LAPORAN

PRAKTIK KERJA INDUSTRI

DI

PT BENTANG INSPIRA TEKNOLOGI

Jl. Merak Ngibing No. 12 Sukaluyu Bandung

INSTALASI DAN KONFIGURASI *INTRUSION DETECTION SYSTEM* (IDS) MENGGUNAKAN SURICATA DENGAN NOTIFIKASI TELEGRAM PADA *SERVER* DINAS TENAGA KERJA BANDUNG DI PT BENTANG INSPIRA TEKNOLOGI

Diajukan untuk memenuhi salah satu syarat kelulusan dari SMK Negeri 1 Cimahi

Oleh :

NAMA : AINI NURUL AZIZAH

NO. INDUK SISWA : 171113256

TINGKAT : IV (EMPAT)

KOMPETENSI KEAHLIAN : SISTEM INFORMATIKA JARINGAN DAN APLIKASI

BIDANG KEAHLIAN : TEKNOLOGI INFORMASI DAN KOMUNIKASI

SEKOLAH MENENGAH KEJURUAN NEGERI 1

CIMAHI

2021

PENGESAHAN DARI PIHAK INDUSTRI

INSTALASI DAN KONFIGURASI *INTRUSION DETECTION SYSTEM* (IDS) MENGGUNAKAN SURICATA DENGAN NOTIFIKASI TELEGRAM PADA *SERVER* DINAS TENAGA KERJA BANDUNG DI PT BENTANG INSPIRA TEKNOLOGI

Laporan ini telah disetujui oleh :

Pembimbing,

MUHAMAD HAMDAN RIFAI, S.T.

Mengetahui,
Direktur Utama

MUHAMAD IKMAL WIAWAN, M.Kom.

PT. BENTANG INSPIRA TEKNOLOGI
BANDUNG
2021

PENGESAHAN DARI PIHAK SEKOLAH

INSTALASI DAN KONFIGURASI *INTRUSION DETECTION SYSTEM* (IDS) MENGGUNAKAN SURICATA DENGAN NOTIFIKASI TELEGRAM PADA *SERVER* DINAS TENAGA KERJA BANDUNG DI PT BENTANG INSPIRA TEKNOLOGI

Laporan ini telah disetujui oleh :

Ketua Kompetensi Keahlian,	Pembimbing.
DIKY RIDWAN, S.Kom.	Dr. Hj. SRI PRIHATININGSIH, M.T.
NIP. 19750703200902001	NIP. 196208041985032005

Mengetahui,
Kepala SMK Negeri 1 Cimahi

Drs. DAUD SALEH, M.M.

NIP. 196307181989021001

SEKOLAH MENENGAH KEJURUAN NEGERI 1
CIMAHI
2021


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

KATA PENGANTAR

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah melimpahkan Rahmat, Rizki, serta karunia-Nya kepada kita semua, khususnya kepada penulis sehingga pelaksanaan Praktek Kerja Industri (Prakerin) serta penulisan laporan Praktek Kerja Industri di PT. Bentang Inspira Teknologi yang berjudul INSTALASI DAN KONFIGURASI *INTRUSION DETECTION SYSTEM* (IDS) MENGGUNAKAN SURICATA DENGAN NOTIFIKASI TELEGRAM PADA *SERVER* DINAS TENAGA KERJA BANDUNG DI PT BENTANG INSPIRA TEKNOLOGI dapat diselesaikan dengan lancar.

Pembuatan dan penulisan laporan Praktik Kerja Industri ini diajukan untuk memenuhi salah satu syarat kelulusan dari Sekolah Menengah Kejurusan Negeri 1 Cimahi sekaligus sebagai bahan pertanggungjawaban terhadap pelaksanaan Praktek Kerja Industri di PT. Bentang Inspira Teknologi selama 6 bulan yang dimulai tanggal 10 Agustus 2020 sampai dengan 10 Januari 2021.

Dalam penyusunan karya tulis ini, penulis mendapat banyak bantuan dan dukungan dari semua pihak. Oleh karena itu pada kesempatan ini penulis ingin menyampaikan rasa hormat dan mengucapkan terimakasih sebesar – besarnya kepada :

1. Orang tua dan seluruh keluarga besar penulis yang senantiasa mendoakan, memberi semangat moral dan spiritual, memberi bantuan baik secara moril maupun materil dengan penuh keikhlasan, serta selalu berharap yang terbaik bagi penulis.
2. Drs. Daud Saleh, M.M, selaku Kepala Sekolah Menengah Kejuruan Negeri 1 Cimahi.
3. Diky Ridwan, S.Kom, selaku Ketua Kompetensi Keahlian Sistem Informatika Jaringan dan Aplikasi Sekolah Menengah Kejuruan Negeri 1 Cimahi.
4. Dr. Hj. Sri Prihatiningsih, M.T, selaku Wali Kelas IV Sistem Informatika Jaringan dan Aplikasi A sekaligus selaku pembimbing dari penulis dari pihak sekolah.

5. Seluruh staf, guru, dan karyawan kompetensi keahlian Sistem Informatika Jaringan dan Aplikasi.
6. Muhamad Ikmal Wiawan, M.Kom, selaku Direktur PT. Bentang Inspira Teknologi.
7. Muhammad Hamdan Rifai, S.T, selaku pembimbing dari pihak PT. Bentang Inspira Teknologi.
8. Rekan – rekan prakerin di PT. Bentang Inspira Teknologi.
9. Seluruh rekan seperjuangan tingkat IV angkatan 15 komptensi keahlian Sistem Informatika Jaringan dan Aplikasi Sekolah Menengah Kejuruan Negeri 1 Cimahi.

Serta semua pihak yang tidak dapat penulis sebutkan satu – persatu yang telah membantu dan memberikan dorongan serta masukkan-masukkan sebagai petunjuk dalam pembuatan karya tulis ini.

Penulis juga sudah berusaha semaksimal mungkin dalam penyusunan karya tulis ini. Namun, apabila terdapat kesalahan dan kekurangan, penulis sangat mengharapkan kritik dan saran yang sifatnya membangun dan mengarah kepada yang lebih baik untuk kedepannya.

Semoga karya tulis ini dapat bermanfaat khususnya bagi penulis, umumnya bagi pembaca, serta dapat memberikan semangat untuk terus menggali ilmu terutama dalam bidang ilmu pengetahuan dan teknologi.

Cimahi, Januari 2021

Penulis


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

BAB I
PENDAHULUAN

1.1. Latar Belakang Masalah

PT Bentang Inspira Teknologi (BIT) merupakan perusahaan yang bergerak di bidang pengembangan produk IT, seperti aplikasi yang berbasis *website* ataupun *desktop,* serta pemeliharaan dan topografi jaringan komputer, internet, dan *server* bagi lembaga maupun perusahaan swasta. Salah satu produk yang telah dikelola oleh PT BIT adalah *website* Dinas Tenaga Kerja (Disnaker) Kota Bandung yang mana *website* ini dibuat untuk memberikan informasi mengenai lowongan kerja kepada para pencari kerja di Kota Bandung.

Sebagai *server* yang tentunya memerlukan keamanan yang baik agar dapat terus berjalan, *server* Disnaker Kota Bandung belum memiliki sistem yang dapat mendeteksi aktivitas mencurigakan yang dapat berpotensi sebagai serangan. Pasalnya biasanya ketika terjadi serangan atau aktivitas mencurigakan terhadap server, administrator tidak bisa langsung mendeteksi dan menindaklanjuti serangan tersebut. Diperlukan analisis terlebih dahulu untuk mengetahui adanya serangan atau aktivitas yang mencurigakan terhadap *server*. Keterlambatan administrator dalam mengetahui adanya serangan terhadap *server* bisa saja memberikan dampak buruk seperti *hang, error,* bahkan sampai kehilangan data.

Maka dari itu, agar aktifitas ancaman serangan yang terjadi dapat langsung terdeteksi, penulis menerapkan sebuah sistem pendeteksi gangguan jaringan yang disebut *Intrusion Detection System* (IDS) pada *server* Disnaker Kota Bandung. IDS adalah sebuah sistem yang dapat secara otomatis memonitor kejadian pada jaringan komputer dan dapat menganalisa masalah keamanan jaringan.

Penerapan IDS pada *server* Disnaker Kota Bandung di PT Bentang Inspira Teknlogi adalah dengan menggunakan aplikasi Suricata. Nantinya *alert* atau pemberitahuan serangan akan langsung dikirimkan ke ponsel administrator melalui aplikasi telegram.

Dengan begitu, administrator akan langsung mengetahui jika terjadi aktifitas mencurigakan yang berpotensi sebagai ancaman serangan terhadap *server* Disnaker Kota Bandung di PT. Bentang Inspira Teknologi.

1.2. Tujuan

Tujuan dari penerapan *Intrusion Detection System* pada *server* Disnaker Kota Bandung di PT Bentang Inspira Teknologi adalah untuk mendeteksi adanya aktivitas mencurigakan yang berpotensi sebagai serangan. Sedangkan notifikasi suricata ke telegram administrator dibuat agar administrator dapat langsung mengetahui jika terjadi serangan pada *server* Disnaker Kota Bandung di PT Bentang Inspira Teknlogi.

