C_i)\ x\ \dots x\ P(X_n|C_i)$$

atau

$$P(X|C_i) = \prod_{k=1}^{n} P(X_k|C_i)$$

Rumus 3. 2 Rumus *Naive Bayes* jika Memiliki Banyak Atribut

Apabila saat melakukan perhitungan dan ada yang memiliki hasil angka nol, maka gunakan *Laplacian Correction* atau biasa disebut juga sebagai *Laplacian Estimator* untuk menghindari perhitungan nilai yang probabilitas nol. Caranya yaitu dengan menambah 1 pada sampel setiap atribut dan pada menambah jumlah atribut (pada rumus disebut k) pada jumlah sampel pada kelas. Rumusnya yaitu:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i) + 1}{P(X) + k}$$

Rumus 3. 3 Rumus *Laplacian Correction*

3.5 Metode Evaluasi

Penelitian ini menggunakan metode *confusion matrix* untuk melakukan evaluasi. *Confusion matrix* adalah alat yang berguna untuk menganalisis seberapa baik *classifier* mengenali *tuple* dari kelas yang berbeda. (Han et al., 2012) Nilai dari *True-Positive* dan *True-Negative* memberikan informasi ketika *classifier* melakukan klasifikasi data bernilai benar, sedangkan *False-Positive* dan *False-Negative* memberikan informasi ketika *classifier* salah dalam melakukan klasifikasi data.

Tabel 3. 2 Tabel *Confusion Matrix*

Prediksi / Aktual	Normal	DoS	Total
Normal	TP	FN	P
DoS	FP	TN	N
Total	P’	N’	P + N or P’ + N’

Dari tabel diatas dapat diterangkan bahwa :

1. TP (*True Positives*) = Merujuk pada tupel positif yang diberi label dengan benar oleh pengklasifikasi, maka TP menjadi jumlah positif sebenarnya.
2. TN (*True Negatives*) = Merujuk pada tupel negatif yang diberi label dengan benar oleh penggolong, maka TN menjadi jumlah negatif yang sebenarnya.

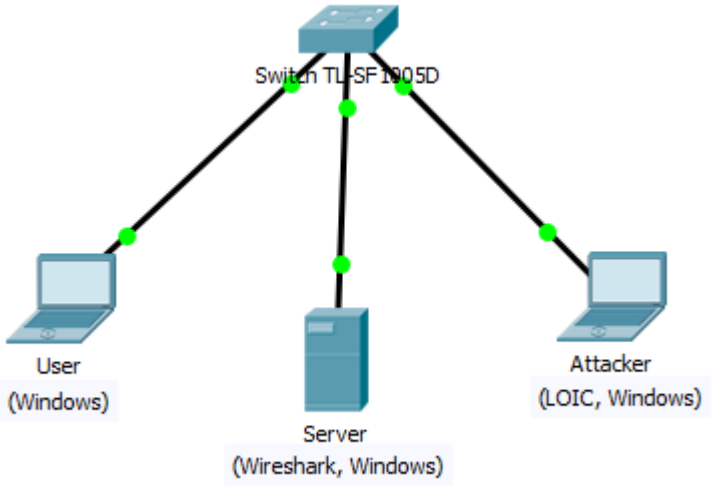
3. FP (*False Positives*) = Merujuk pada tupel negatif yang salah diberi label sebagai positif, maka FP menjadi jumlah positif yang palsu.
4. FN (*False Negatives*) = Merujuk pada tupel positif yang disalahartikan sebagai negatif, maka FN menjadi jumlah negatif yang palsu
5. P' = Total jumlah dari TP dan FP.
6. N' = Total jumlah dari FN dan TN.

Dari hasil *confusion matrix*, dapat diperhitungkan *recall*, akurasi, *F-measure*, dan *precision* untuk menganalisa kinerja algoritma dalam melakukan klasifikasi untuk mendeteksi serangan DoS dengan persamaan :

1. *Recall* $= \frac{TP}{P}$ **Rumus 3. 4** Rumus *Recall*
2. Akurasi $= \frac{TP+TN}{P+N}$ **Rumus 3. 5** Rumus Akurasi
3. *F-measure* $= \frac{2\ x\ Precision\ x\ Recall}{Precision+Recall}$ **Rumus 3. 6** Rumus *F-Measure*
4. *Precision* $= \frac{TP}{TP+FP}$ **Rumus 3. 7** Rumus *Precision*

Akurasi menunjukkan kedekatan hasil pengukuran dengan nilai sesungguhnya; *precision* merupakan tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem; *recall* ialah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi; sedangkan *F-Measure* merupakan salah satu perhitungan evalusasi dalam informasi temu kembali yang mengkombinasikan *recall* dan *precission*. (Han et al., 2012)

3.6 Topologi Jaringan Pengujian Serangan DoS



Gambar 3. 3 Topologi Jaringan untuk Melakukan Pengujian

Topologi jaringan yang digunakan untuk pengujian serangan DoS yaitu topologi *star* dimana topologi jaringan ini terdiri dari 1 *switch* TL-SF1005D, 3 laptop, dan kabel UTP. Pada pengujian ini, setiap laptop memiliki peran yang berbeda yaitu ada yang sebagai *user*, *server*, dan *attacker*. Aplikasi *wireshark* dijalankan di laptop server, kemudian pada laptop *attacker* menggunakan aplikasi LOIC untuk melakukan serangan DoS ke laptop server. Setelah melakukan serangan maka akan terekam pada *wireshark*, hasil capture *wireshark* dilakukan pengujian di aplikasi WEKA memakai algoritma *naive bayes* untuk pendeteksian serangan serta mendapatkan hasil akurasi dan klasifikasi. IP pada setiap laptop dirincikan sebagai berikut:

Tabel 3. 3 Alamat IP setiap Laptop

No	Laptop	IP
1	User	169.254.62.38
2	Server	169.254.74.248
3	*Attacker*	169.254.3.22

3.7 Lokasi dan Jadwal Penelitian

Pada penelitian ini, peneliti melakukan penelitian di rumah. Penelitian ini dilaksanakan mulai dari bulan april 2019 sampai dengan bulan juli 2019, rincian jadwal penelitian yang dilakukan dapat dilihat pada tabel jadwal penelitian berikut ini:

Tabel 3. 4 Jadwal Penelitian

No	**Kegiatan**	**April**				**Mei**				**Juni**				**Juli**			
		1	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1	Identifikasi Masalah																
2	Buat Kerangka Pemikiran																
3	Buat Metode Penelitian																
4	Menganalisa Jaringan																
5	Membuat Kesimpulan																
6	Penyusunan Skripsi																