IMPLEMENTASI MONITORING KEAMANAN JARINGAN MENGGUNAKAN *SNORT* DAN TELEGRAM *BOT* SEBAGAI *NOTIFICATION ALERT*

Disusun dan Diajukan untuk memenuhi Salah Satu Persyaratan Ujian Akhir Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan

SKRIPSI

OLEH

NAMA	: MELFA EVY BELLMONDO
N. P. M	: 1714370372
PROGRAM STUDI	: SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2021

Halaman Pengesahan

PENGESAHAN SKRIPSI

JUDUL : IMPLEMENTASI MONITORING KEAMANAN JARINGAN MENGGUNAKAN SNORT DAN TELEGRAM BOT SEBAGAI NOTIFICATION ALERT

NAMA	: MELFA EVY BELL MONDO
N.P.M	: 1714370372
FAKULTAS	: SAINS & TEKNOLOGI
PROGRAM STUDI	: Sistem Komputer
TANGGAL KELULUSAN	: 21 Agustus 2021

DIKETAHUI

DEKAN

Hamdani, ST., MT.

KETUA PROGRAM STUDI

Eko Hariyanto, S.Kom., M.Kom

DISETUJUI
KOMISI PEMBIMBING

PEMBIMBING I

Akhyar Lubis, S.Kom., M.Kom

PEMBIMBING II

Dian Kurnia, S.Kom., M.Kom

Note : dokumen ini telah diproses secara online melalui sistem informasi akademik UNPAB

UNIVERSITAS PEMBANGUNAN PANCA BUDI
FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO	(TERAKREDITASI)
PROGRAM STUDI ARSITEKTUR	(TERAKREDITASI)
PROGRAM STUDI SISTEM KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI TEKNIK KOMPUTER	(TERAKREDITASI)
PROGRAM STUDI AGROTEKNOLOGI	(TERAKREDITASI)
PROGRAM STUDI PETERNAKAN	(TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

Saya yang bertanda tangan di bawah ini :

Nama Lengkap	: MELFA EVY BELLAMONDO
Tempat Tgl. Lahir	: TUNTUNGAN / 13 Maret 1999
Nomor Pokok Mahasiswa	: 1714370372
Program Studi	: Sistem Komputer
Konsentrasi	: Keamanan Jaringan Komputer
Jumlah Kredit yang telah dicapai	: 141 SKS, IPK 3.86
Nomor Hp	: 082161354804
Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut	:

No.	Judul
1.	Implementasi Monitoring Keamanan Jaringan Menggunakan Snort dan Telegram Bot Sebagai Notification Alert©

Catatan : Diisi Oleh Dosen Jika Ada Perubahan Judul

*Coret Yang Tidak Perlu

Rektor I,

(Cahyo Prayogo, S.E., M.M.)

Medan, 31 Mei 2021

Pemohon

(Melfa Evy Bellamondo)

Tanggal : ..... Disahkan oleh : Dekan	Tanggal : ..... Disetujui oleh : Dosen Pembimbing I :
(Hamdani, ST., MT.)	(Akhmad Lubis, S.Kom., M.Kom)
Tanggal : ..... Disetujui oleh: Ka. Prodi Sistem Komputer	Tanggal : ..... Disetujui oleh: Dosen Pembimbing II:
(Eko Hariyanto, S.Kom., M.Kom)	(Dian Kurnia, S.Kom., M.Kom)

FM-BPAA-2012-041

Hal : Permohonan Meja Hijau

Medan, 14 Februari 2022
Kepada Yth : Bapak/Ibu Dekan
Fakultas SAINS & TEKNOLOGI
UNPAB Medan
Di -
Tempat

Dengan hormat, saya yang bertanda tangan di bawah ini :

Nama	: MELFA EVY BELL MONDO
Tempat/Tgl. Lahir	: Tuntungan / 13 Maret 1999
Nama Orang Tua	: SUTRISNO
N. P. M	: 1714370372
Fakultas	: SAINS & TEKNOLOGI
Program Studi	: Sistem Komputer
No. HP	: 082161354884
Alamat	: JL.BESAR TUNTUNGAN DUSUN III NO:126

Datang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Implementasi Monitoring Keamanan Jaringan Menggunakan Snort dan Telegram Bot sebagai Notification Alert**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntut ujian perbaikan nilai mata kuliah untuk perbaikan indek prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkipnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 examplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 examplar untuk penguji (bentuk dan warna penjilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangani dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	1,000,000
2. [170] Administrasi Wisuda	: Rp.	1,750,000
Total Biaya	**: Rp.**	**2,750,000**

Ukuran Toga : XL

Diketahui/Disetujui oleh :

Hamdani, ST., MT.
Dekan Fakultas SAINS & TEKNOLOGI

Hormat saya

MELFA EVY BELL MONDO
1714370372

Catatan :

- 1.Surat permohonan ini sah dan berlaku bila ;
  - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
  - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2.Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Melfa Evy Bellmondo

NPM : 1714370372

Prodi : Sistem Komputer

Judul Skripsi : **Implementasi Monitoring Keamanan Jaringan Menggunakan *Snort* dan *Telegram Bot* Sebagai *Notification Alert***

Dengan ini menyatakan bahwa :

1. Tugas Akhir/Skripsi saya bukan hasil plagiat
2. Saya tidak akan menuntut perbaikan nilai indeks prestasi (IPK) setelah ujian sidang meja hijau
3. Skripsi saya tidak dapat dipublikasikan oleh pihak lembaga dan saya tidak akan menuntut akibat publikasi tersebut

Demikian pernyataan ini saya buat dengan sebenar-benarnya, terimakasih.

Medan, 27 Januari 2022

Melfa Evy Bellmondo

NPM : 1714370372

SURAT PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar kesarjanaan didalam perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah di tulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis di dalam skripsi ini dan disebutkan dalam daftar pustaka.

Medan, 27 Januari 2022

Melfa Evy Bellmondo

NPM : 1714370372

ABSTRAK

MELFA EVY BELLMONDO

IMPLEMENTASI MONITORING KEAMANAN JARINGAN MENGGUNAKAN *SNORT* DAN TELEGRAM *BOT* SEBAGAI *NOTIFICATION ALERT*

2021

Monitoring Keamanan Jaringan adalah Sebuah kegiatan yang berguna ataupun berfungsi untuk dapat melakukan sebuah pengawasan, pemeriksaan ataupun pengecekan terhadap suatu keamanan jaringan sehingga untuk memastikan ataupun mengontrol jaringan tersebut aman dari sebuah ancaman serangan yang dapat mengancam setiap waktu. Salah satu cara untuk dapat melakukan monitoring terhadap jaringan dapat dilakukan dengan menggunakan sebuah *tools Intrussion Detection System* yaitu *Snort.* Dimana *tools snort* ini dapat digunakan untuk mendeteksi adanya serangan yang terjadi dalam sebuah jaringan seperti ancaman *DOS Attack* berupa *TCP Flooding* ataupun *UDP Flooding.* Jika terdeteksi serangan pada keamanan jaringan, maka Telegram *bot* yang sudah terhubung dengan *tools snort* akan memberitahukan administrator jaringan bahwa keamanan jaringannya sedang tidak baik ataupun sedang mengalami serangan.

Kata Kunci: *Intrussion Detection System, Snort, DOS Attack, Telegram Bot*

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, karena dengan berkat dan rahmat serta kasih anugerah-Nya penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini sebagaimana mestinya. Judul skripsi ini adalah " **Implementasi Monitoring Keamanan Jaringan Menggunakan *Snort* dan Telegram *Bot* Sebagai *Notification Alert*** ". Dalam kesempatan ini, penulis mengucapkan rasa terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini. Penulis ingin mengucapkan terima kasih kepada:

1. Kedua Orang tua saya yang selalu memberikan doa, semangat, dukungan dan motivasi dalam penyusunan skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E., M.M. selaku Rektor Universitas Pembangunanan Panca Budi Medan.
3. Bapak Hamdani, S.T., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
4. Bapak Eko Hariyanto, S.Kom., M.Kom., selaku Ketua Program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
5. Bapak Akhyar Lubis, S.Kom., M.Kom., selaku Dosen Pembimbing I yang telah memberikan arahan serta bimbingan dalam penyelesaian skripsi ini.
6. Bapak Dian Kurnia, S.Kom., M.Kom., selaku Dosen Pembimbing II yang telah memberikan ilmu pengetahuan, serta bimbingan dalam penyelesaian skripsi ini.
7. Seluruh Dosen-dosen Program Studi Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
8. Seluruh staff dan karyawan pada Universitas Pembangunan Panca Budi Medan.
9. Teman-teman penulis dari program studi Sistem Komputer, Fakultas Sains dan Teknologi, Universitas Pembangunan Panca Budi Medan. Terutama yaitu Kak Megawati, Hesti Wulandari, Eki A, Rendi F, Kamaluddin, Mhd Dwi, Ayu K.
10. Terima kasih juga saya ucapkan untuk temen-teman jauh saya yang selalu memberikan Support, doa serta semangat yaitu Diaz Aztisyah, Katherina, Rut, Kathleen, Yulia Wahyuni, Jenice, Nia A, Egia, Mhd Alvin, dan adik Sepupu saya Dianita Sari.

Penulis juga menyadari bahwa penyusunan skripsi ini belum sempurna baik dalam penulisan maupun isi disebabkan keterbatasan kemampuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang sifatnya membangun dari pembaca untuk kesempurnaan isi skripsi ini.

Medan, 31 Agustus 2021

Penulis

Melfa Evy Bellmondo

1714370372

DAFTAR ISI

ABSTRAK

KATA PENGANTAR i

DAFTAR ISI ii

DAFTAR GAMBAR iiv

DAFTAR TABEL v

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Rumusan Masalah 4

1.3 Batasan Masalah 4

1.4 Tujuan Penelitian 5

1.5 Manfaat Penelitian 5

BAB II LANDASAN TEORI 6

2.1 Implementasi 6

2.2 Monitoring 7

2.3 Keamanan Jaringan Komputer 8

2.4 *Intrussion Detection System (IDS)* 8

2.5 *Snort* 10

2.6 Jenis Serangan 11

2.7 Telegram 13

2.8 *Bot* 14

2.9 Telegram *Bot* 15

2.10 *Application Programming Interface (API)* 15

2.11 *Notification Alert* 16

2.12 *Wireless Local Area Network (WLAN)* 16
2.13 *Linux Ubuntu* 17
2.14 *IP Address* 19
2.15 *Virtual Mechine* 20
2.16 *Bash Shell* 21
2.17 *Putty* 22
2.18 *Flowchart* 23
BAB III METODE PENELITIAN 26
3.1 Tahapan Penelitian 26
3.2 Metode Pengumpulan Data 29
3.3 Rancangan Penelitian 30
3.4 Manajemen Biaya 35
3.5 Rancangan *Security* 36
BAB IV HASIL DAN PEMBAHASAN 39
4.1 Kebutuhan Spesifikasi *Hardware* 39
4.2 Kebutuhan Spesifikasi *Software* 41
4.3 Implementasi Sistem 42
4.4 Pengujian dengan Serangan *Denial of Service (DOS)* 43
4.5 Hasil Pengujian Serangan pada server 58
BAB V PENUTUP 60
5.1 Kesimpulan 60
5.2 Saran 61
DAFTAR PUSTAKA 62
LAMPIRAN-LAMPIRAN 67

DAFTAR GAMBAR

Gambar 2.1 Aplikasi Putty.....22

Gambar 3.1 Metode Penelitian menggunakan metode *Waterfall* 26

Gambar 3.2 Topologi Jaringan dari Sistem *IDS* yang dirancang.....31

Gambar 3.3 *Flowchart* Perancangan *Snort IDS* dan Telegram *Bot* 33

Gambar 3.4 *Flowchart* rancangan *security IDS Snort* 37

Gambar 4.1 Tampilan *Software Low Orbit Ion Cannon (LOIC)* 44

Gambar 4.2 Tampilan *tools snort* yang sudah diinstall pada server 45

Gambar 4.3 *Rules snort.* 46

Gambar 4.4 Serangan *TCP flood* terhadap server. 48

Gambar 4.5 Serangan *TCP* terdeteksi *Snort.* 48

Gambar 4.6 Serangan *UDP flood* terhadap server. 49

Gambar 4.7 Serangan *UDP flood* terdeteksi *snort.* 50

Gambar 4.8 Token *API bot.* 53

Gambar 4.9 Program dalam *bot Telegram.* 55

Gambar 4.10 Program dalam *bot Telegram.* 55

Gambar 4.11 *Alert* terkirim ke Telegram *bot.* 56

Gambar 4.12 Notifikasi Telegram atas Serangan *TCP flood.* 57

Gambar 4.13 Notifikasi Telegram atas Serangan *UDP flood.* 57

Gambar 4.11 *Alert* terkirim ke Telegram *bot.* 56

Gambar 4.11 *Alert* terkirim ke Telegram *bot.* 56

DAFTAR TABEL

Tabel 2.1 Simbol *Flowchart* 23

Tabel 3.1 Tabel Pengalamatan *IP Address* 31

Tabel 3.2 Manajemen Biaya 36

Tabel 4.1 Komponen *Hardware* atau perangkat keras.....39

Tabel 4.2 Komponen *Software* atau Perangkat Lunak.....41

Tabel 4.3 *Snort Default Classification*.....51

Tabel 4.4 Informasi *bot* telegram.....54

Tabel 4.5 Hasil Pengujian Serangan 58

Tabel 4.6 Hasil Deteksi pengujian berdasarkan waktu 59

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring perkembangan teknologi yang semakin maju dengan pesat dari tahun ke tahun tentunya sangat membantu masyarakat dalam melakukan kegiatan dengan efektif dan efisien. Seperti : penggunaan teknologi *remote* server, dimana dengan menggunakan teknologi ini, maka server tersebut dapat dikontrol dari jarak jauh. Dengan kemajuan teknologi ini, pasti masih saja ada aspek-aspek yang menjadi perhatian. Salah satu aspek yang paling penting dalam sebuah sistem teknologi adalah aspek keamanan. Sehingga banyak pengembang dan pengguna teknologi yang menaruh perhatian besar terhadap suatu sistem keamanan sebuah teknologi. Karena sebaik apa pun sebuah sistem dibuat, tidak ada yang dapat menjamin keamanan sistem tersebut, karena ancaman terhadap sebuah sistem pasti selalu ada.