1.3. Pembatasan Masalah

Adapun dalam pembuatan karya tulis ini terdapat batasan-batasan masalah sebagai berikut :

1. Sistem Operasi yang digunakan untuk melakukan instalasi dan konfigurasi IDS adalah Centos 7, sisi *remote access* menggunakan Windows 10, serta sisi penyerang menggunakan sistem operasi Ubuntu 16.04.
2. Konfigurasi yang dilakukan pada server dengan sistem operasi Centos 7 menggunakan konfigurasi berbasis CUI (*Command User Interface*).
3. Aplikasi *Intrusion Detection System* yang digunakan adalah Suricata versi 3.1.
4. *Rule* yang digunakan untuk *Intrusion Detection System* adalah *local rule* atau *rule* yang dibuat secara manual.
5. Penulis tidak membandingkan aplikasi yang digunakan dengan aplikasi lain.
6. Jenis serangan yang akan dideteksi sebagai bahan pengujian instalasi IDS adalah DoS dengan tipe *SYN Flood* pada port 80.

7. Media notifikasi telegram untuk administrator adalah melalui grup telegram yang didalamnya terdapat bot notifikasi suricata.
8. Pengujian yang dilakukan meliputi melakukan serangan dari mesin penyerang terhadap *server*, pengecekkan pada *log file* suricata (*fast.log*), dan pengecekkan notifikasi IDS Suricata pada telegram administrator.

1.4. Sistematika Pembahasan

Karya tulis ini terdiri atas 5 bab yang tersusun secara sistematis dan di uraikan sebagai berikut :

BAB I PENDAHULUAN

Pada bab ini penulis memaparkan latar belakang masalah yakni hal-hal yang melatarbelakangi pembuatan laporan, tujuan pembuatan laporan, pembatasan masalah dan sistematika pembahasan dalam laporan.

BAB II PT BENTANG INSPIRA TEKNOLOGI

Bab ini menjelaskan secara terurai informasi perusahaan seperti sejarah perusahaan, perkembangan perusahaan, gambaran umum, lokasi perusahaan, struktur perusahaan, visi dan misi perusahaan, produk yang ditawarkan, dan jasa serta hal lainnya yang berkaitan dengan PT. Bentang Inspira Teknologi.

BAB III LANDASAN TEORI

Bab ini memaparkan tentang konsep dan teori penunjang yang dikaji untuk memahami semua hal yang berkaitan dengan instalasi dan konfigurasi *Intrusion Detection System* dan jenis serangan yang sangat berkaitan dengan pembuatan *rule* pada IDS agar serangan-serangan tersebut dapat terdeteksi oleh sistem IDS.

BAB IV PEMBAHASAN

Pada bab ini memaparkan tahap-tahap perencanaan serta tahapan instalasi dan konfigurasi *Intrusion Detection System* pada *server* Disnaker Kota Bandung di PT Bentang Inspira Teknologi yang mana meliputi topologi, alat dan bahan yang dibutuhkan, langkah kerja dalam instalasi dan konfigurasi, serta pengujian terhadap konfigurasi yang telah dilakukan.

BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil analisis pada konfigurasi dan relevansinya dengan teori – teori penunjang serta saran dari penulis untuk pembaca yang ingin mengimplementasikannya sendiri.


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

BAB II
PT BENTANG INSPIRA TEKNOLOGI

2.1. Sejarah Umum

Bentang Inspira Teknologi (BIT) merupakan perusahaan yang bergerak di bidang Teknologi Informasi. Secara legal perusahaan ini berdiri pada bulan Februari 2009. Sejarah pendirian BIT berawal dari tahun 1998, dimana saat itu pendirinya telah membentuk suatu tim yang mengelola jaringan internet atau administrator di Institut Teknologi Bandung tepatnya di PPAU ITB sebagai pusat *incubator*. Tahun 2001, tim tersebut mencoba mengembangkan TI di luar kampus ITB dengan menggunakan nama CITS atau *Centre Information Technology Services.* Pada tahun 2004, *Centre Information Technologi Services* berubah menjadi Cipta Informasi Teknologi Solusi Indonesia (CITS Indonesia) dan mulai dipercaya sebagai salah satu konsultan TI di pemerintah provinsi Jawa Barat. Dengan pengalaman dan kepercayaan yang diberikan oleh sejumlah Satuan Kerja Perangkat Daerah (SKPD) maka pada tahun 2009 bulan Februari, Bentang Inspira Teknologi (BIT) mengukuhkan diri secara legal sebagai perusahaan yang fokus terhadap Teknologi Informasi.

Gambar 2. 1 Logo PT Bentang Inspira Teknologi

Bentang Inspira Teknologi didukung oleh infrastruktur yang sangat baik dengan sumber daya manusia yang profesional, berpengalaman, dan handal. Dengan berbekal semangat, inovatif, kreatif, serta jaringan yang luas, Bentang Inspira Teknologi sanggup bersaing di bisnis teknologi informasi. Bentang Inspira Teknologi selalu mengutamakan hubungan baik dengan klien dengan berkomitmen melayani klien dan memberikan pelayanan yang terbaik dan profesional.

2.2. Visi dan Misi

a. Visi Perusahaan

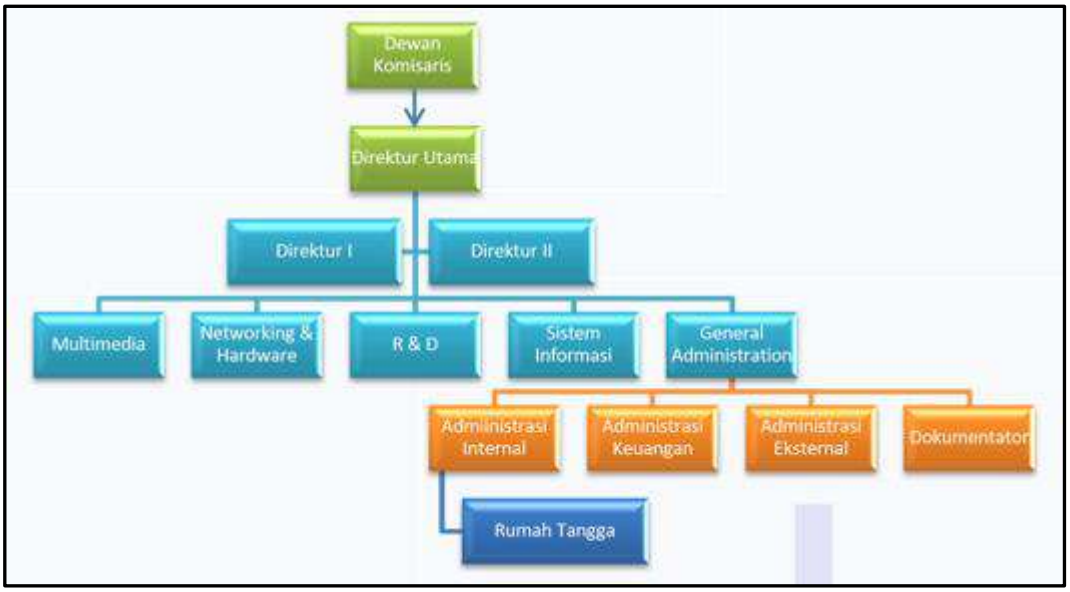
"*Menjadikan Perusahaan IT Yang Profesional dan Inovatif dalam Memberikan Solusi Terbaik Bagi Pemanfaatan Teknologi Informasi*"

b. Misi Perusahaan

1. Mengembangkan produk IT yang tepat guna.
2. Mengedepankan profesionalisme dalam menghasilkan layanan yang berkualitas.
3. Memberikan layanan yang maksimal bagi klien.
4. Mengembangkan inovasi teknologi yang tebaik dan terkini dalam setiap produk yang dihasilkan.

2.3. Struktur Organisasi

Struktur organisasi PT Bentang Inspira Teknologi secara garis besar terlihat pada gambar 2.2.



Gambar 2. 2 Struktur Organisasi PT Bentang Inspira Teknologi

2.4. Layanan dan Produk

Bisnis utama Bentang Inspira Teknologi adalah konsultan dan pengembangan produk IT, baik itu bagi lembaga pemerintah maupun perusahaan swasta. Dalam perjalanannya Bentang Inspira Teknologi berpengalaman dalam bidang pengembangan *software* dan *hardware*. Untuk *software* Bentang Inspira Teknologi mengembangkan aplikasi, baik yang berbasis *website* maupun *desktop*. Sedangkan untuk *hardware* Bentang Inspira Teknologi melakukan pemeliharaan dan topografi jaringan komputer, internet, maupun *server*. Berbagai macam produk dan jasa lainnya telah dihasilkan oleh Bentang Inspira Teknologi, diantaranya:

2.4.1. *Application*

a. *Website Application*

Website yang Bentang Inspira Teknologi buat menggunakan bahasa pemrograman PHP, HTML, CSS, Javascript, AJAX maupun Flash. Dan pengolahan *database*-nya menggunakan MySQL. Bentang Inspira Teknologi telah mengembangkan teknologi *website* yang terintegrasi dengan media lain, diantaranya :

1) *Video & Audio Streaming*
2) *Database Internal*
3) *E-Commerce*
4) *Internet Banking*

b. *Desktop Application*

Aplikasi *desktop* yang Bentang Inspira Teknologi rancang menggunakan bahasa pemrograman Delphi, Visual Basic, C++, maupun bahasa pemrograman aplikasi *desktop* lainnya. Beberapa produk aplikasi *desktop* yang telah dikembangkan diantaranya :

1) Sistem Informasi Pengawasan
2) Sistem Informasi Kepegawaian
3) Sistem Informasi Pewilayahan dan Pemetaan
4) Sistem Transportasi Perkotaan
5) Sistem Informasi Keuangan

2.4.2. *Geographical Information System*

Geographical Information System (GIS) atau Sistem Informasi Geografis merupakan sistem informasi berbasis komputer yang menggabungkan antara unsur peta (geografis) dan informasinya tentang peta tersebut (data atribut) yang dirancang untuk mendapatkan, mengolah, memanipulasi, analisa, memperagakan, dan menampilkan data spatial untuk menyelesaikan perencanaan, mengolah dan meneliti permasalahan.

2.4.3. *Multimedia*

Bentang Inspira Teknologi menganggap bahwa suatu produk yang bagus bukan hanya dinilai dari segi isinya saja, melainkan kemasan yang bagus dan menarik pun menjadi syarat utama dalam suatu produk. Oleh karena itu bagi Bentang Inspira Teknologi elemen multimedia dan desain grafis merupakan suatu hal yang penting. Beberapa produk yang telah dikembangkan Bentang Inspira Teknologi dalam bidang multimedia :

1) Animasi 2D dan 3D
2) Profil Multimedia
3) *Web Design*
4) *Touch Screen*
5) *Video Company Profile*
6) *TV Advertising*

2.4.4. *Networking*

Bentang Inspira Teknologi memiliki sumber daya yang handal dan professional dalam merancang topografi jaringan bagi perangkat IT, seperti :

1) *Server*
2) *Router*
3) *Switch*
4) *Modem External ADSL (Internet)*
5) Kabel RJ
6) Dan perangkat pendukung lainnya.

Bentang Inspira Teknologi berpengalaman dalam menangani permasalahan-permasalahan koneksi *Local Area Network* (LAN) baik untuk kelas menengah maupun gedung-gedung bertingkat.

2.4.5. *Maintenance Server*

Bentang Inspira Teknologi sadar bahwa penanganan dan pemeliharaan *server* tidak bisa dikerjakan oleh sembarang orang, oleh karena itu Bentang Inspira Teknologi memiliki tenaga-tenaga ahli yang professional dan berpengalaman dalam bidangnya.

Bentang Inspira Teknologi bukan hanya melayani pemeliharaan dan instalasi *server*, Bentang Inspira Teknologi pun memliki solusi-solusi terbaik bagi penanganan *server* agar pengguna dapat bekerja secara efektif dan efesien.

2.4.6. Perancangan dan Perencanaan

Melakukan perancangan dan perencanaan terhadap aplikasi yang harus dibangun dalam lingkup global maupun pentahapan, seperti rencana induk IT, kajian pembangunan suatu aplikasi, agar hasil implementasi yang didapatkan akan maksimal dengan mengacu pada rencana induk maupun kajian yang telah disusun tersebut. Perancangan dan perencanaan ini pun bertujuan untuk menyediakan dokumen acuan dalam pengembangan dan penerapan Sistem Informasi, Infrastruktur pada suatu instansi.

Sasaran yang ingin dicapai dari adanya kegiatan ini diantaranya :

1) Terdefinisikannya berbagai kebutuhan proses, data, aplikasi maupun teknoligu dalam rangka mendungng pengembangan dan penerapan sistem informasi.
2) Tersusunnya cetak biru (*blueprint*).
3) Tersusunnyasolusi pentahapan untuk pengembangan dan penerapan suatu sistem informasi.
4) Tersusunnya strategi dalam implementassi sistem informasi.

2.5. Hubungan Kerja Sama dan Pengalaman Kerja

2.5.1. Dinas Tanaman Pangan dan Holtikultura Provinsi Jawa Barat

1) Pemeliharaan Jaringan Komputer dan *Website*, tahun 2017.
2) Jasa Pemeliharaan Jaringan Komputer/LAN, tahun 2018.
3) Pengadaan Perbaikan Jaringan Komputer/LAN, tahun 2018.
4) Pemeliharaan Perabotan, Fasilitas, dan Gedung Kantor TPH, tahun 2019.

2.5.2. Konsil Kedokteran Indonesia

1) Pemeliharaan Jaringan, *Server*, dan *Webiste* Konsil Kedokteran Indonesia, tahun 2017.
2) Pengadaan Pemeliharaan Jaringan, *Server*, dan *Webiste* Konsil Kedokteran Indonesia, tahun 2018.
3) Pengembangan Aplikasi Administrasi Penegakkan Disiplin Berbasis Web. tahun 2019.

2.5.3. Badan Kepegawaian Daerah Provinsi Kalimantan Tengah

1) Pengadaan Sistem Aplikasi Absensi Elektronik Terintegrasi, tahun 2017.

2.5.4. Lembaga Penjaminan Mutu Pendidikan Jawa Barat

1) Pemeliharaan Kabel Jaringan Gedung LPMP Jawa Barat, tahun 2018.
2) Instalasi Perangkat Jaringan Dalam Rangka Pemeliharaan Aplikasi Kantor, tahun 2018.

2.5.5. PT Pertamina *Training* & *Consulting*

1) Jasa Konsultan Visualisasi Komunikasi dan Viralisasi untuk Media Sosial, tahun 2020.

2.5.6. Badan Nasional Pengelolaan Perbatasan

1) *Updating Database* Pengelolaan Wilayah Darat, tahun 2019.

2.5.7. Pusat Informasi Kriminal Nasional Bareskrim POLRI

1) Pemeliharaan Peralatan Fungsional Sistem Piknas Pusiknas Baareskrim Polri T.A. 2019, tahun 2019


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

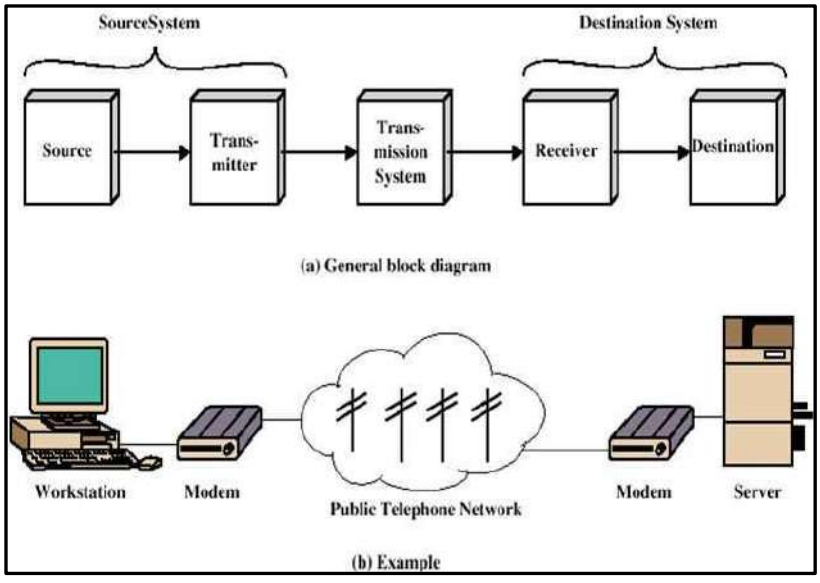
BAB III
LANDASAN TEORI

3.1. Komunikasi Data

Komunikasi data dalam jaringan merupakan sebuah proses pertukaran data antara dua perangkat yang perpindahannya melalui media transmisi tertentu. Media transmisi yang dapat digunakan sebagai media pertukaran salah satunya adalah kabel.

3.1.1. Komponen Komunikasi Data

Dikutip dari (Pengertian Komunikasi Data, 2020) komunikasi data terdiri dari beberapa komponen seperti sumber atau *source*, pengirim atau *transmitter*, sistem transmisi, penerima atau *reciever* dan tujuan atau *destination*. Model dari komponen komunikasi data dapat dilihat pada gambar 3.1.



Gambar 3. 1 Komponen Komunikasi Data
(Sumber: www.jagad.id)

a. Sumber (*Source*)

Sumber atau *source* merupakan komponen yang membangkitkan data atau informasi yang nantinya akan ditransmisikan bisa berbentuk alat input di komputer. Alat ini bisa mengubah informasi audio atau suara, video atau teks menjadi satuan data untuk diproses di sistem komputer seperti contohnya telepon dan komputer.

b. Pengirim (*Transmitter*)

Pengirim atau *transmitter* merupakan alat yang berguna untuk memproses data yang berasal dari sumber atau *source* dan nantinya akan disalurkan menuju ke sistem transmisi. Untuk bentuk fisiknya bisa berupa komputer personal yang bisa mengolah semua pesawat telepon untuk berkomunikasi dengan informasi berbentuk audio atau suara, contoh lainnya yaitu modem memiliki fungsi menyalurkan satu digital *bit stream* dari sebuah alat yang sudah dipersiapkan.

c. Sistem Transmisi

Sistem transmisi merupakan jalur penghubung antara sistem sumber dengan sistem tujuan media yang dipakai, sebagai contoh yaitu kabel dan juga gelombang elektro magnetik.

d. Penerima (*Receiver*)

Penerima atau *receiver* merupakan alat yang berguna untuk menerima sinyal dari sistem transmisi dan nantinya akan diproses untuk dijadikan sebuah informasi. Salah satu contoh yaitu modem yang difungsikan sebagai pesawat penerima yang nantinya akan menerima sinyal analog dari jaringan transmisi yang kemudian mengubahnya menjadi aliran bit digital supaya dapat diterjemahkan dan dibaca oleh komputer.

e. Tujuan (*Destination*)

Tujuan atau destination merupakan salah satu komponen yang menerima informasi yang sudah dikirimkan oleh *receiver* atau penerima kemudian diubah menjadi informasi yang sama ketika akan dikirimkan.