Salah satu perusahaan yang bergerak dalam keamanan jaringan komputer memprediksi kejahatan *cyber* seperti : *Denial of Service (DOS), spoofing, sniffing, spamming,* dan lainnya akan mengemuka di tahun 2019 dipengaruhi dengan semakin terbukanya pengetahuan tentang *hacking* dan *cracking* serta *tools* pendukung yang mudah dan gratis untuk didapatkan.(Adesty et al., 2020)

Dengan banyaknya ancaman kejahatan yang dapat menyerang sebuah sistem keamanan teknologi kapan saja, maka mengelola keamanan sistem adalah suatu

hal yang harus di persiapkan karena keamanan data harus di jaga, salah satunya menerapkan penggunaan *tools Intrussion Detection System* yang dapat digunakan untuk membantu melakukan proses monitoring terhadap keamanan jaringan. Dalam salah satu jurnal penelitian, dimana penelitian dilakukan menggunakan *Intrussion Detection Prevention System (IDPS)* serta menggunakan *SMS Gateway* untuk penyebaran Informasi hasil dari monitoring jaringan. Dengan menggunakan *SMS Gateway* dapat menyebarkan pesan ke ratusan nomor yang langsung terhubung dengan *database* nomor-nomor ponsel saja tanpa harus mengetik ratusan nomor dan pesan di ponsel karena semua nomor akan diambil secara otomatis dari *database* tersebut. (Syahputra Daniel, 2019)

Sedangkan dalam penelitian ini, komponen *Intrussion Detection System (IDS)* digunakan sebagai *tools* yang dapat dimanfaatkan untuk mendekteksi aktivitas mencurigakan dalam sebuah sistem tersebut serta menggunakan telegram *bot* yang dapat mengirimkan pesan otomatis sesuai dengan fungsi dari *bot* tersebut, maka admin akan dapat segera mengetahui bagaimana keadaan sistem server jika terjadi hal – hal yang mencurigakan. Penggunaan aplikasi telegram dikarenakan telegram adalah aplikasi berbasis *online* serta memilik *Application Programming Interface* yang dapat digunakan secara publik dan tidak terbatas.

Dalam jurnal penelitian " Monitoring Sistem Keamanan Jaringan Berbasis Telegram *Bot* pada *Local Area Network* ", dijelaskan bahwa penggunaan telegram sebagai alat monitoring jaringan karena aplikasi pesan telegram memiliki *Application programing Interface (API)* yang dapat di gunakan oleh publik. Berbeda dengan aplikasi pesan lainya seperti *Whatsapp* dan *LINE*. *API* yang di

sediakan oleh Telegram dapat gunakan oleh siapapun dan tanpa batas yang memungkinkan untuk dengan mudah membuat program yang menggunakan pesan telegram sebagai antarmuka.(Abdullah & Nurhayati, 2019)

Keamanan suatu sistem sangat berpengaruh terhadap pengembang sistem dan juga pengguna sistem tersebut. Jadi, dibutuhkan juga sistem monitoring untuk memaksimalkan sistem keamanan. Jika hanya mengandalkan seorang server administrator maka kurang maksimal, maka dibutuhkan sebuah cara cepat untuk dapat terhubung dengan server sehingga dapat memberikan pemberitahuan tentang server tersebut dengan cepat. Karena administrator server pasti tidak bisa selalu memantau keadaan keamanan server setiap waktu. Sehingga dalam penelitian ini diharapkan dapat mempermudah proses monitoring keamanan jaringan komputer agar lebih efektif dan efisien.

Penulis dalam Penelitian ini ingin membangun sebuah sistem untuk melakukan monitoring jaringan . Berdasarkan latar belakang yang telah dibuat, dengan adanya pemikiran tersebut muncul suatu ide dengan judul skripsi yaitu **“IMPLEMENTASI MONITORING KEAMANAN JARINGAN MENGGUNAKAN *SNORT* DAN TELEGRAM *BOT* SEBAGAI *NOTIFICATION ALERT* “.**

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah diatas maka rumusan masalah dapat di susun sebagai berikut :

1. Bagaimana cara melakukan monitoring terhadap suatu sistem jaringan menggunakan *snort* ?
2. Bagaimana pemanfaatan aplikasi telegram sebagai *notification alert* dalam monitoring sistem jaringan?

1.3 Batasan Masalah

Untuk menghindari meluasnya pembahasan masalah, maka penulis membatasi masalah yang akan di angkat, yaitu :

1. Melakukan monitoring terhadap sistem jaringan yang dibangun dari ancaman kejahatan komputer.
2. *Tools Intrussion Detection System* yang digunakan dalam monitoring jaringan adalah *Snort.*
3. Monitoring jaringan menggunakan sistem *bot* pada aplikasi telegram.
4. Membahas pemanfaatan aplikasi telegram sebagai *notification alert* dalam monitoring sistem jaringan.
5. Dalam monitoring kemanan jaringan ini dilakukan dengan menggunakan *Linux Ubuntu Server* versi 12.04.2 .
6. Dalam monitoring keamanan jaringan ini dilakukan dengan menggunakan Jaringan *Wireless Local Area Network (WLAN).*

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Penelitian ini bertujuan untuk melakukan monitoring terhadap sistem jaringan yang dibangun agar terhindar dari ancaman kejahatan komputer.
2. Penelitian ini bertujuan untuk dapat melakukan monitoring jaringan menggunakan tools *IDS* yaitu *Snort.*
3. Penelitian ini bertujuan untuk dapat melakukan monitoring jaringan menggunakan *bot* pada aplikasi telegram.

1.5 Manfaat Penelitian

Adapun beberapa manfaat dari penelitian ini adalah sebagai berikut :

1. Penelitian ini bermanfaat untuk menerapkan ilmu pengetahuan selama kuliah tentang monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert.*
2. Penelitian ini bermanfaat untuk dapat memaksimalkan penggunaan aplikasi telegram sebagai *notification alert* dalam monitoring sistem jaringan.
3. Penelitian ini bermanfaat untuk dapat membantu kinerja seorang administrator jaringan yang tidak mungkin setiap saat dapat mengawasi jaringan tersebut.

BAB II

LANDASAN TEORI

2.1 Implementasi

Implementasi mengacu pada tindakan untuk mencapai tujuan-tujuan yang telah ditetapkan dalam suatu keputusan. Tindakan ini berusaha untuk mengubah keputusan-keputusan tersebut menjadi pola-pola operasional serta berusaha mencapai perubahan-perubahan besar atau kecil sebagaimana yang telah diputuskan sebelumnya. Implementasi pada hakikatnya juga merupakan upaya pemahaman apa yang seharusnya terjadi setelah program dilaksanakan.(Apriandi, 2015)

Implementasi merupakan suatu proses yang dinamis, dimana pelaksana kebijakan melakukan suatu aktivitas atau kegiatan, sehingga pada akhirnya akan mendapatkan suatu hasil yang sesuai dengan tujuan atau sasaran kebijakan itu sendiri.(Andhini & Fitri, 2017)

Berdasarkan pengertian implementasi yang sudah dijelaskan oleh beberapa ahli di atas dapat saya simpulkan bahwa implementasi adalah penerapan ataupun pelaksanaan dari suatu ide yang sudah di rancang sebelumnya.

2.2 Monitoring

Monitoring adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan *continue* tentang kegiatan/program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program/kegiatan itu selanjutnya. Monitoring adalah pemantauan yang dapat dijelaskan sebagai kesadaran *(awareness)* tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan kearah tujuan.(Widiastuti & Susanto, 2014)

Monitoring merupakan sebuah kegiatan untuk menjamin akan tercapainya semua tujuan organisasi dan manajemen. Monitoring juga didefinisikan sebagai langkah untuk mengkaji apakah kegiatan yang dilaksanakan telah sesuai dengan rencana, mengidentifikasi masalah yang timbul agar langsung dapat diatasi, melakukan penilaian apakah pola kerja dan manajemen yang digunakan sudah tepat untuk mencapai tujuan, mengetahui kaitan antara kegiatan dengan tujuan untuk memperoleh ukuran kemajuan.(Hiezma, 2015)

Berdasarkan beberapa penjelasan mengenai monitoring diatas, dapat disimpulkan bahwa monitoring adalah sebuah cara yang dapat dilakukan untuk memantau ataupun mengawasi sebuah objek untuk mengetahui informasi tentang objek yang di monitoring tersebut.

2.3 Keamanan Jaringan Komputer

Keamanan jaringan komputer adalah sebuah cara atau upaya yang dapat dilakukan untuk mencegah sebuah sistem jaringan komputer dari suatu ancaman ataupun serangan yang dapat kapan saja mengancam keamanan jaringan tersebut .

Sistem Keamanan Komputer, dalam beberapa tahun ini telah menjadi fokus utama dalam dunia jaringan komputer, hal ini disebabkan tingginya ancaman yang mencurigakan *(suspicious threat)* dan serangan dari Internet. Keamanan Komputer *(Security)* merupakan salah satu kunci yang dapat mempengaruhi tingkat *Reliability* (termasuk *performance* dan *availability*) suatu *internetwork*.(Shahzad et al., 2019)

Seorang ahli dalam *computer security*, menyatakan bahwa komputer dikatakan aman apabila dapat diandalkan serta perangkat lunaknya bekerja sesuai dengan apa yang diharapkan.(Rendro et al., 2020)

Keamanan jaringan merupakan suatu tindakan yang berhubungan dengan deteksi dan pencegahan terhadap tindakan yang dianggap merugikan.(Radhito et al., 2019)

2.4 *Intrussion Detection System (IDS)*

Intrusion Detection System (IDS) merupakan sebuah sistem yang dapat mendeteksi aktivitas yang mencurigakan pada sebuah sistem atau jaringan. Jika ditemukan aktivitas yang mencurigakan pada *traffic* jaringan maka *IDS* akan memberikan sebuah peringatan terhadap sistem atau administrator jaringan dan melakukan analisis dan mencari bukti dari percobaan penyusupan.(Abdullah & Nurhayati, 2019)

Intrusion Detection System (IDS) dalam mendeteksi serangan jaringan komputer. *Intrusion Detection System (IDS)* dapat didefinisikan sebagai *tool*, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer.(Ii et al., 2015)

Intrusion Detection System adalah sebuah alarm keamanan yang dikonfigurasi untuk melakukan pengamatan aktifitas *host* dan kegiatan penyusupan.(Radhito et al., 2019)

Intrusion Detection System (IDS) adalah sebuah sistem yang digunakan untuk mendeteksi adanya serangan pada sebuah komputer *atau server*.(Shahzad et al., 2019)

Penerapan *IDS* dapat dilakukan diberbagai tempat pada suatu jaringan di sebuah instansi atau perusahaan dengan tujuan tercapainya keamanan sistem. *IDS* sendiri dapat diklasifikasikan menjadi dua jenis yaitu *Host-based Intrusion Detection System (HIDS)* dan *Network-based Intrusion Detection System (NIDS).* Kedua jenis *IDS* adalah sebagai berikut :

1. *Host-based Instrusion Detection System (HIDS) IDS* tipe ini diterapkan dan beroperasi pada sebuah komputer server yang dianggap kritis atau rawan. Dalam pengertian lainnya, *HIDS* sesuai untuk arsitektur yang berupa *single server* yang memberikan layanan seperti web server, mail server, maupun layanan lainnya. Tujuan *HIDS* untuk memantau serta mendeteksi aliran paket-paket yang masuk dan keluar yang terindikasi berbahaya pada host sehingga tipe ini disebut juga *hostbased IDS.*

2. *Network-based Instrusion Detection System (NIDS)* Pada *IDS* jenis ini diterapkan dan beroperasi dengan melihat semua lalu lintas aliran yang melewati jaringan sehingga disebut *network-based IDS*. Pada klasifikasi ini semua paket yang keluar maupun masuk pada sebuah jaringan komputer akan terlebih dahulu dianalisa dengan tujuan untuk menemukan adanya percobaan penyusupan ke dalam sistem jaringan. Hal ini efektif untuk menganalisa *traffic* diantara *host* maupun segmen jaringan lokal. Berbeda dengan *HIDS*, pada jenis ini *NIDS* akan ditempatkan pada pintu masuk jaringan *(gateway).*(Tambunan et al., 2020)

2.5 *Snort*

Snort merupakan sebuah aplikasi atau *tool* sekuriti yang berfungsi untuk mendeteksi intrusi-intrusi jaringan (penyusupan, penyerangan, pemindaian, dan beragam bentuk ancaman lainnya), sekaligus juga melakukan pencegahan. Dalam praktiknya, *snort* sangat andal untuk membentuk logging paket-paket dan analisis trafik-trafik secara *real-time* dalam jaringan berbasis *TCP/IP*.(Radhito et al., 2019)

Snort adalah *Intrusion Detection System* jaringan *open source* yang mampu menjalankan analisis *real-time* dan paket *logging* pada *IP network. Snort* merupakan suatu perangkat lunak untuk mendeteksi penyusup maupun menganalisa paket yang melintasi jaringan komputer secara *realtime traffic* dan *logging* ke dalam *database* serta mampu mendeteksi berbagai serangan yang berasal dari luar jaringan.(Sudradjat, 2017)

Snort merupakan aplikasi atau perangkat lunak berbasis opensource yang memiliki keunggulan untuk mengetahui adanya indikasi penyusupan pada jaringan berbasis *TCP/IP* secara *real time*.(Tri Atmaja et al., 2018)

Snort merupakan bagian dari *Intrusion Detection System* yang terdiri dari beberapa komponen, dimana *Snort* dioperasikan dalam 4 buah mode, yaitu :

1. *Snifer mode,* melihat paket yang lewat di jaringan.
2. *Packet logger mode*, mencatat semua log dari paket-paket ke dalam disk.
3. *Intruction Detection Mode*, snort akan berfungsi untuk mendeteksi serangan yang dilakukan melalui jaringan komputer.
4. *Inline Mode,* mengambil paket dari iptable dan menginstrusikan iptable untuk meneruskan paket tersebut berdasarkan jenis rules dari snort yang digunakan.(Ii et al., 2015)

2.6 Jenis Serangan

Flooding Data adalah sejenis serangan *Denial of Service (DOS),* dimana *flooding* data melakukan serangan terhadap sebuah komputer atau server di dalam jaringan lokal maupun internet dengan cara menghabiskan sumber *(resource)* yang dimiliki oleh komputer tersebut**.** Pengiriman data yang berlebihan baik dari besar paket maupun jumlah paket kedalam suatu jaringan dan umumnya merupakan data yang tidak berguna disebut dengan *Flooding* Data, ada kalanya data yang berbeda dalam *traffic* merupakan data yang tidak perlu. Data tersebut memang sengaja dikirim oleh seseorang meneruskan jaringan data yang ada.