3.2. Jaringan Komputer

3.2.1. Konsep Jaringan Komputer

Jaringan komputer adalah kumpulan dari beberapa komputer yang saling terhubung melalui media trasnmisi untuk melakukan komunikasi data dan bertukar informasi (Sukmaaji & Rianto, 2008).

Komunikasi data yang bisa dilakukan melalui jaringan komputer dapat berupa teks, gambar, video, dan suara yang mana komunikasi ini tidak hanya dihubungkan oleh media transmisi tapi juga perlu adannya perangkat lunak dalam memfasilitasi komunikasi antara komputer-komputer tersebut.

Dalam jaringan komputer masing-masing dari sekumpulan komputer berdiri secara terpisah namun saling berhubungan dalam melaksanakan tugasnya. Dua komputer misalnya dihubungkan melalui media komunikasi contohnya kabel. Setelah terhubung kedua komputer ini saling berbagi atau bertukar informasi seperti membagikan foto, teks, atau video. Dalam berkomunikasi jaringan komuter menggunakan protokol komunikasi. Secara garis besar seperti itulah cara kerja dari jaringan komputer.

Tujuan dari dibangunnya jaringan komputer adalah untuk membawa informasi dalam layanan (*service*) secara tepat tanpa adanya kesalahan dari sisi pengirim (*transmitter*) maupun sisi penerima (*receiver*).

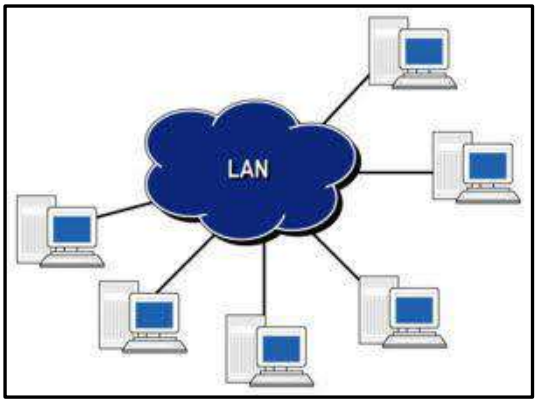
3.2.2. Klasifikasi Jaringan Komputer

Pengklasifikasian jaringan komputer ini didasarkan pada luas area yang dapat dijangkau atau dilayani. Secara umum klasifikasi jaringan komputer berdasarkan daerah jangkauan dibagi menjadi 3 macam, yaitu *Local Area Network* (LAN), *Metropolitan Area Network* (MAN), dan *Wide Area Network* (WAN).

a. *Local Area Network* (LAN)

Local Area Network (LAN) merupakan cakupan jaringan komputer yang menjangkau area yang terbatas, misalnya satu kantor, gedung, atau pabrik dimana komputer mempunyai jaringan yang secara fisik berdekatan satu dengan yang lainnya. Jarak dari LAN kurang lebih sampai dengan 10 km. Ciri-ciri LAN adalah :

1) Mempunyai pesat data yang lebih tinggi.
2) Meliputi wilayah yang lebih kecil.
3) Tidak membutuhkan jalur telekomunikasi yang disewa dari operator telekomunikasi.



Gambar 3. 2 *Local Area Network*
(Sumber: www.temukanpengertian.com)

b. *Metropolitan Area Network* (MAN)

Metropolitan Area Network (MAN) pada dasarnya merupakan LAN dengan versi yang berukuran lebih besar. Dua atau lebih LAN disebut juga dengan MAN yang mana ini dihubungkan bersama-sama dalam batas-batas kira-kira suatu kawasan metropolitan atau satu kota. Jarak maksimum yang dijangkau MAN kira-kira 80 km. Ciri-ciri MAN adalah :

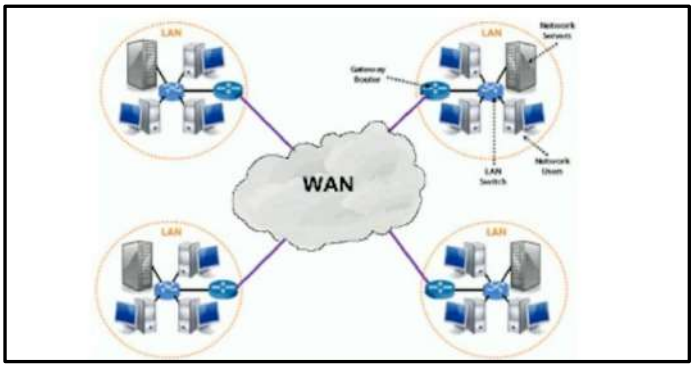
1) Cakupan wilayah jaringan yang lebih luas.
2) Pemeliharaan jaringan memerlukan waktu yang tidak sebentar.
3) Mempermudah dalam hal berbisnis dan juga keamanan dalam jaringan menjadi lebih baik.

c. *Wide Area Network* (WAN)

Wide Area Network (WAN) adalah jaringan komputer yang jangkauan areanya paling luas, bisa antar pulau, negara, bahkan benua. Contoh terbaik dan sangat terkenal dari WAN adalah internet. Tetapi, WAN juga dapat berupa jaringan pribadi. Sebagai contoh, sebuah perusahaan dengan kantor-kantor di berbagai negara dapat memiliki WAN yang menghubungkan berbagai lokasi dari kantor perusahaan tersebut.

Ciri-ciri dari WAN :

1) Cakupan wilayah yang sangat luas, melebihi LAN dan MAN.
2) Biaya operasional dan perawatan yang dibutuhkan lebih tinggi.
3) Sangat rentan terhadap bahaya pencurian data-data penting.



Gambar 3. 3 *Wide Area Network*
(Sumber: www.minatbelajar.com)

3.2.3. Perangkat Jaringan Komputer

Perangkat jaringan komputer adalah perangkat yang digunakan untuk mencapai tujuan dari jaringan komputer. Perangkat jaringan ada banyak jenisnya, beberapa diantaranya adalah :

a. *Network Interface Card*

Network Interface Card dibuat pada sebuah papan PCB yang akan melakukan konversi sinyal sehingga sebuah *workstation* bisa mengirim dan menerima data dalam jaringan. Sering disebut juga dengan Ethernet card, atau sering juga disebut *LAN Card*. NIC merupakan kartu jaringan yang dipasang pada slot ekspansi pada komputer. Slot yang diperlukan bisa

berupa slot PCI atau ISA. Selain itu terdapat juga beberapa *card* yang diperuntukkan khusus bagi laptop atau *notebook* dengan *socket* PCMCIA.

b. Modem

Modem merupakan perangkat keras yang dapat menghantarkan perubahan data sinyal menjadi analog untuk selanjutnya kembali menjadi data sinyal digital sehingga komputer dapat dijalankan. Biasanya ketika modem mendapatkan sinyal analog maka ia akan merubahnya menjadi signal digital dan menghantarkannya ke komputer. Dalam artian, modem itu bisa membuat komputer/PC terkoneksi dengan jaringan internet.

c. Hub/Switch

Hub atau *Switch* digunakan untuk menghubungkan setiap *node* dalam jaringan komputer, terutama dalam jaringan LAN. Ukuran sebuah *hub/switch* ditentukan oleh jumlah *port* (8 *port*, 16 *port*, 32 *port*, atau 48 *port*). Semakin banyak *port* yang dimiliki *hub/switch* maka semakin banyak komputer yang dapat dihubungkan dalam jaringan.

d. Router

Router adalah sebuah perangkat yang berfungsi untuk menghubungkan dua jaringan atau lebih sehingga pengiriman data dari satu perangkat ke perangkat lain bisa diterima meski berada dalam jaringan yang berbeda. *Router* akan menerima paket-paket data dari internet dan mengirimkan paket-paket data tersesbut menuju sebuah alamat IP tertentu.

3.2.4. Media Transmisi Jaringan Komputer

Media transmisi merupakan media yang menghubungkan antar komputer ataupun jaringan. Media transmisi menghubungkan pengirim dan penerima dalam komunikasi data dan pertukaran informasi. Media transmisi dibagi menjadi dua jenis, yakni kabel dan tanpa kabel (nirkabel).

a. Kabel (*Wired*)

1) *Twisted Pair Cable*

Twisted pair cable merupakan sepasang kabel tembaga yang dipilin berbentuk spiral dan dibungkus dengan lapisan plastik. *Twisted pair cable* dibedakan menjadi dua jenis yaitu kabel UTP (*Unshielded Twisted Pair*) dan STP (*Shielded Twisted Pair*). Diameter kabel kabel *twisted pair* sekitar 0,4 mm hingga 0,8 mm.

2) *Coaxial Cable*

Kabel koaksial adalah kabel dua konduktor yang mana satu konduktor berada di rongga luar mengelilingi satu konduktor tunggal yang dipisahkan oleh bahan isolator. Kabel jenis ini memiliki ukuran hambatan dalam arus bolak-balik yang konstan serta tidak menghasilkan medan magnet sehingga cocok untuk mentransmisikan sinyal frekuensi tinggi.

3) *Fiber Optic Cable*

Kabel serat optic merupakan jenis kabel yang terbuat dari serat kaca atau plastic halus yang dapat mentransmisikan sinyal cahaya dari satu tempat ke tempat lainnya. Sumber cahayanya dapat berupa sinar laser ataupun sinar LED. Diameter kabel serat optic sekitar 120 mikrometer.

b. Nirkabel (*Wireless*)

1) Frekuensi Radio

Frekuensi radio adalah media transmisi yang menggunakan gelombang elektromagnetik dengan kisaran frekuensi diantara 3kHz hingga 300GHz. Frekuensi radio pada umumnya meneggunakan antenna untuk menyebarkan gelombang elektromagnetiknya. Media transmisi ini banyak diaplikasikan pada televisi dan Radio FM.

2) Gelombang Mikro (*Microwave*)

Gelombang mikro adalah media transmisi yang menggunakan gelombang elektromagnetik dengan frekuensi super tinggi yaitu kisaran

3GHz hingga 30GHz dengan panjang gelombang sekitar 1 mm hingga 1 m untuk mentransmisikan sinyal dari pengirim ke penerima.

3) Infra Merah (*Infrared*)

Infra merah merupakan media transmisi yang menggunakan radiasi elektromagnetik. Panjang gelombang infra merah lebih panjang dari cahaya tampak, tapi lebih pendek dari radiasi gelombang radiasi. Infra merah biasanya digunakan pada komunikasi jarak dekat seperti *remote* televisi.

4) Satelit

Media transmisi ini menggunakan satelit sebagai penerima sinyal dari stasiun bumi dan memancarkannya ke stasiun bumi lainnya. Pada umumnya satelit mengorbit pada ketinggian 36.000 km dari permukaan bumi. Satelit sering digunakan untuk siaran televise, telepon jarak jauh, dan jaringan bisnis privat (*private business network*).

5) *Bluetooth*

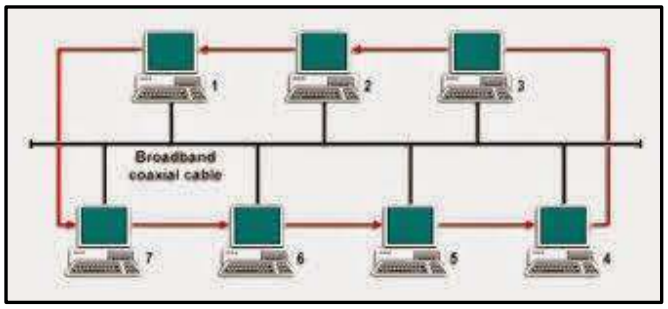
Bluetooth menggunakan fruekuensi 2.4 GHz untuk mengirim dan menerima data. Media transmisi ini sudah sering dijumpai pada fitur di *smartphone*, tablet, dan laptop yang mana *Bluetooth* ini dapat digunakan untuk berbagi *file* antar satu sama lain dengan jarak maksimal 10 m.

3.3. Topologi Jaringan

Topologi jaringan atau arsitektur jaringan adalah gambaran perencanaan hubungan antarkomputer yang melakukan komunikasi satu sama lain. Topologi jaringan mengatur bagaimana setiap *host* terhubung dengan *host* lainnya secara fisik menggunakan media transmisi tertentu. Ada beberapa jenis tolopogi jaringan dalam jaringan komputer, yakni sebegai berikut.

3.3.1. Topologi Bus

Topologi bus merupakan topologi jaringan yang hanya memakai satu kabel untuk media transmisi dimana sepanjang kabel terdapat *node-node* yang terhubung ke kabel utama. Kabel utama dalam topologi ini kedua ujungnya ditutup dengan terminator.



Gambar 3. 4 Topologi Bus
(Sumber: www.it-jurnal.com)

a. Kelebihan

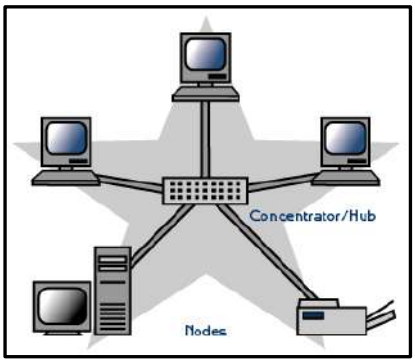
1) Tidak perlu sumber daya kabel yang banyak.
2) Biaya lebih murah.
3) Tidak rumit jika ingin menambah jangkauan jaringan.

b. Kekurangan

1) Tidak cocok untuk *traffic* jaringan yang padat.
2) Setiap *barrel connector* sebagai penghubung memperlemah sinyal elektrik yang dikirimkan.
3) Sulit melakukan *troubleshooting* pada topologi bus.
4) Jika satu *node* rusak semua akan rusak.
5) Lebih lambat.

3.3.2. Topologi Star

Topologi star yaitu bentuk topologi jaringan yang mana terdapat satu penghubung (*hub* / *switch*) sebagai pusat jaringan dan setiap komputer terhubung ke penghubung tersebut. Berbentuk seperti bintang. Jika satu komputer ingin mengirim data ke komputer lain, maka data akan dikirim ke pursat lalu baru dikirimkan ke tujuan.



Gambar 3. 5 Topologi Star
(Sumber: www.it-jurnal.com)

a. Kelebihan

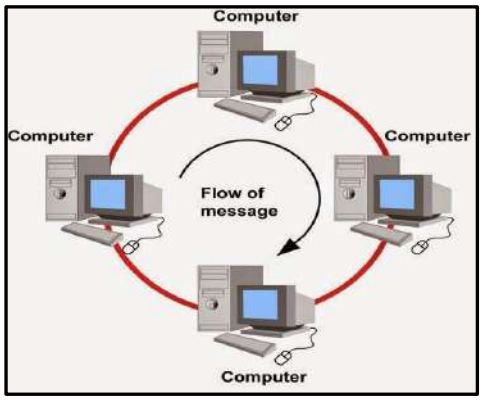
1) Cukup mudah mengubah dan menambah komputer ke komputer.
2) Bila satu rusak tidak akan mengganggu yang lain.
3) Dapat menggunakan beberapa tipe kabel dalam satu jaringan.

b. Kekurangan

1) Jika pusat konsentrator mengalami kegagalan maka jaringan akan gagal berfungsi.
2) Membutuhkan banyak kabel.
3) Jumlah terminal terbatas tergantung *port* konsentrator.
4) Lalu lintas data yang padat dapat memperlambat kinerja.

3.3.3. Topologi Ring

Topologi ring yaitu topologi jaringan yang rangakiannya berupa *node* dimana masing-masing *node* bagian kanan dan kirinya terhubung ke dua *node* lainnya sampai ke komputer yang pertama dan membentuk cicin. Masing-masing *node* ini berfungsi sebagai *repeater* yang memperkuat sinyal. Paket-paket data yang mengalir pada topologi jaringan ini adalah dalam satu arah.



Gambar 3. 6 Topologi Ring
(Sumber: www.it-jurnal.com)

a. Kelebihan

1) Data mengalir satu arah sehingga *collision* dapat dihindarkan.
2) Aliran data mengalir lebih cepat karena dapat melayani data dari kiri atau kanan *server*.
3) Dapat melayani aliran lalu lintas data yang padat.
4) Waktu untuk mengakses data lebih optimal.

b. Kekurangan

1) Apabila ada satu komputer yang gagal berfungsi, maka akan mempengaruhi jaringan.
2) Menambah atau mengurangi komputer akan mengacaukan jaringan.
3) Sulit melakukan konfigurasi ulang.

3.3.4. Topologi Pengembangan

Topologi pengembangan merupakan gabungan beberapa topologi dasar yang dikenal dengan istilah Topologi *Hybrid*/gabungan. Pembuatan topologi *hybrid* dilakukan karena keterbatasan karakteristik topologi dasar yang sebetulnya diperlukan pada implementasi jaringan komputer sehingga untuk melengkapinya digabungkan dengan topologi dasar lain. Beberapa contoh topologi pengembangan, antara lain, Topologi *Tree*, Topologi *Mesh*, dan Topologi *Cell*.

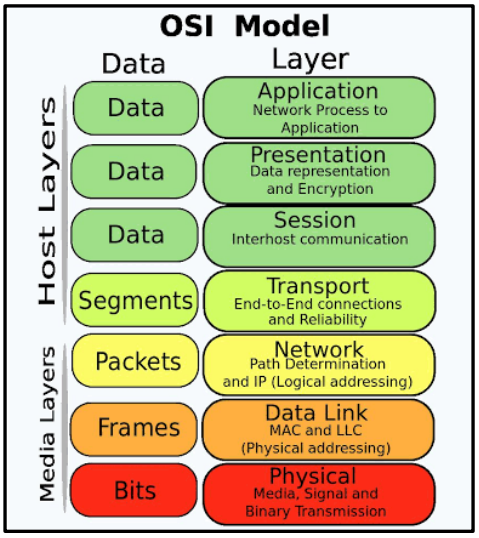
3.4. Model Referensi Komunikasi Data

Model referensi komunikasi data adalah model referensi yang dijadikan sebagai standar dalam implementasi komunikasi data.. Dengan adanya model referensi ini, komunikasi data dapat distandarkan, walaupun *peripheral* yang digunakannya untuk fungsi pengirim, penerima, maupun media yang menghubungkannya diproduksi oleh vendor yang berbeda – beda.