Pengiriman data tersebut dapat mengakibatkan lambatnya jalur *traffic* yang ada dalam jaringan

Macam – macam *Flooding Attack* dalam serangan *DoS Attack* :

1. *Ping of Death* Pengiriman paket *echo request ICMP* kedalam suatu jaringan secara berlebihan. Pengiriman paket ini dapat mengakibatkan sistem *crash, hang* atau *reboot*.
2. *Smurf Attack* Hampir sama dengan *ping of death* tetapi untuk *smurf attack* paket *ICMP* tidak dikirim secara langsung ke korban, melainkan melalui perantara. Pada awalnya dikirim sebuah paket *ICMP echo request* kesebuah host lain, paket ini bertujuan agar host tersebut mengirimkan paket *ICMP ping* secara terus menerus ke korban terakhirnya.
3. *SYN Flooding, SYN Flooding* terjadi bila suatu host hanya mengirimkan paket *SYN TCP* saja secara kontinyu tanpa mengirimkan paket *ACK* sebagai konfirmasinya. Hal ini akan menyebabkan host tujuan akan terus menunggu paket tersebut dengan menyimpannya kedalam *backlog*. Meskipun besaran paketnya kecil, tetapi apabila pengiriman *SYN* tersebut terus menerus akan memperbesar *backlog*.
4. *UDP Flooding* Pengiriman data *UDP* secara berlebihan kedalam suatu jaringan, pengiriman *UDP flood* ini akan membentuk suatu jalur hubungan dengan suatu servis *UDP* dari host tujuan.(Hambali & Nurmiati, 2018)

2.7 Telegram

Telegram adalah aplikasi pesan instan berbasis *cloud* yang fokus pada kecepatan dan keamanan. Telegram dirancang untuk memudahkan pengguna saling berkirim pesan teks, audio, video, gambar dan *sticker* dengan aman.(Sari, 2018)

Telegram merupakan aplikasi berbasis *cloud*, yang memudahkan penggunanya dapat mengakses satu *account* telegram dari perangkat yang berbeda dan secara bersamaan. Serta dapat membagikan jumlah berkas yang tak terbatas hingga 1,5 GB.(Sari, 2018)

Telegram merupakan aplikasi berbasis *cloud*, yang memudahkan penggunanya dapat mengakses satu account telegram dari perangkat yang berbeda dan secara bersamaan. Serta dapat membagikan jumlah berkas yang tak terbatas hingga 1,5 GB. Aplikasi telegram diprakasai oleh dua bersaudara asal Rusia, Nikolai Durov dan Pavel Durov. Keduanya saling berbagi tugas, Nikolai fokus pada pengembangan aplikasi dengan menciptakan protokol *MTProto* yang menjadi motor bagi telegram. Sementara Pavel bertanggung jawab dalam hal pendanaan dan infrastruktur melalui pendanaan Digital Fortress.(Fifit, 2020)

Telegram merupakan alternatif layanan aplikasi perpesanan untuk ponsel *(mobile)* maupun *desktop* yang berbasis *cloud* dengan keamanan tingkat tinggi serta kecepatan aksesnya.(Tri Atmaja et al., 2018)

2.8 *Bot*

Bot adalah program komputer yang melakukan pekerjaan tertentu secara otomatis.

Bot adalah sebuah mesin, dibuat memudahkan kehidupan keseharian kita tanpa harus terpaku di depan komputer.(Utomo et al., 2017)

Bot (kependekan dari kata "robot") adalah sebuah program yang beroperasi sebagai agen untuk seorang user atau untuk program yang lain. Di Internet, *bot* yang paling banyak ditemui adalah program, yang juga disebut *spider* atau *crawler*, yang mengakses situs Web dan mengumpulkan kontennya untuk indeks mesin pencari. Telegram adalah program perangkat lunak olah pesan yang berfokus pada kecepatan dan keamanan, telegram bersifat *free* (gratis). Dengan Telegram, Anda dapat mengirim pesan teks, gambar, video, file *(doc, zip, mp3)* atau secara sederhananya telegram itu seperti kombinasi antara SMS dan *Email*. Salah satu fitur unik yang dimiliki telegram adalah pengguna dapat membuat *bot*, *bot* telegram ini dapat bertindak seperti akun telegram manusia yakni untuk mengirim dan menerima pesan. *Bot* telegram dapat dibuat dengan cara mendaftarkannya ke *@botfather* pada telegram.(Fathroni et al., 2017)

Bot internet atau yang lebih kita kenal dengan robot web, adalah sebuah aplikasi perangkat lunak yang berbasis otomotis yang menjalankan semua perintah melalui internet. *Bot* biasanya menjalankan sebuah perintah yang pada dasarnya mudah dan secara terstruktur, namun dengan tingkatan yang lebih tinggi dibandingkan dengan yang hanya manusia saja.(Bahrurozi, 2005)

2.9 Telegram *Bot*

Telegram Bot API (Application Progamming Interface) adalah sebuah perangkat lunak atau aplikasi yang digunakan untuk berinteraksi antara *Bot* dengan penggunannya maka dari itu dibutuhkanlah sebuah *API.*(Tri Atmaja et al., 2018)

Telegram Bot Application Programming Interface (API) adalah sebuah teknologi *open source* yang disediakan oleh Telegram untuk membangun aplikasi bot Telegram bagi para pengembang. *Bot API* ini merupakan interface berbasis *HTTP* untuk menghubungkan bot yang dikembangkan oleh para pengembang dengan sistem Telegram.(Risanty & Sopiyan, 2017)

Bot Telegram merupakan sebuah akun khusus yang tidak memerlukan nomer telepon. Akun ini berfungsi sebagai interface untuk menjalakan *code* yang sudah dibangun. Untuk keamanan data, server perantara pada *Telegram* akan menangani semua enkripsi dan komunikasi dengan *Bot API.*(Risanty & Sopiyan, 2017)

2.10 *Application Programming Interface (API)*

API merupakan sekumpulan intruksi program atau protokol yang digunakan untuk membangun aplikasi perangkat lunak serta berperan sebagai pesan yang menerima permintaan pengguna dan memberitahu sistem sesuai permintaan sehingga terjadi konektifitas antar sistem. Berupa *API Key, API secret, Access Token,* dan *Access Token Secret.*(Pesantren et al., 2017)

API dapat memungkinkan seorang developer menghubungkan dua bagian dari sebuah aplikasi atau dengan aplikasi yang berbeda secara serentak. *API* terdiri

dari beberapa elemen yaitu *function*, *protocol*, dan *tools* lainnya diamana seorang developer dapat menciptakan sebuah aplikasi. Tujuan dari penggunaan *API* adalah untuk mempersingkat pengembangan dengan menyediakan fungsi secara terpisah agar developer tidak perlu membuat fitur yang sama.(Bahrurozi, 2005)

2.11 *Notification Alert*

Notification Alert adalah penggabungan dari kata " notifikasi " dan juga kata *" alert* " memiliki pengertian yaitu sebuah tanda peringatan ataupun tanda peringatan pemberitahuan tentang sebuah informasi kepada orang yang terkait.

2.12 *Wireless Local Area Network (WLAN)*

Wireless Local Area Network (WLAN) adalah jaringan komputer yang menggunakan gelombang elektromagnetik dan *infrared* sebagai media untuk men-transfer data. *WLAN* juga sering dikenal dengan istilah jaringan *nirkabel* atau *wireless.* Pada umumnya, *WLAN* hampir sama dengan jaringan *LAN*, hanya saja *WLAN* menggunakan *wireless device* dalam berhubungan dengan jaringan dan bukan berupa kabel.(Taufiq Rohman, S.Pd.I, 2019)

Jaringan *wireless* memiliki dua mode, yaitu infrastruktur dan *Ad-Hoc*. Konfigurasi infrastruktur adalah komunikasi yang terjadi diantara masing-masing komputer melalui *access point* pada *WLAN* atau *LAN*. Mode infrastruktur ini digunakan ketika komputer ingin mengakses jaringan kabel contohnya melakukan *sharing printer*. Pada mode ini, *access point* merupakan hal yang penting, dikarenakan berfungsi untuk melayani komunikasi yang ada pada jaringan

wireless. Oleh karena itu, jumlah dan lokasi *access point* sangat berpengaruh pada kelancaran jaringan *wireless* yang ada.(Taufiq Rohman, S.Pd.I, 2019)

2.13 *Linux Ubuntu*

Linux adalah sistem operasi berbasis *GNU/Linux* yang bersifat *Open Source* dan memiliki banyak varian seperti *Debian, Slackware, Open Suse, Archlinux, Redhat* dan sebagainya. Walaupun sangat banyak varian *GNU/Linux* hanya menyediakan aplikasi yang sudah ditentukan yang mungkin kurang bermanfaat oleh pengguna sehingga hal ini mengakibatkan banyak pengguna yang melakukan remastering untuk memenuhi kebutuhannya. *Remastering* adalah proses membuat sistem operasi baru dengan mengurangi atau menambahkan fitur-fiturnya dari distro *GNU/Linux* yang telah ada. *Linux* adalah sebuah aplikasi atau program yang menggunakan kernel sebagai sistem operasi. *Script* pertama *Linux* dirancang dan ditulis oleh seorang mahasiswa dari Finlandia bernama "*Linus Torvalds"* untuk *Intel 80386* arsitektur. *Script* lain dari Linux yang tersedia di Internet pada tahun 1991. Setelah itu, banyak orang bermain peran penting dalam mengembangkan dan memperluas *Linux* di berbagai belahan dunia. Sistemnya, peralatan sistem dan pustakanya umumnya berasal dari sistem operasi *GNU,* yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi *GNU* adalah dasar dari munculnya nama alternatif *GNU/Linux.* Dia menggunakan alat proyek *GNU* dan dengan demikian sistem operasi dikembangkan melalui proyek *GNU / Linux.*(Harjono et al., 2019)

Ubuntu merupakan salah satu distribusi *Linux* yang berbasiskan debian dan didistribusikan sebagai perangkat lunak bebas. Nama *Ubuntu* berasal dari filosofi dari Afrika Selatan yang berarti "kemanusiaan kepada sesama". *Ubuntu* dirancang untuk kepentingan penggunaan pribadi, namun versi server *Ubuntu* juga tersedia, dan telah dipakai secara luas. Proyek *Ubuntu* resmi disponsori oleh *Canonical Ltd*. yang merupakan sebuah perusahaan yang dimiliki oleh pengusaha Afrika Selatan Mark Shuttleworth. Tujuan dari distribusi *Linux Ubuntu* adalah membawa semangat yang terkandung di dalam filosofi *Ubuntu* ke dalam dunia perangkat lunak. *Ubuntu adalah* sistem operasi lengkap berbasis *Linux*, tersedia secara bebas, dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional.(Velasquez-Valencia et al., 2018)

Linux ubuntu adalah salah satu sistem operasi yang banyak digunakan oleh orang-orang selain operasi *windows* di era ini. Dimana, alasan dalam penggunaan sistem operasi *linux* ini adalah karena dapat berkerja lebih baik daripada sistem operasi *windows* karena rentan terhadap serangan *malware* ataupun virus.

Selain salah satu kelebihan diatas, masih terdapat beberapa kelebihan dari sistem operasi linux yaitu diantaranya :

1. Bersifat *open source*, dimana sistem operasi *linux* memiliki kode program yang terbuka sehingga dapat dikembangkan oleh siapa saja.
2. Sistem operasi *linux* dapat digunakan secara gratis atau tidak berbayar.

3. Sistem operasi linux memiliki keamanan yang lebih baik karena dirancang *multiuser* sehingga apabila virus menyerang *user* tertentu, akan sangat sulis untuk menyebar ke *user* yang lain.
4. Sistem operasi linux dapat digunakan pada komputer dengan spesifikasi yang masih rendah.

Akan tetapi, tentunya ditengah beberapa kelebihan yang dimiliki sistem operasi *linux* ini, pasti masih ada saja kekurangan yang dimilikinya yaitu antara lain :

1. Masih banyak *user* ataupun pengguna yang belum terbiasa dalam menggunakan sistem operasi *linux* ini.
2. Tampilan dari sistem operasi *linux ubuntu* ini masih kurang menarik.
3. Sistem operasi ini, masih terlalu sulit digunakan bagi para pemula karena sistem operasi ini masih bersifat *Command Line Interface (CLI).*

2.14 *IP Address*

IP Address merupakan indetifikasi unik dari sebuah komputer, yang berupa *logical number address*. *IP address* sendiri berisikan alamat informasi berharga yang dikodekan dan juga menyediakan kompleksitas *routing.*

Internet Protocol (IP) adalah protokol yang paling banyak digunakan sebagai sarana untuk melakukan pengalamatan dalam sebuah jaringan.(Velasquez-Valencia et al., 2018)

Jadi, *IP address* merupakan sebuah identitas yang terdiri dari angka yang menjadi sebuah pengenal perangkat komputer ataupun alamat dari sebuah perangkat komputer.

2.15 *Virtual Mechine*

Mesin Virtual sering disebut sebagai *virtual machine* merupakan penerapan perangkat lunak ke sebuah mesin komputer yang bisa menjalankan program yang serupa layaknya kayak sebuah komputer asli.