Saat ini, terdapat dua model referensi yang diakui untuk implementasi komunikasi data, yaitu Model Referensi *Open System Interconnection* (OSI) yang dikeluarkan *International Standard Organization* (ISO), dan Model Referensi *Transmission Control Protocol* (TCP/IP) yang pada awalnya dikemukakan oleh Departemen Pertahanan Amerika Serikat.

3.4.1. Model Referensi OSI

Model Referensi *Open System Interconnection* (OSI) adalah model referensi yang ditetapkan oleh *International Standard Organization* (ISO) sebagai standard pertukaran data (arsitektur komunikasi data) antarkomputer. OSI *Reference Model* merupakan model referensi standard yang mempresentasikan komunikasi data antarperalatan jaringan dan antarjaringan. Susunan layer dari model referensi OSI tergambar pada gambar 3.7.



Gambar 3. 7 Model Referensi OSI
(Sumber: www.lifewire.com)

Keuntungan menggunakan OSI *Reference Model*, diantaranya:

a. Jaringan dibagi menjadi bagian-bagian yang lebih kecil sehingga dapat lebih mudah untuk diatur dan dipelajari.
b. Standarisasi *interface* yang digunakan sehingga membantu vendor-vendor perangkat jaringan yang berbeda dalam membangun dan mendukung pengembangan setiap perangkat.

Penjelasan dari lapisan Model Referensi OSI sebagai berikut.

a. Lapisan Aplikasi (*Application Layer*)

Aplikasi atau servis yang melakukan pengolahan data untuk pemakai, seperti *Electronic Mail, File Transfers, Browser*, dan lain-lain.

b. Lapisan Presentasi (*Presentation Layer*)

Menjamin data dapat dibaca sistem yang menerima data, menentukan format data yang dikirimkan atau diterima, menentukan struktur data, mengatur sintaks transfer data bagi Lapisan Aplikasi.

c. Lapisan Sesi (*Session Layer*)

Membangun (*establish*), mengatur (*manage*), dan menghentikan (*terminate*) sesi (*session*) antar aplikasi.

d. Lapisan Transport (*Transport Layer*)

Menentukan metode dan kehandalan pengiriman (*transport*) data antar*hosts*. Membangun (*establish*), menjaga (*maintain*), dan menghentikan (*terminate*) perangkat-perangkat virtual (*virtual circuit*) antar *hosts* atau jaringan.

e. Lapisan Jaringan (*Network Layer*)

Mengatur penentuan jalur (*path*) pengiriman data antara komputer-komputer yang berkomunikasi, yang mempresentasikan komunikasi data antarperalatan jaringan dan antarjaringan. Protokol komunikasi data yang digunakan pada proses ini disebut protokol TCP/IP.

f. Lapisan Datalink (*Datalink Layer*)

Bertanggung jawab dalam meyediakan *link* untuk data. Selain itu, lapisan ini juga menentukan bagaimana bit-bit data dikelompokan menjadi format yang disebut dengan frame. *Datalink Layer* terbagi menjadi dua sub *layer*, yaitu:

1) *Logical Link Control* (LLC)

LLC berfungsi untuk mendefinisikan protokol *network layer* dan kemudian melakukan enkapsulasi protokol-protokol *transport* serta mengkonversi satuan bit data yang dapat dibaca oleh *host* ke satuan sementara yang ada di lapisan *datalink.*

2) *Media Access Control* (MAC)

MAC berfungsi untuk mendefinisikan bagaimana paket ditempatkan pada sebuah media dalam sub *layer* serta mengkonversi satuan data sementara tersebut ke satuan data yang dapat dibaca oleh media transmisi seperti volt, hertz, dan lain-lain.

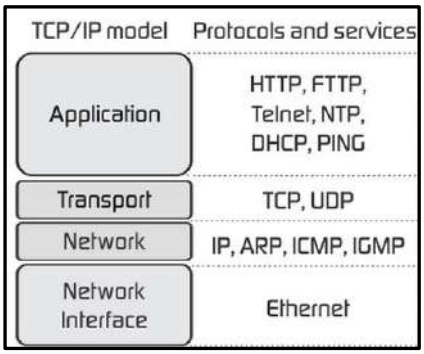
g. Lapisan Physical (*Physical Layer*)

Bertanggung jawab dalam mengirimkan data bit melalui media seperti kabel dan menjaga koneksi fisik antar sistem, lapisan ini juga berfungsi untuk mendefinisikan bagaimana *Network Interface Card* (NIC) dapat berinteraksi dengan media kabel atau radi, metode pensinyalan, sinkronisasi bit, serta arsitektur jaringan seperti *Ethernet* atau *Token Ring*.

3.4.2. Model Referensi TCP/IP

Model Referensi *Transmission Control Protocol/Internet Protocol* (TCP/IP) dibuat oleh *Departement of Defense* (DoD) untuk memastikan dan menjaga integritas data, serta menjaga komunikasi jika terjadi perang bencana. TCP/IP membuat standarisasi dalam interkoneksi jaringan kedalam sebuah lapisan dan juga membuat pengalamatan TCP/IP agar *host-host* yang ada disebuah jaringan dapat saling berkomunikasi.

Susunan struktur lapisan pada Model Referensi TCP/IP dapat dilihat pada gambar 3.8.



Gambar 3. 8 Referensi TCP/IP
(Sumber: fiberbit.com.tw)

Penjelasan dari lapisan Model Referensi OSI sebagai berikut

a. Lapisan Aplikasi (*Application Layer*)

Berperan sebagai protokol *high-level* yang melakukan proses representasi, *encoding*, dan pengendalian dialog antarpemakai. Aplikasi yang bekerja di lapisan ini diantaranya *File Trannsfers, Email, Remote Login, Network Management, DNS,* dan *Browser*.

b. Lapisan Transport (*Transport Layer*)

Lapisan ini mengubah data menjadi suatu paket data dan menentukan metode pengiriman, kendali aliran dan koreksi kesalahan terhadap paket data. Dua protokol yang bekerja di lapisan ini adalah TCP dan UDP.

c. Lapisan Internet (*Internet Layer*)

Berperan memberikan infomasi alamat asal dan tujuan dari paket data dan menentukan jalur atau rute (*routing*) pengiriman paket data. Protokol yang bekerja dalam lapisan ini mengatur kinerja untuk lalulintas jaringan, yaitu IP, ICMP, ARP, dan RARP.

d. Lapisan Akses Jaringan (*Network Access layer*)

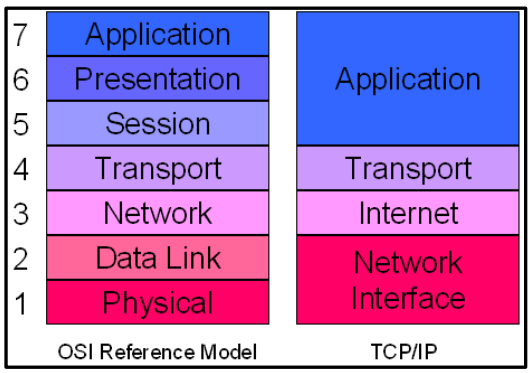
Lapisan ini sering disebut lapisan *host-to-network.* Lapisan ini menangani komponen dan proses yang berkaitan dengan sambungan fisik (*physical link*), baik secara fisik maupun logika. Informasi mengenai teknologi jaringan yang digunakan juga ditentukan pada lapisan ini.

Agar beberapa komputer dapat berkomunikasi menggunakan protokol TCP/IP ada dua pengaturan pokok, yaitu :

1) Pengaturan aplikasi yang digunakan untuk akses komunikasi, dilakukan melalui protokol aplikasi yang tergabung dalam kelompok TCP (*Transmission Control Protocol*), seperti pengaturan *software* pada lapisan aplikasi, juga pengaturan dalam penyajian aplikasi pada lapisan transport.
2) Pengaturan jaringan yang bertanggung jawab untuk penyelenggaraan koneksi antarkomputer yang saling berkomunikasi satu sama lain. Pengaturan dilakukan melalui protokol jaringan yang bertanggung jawab dalam kelompok IP (*Internet Protocol*). Kegiatannya meliputi proses pengalamatan (*addressing*) dan proses *routing* paket data.

Dalam aplikasinya, subprotokol yang merupakan turunan dari TCP/IP diterjemahkan lagi oleh masing-masing subprosesnya. Misalnya, aplikasi *browser* ada http (*hyper text transfer protocol*), sedangkan layanan *email* dapat digunakan SMTP dan POP, dan sebagainya. Tanggung jawab penyelenggaraan koneksi aplikasi tersebut ada kelompok protokol yang merupakan bagiandari TCP dan ada yang merupakan bagian dari UDP.

Perbandingan antara kelompok protokol TCP/IP dengan penerapan fungsi model referensi OSI dapat dilihat pada gambar.



Gambar 3. 9 Perbandingan OSI dan TCP/IP
(Sumber: www.gemaroprek.com)

Ada sebuah metode yang digunakan dalam jaringan TCP / IP untuk membuat koneksi antar *host* yang akan saling bertukar informasi ataupun antar *client* dan *server* yakni, *Three Way Handshake* atau dikenal juga sebagai jabat tangan TCP. Ini adalah metode tiga langkah yang membutuhkan baik klien dan *server* untuk bertukar SYN dan ACK (pengakuan) paket sebelum komunikasi data aktual dimulai.

Proses terjadinya *three way handshake* dijelaskan seperti berikut (ismitggwp, 2018) :

1. *Host* sumber akan mengirim segment bernama ***synchronization (SYN)*** sebagai ***"connection agreement"***.
2. *Host* tujuan akan membalas mengirim segment ***synchronization (SYN)*** dan segment ***acknowledge (ACK)*** sebagai tanda bahwa segment ***SYN*** sudah diterima.
3. Host sumber akan mengirim segment ***ACK*** sebagai tanda bahwa segment ***SYN*** sudah diterima dan koneksi sudah terbuat.
4. Ketika koneksi sudah terbuat, Host sumber dan tujuan dapat saling bertukar data.

3.4.3. Enkapsulasi dan Dekasulapsi

Enkapsulasi merupakan sebuah proses untuk membuat satu jenis paket data jaringan menjadi jenis data lainnya. Enkapsulasi terjadi ketika sebuah protokol yang berada pada lapisan yang lebih rendah menerima data dari protokol yang berada pada lapisan yang lebih tinggi dan meletakkan data ke format data yang lebih dipahami oleh protokol tersebut (Sukmaaji & Rianto, 2008).

Enkapsulasi data pada layer 4 disebut sebagai *segment*. *Segment* selanjtunya dikirim ke lapisan *network* sebagai data. Pada lapisan *network* data kembali dikemas dengan informasi yang relevan untuk lapisan 3 berupa *header*. Pada lapisan *network*, hasil enkapsulasi data disebut sebagai paket. Paket diteruskan ke lapisan *datalink* dan diberi informasi yang disebut sebagai header *layer 2*. Setelah mendapat informasi *header* lapisan dua, kemudian disebut sebagai *frame*. *Frame*

kemudian memasuki *layer* satu yaitu *physical layer* dan diubah menjadi *bitstream* yang akhirnya ditransmisikan ke tujuan.

Proses transmisi data pada lapisan fisik, bentuk transmisi datanya dipengaruhi oleh media yang digunakan dan media inilah yang disebut media transmisi. Sesampai di tujuan, *bitstream* ini kemudia dubah menjadi *frame* dengan melepas *frame-header* dan dikirim ke *layer-3* sebagai paket. Paket selanjutnya melepas *header* dan mengirim data tersebut ke *layer-4* sebagai *segment*. *Segment* kemudian melepas header lapisan ke 4 dan memberikan data ke lapisan 5, 6, 7 yang akhirnya diterima oleh pengguna sebagai data. Proses pelepasan *header* dari lapisan ke lapisan ini disebut sebagai dekapsulasi.

3.5. IP *Address*

IP *Address* merupakan alamat identifikasi unik yang dimiliki oleh setiap komputer dan perangkat lainnya. IP *Address* harus bersifat unik pada tiap perangkat. Terdapat dua jenis alamat IP yaitu :

a. IPv4, merupakan alamat IP yang terdiri dari 32 bit. Dibagi menjadi 4 segmen yang dipisahkan oleh titik yang mana setiap segmennya berukuran 8 bit.
b. IPv6, merupakan alamat IP yang terdiri dari 128 bit. Dibagi menjadi 8 segmen yang dipisahkan oleh titik dua yang mana setiap segmennya berukuran 16 bit.

3.5.1. Pembagian Kelas IPv4

Secara teoritis alamat IP versi 4 dengan susunannya sebanyak 32 bit berarti mampu mengintegrasikan sebanyak 4.294.967.296 (2^{32}). Empat milyar lebih alamat IP ini diklasifikasikan menjadi beberapa kelas tertentu. Pembagian kelas-kelas ini bertujuan untuk mempermudah alokasi alamat IP , baik untuk *host*/jaringan tertentu maupun untuk keperluan tertentu (Daryanto, 2010). Pembagian kelas alamat IP versi 4 dibagi menjadi 5 kelas seperti yang tercantum pada gambar tabel berikut :

Tabel 3. 1 Pembagian Kelas IPv4

Kelas	Oketet pertama dalam desimal	Oketet pertama dalam biner	Penggunaan
Kelas A	1 – 126	0xxx xxxx	Jaringan komputer dengan sekal besar
Kelas B	128–191	10xx xxxx	Jaringan komputer dengan skala menengah sampai besar
Kelas C	192–223	110x xxxx	Jaringan komputer dengan skala kecil
Kelas D	224–239	1110 xxxx	Alamat multicast
Kelas E	240–255	1111 xxxx	Alamat percobaan atau eksperimen

(Sumber : www.yuksinau.id)

3.5.2. IP *Address Public* dan *Private*

a. IP *Public*

Alamat IP *Public* merupakan alamat IP yang digunakan dalam komunikasi jaringan internet. Kepemilikan alamat IP *Public* ini diatur oleh *Internet Service Provider* secara langsung ketika perangkat terhubung ke *gateway* internet.

b. IP *Private*

IP *Private* merupakan alamat IP yang bersifat lokal dan pribadi karena digunakan sebagai identifikasi komputer pada keperluan jaringan berskala lokal (LAN) / intranet. Daftar IP *Private* dapat dilihat pada tabel 3.2.

Tabel 3. 2 *Range* IP *Private*

Kelas	Range
A	10.0.0.0 s.d 10.255.255.255
B	172.16.0.0 s.d 172.31.255.255
C	192.168.0.0 s.d 192.168.255.255

3.5.3. NetID dan HostID

Secara umum, sebuah alamat IP tersusun ataas 32 *bit* yang mendefinisikan koneksi sebuah *host* ke jaringan. Dalam hal ini, terdapat dua macam identitas yaitu:

1. NetID, yaitu mengindetifikasikan jaringan (*network*). Jika diibaratkan, NetID ini merupakan alamat rumah.
2. HostID, yaitu mengindetifikasikan sebuah *host* ke jaringan. Istilah host sama dengan stasuin (*station*) atau titik (*node*). Jika diibaratkan, HostID ini merupakan nomor dari sebuah rumah.

3.6. Protokol Jaringan

Protokol jaringan adalah sebuah aturan yang mendefinisikan beberapa fungsi yang ada dalam sebuah jaringan komputer, misalnya mengirim pesan, data, informasi, dan fungsi lain yang harus dipatuhi oleh pengirim (*transmitter*) dan penerima (*receiver*) agar komunikasi dapat berlangsung dengan benar (Sukmaaji & Rianto, 2008).

Protokol berfungsi agar komputer yang berada dalam jaringan berkomunikasi dengan bahasa yang sama. Penggunaan protokol erat kaitannya dengan aplikasi yang digunakan dalam suatu jaringan komputer. Masing-masing protocol memiliki definisi berdasarkan nomor *port*. Beberapa jenis protokol jaringan adalah :

3.6.1. TCP

TCP atau *Transmission Control Protocol* adalah protocol yang berada di lapisan *transport*. TCP dapat mentransmisikan data per segmen, artinya paket data dipecah dalam jumlah yang sesuai dengan besaran paket, kemudian dikirim satu persatu hingga selesai. TCP memiliki konsep *connection-oriented*, yang artinya koneksi *end-to-end* harus dibangun dulu di kedua ujung terminal sebelum mengirimkan data.

3.6.2. ICMP

ICMP atau *Internet Control Message Protocol* merupakan bagian dari *Internet Protocol*. ICMP digunakan peralatan-peralatan yang terhubung melalui jaringan internet untuk keperluan analisa jaringan. Penggunaan ICMP yang terkenal adalah *ping* dan *traceroute*.

Pesan-pesan ICMP umumnya dibuat sebagai jawaban atas kesalahan di datagram IP atau untuk kegunaan pelacakan atau *routing* (Sukmaaji & Rianto, 2008).

3.6.3. HTTP

HTTP atau *Hyper Text Transfer Protocol* merupakan protokol utama yang digunakan untuk mengakses data melalui *World Wide Web* (WWW). Protokol ini dapat digunakan unutk mentrasfer data dalam format *plaintext, hypertext,* audio, video, dan lain-lain (Sukmaaji & Rianto, 2008).

Cara kerja protokol ini cukup sederhana. Jika ada permintaan (*request*) dari sisi *client*, maka *server* akan menanggapi permintaan tersebut dengan mengirimkan hasilnya dalam bentuk beberapa halaman web atau dokumen HTML.

3.6.4. SSH

SSH atau *Secure Shell* adalah protokol jaringan kriptografi yang mengoperasikan layanan jaringan melalui saluran yang aman yang menghubungkan aplikasi SSH pada *client* dengan SSH pada *server*. SSH ini banyak digunakan pada sistem berbasis Linux dan Unix untuk mengakses akun *shell*.

3.6.5. DNS

DNS atau *Domain Name System* adalah metode agar sebuah situs web dapat ditemukan tanpa harus mengetahui letak secara fisik situs tersebut di dalam internet. Fungsi DNS adalah menerjemahkan nama komputer ke dalam alamat IP sehingga pengguna dalam mencari sebuah web tidak perlu mengetahui alamat IP sebenarnya

melainkan hanya dengan mengetahui nama domain yang dibuat untuk menerjemahkan alamat IP tersebut.