Virutal machine berupa sistem atau *tools* yang kita gunakan saat kita ingin melakukan virtualisasi. *Tools* ini berfungsi sebagai virtualisasi pada saat kita ingin mengeksekusi sistem operasi dari dalam sistem operasi utamanya sehingga kita bisa membuat sebuah percobaan serta mengoperasikan sistem operasi kita di *virtual machine* tersebut tanpa mengganggu sistem operasi yang sebenarnya.(Ii, 2020)

Tools yang digunakan dalam proses uji coba implementasi tugas ini, menggunakan mesin virtual *"virtualbox"* yang digunakan untuk melakukan virtualisasi sistem operasi.

2.16 *Bash Shell*

Bash adalah *shell,* atau *command language interpreter* atau dalam bahasa Indonesia adalah penerjemah bahasa perintah, untuk sistem operasi *GNU*. *Bash* sendiri adalah singkatan untuk *'Bourne-Again Shell'*, dimana Stephen Bourne

adalah penulis pertama untuk *Unix shellsh*, yang muncul dalam *Seventh Edition Bell Labs Research* versi *Unix.*

Pada dasarnya, *shell* hanyalah sebuah prosesor makro yang mengeksekusi perintah. Istilah prosesor makro berarti fungsi di mana teks dan simbol diperluas untuk membuat ekspresi yang lebih besar. Sebuah *shell Unix* merupakan sebuah interpreter perintah dan bahasa pemrograman. Sebagai penerjemah perintah, *shell* menyediakan antarmuka pengguna untuk set kaya utilitas *GNU.* Fitur bahasa pemrograman memungkinkan utilitas ini untuk digabungkan. File yang berisi perintah dapat dibuat, dan menjadi perintah sendiri. Ini perintah baru memiliki status yang sama seperti perintah sistem di direktori seperti */bin*, memungkinkan pengguna atau kelompok untuk membangun lingkungan kustom untuk mengotomatisasi tugas umum mereka.

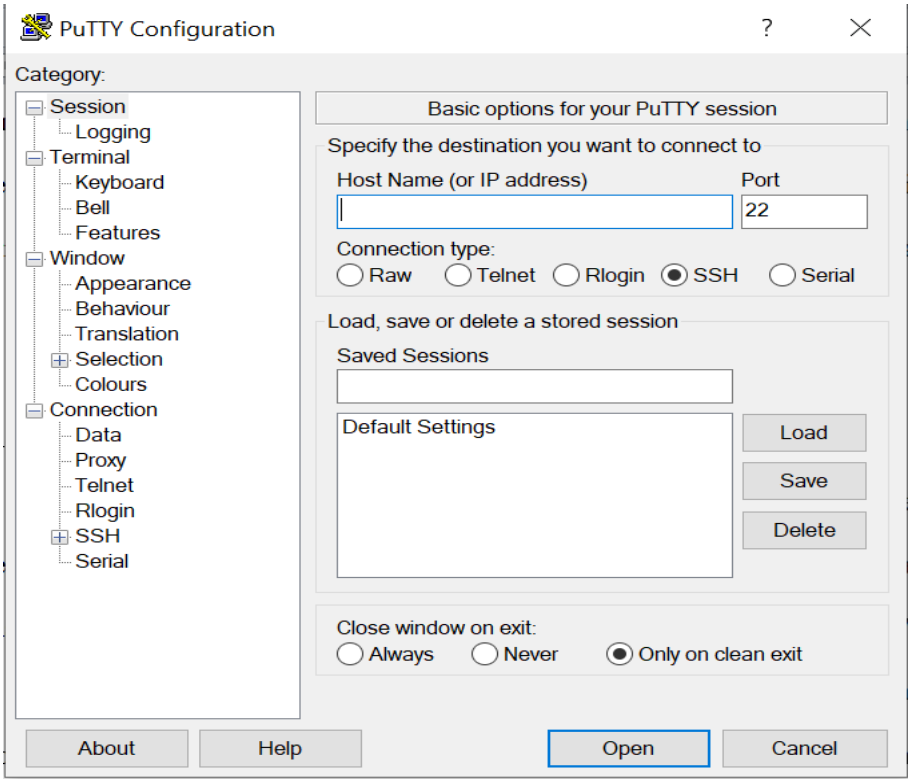
Shell adalah sebuah *user interface* untuk sistem operasi *UNIX*, dalam pemrograman apapun yang perlu mengambil masukan atau input dari pengguna, akan diterjemahkan ke dalam instruksi sehingga sistem operasi dapat mengerti, dan menyampaikan output sistem operasi kembali ke pengguna.

Terdapat berbagai jenis antarmuka pengguna, bash termasuk dalam kategori yang paling umum, yang dikenal sebagai *character-based user interface*. Antarmuka ini menerima jenis baris perintah tekstual yang pengguna ketikkan; antarmuka ini biasanya diproduksi keluaran berbasis teks.(Suri, 2019)

2.17 *Putty*

Putty adalah sebuah program *open source* yang dapat Anda gunakan untuk melakukan protokol jaringan *SSH, Telnet* dan *Rlogin*. Aplikasi ini merupakan

aplikasi portable sehingga tidak perlu di install. Protokol ini dapat digunakan untuk menjalankan sesi remote pada sebuah komputer melalui sebuah jaringan, baik itu *LAN*, maupun internet. Program ini banyak digunakan oleh para pengguna komputer tingkat menengah ke atas, yang biasanya digunakan untuk menyambungkan, mensimulasi, atau mencoba berbagai hal yang terkait dengan jaringan. Program ini juga dapat Anda gunakan sebagai tunnel di suatu jaringan.(Ii & Pustaka, 2013)



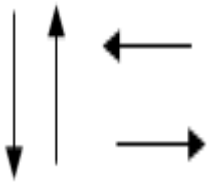
Gambar 2.1 Aplikasi Putty.(Ii & Pustaka, 2013)

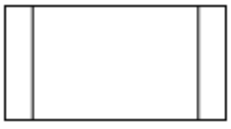
2.18 *Flowchart*

Flowchart adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program".(Suhendro, 2017)

Dari pendapat ahli diatas, dapat disimpulkan bahwa *flowchart* merupakan sebuah penggambaran alur yang dibentuk dengan simbol-simbol tertentu secara sistematis, sehingga menggambarkan proses dari suatu program.

Tabel 2.1 Simbol *Flowchart.*(Syahputra Daniel, 2019)

Simbol	Nama	Fungsi
	***Terminal Point Symbol*** **/ Simbol Titik Terminal**	Menunjukkan permulaan (start) atau akhir (stop) dari suatu proses.
	***Flow Direction Symbol*** **/ Simbol Arus**	Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (connecting line). Simbol ini juga berfungsi untuk menunjukkan garis alir dari proses.
	Processing Symbol **/ Simbol Proses**	Digunakan untuk menunjukkan kegiatan yang dilakukan oleh komputer. Pada bidang

Simbol	Nama	Fungsi
		industri (proses produksi barang), simbol ini menggambarkan kegiatan inspeksi atau yang biasa dikenal dengan simbol inspeksi.
	***Decision Symbol*** **/ Simbol Keputusan**	Simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada. Simbol ini biasanya ditemui pada flowchart program.
	***Input-Output*** **/ Simbol Keluar-Masuk**	Menunjukkan proses input-output yang terjadi tanpa bergantung dari jenis peralatannya.
	***Predefined Process*** **/ Simbol Proses Terdefinisi**	Simbol yang digunakan untuk menunjukkan pelaksanaan suatu bagian prosedur (sub-proses).

Simbol	Nama	Fungsi
		Dengan kata lain, prosedur yang terinformasi di sini belum detail dan akan dirinci di tempat lain.
	***Connector (On-page)***	Simbol ini fungsinya adalah untuk menyederhanakan hubungan antar simbol yang letaknya berjauhan .

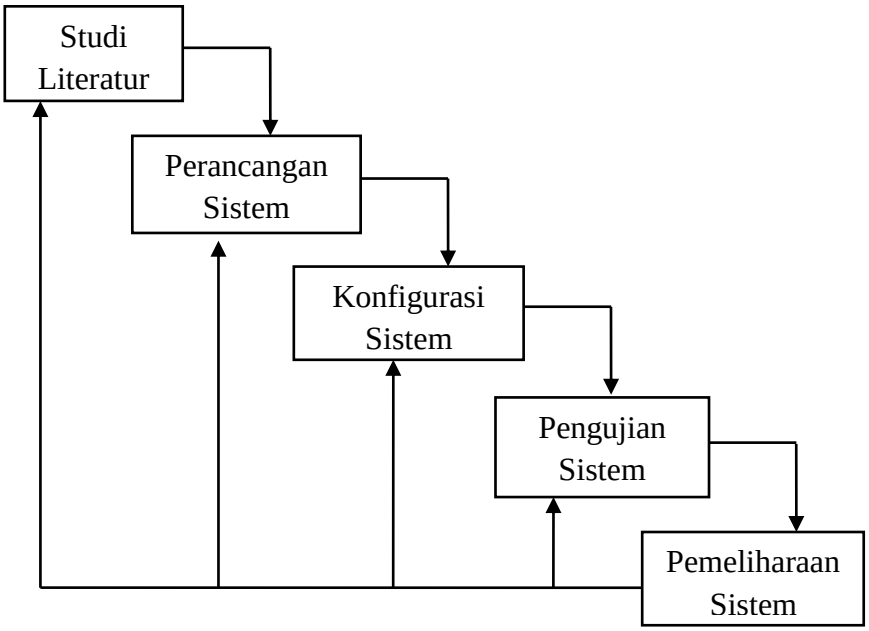
BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Dalam Penelitian ini, metode penelitian yang digunakan dalam membangun monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert* adalah metode air terjun *(waterfall),* dimana alur tahapan penelitian terurut .

Dalam Metode air terjun *(waterfall)* terdapat beberapa tahapan yang ada didalam metode ini, antara lain : Studi Literatur, Perancangan Sistem, Konfigurasi Sistem, Pengujian Sistem, dan Evaluasi Sistem . Adapun Metode dalam penelitian ini yaitu sebagai berikut :



Gambar 3.1 Metode Penelitian menggunakan metode *Waterfall.*

Beberapa tahapan yang ada dalam metode *waterfall* yaitu :

1. Studi Literatur

   Studi Literatur dilakukan dengan cara melakukan pencarian informasi terkait pembahasan penelitian dari berbagai sumber tertulis, baik berupa buku-buku, arsip, artikel, dan jurnal, yang relevan dengan permasalahan yang dikaji.

2. Perancangan Sistem

   Perancangan Sistem adalah penggambaran, perencanaan tentang monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert.* Dimana, dalam penelitian ini sistem snort di install dalam server jaringan yang dimanfaatkan untuk melakukan monitoring terhadap keamanan jaringan . Kemudian, *snort* yang sudah terinstall dalam server jaringan akan coba di uji dengan serangan seperti: *Denial of Service (DOS)* untuk mencoba apakah monitoring jaringan dapat berjalan sesuai dengan apa yang diharapkan, lalu jika tahapan tersebut berhasil, maka selanjutnya akan di coba untuk dapat menghubungkan antara *snort* sebagai *tools IDS* dengan telegram bot yang berfungsi sebagai *notification alert* .

   Meliputi beberapa tahap yang terstruktur sebagai berikut :

   a. Sistem yang dirancang menggunakan sistem operasi dan konfigurasi *linux* dan *Snort* serta menggunakan sistem operasi *windows* dalam proses pengujian.

b. Hasil dan pembahasan dengan cara Implementasi perangkat dan pengujian sistem yang sudah dibuat apakah sesuai dengan harapan dari seorang peneliti.

3. Konfigurasi Sistem

Dalam penelitian ini sistem yang dikonfigurasi yaitu pertama-tama akan menginstall *linux ubuntu dekstop versi 12.04.2*. Kemudian, setelah linux ubuntu dekstop *12.04.2* terinstall, maka akan diinstall *Snort* sebagai sistem yang dapat mendeteksi adanya serangan. Setelah itu, maka akan dihubungkan dengan telegram *bot* yang sudah dibuat sebagai *tools* yang digunakan untuk melakukan *notification alert.*

4. Pengujian Sistem

Dimana pengujian sistem dilakukan untuk dapat melihat apakah sistem yang telah dibuat dapat berjalan dengan baik sesuai dengan rencana yang telah dibuat. Pengujian sistem ini, nantinya dilakukan dengan mencoba melakukan serangan, seperti serangan *Denial of Service (DOS)* terhadap server jaringan. Lalu, memastikan bahwa tools *IDS* yaitu *snort* yang sudah terpasang pada server jaringan dapat membaca serangan yang dilakukan atau tidak, jika dapat terbaca maka sistem monitoring sudah dapat berjalan dengan baik sehingga selanjutnya dapat dilakukan penghubungan terhadap *bot* telegram, agar serangan tersebut dapat diketahui oleh administrator jaringan dengan adanya *notification alert* yang diterima melalui *bot* telegram.

5. Pemeliharaan Sistem

Pada tahapan pemeliharaan ataupun perawatan sistem ini, akan mencoba untuk melakukan *maintanence* ataupun pemeliharaan terhadap bagaimana sistem yang sudah selesai dibuat, sehingga sistem tersebut dapat terus digunakan dalam jangka waktu yang lama ataupun dengan dilakukannya pemeliharaan sistem ini dapat mengetahui kekurangan sistem sehingga sistem dapat ditingkatkan lagi atau dapat dikembangkan lagi sistem yang sudah dibuat agar lebih baik.

3.2 Metode Pengumpulan Data

Untuk dapat mendukung pembuatan laporan penelitian ini, dibutuhkan metode pengumpulan data sebagai landasan dari isi laporan ini. Pengumpulan data adalah proses pengumpulan informasi dan fakta yang dikumpulkan oleh peneliti untuk kepentingan memecahkan masalah atau menjawab pertanyaan penelitian. Dimana dalam laporan ini, pengumpulan data dilakukan dengan menggunakan beberapa teori yang ada pada website-website yang sesuai dengan pembahasan , jurnal-jurnal teknologi yang dapat dilihat di *google schoolar*, dan berbagai penelitian lainnya yang berhubungan dengan permasalahan yang di bahas dalam penelitian ini.

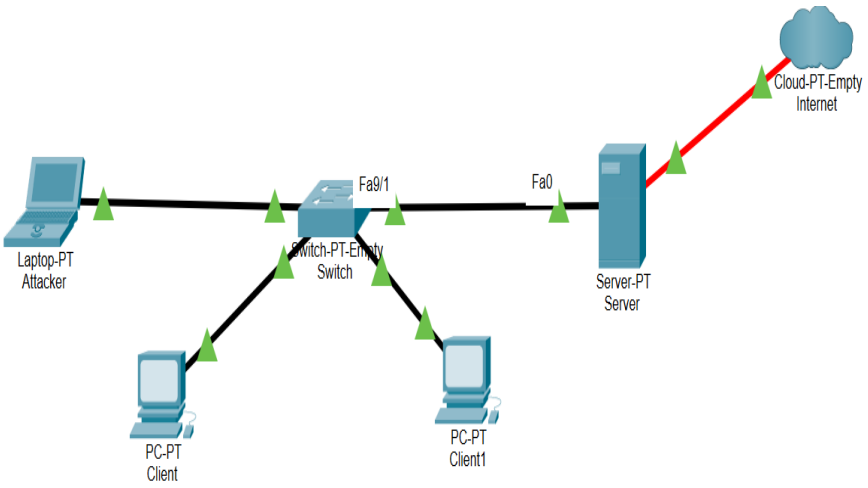
3.3 Rancangan Penelitian

Dalam tugas akhir ini, penelitian yang akan dilakukan adalah membangun sebuah sistem monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert*.

Sistem monitoring keamanan jaringan yang dibuat, nantinya akan dikonfigurasi, sehingga sistem ini akan dapat melakukan monitoring terhadap jaringan tersebut dan dapat mengetahui apakah terdapat serangan yang menyerang jaringan tersebut sehingga serangan itu dapat terdeteksi *(detection).*

Pendeteksian serangan ini nantinya akan dilakukan oleh *tools Intrussion Detection System (IDS)* yaitu *Snort*. Kemudian informasi bahwasanya terjadi serangan pada sebuah sistem jaringan akan dikirimkan melalui telegram *bot* sebagai *notification alert* kepada seorang administrator jaringan, dimana telegram *bot* ini terhubung dengan sistem *Intrussion Detection System (IDS)* yaitu *Snort* menggunakan *Application Program Interface (API)* sebagai antar muka aplikasi yang berisi *chat id*, *token api* yang ada di telegram *bot.*

Topologi Sistem Monitoring Keamanan Jaringan yang akan dibangun seperti dibawah ini :



Gambar 3.2 Topologi Jaringan dari Sistem *IDS* yang dirancang.

Dari gambar perancangan topologi jaringan sistem *Intrussion Detection System (IDS)* diatas dapat menjelaskan bahwa dalam jaringan yang akan dibuat nantinya akan terdapat sebuah server yang akan terinstall *tools snort* sebagai *Intrussion Detection System (IDS)* dan juga terdapat komputer *client* yang akan melakukan serang terhadap keamanan jaringan, lalu ada telegram *bot* yang nanti nya dihubungkan dengan snort akan memberitahu administrator bahwa ada serangan sehingga berfungsi sebagai *notification alert* . Dalam jaringan menggunakan pengalamatan IP yang dapat dilihat dalam tabel pengalamatan *IP Address* dibawah ini.

Tabel 3.1 Tabel Pengalamatan *IP Address* .

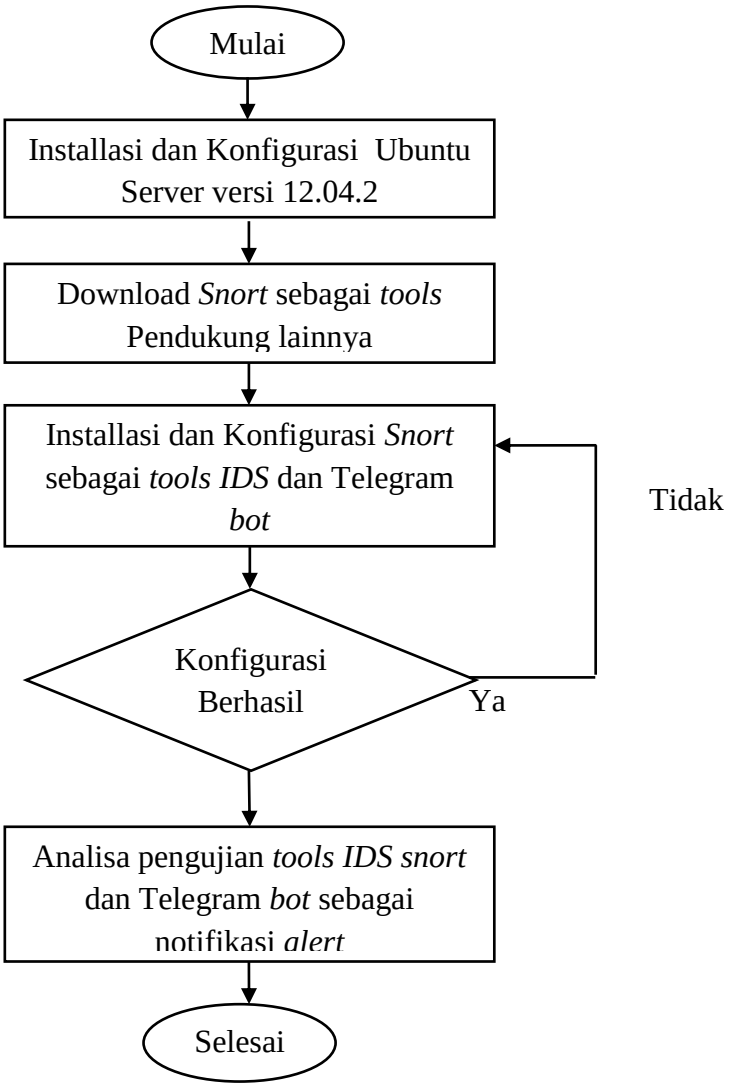
No	Perangkat dalam Jaringan	*Port Ethernet*	*IP Address*
1.	Internet *Hotspot*	-	192.168.140.152/24

No	Perangkat dalam Jaringan	*Port Ethernet*	*IP Address*
2.	Server	*WLAN*	*Dynamic Host Configuration Protocol (DHCP)*
		Eth 0	192.168.140.98/24
3.	*Attacker*	Eth 0	*Dynamic Host Configuration Protocol (DHCP)*
		Eth 1	-

Dalam tabel pengalamatan IP address diatas, dapat dilihat bahwa koneksi internet yang digunakan dalam pelaksanaan sistem yang akan dibuat ini berasal dari *hotspot* ataupun koneksi dari *wi-fi* yang terhubung dengan server.

3.3.1 Flowchart Intrussion Detection System (IDS)

Dalam proses penelitian ini, untuk memudahkan proses pembuatan sistem monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert* maka dibuatlah diagram alur atau *flowchart* seperti dibawah ini yang menggambarkan alur sistem yang akan dibuat.



Gambar 3.3 *Flowchart* Perancangan *Snort IDS* dan Telegram *Bot.*

Dari diagram alur diatas, dapat dijelaskan bahwa :

1. Pertama, dalam proses perancangan sistem ini akan melakukan installasi *Linux Ubuntu Server* 12 *versi* 12.04.2. Lalu, melengkapi alur installasi hingga penginstallan *Linux* selesai.
2. Saat telah selesai penginstallan *Linux Ubuntu Server* 12 *versi* 12.04.2, lakukan penyesuaian *IP Address* .

3. Kemudian Setelah selasai dalam penyetingan *IP Address*, lakukan lah penginstallan terhadap paket-paket yang dibutuhkan dalam mendukung kinerja *Snort* agar penginstallan *Snort* nanti tidak terjadi kesalahan . Setelah semua paket di install kemudian tahap *download* dan *install SNORT, SSH, Telegram bot.* dan mengkonfigurasinya.
4. Bila semua tahap telah berhasil, lakukan tahap akhir yaitu pengujian sistem yang telah dibangun dengan mencoba melakukan pengujian terhadap sistem yang telah dibuat . Serangan terhadap server dapat dilakukan dengan serangan *DoS* berupa *flooding* paket data contohnya.

Komponen Kerja *Intrusion Detection System (IDS)* yaitu *Snort Engine* :

1. *Library Packet Capture* atau *(libpcap)* berfungsi untuk dapat meng*capture* lalu memisahkan paket data yang melalui *ethernet card* yang akan digunakan *Snort*.
2. *Packet Decoder* berfungsi untuk mengambil paket dari layer 2 yang dikirim *libpcap.* Dengan memisahkan *Data Link, Protocol IP*, paket *TCP* dan *UDP Snort* memiliki informasi protokol yang akan di proses lebih lanjut.
3. *Preprocessor* adalah komponen yang berfungsi untuk menyusun atau mengubah paket data sebelum menuju ke *detection engine* dan beroperasi untuk mencari tahu bila paket data terjadi serangan.

4. *Detection Engine* adalah bagian terpenting dari Snort. Berfungsi untuk mendeteksi bila terjadinya kegiatan penyerangan. *Detection Engine* memproses *rule snort* untuk membaca strutur data internal yang di cocokkan dengan paket yang ada. Bila paket cocok dengan *rule* yang ada, tindakan yang di ambil berupa logging paket atau *alert*, jika tidak cocok paket akan di biarkan begitu saja.
5. Sistem *Log* dan *Alert* yang ada pada *detection engine* berfungsi untuk melihat paket, dimana jika paket cocok dengan *rule* yang ada, tindakan yang akan di ambil berupa logging paket atau *alert* dan log disimpan pada format teks didalam penyimpanan.
6. *Modul Output* berfungsi untuk mengatur jenis keluaran yang dihasilkan oleh sistem *log* dan *alert.*

3.4 Manajemen Biaya

Dalam melakukan sebuah penelitian, tentunya masalah biaya adalah salah satu bagian yang menjadi hal terpenting, karena sedikit banyaknya dalam sebuah penelitian dibutuhkan biaya, seperti biaya untuk memenuhi kebutuhan – kebutuhan yang dibutuhkan dalam melakukan penelitian ataupun biaya lainnya yang menunjang proses penelitian ini. Seperti dalam penelitian ini, dapat dilihat dibawah ini manajemen biaya yang dibutuhkan untuk dapat melakukan sebuah penelitian ini.

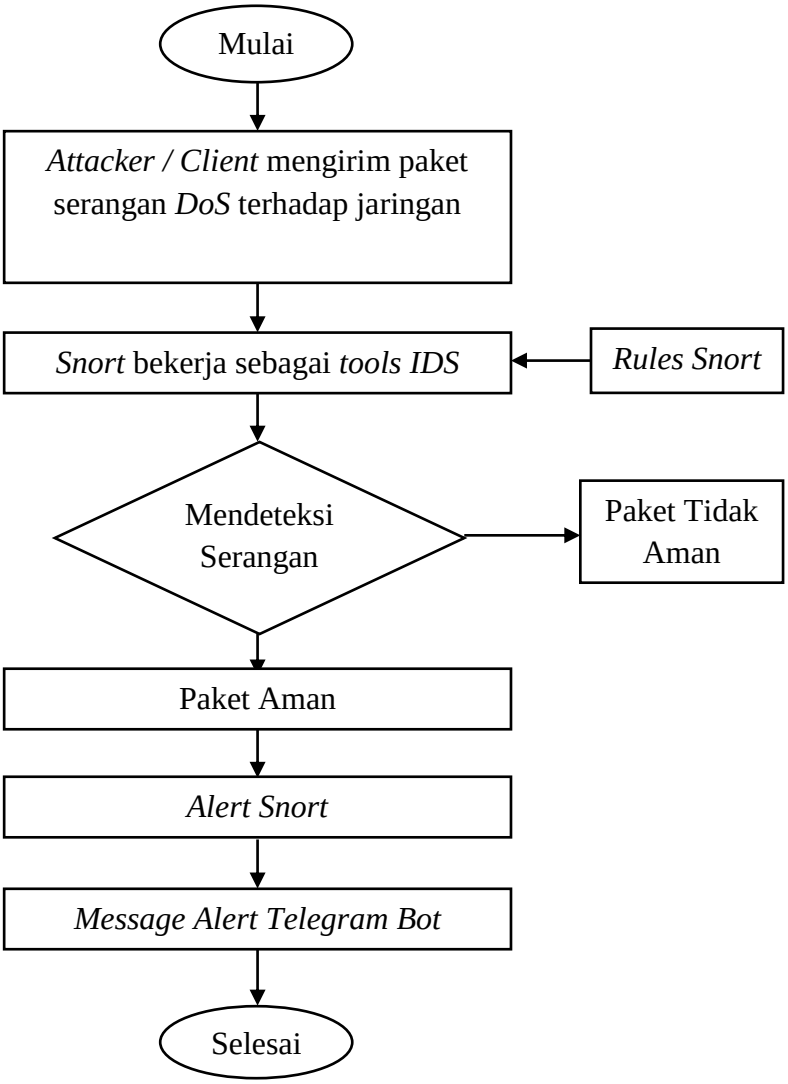
Tabel 3.2 Manajemen Biaya

No.	Jenis Pengeluaran	Biaya yang dibutuhkan (Rp)
1.	*Internet Hotspot*	Rp. 70.000-,
2.	Server	-
3.	*Attacker*	-
4.	Lain-lain	00.000
Jumlah		Rp. 70.000-,

Dari tabel diatas, dapat dilihat bahwa dalam penelitian ini, biaya yang dibutuhkan untuk dapat melakukan penelitian Sistem monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai *notification alert*.

3.5 Rancangan *Security*

Dalam proses pembuatan sistem monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai notification *alert* ini, tentunya terdapat rancangan keamanan sistem yang akan digunakan nantinya. Rancangan sistem ini dapat dilihat dalam diagram dibawah ini.



Gambar 3.4 *Flowchart* rancangan *security IDS Snort.*

Dari *Flowchart* diagram diatas, dapat dilihat bahwa dalam rancangan security ataupun rancangan keamanan sistem monitoring jaringan ini, bahwa saat terjadi paket serangan seperti serangan *DoS* contohnya yang dilakukan *attacker*, maka *snort* sebagai *tools Intrussion Detection System* yang sudah diinstall dalam *snort* akan melakukan tugasnya untuk mendeteksi serangan yang terjadi dalam jaringan server, dimana dalam sistem *snort* ini, akan terdapat beberapa *rules*, seperti rules untuk

pendeteksian serangan *DoS* yang akan di masukkan untuk mendeteksi serangan yang mengancam keamanan jaringan server. Setelah terdeteksi paket serangan tidak aman, maka akan muncul *alert* dari *snort* yang menggambarkan bahwa serangan tersebut berbahaya terhadap keamanan jaringan,dan *alert* ini akan terhubung dengan telegram *bot* yang sudah dikonfigurasikan dengan *snort*. Sehingga telegram *bot* inilah, nantinya yang akan otomatis memberitahukan administrator jaringan dari jaringan tersebut, bahwa ada ancaman serangan yang terjadi dalam jaringan .

Serangan yang akan dicoba dalam proses pengujian sistem monitoring ini adalah salah satunya serangan *DoS* seperti yang sudah dijelaskan diatas, dimana serangan *DoS* adalah sebuah serangan yang membanjiri lalu lintas jaringan dengan membuat beban lalu lintas jaringan tersebut menjadi lambat, salah satu jenis serangan *DoS* yang akan dicoba seperti serangan *TCP flood, UDP flood* yang merupakan serangan dengan cara membanjiri lalu lintas data jaringan dengan paket *TCP* atau *UDP* ke sistem jaringan yang dituju.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Spesifikasi *Hardware*

Perangkat Keras atau *hardware* yang dibutuhkan dalam melakukan pengujian terhadap materi penelitian mengenai monitoring jaringan menggunakan *snort* dan *telegram bot* sebagai *notifiation alert* adalah dapat dilihat seperti pada tabel dibawah ini.

Tabel 4.1 Komponen *Hardware* atau perangkat keras.

No	Perangkat Keras	Keterangan	Jumlah
1.	*Attacker*	HP 14s DK0073AU, Spesifikasi: Device name LAPTOP-AS320HP Processor AMD Ryzen 3 3200U with Radeon Vega Mobile Gfx 2.60 GHz, Installed RAM 8,00 GB (5,92 GB usable), Device : IA49C18EE-763A-41CD-9DE6-B89D5DDDE1CF, Product: ID00327-35866-28926-AAOEM,	1 Unit

No	Perangkat Keras	Keterangan	Jumlah
		System type 64-bit operating system, x64-based processor	
2.	*Server*	Name : UBUNTU-MELFA, Operating System : Ubuntu (64 bit). Sistem : *Base memory* : 2024 MB. Display :Video Memory : 16 MB, *Graphic Controller* : *VMSVGA*. Network : Intel PRO / 1000 MT Dekstop (Bridge Adapter, Realtek RTL8723DE 802.11 b/g/n PCIe Adapter, Intel PRO / 1000 MT Dekstop (Internal Network, 'PORT2').	1 unit

4.2 Kebutuhan Spesifikasi *Software*

Perangkat Lunak atau *software* yang dibutuhkan dalam melakukan uji coba terhadap penelitian mengenai monitoring jaringan menggunakan *snort* dan *telegram bot* sebagai *notifiation alert* adalah dapat dilihat seperti pada tabel dibawah ini.

Tabel 4.2 Komponen *Software* atau Perangkat Lunak.

No	Perangkat Lunak	Keterangan
1.	Sistem Operasi *Linux Ubuntu Dekstop 12.04.2*	Sistem Operasi ini bekerja sebagai Server yang akan menjadi target penyerangan.
2.	*Snort 2.9.8.0*	*Tools* ini akan berfungsi sebagai *tools Intrusion Detection System (IDS).*
3.	Sistem Operasi *Windows 7*	Sistem Operasi ini dipergunakan pada komputer *Attacker* dan *Client.*
4.	*Telegram Bot*	Aplikasi ini akan dipergunakan sebagai *tools notification alert.*
5.	*Low Orbit Ion Cannon (LOIC)*	Perangkat Lunak *Attacker* yang digunakan untuk melakukan penyerangan terhadap server.

4.3 Implementasi Sistem

Dengan sistem yang telah dikonfigurasi, maka selanjutnya sistem dioperasikan kemudian melakukan pengujian untuk melihat hingga sampai mana sistem yang dibuat dapat berjalan dengan baik sesuai dengan apa yang diharapkan. Implementasi *Security System* menggunakan *Snort Instrusion Detection System (IDS)* terdapat bagian utama yang akan berperan yaitu:

1. Implementasi *Snort* yang digunakan untuk dapat memonitoring jalur paket data .
2. Implementasi *Rule Snort* untuk dapat mendeteksi dan mengolah paket data yang melewati *snort* sehingga dapat diketahui apakah paket data terdapat ancaman atau tidak.
3. Implementasi *Output Rule Snort,* dimana *snort* yang berfungsi untuk mendeteksi serangan dalam sebuah jaringan, lalu telegram *bot* yang sudah dihubungkan dengan telegram *bot* akan mengirimkan *alert* dalam bentuk pesan atau *message* pada aplikasi telegram kepada administrator jaringan .

Implementasi *Security System* menggunakan *Snort Instrusion Detection System (IDS)* nantinya akan mendapatkan hasil dan batasan yaitu pencegahan serangan terhadap sebuah server dan menampilkan hasil monitoring kinerja server berupa notifikasi pesan *alert* ke administrator jika terjadi ancaman, untuk batasannya, dalam uji coba pengujian ini hanya menggunakan serangan *Denial of Service (DoS)* yaitu *TCP Flood, serta UDP Flood* yang dilakukan menggunakan *software* ataupun perangkat lunak *Low Orbit Ion Cannon (LOIC).*

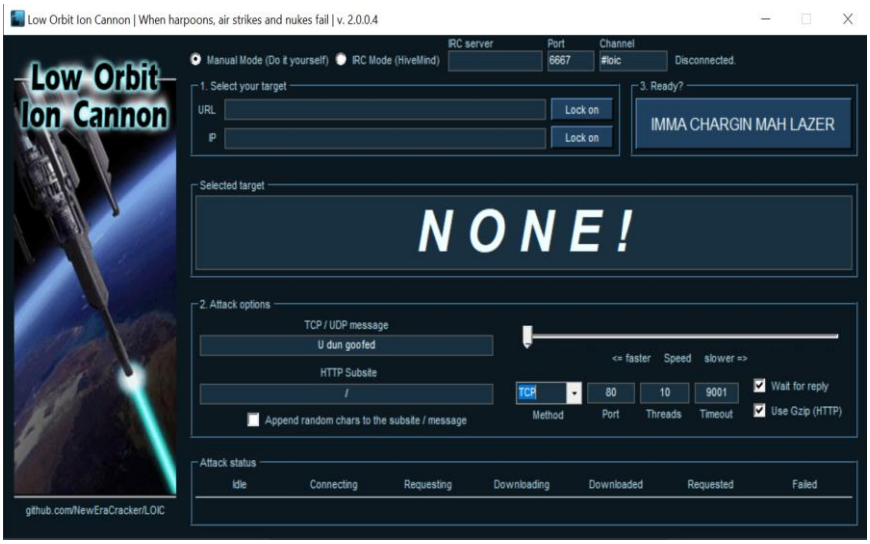
4.4 Pengujian dengan Serangan *Denial of Service (DOS)*

Dalam pengujian sistem yang akan dilakukan ini, pengujian sistem dilakukan dengan melakukan serangan *Denial of Service* yaitu *Transmission Control Protocol (TCP) flood* dan *User Datagram Protocol (UDP) flood* yang bekerja dengan mengirimkan paket dengan jumlah yang sangat besar atau berlebih ke server sehingga dapat mampu membuat sebuah server terganggu ataupun *crash* sehingga server tidak dapat bekerja dengan baik untuk melayani permintaan dari komputer *attacker*.

4.4.1 Penggunaan *Software Low Orbit Ion Cannon (LOIC)*

Software ataupun perangkat lunak *Low Orbit Ion Cannon (LOIC)* merupakan *tools* yang nantinya akan digunakan oleh *attacker* dalam melakukan penyerangan *Denial of Service* terhadap server. Dalam proses penyerangan sebuah server, *tools* ini bekerja dengan menyerang alamat *URL* ataupun alamat *IP* dari server tersebut melalui metode *TCP flood, UDP flood*. *Software LOIC* dapat digunakan dengan mudah dan gratis .

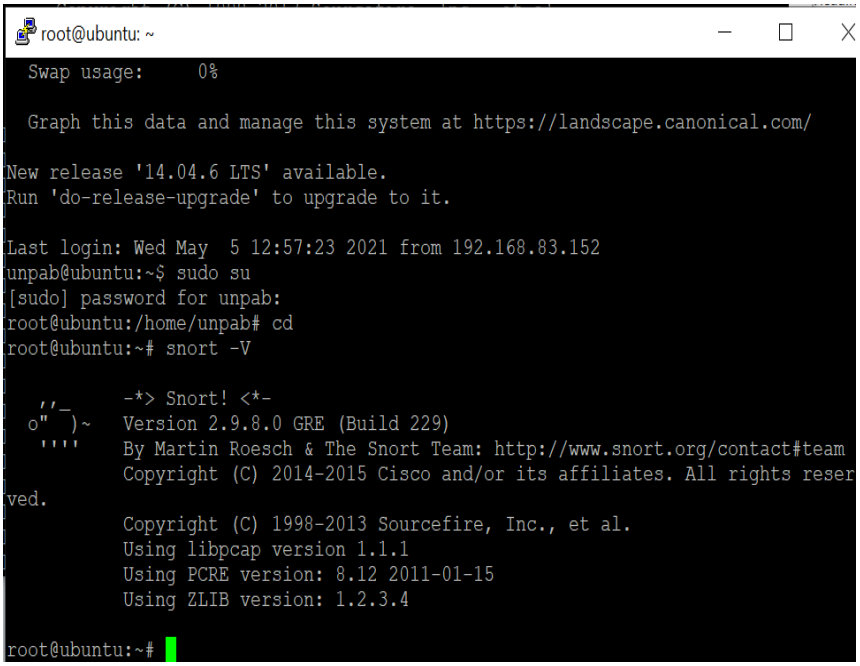
Dalam penelitian ini, *tools LOIC* digunakan untuk melakukan pengujian terhadap sistem keamanan sebuah server dari ancaman serangan *Denial of Service (Dos).* Untuk tampilan *software* dari *LOIC* ini dapat dilihat seperti dibawah ini :



Gambar 4.1 Tampilan *Software Low Orbit Ion Cannon (LOIC).*

4.4.2 *Monitoring Keamanan Jaringan menggunakan Intrussion Detection System Snort*

Dalam proses monitoring keamanan jaringan pada sebuah server disini, menggunakan *tools Intrussion Detection System* yaitu *Snort.* Dimana, snort berfungsi sebagai *tools* yang mendeteksi serangan yang masuk ke dalam jaringan server. Dapat dilihat digambar dibawah ini, *tools snort* yang sudah di install dalam sistem server .



Gambar 4.2 Tampilan *tools snort* yang sudah diinstall pada server.

Dalam tampilan *console snort* diatas, dapat dilihat bahwa *snort* versi yang sudah diinstall di dalam server sebagai *tools Intrussion Detection System* adalah *snort* versi 2.9.8.0. Setelah *snort* terinstall pada server, maka *tools* ini akan bekerja dalam mendeteksi serangan yang terjadi pada server ataupun melakukan monitoring pada keamanan jaringan sebuah server.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"NMAP scan SYN"; flags:S,12; ack:0; threshold: type both, track by_dst, count 1, seconds 60; reference:arachnids,27; GID:1; sid:10000001; rev:001; classtype: attempted-recon;)
alert tcp $EXTERNAL_NET any -> $HOME_NET [21,22,80] (msg:"NMAP scan FIN"; flow:stateless; flags:F,12; ack:0; threshold: type both, track by_dst, count 3, seconds 10; reference:arachnids,27; GID:1; sid:10000002; rev:001; classtype:attempted-recon;)
alert tcp $EXTERNAL_NET any -> $HOME_NET [21,22,80] (msg:"SCAN nmap XMAS"; flow:stateless; flags:FPU,12; threshold: type both, track by_dst, count 3, seconds 10; reference:arachnids,30; classtype:attempted-recon; sid:1228; rev:7;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (flags:S;msg:"DdoS Detected"; flow:stateless; threshold: type both, track by_dst, count 70, seconds 10; classtype:bad-unknown; GID:1; sid:10000007; rev:001;)
alert udp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"DdoS UDP"; flow:stateless; threshold:type both,track by_dst, count 70, seconds 10; classtype:bad-unknown; GID:1; sid:10000012; rev:001;)
```

Gambar 4.3 *Rules snort.*

Meskipun *snort* sebagai *tools* yang dapat mendeteksi sebuah serangan, tentunya harus ditambahkan sebuah *rules-rules snort* sebagai pedoman ataupun sebuah perintah yang digunakan untuk mendeteksi jenis serangan apa yang sudah menyerang keamanan jaringan server tersebut, apakah itu termasuk serangan *TCP, UDP.* Rules snort dapat dilihat pada gambar yang ada diatas, dimana dalam rules snort tersebut dibuat untuk pendeteksian serangan *TCP, UDP flood.*

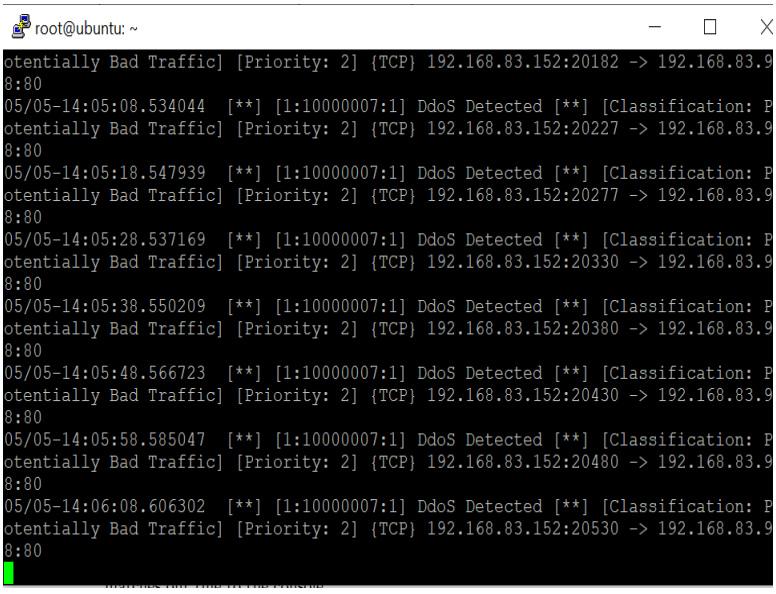
Seperti yang sudah dijelaskan diatas bahwa dalam proses pengujian untuk membuktikan apakah snort dapat berfungsi untuk mendeteksi sebuah serangan, maka akan dibuktikan dengan melakukan penyerangan terhadap server melalui aplikasi *Low Orbit Ion Cannon (LOIC)* dengan beberapa metode seperti dibawah ini :

1. Serangan Menggunakan Metode *TCP flood*

   Dalam metode *TCP flood* ini, paket atau data *Transfer Control Protocol* akan dikirim secara berlebihan ke dalam jaringan server sehingga membanjiri lalu lintas data jaringan server tersebut.

Gambar 4.4 Serangan *TCP flood* terhadap server.

Setelah melakukan penyerangan terhadap server dengan menggunakan metode *TCP flood*, maka akan dilihat bahwa apakah *tools snort* yang sudah diinstall pada server dapat berkerja dengan baik untuk mendeteksi sebuah serangan *TCP flood* ini.

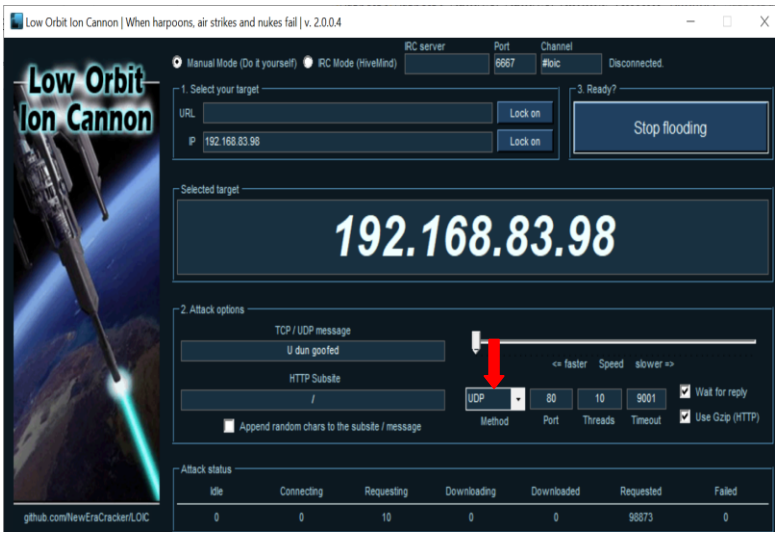


Gambar 4.5 Serangan *TCP* terdeteksi *Snort.*

Dari gambar *console snort* diatas dapat dilihat bahwa *snort* dapat melakukan pendeteksian terhadap serangan DoS yaitu *TCP flood* yang menyerang ke dalam sistem keamanan server. Dalam proses terjadinya serangan pada server dengan serangan *DoS* tersebut dapat dilihat bahwa serangan *TCP flood* tersebut memiliki klasifikasi *Potentially Bad Traffic*, dengan priority medium, dan serangan ini bersumber dari alamat IP 192.168.83.152. Serangan ini juga diidentifikasi dengan *class type Attempt Denial of Service* dengan *priority* medium pada port *TCP/IP* dikarenakan serangan berfokus untuk menyerang jalur protokol ini.

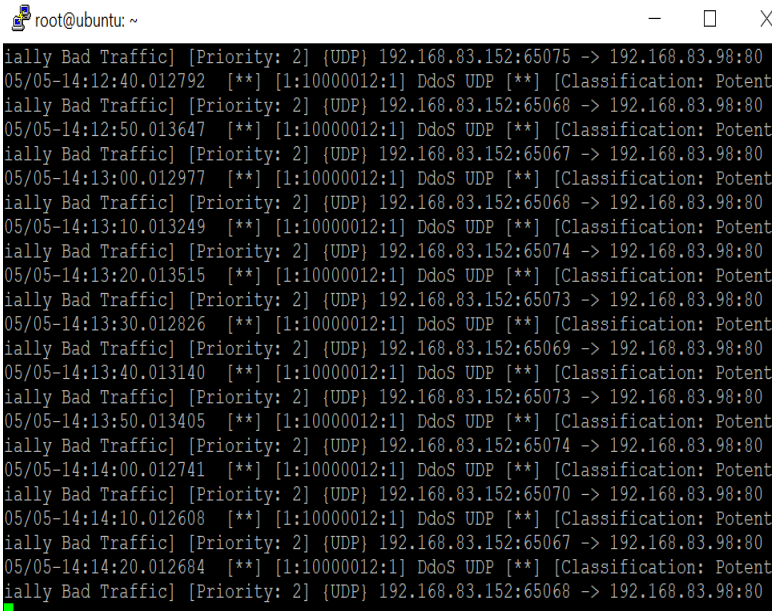
2. Serangan Menggunakan Metode *UDP flood*

Dalam metode *UDP flood* ini, paket atau data *UDP* akan dikirim secara berlebihan ke dalam jaringan server.



Gambar 4.6 Serangan *UDP flood* terhadap server.

Setelah melakukan penyerangan terhadap server dengan menggunakan metode *UDP flood*, maka akan dilihat bahwa apakah *tools snort* yang sudah diinstall pada server dapat berkerja dengan baik untuk mendeteksi sebuah serangan *UDP flood* ini.



Gambar 4.7 Serangan *UDP flood* terdeteksi *snort.*

Dari gambar diatas dapat dilihat bahwa *snort* dapat melakukan pendeteksian terhadap serangan *DoS* yaitu *UDP flood* yang menyerang ke dalam sistem keamanan server. Dalam proses terjadinya serangan pada server dengan serangan *DoS* tersebut dapat dilihat bahwa serangan *UDP flood* tersebut memiliki klasifikasi *Potentially Bad Traffic*, dengan priority medium, dan serangan ini bersumber dari alamat IP 192.168.83.152. Serangan ini juga diidentifikasi dengan *class type Attempt Denial of Service* dengan *priority*

medium pada port *UDP/IP* dikarenakan serangan berfokus untuk menyerang jalur protokol ini. Keterangan dari beberapa serangan yang sudah dilakukan di atas dapat dilihat pada tabel yang ada di bawah ini :

Tabel 4.3 *Snort Default Classification.*(Kurnia Dian, 2020)

Class Type	Description	Priority
Web application-attack	Web application attack	High
Attempted-dos	Attempted Denial of Service	Medium
Attempted-recon	Attampted Information Leak	Medium
Bad-unknown	Potentially bad traffic	Medium
Denial-of-Service	Detection of a Denial of service attack	Medium
Misc-attack	Misc attack	Medium
Successful-dos	Denial of Service	Medium
Successful-recon-limited	Information Leak	Medium
Web-application-activity	Access to a potentially vulnerable web application	Medium
icmp-event	Generic ICMP event	Low
misc-activity	Misc activity	Low
network-scan	Detection of a Network	Low

Class Type	Description	Priority
	Scan	
Unknown	Unknown Traffic	Low

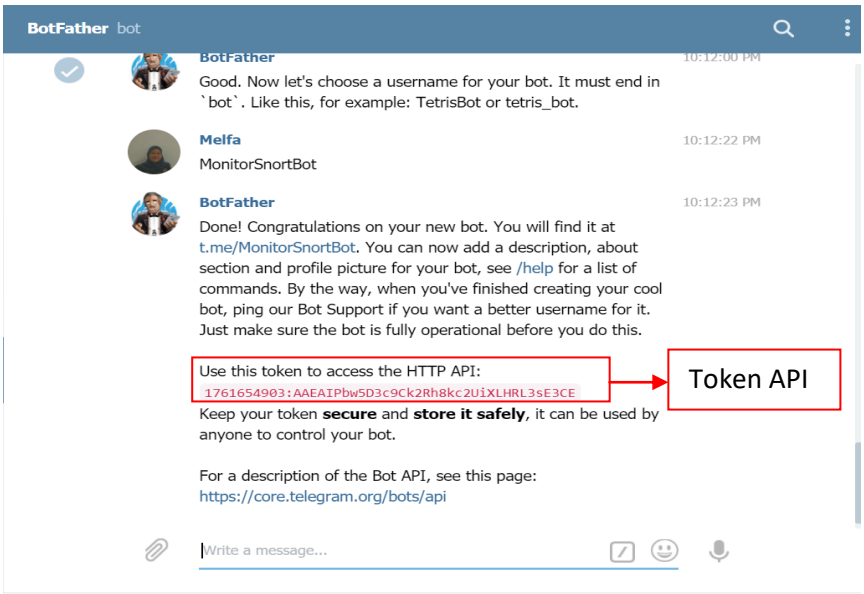
Dari tabel yang ada di atas yaitu tabel *snort default classification* dapat dilihat beberapa keterangan-keterangan yang berkaitan dengan serangan-serangan yang terjadi kepada server di dalam suatu jaringan sehingga dapat diketahui maksud atau arti dari sebuah istilah serangan yang ada terjadi dalam jaringan tersebut.

4.4.3 Pemanfaatan *Telegram bot* sebagai *notification alert* dalam proses monitoring Keamanan Jaringan

Setelah melakukan proses monitoring terhadap serangan yang terjadi pada sistem server yang mengancam sistem keamanannya, maka sekarang akan mencoba cara agar pesan ataupun *message* dari serangan yang terjadi pada server tersebut dapat terkirim dan muncul sebagai notifikasi ataupun *alert* untuk memberitahukan bagaimana keadaan server tersebut kepada administrator jaringan server tersebut sehingga dapat diketahui dan dilakukan penindakan lebih lanjut atas serangan yang terjadi pada server. Pesan notifikasi ini, akan disampaikan melalui aplikasi pesan singkat berbasis telegram dengan *bot* telegram.

Untuk dapat membuat telegram *bot*, pertama-tama harus membuat akun *bot* melalui akun *"BotFather"*, sehingga nantinya juga akan sekaligus mendapatkan akses Token *Application*

Programming Interface (API). Pembuatan *bot* dapat dilihat gambar dibawah ini, dimana pertama harus mengetikkan perintah *"/start"* lalu dilanjutkan dengan membuat nama *bot* sesuai dengan yang diinginkan .



Gambar 4.8 Token *API bot*.

Dalam gambar diatas, dapat dilihat bahwa *bot* yang dibuat memiliki nama yaitu *"MonitorSnortBot"* kemudian langsung mendapatkan *token API* yaitu : *1761654903:AAEAIPbw5D3c9Ck2Rh8kc2UiXLHRL3sE3CE*.

Setelah melakukan pembuatan *bot* telegram, maka dapat dilihat informasi dari bot tersebut seperti pada tabel yang ada dibawah ini :

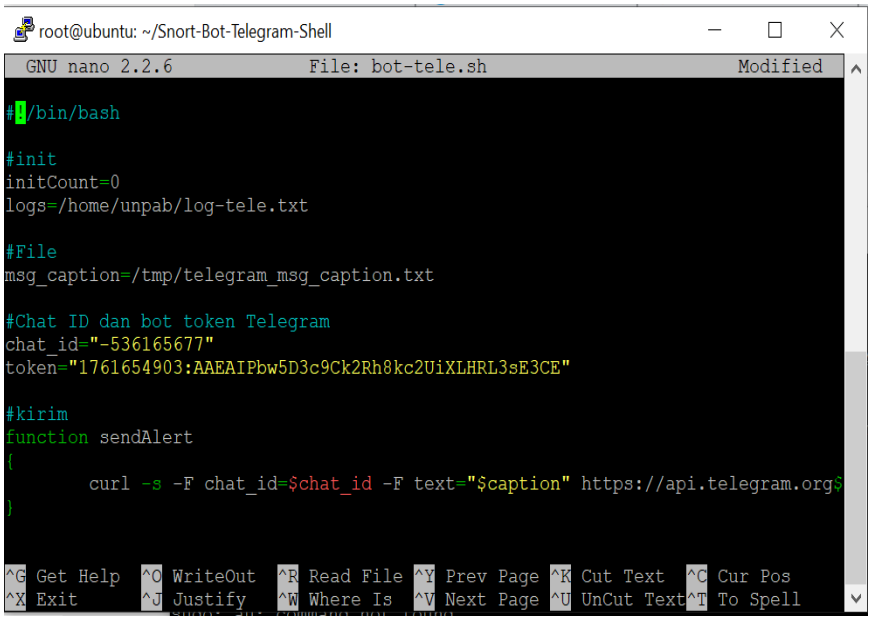
Tabel 4.4 Informasi *bot* telegram.

Parameter	Value

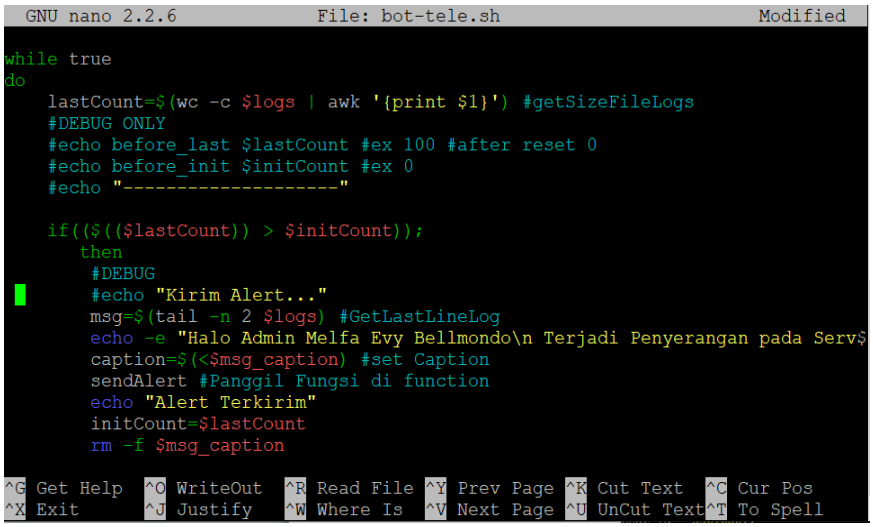
Parameter	Value
Nama	MonitorSnort
Username	MonitorSnortBot
Token API	*1761654903:AAEAIPbw5D3c9Ck2Rh8kc2UiXLHRL3sE3CE*

Dari tabel diatas, dapat dilihat informasi mengenai *bot* Telegram yang sudah dibuat sebelumnya. Dimana terdapat nama *bot*, *username bot*, dan juga Token *API* yang telah didapatkan dari proses pembuatan akun *bot* yang baru.

Setelah itu, maka akan dilanjutkan dengan membuat program yang akan menghubungkan *bot* telegram yang sudah ada dengan *snort* server, agar pesan atau *message* yang menggambarkan keadaan server tersebut jika terjadi serangan dapat diketahui oleh administrator server tersebut jika berada di jarak yang jauh dari server. Perintah yang akan dibuat dalam *bot* telegram yang akan digunakan dapat dilihat dalam gambar dibawah ini .

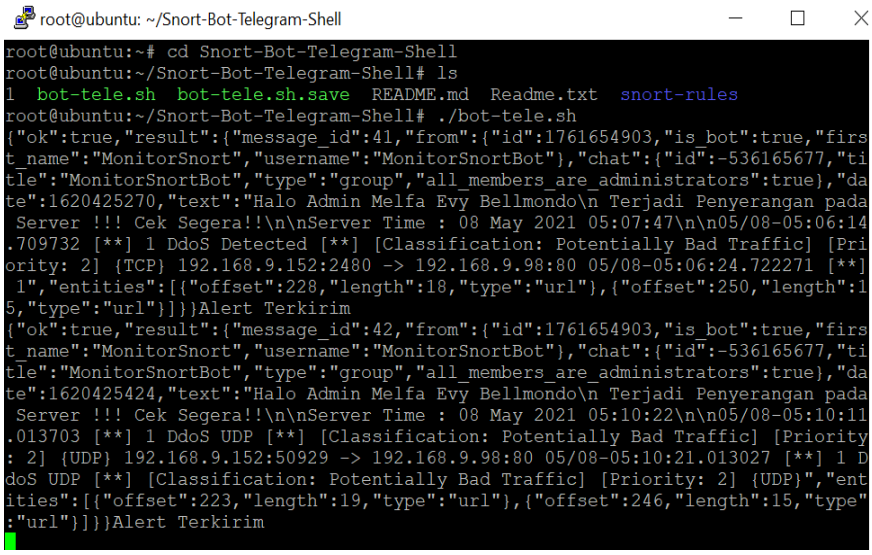


Gambar 4.9 Program dalam *bot Telegram* .



Gambar 4.10 Program dalam *bot Telegram*

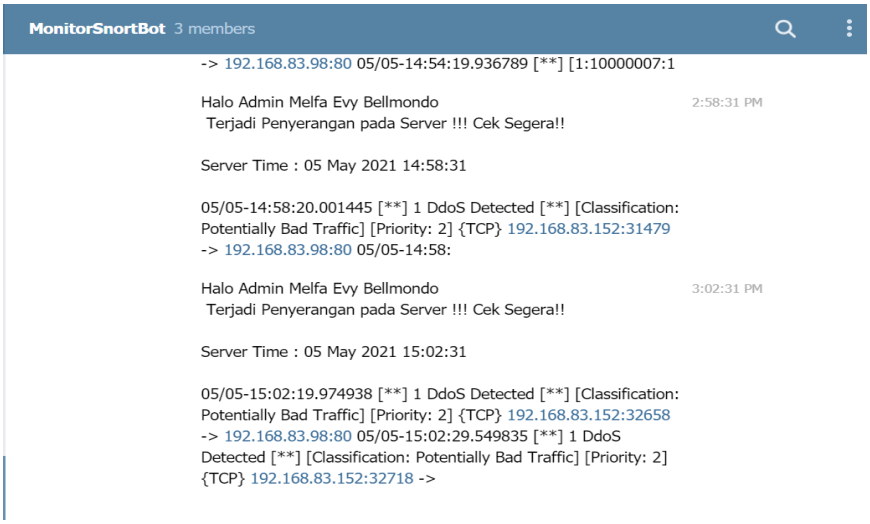
Program atau *command* yang akan dibuat dalam telegram *bot* akan menggunakan program *bash shell linux. Bash shell* merupakan bahasa pemrograman yang berjalan dalam kernel *linux*, dan berfungsi sebagai penerjemah antara *user* dengan sistem operasi.



Gambar 4.11 *Alert* terkirim ke Telegram *bot*.

Gambar di atas menggambarkan bahwa ketika telegram *bot* yang akan digunakan sudah terhubung dengan *snort* sebagai *tools* yang digunakan untuk memonitoring jaringan, maka otomatis jika terjadi serangan terhadap server dan dapat terdeteksi oleh *snort*, maka pesan bahwasanya terjadi serangan pada server bisa dapat langsung juga diketahui oleh administrator jaringan server tersebut.

Notifikasi yang akan memberitahukan bagaimana keadaan server ketika ada serangan yang masuk dalam jaringan server sangat dibutuhkan untuk dapat membantu kinerja dari seorang adminitrator jaringan server tersebut yang tidak dapat selalu mengawasi server dari jarak yang dekat.



Gambar 4.12 Notifikasi Telegram atas Serangan *TCP flood*.

Gambar 4.13 Notifikasi Telegram atas serangan *UDP flood.*

Pada gambar diatas, dapat dilihat bahwa monitoring keamanan jaringan menggunakan *snort* dan telegram *bot* sebagai notifikasi *alert* memberi kemudahan kepada seorang administrator jaringan karena dapat melakukan monitoring secara *online mobile.*

Dari kedua gambar diatas yang menggambarkan notifikasi pada aplikasi *snort*, dimana terdapat serangan *TCP* dan *UDP flood* beserta waktu yang mencakup tanggal hingga waktu dari terjadinya

penyerangan terhadap server sama persis seperti apa yang ada pada terminal *console snort*.

4.5 Hasil Pengujian Serangan pada server

Hasil dari pengujian serangan yang dilakukan pada server berisikan apakah serangan yang dilakukan terhadap server berhasil atau tidak dilakukan, serta gambaran waktu terdeteksi nya serangan pada server tersebut dapat dilihat dari tabel hasil pengujian yang sudah dilakukan seperti dibawah ini.

Tabel 4.5 Hasil Pengujian Serangan.

No	Jenis Serangan	Hasil yang di dapatkan	Kesimpulan
1.	*TCP flood*	Terdeteksi	Berhasil
2.	*UDP flood*	Terdeteksi	Berhasil

Pada tabel yang ada di atas dapat dilihat bawah dalam proses pengujian keamanan server ini, menggunakan jenis serangan *Denial of Service* yaitu *TCP flooding* dan juga *UDP flooding.* Semua serangan yang dilakukan kepada server tersebut berhasil terdeteksi oleh *tools snort* selaku *tools Intrussion Detection System* yang digunakan dalam uji coba ini.

Tabel 4.6 Hasil Deteksi pengujian berdasarkan waktu

No	Jenis Serangan	Hasil deteksi pengujian (waktu)		
		Awal Serangan	Terdeteksi	NotifikasiTerkirim
1.	*TCP flood*	14:06:08	14:06:18	14:10:10
2.	*UDP flood*	14:15:30	14:15:40	14:20:21

Dari gambar tabel diatas dapat dilihat bahwa waktu pendeteksian serangan yang terjadi dari awal serangan dilakukan dalam proses uji coba ini dilakukan. Rata-rata waktu pendeteksian awal serangan terjadi sampai serangan tersebut terdeksi oleh *snort* membutuhkan waktu sekitaran 10 detik, dan kemudian untuk serangan yang terdeteksi lalu terkirim dalam notifikasi telegram memiliki rentan waktu rata-rata 4 sampai 5 menit yang artinya waktu dalam terjadinya setiap serangan hingga pesan yang terkirim pada telegram memiliki renggang waktu yang bervariatif.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pada pembahasan dalam bab-bab sebelumnya yang berisi penjabaran tentang materi yang berkaitan dengan tugas akhir ini, mulai dari pembahasan konfigurasi serta implementasi sistem ini dapat diambil kesimpulan bahwa :

1. Sistem *Intrussion Detection System (IDS) Snort* dapat berkerja secara efektif dan efesien dalam mendeteksi serangan yang menyerang sistem dalam suatu jaringan server.
2. Dari proses pengujian yang dilakukan dalam uji coba ini dengan melakukan serangan *TCP* dan *UDP flooding* terhadap keamanan sistem server ini dan berhasil terdeteksi oleh sistem *IDS Snort.*
3. Serangan yang sudah terdeteksi oleh *Snort*, dapat dikirimkan oleh *bot telegram* sebagai *notification alert* kepada administrator jaringan sehingga dapat diketahui secara cepat jika terjadi serangan pada sistem server tersebut.

5.2 Saran

Dalam penelitian yang sudah dilakukan ini, tentunya masih terdapat berbagai kekurangan di dalam nya, sehingga diharapkan dalam penelitian yang akan dilakukan kedepannya dapat dilakukan lebih baik lagi dari penilitiaan yang sudah ada sekarang. Untuk itu, penulis memberi saran seperti :

1. Diharapkan dalam penelitian selanjutnya, agar menambahkan sebuah sistem pencegahan atau *prevention* untuk dapat mencegah serangan-serangan yang dapat menyerang keamanan sistem sebuah server sehingga sistem keamanan yang ada pada sistem tersebut ada sistem pendeteksian dan pencegahan (IDPS).
2. Sebaiknya dalam penelitian selanjutnya, dalam uji coba yang akan dilakukan ada baiknya untuk dapat melibatkan lebih banyak lagi serangan dan *rules-rules* serangan yang lain, agar dapat lebih baik lagi sistem keamanannya kedepannya.

DAFTAR PUSTAKA

Andi. S, Rosa A. dan M. Shalahuddin, 2013. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika. Sidik

Andriani, Y., Ramli, N. M., Syamsumir, D. F., Kassim, M. N. I., Jaafar, J., Aziz, N. A., ... & Mohamad, H. (2019). Phytochemical analysis, antioxidant, antibacterial and cytotoxicity properties of keys and cores part of Pandanus tectorius fruits. Arabian Journal of Chemistry, 12(8), 3555-3564.

Betha, 2012. Pemrograman Web PHP, Bandung : Informatika.

B. Nugroho. (2005). Database Relasional dengan MySQL. C.V Andi Offset : Yogyakarta. Fahmi, I. (2016). *Teori dan Teknik Pengambilan Keputusan: Kualitatif dan Kuantitatif*. Jakarta: Rajawali Pers.

Fowler, Martin. 2004. UML Distilled. Yogyakarta : Andi.

Hanum, Yuhilza. Software Engineering (Rekayasa Perangkat Lunak). Jakarta :Erlangga. Komang, I Setia Buana.2014. Jago Pemrograman PHP. Bandung: Dunia Komputer.

Magdalena, Hilyah. 2012. Sistem pendukung keputusan untuk menentukan mahasiswa lulusan terbaik di perguruan tinggi (studi kasus STMIK ATMA LUHUR Pangkal Pinang).Yogyakarta: Seminar Nasional Teknologi Informasi dan Komunikasi.

Mulyanto, Aunur R. 2008. Rekayasa Perangkat Lunak, Jilid 1. Jakarta: Direktur Pembinaan Sekolah Menengah Kejuruan. Oktavian, Diar, Puji. 2013. Membuat Powerfull Menggunakan PHP. Yogyakarta: Mediakom.

Peranginangin, Kasiman. 2014. Aplikasi Web dengan PHP & MySQL. Yogyakarta

Ramadhan, Z., Zarlis, M., Efendi, S., & Siahaan, A. P. U. (2018). Perbandingan Algoritma Prim dengan Algoritma Floyd-Warshall dalam Menentukan Rute Terpendek (Shortest Path Problem). JURIKOM (Jurnal Riset Komputer), 5(2), 135-139.

Rusdina, R., Syafarina, G. A., & Amin, M. (2020). PROTOTYPE APLIKASI BANJAR BERBASIS ANDROID STUDIO SEBAGAI SALAH SATU PETUNJUK WISATA DI BANJARMASIN. Technologia: Jurnal Ilmiah, 11(1), 59-63

Santosa, A., Sitopu, M. W., Sirait, D. N., & Nasution, D. (2021). Analysis of Damage to Localizer Equipment (Case Study of Sultan Iskandar Muda Airport, Banda Aceh). Budapest International Research and Critics Institute (BIRCI-Journal): Humanities and Social Sciences, 4(3), 7054-7061.

Subakti, Irfan. 2002. Sistem Pendukung Keputusan (Decision Support System). Teknik Informatika Institut Teknologi Sepuluh Nopember.

Syukur, Abdul, Tyas Catur P dkk., 2010. Penerapan Metode Analytical Hierarchi Process dalam penerimaan karyawan pada PT. Pasir Besi Indonesia. Pascasarjana Teknik Informatika Univeristas Dian Nuswantoro

Syafi'I, M. (2005). Aplikasi Database Dengan PHP 5 MySQL PostgreSQL Oracle. Yogyakarta : Andi.

Trisnani, A. A., Anwar, D. U., Ramadhani, W., Manurung, M. M., & Siahaan, A. P. U. (2018). Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi Menerapkan Metode Vise Kriterijumska Optimizajica I Kompromisno Resenje (VIKOR). JURIKOM (Jurnal Riset Komputer), 5(2), 85-90.

Triyono, S., Putra, R. M., Waluyo, S., & Amin, M. (2019, November). The effect of three different containers of nutrient solution on the growth of vegetables cultured in DFT hydroponics. In IOP Conference Series: Earth and Environmental Science (Vol. 355, No. 1, p. 012092). IOP Publishing.