3.7. Port

Port adalah mekanisme yang mengizinkan sebuah komputer untuk mendukung beberapa sesi koneksi dengan komputer lainnya dan program di dalam jaringan. *Port* dapat mengidentifikasikan aplikasi dan layanan yang menggunakan koneksi di dalam jaringan TCP/IP. Sehingga, *port* juga mengidentifikasikan sebuah proses tertentu di mana sebuah *server* dapat memberikan sebuah layanan kepada klien atau bagaimana sebuah klien dapat mengakses sebuah layanan yang ada dalam *server* (Kanigoro, 2019).

3.8. Remote Access

Remote access adalah sebuah kemampuan yang dimiliki perangkat untuk dapat tersambung dengan *resource* pada satu network atau jaringan dari satu tempat tertentu. Hal ini dapat memungkinkan pengguna dari suatu perangkat mengakses komputer lain dan melakukan semua aktifitas pada komputer tersebut seolah-olah sedang menggunakan komputer tersebut secara fisik. Ada dua peran dalam *remote access*. *Host* yaitu komputer yang dikendalikan, dan *remote* sebagai pengendali.

3.8.1. PuTTY

PuTTY adalah sebuah aplikasi *open-source* yang memanfaatkan protokol jaringan seperti SSH dan Telnet. PuTTY memanfaatkan protokol tersebut untuk mengaktifkan sesi *remote* pada komputer. PuTTY biasanya dipakai oleh para pemilik *server* untuk berkomunikasi dengan *server*-nya menggunakan *command teks* guna menjalankan perintah tertentu pada *server*. PuTTY juga mendukung banyak protokol jaringan lain, seperti SCP, rlogin, *serial port*, dan *raw socket connection* (Cara Menggunakan PuTTY, 2020).

3.9. Server

Server atau dalam bahasa Indonesia disebut peladen merupakan suatu sistem komputer yang memiliki layanan khusus berupa penyimpanan data. Data yang disimpan melalui *server* berupa informasi dan beragam jenis dokumen yang kompleks. Layanan tersebut ditujukan khusus untuk *client* yang berkebutuhan dalam menyediakan informasi untuk pengguna atau pengunjungnya (Apa itu *Server*, 2020).

Server berperan penting dalam menyediakan layanan akses lebih cepat untuk mengirim atau menerima data maupun informasi yang tersedia pada *server*. Dalam bentuk fisiknya, *server* berwujud jaringan komputer dan memiliki ukuran yang sangat besar dengan beberapa komponen pendukung prosesor dan RAM yang berkapasitas besar.

3.10. Sistem Operasi

Sistem Operasi adalah salah satu perangkat lunak atau *software* yang bertanggung jawab dalam mengatur atau mengontrol kerja perangkat keras atau *hardware* dan menjalankan aplikasi atau *software* di dalam suatu jaringan kmputer. Dengan kata lain sistem operasi adalah sistem yang mengendalikan operasi dasar dan memastikan sistem dalam komputer dapat berjalan dengan semestinya. Sistem Operasi memungkinkan suatu aplikasi dapat berfungsi sehingga sistem operasi juga disebut sebagai *essential component*. Komputer dan sistem hanya dapat berfungsi dengan adanya sistem operasi kecuali jika komputer sedang dalam keadaan *booting* (Zakaria, 2019) .

Bisa dipahami bahwa sistem operasi adalah penghubung antara *hardware* dan *software*. Saat komputer pertama kali dinyalakan, yang sedang berjalan itulah sistem operasi. Dan setelah komputer menyala barulah program dan aplikasi dapat berjalan. Saat ini beberapa sistem operasi yang dikenal antara lain *Windows*, *Linux*, *MacOS*, dan lain-lain.

3.10.1. CentOS

CentOS adalah sistem operasi bebas yang dikembangkan oleh *Red Hat Enterprise Linux* (RHEL). Sistem operasi ini adalah distribusi linux sebagai bentuk dari usaha untuk menyediakan sebuah *platform* komputasi yang berkelas *enterprise* dan memiliki kompatibilitas kode biner sepenuhnya dengan kode sumber yang menjadi induknya (Pengertian CentOS Linux, 2019).

Gambar 3. 10 Logo CentOS
(Sumber: commons.wikimedia.org)

CentOS dirilis pada tahun 2004 dan sudah banyak pengguna yang memanfaatkannya. Kelebihan CentOS yaitu sangat mudah dimodifikasi, aman, dan stabil. Karena masih berada dibawah RHEL, CentOS memiliki tingkat keamanan yang terus-menerus diperbarui.

3.10.2. *Shell Script*

Dikutip dari (Mufrizal, 2016) *Shell Script* adalah sebuah bahasa pemrograman yang disusun berdasarkan perintah - perintah shell. Shell script berisi kumpulan beberapa *command* yang ditulis pada sebuah file yang nantinya akan diexecute oleh *shell*. Berikut beberapa keuntungan menggunakan *shell script* :

1) Mengeliminasi task yang berulang-ulang.
2) Menghemat waktu.
3) Menghadirkan urutan aktivitas yang terstruktur, modular, dan terformat.
4) Menyederhanakan *command* yang kompleks menjadi satu *command* aktif yang bisa dijalankan.
5) Digunakan sesering mungkin oleh user. Satu *shell script* untuk berkali-kali pemakaian.

3.11. Keamanan Jaringan

Keamanan jaringan adalah suatu sistem yang memiliki tugas untuk melakukan pencegahan dan identifikasi kepada pengguna yang tidak sah dalam jaringan komputer. Langkah pencegahan ini berfungsi untuk menghentikan penyusup untuk mengakses lewat sistem jaringan komputer. Tujuan dari dilakukan sistem keamanan jaringan komputer adalah untuk antisipasi dari ancaman dalam bentuk fisik maupun *logic* baik secara langsung atau tidak langsung yang mengganggu sistem keamanan jaringan (Eril, 2020).

Satu hal yang perlu dicatat bahwa tidak ada jaringan komputer yang benar-benar aman dan untuk mecapai suatu keamanan yang benar-benar aman itu adalah suatu hal yang mustahil. Namun gangguan keamanan tersebut bisa dicegah dan dikurangi, salah satunya adalah dengan meningkatkan sistem keamanan jaringan.

3.11.1. Aspek Keamanan Jaringan

Keamanan jaringan meliputi beberapa aspek (Ariyus, 2005), diantaranya :

1. *Authentication* : agar penerima informasi dapat memastikan keaslian pesan tersebut datang dari orang yang dimintai informasi. Dengan kata lain informasi tersebut benar-benar dari orang yang dikehendaki.
2. *Integrity* : keaslian pesan yang dikirim melalui sebuah jaringan dan dapat dipastikan bahwa informasi yang dikirim tidak dimodifikasi oleh orang yang tidak berhak dalam perjalanan informasi tersebut.
3. *Nonrepudiation* : atau tidak terbantahkan merupakan hal yang bersangkutan dengan si pengirim. Si pengirim tidak dapat mengelak bahwa dialah yang mengirim informasi tersebut.
4. *Authority* : informasi yang berada pada sistem jaringan tidak dapat dimodifikasi oleh pihak yang tidak berhak atas akses tersebut..
5. *Confidentiality* : merupakan usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. Biasanya berhubungan dengan informasi yang diberikan kepada pihak lain.
6. *Privacy* : lebih ke arah data-data yang sifatnya pribadi.

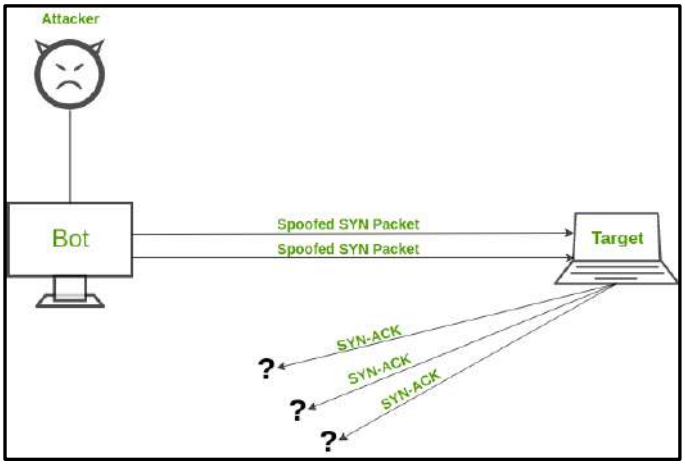
7. *Availability* : aspek ketersediaan berhubunagn dengan ketersediaan informasi ketika dibutuhkan. Sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.
8. *Access Control* : aspek ini berhubungan dengan cara pengaturan akses kepada informasi. Hal itu biasanya berhubungan dengan masalah *authentication* dan juga *privacy*. *Access control* seringkali dilakukan menggunakan kombinasi *user id* dan *password* atau dengan menggunakan mekanisme lainnya.

3.12. DoS

DoS atau *Denial of Service* merupakan suatu istilah yang diberikan sebagai upaya serangan dengan cara menurunkan kinerja *server* secara terus menerus untuk mengulangi permintaan (*request*) ke *server* dari beberapa sumber secara simultan.

Serangan seperti itu bertujuan membuat *server* korban menjadi kewalahan untuk melayani permintaan yang terkirim dan berakhir dengan penghentian aktivitas atau berhenti dengan sendirinya. DOS juga merupakan serangan yang dilancarkan melalui paket-paket tertentu, biasanya paket-paket sederhana dengan jumlah yang sangat banyak yang bermaksud mengacaukan keadaan jaringan target (Ariyus, 2005).

3.12.1. SYN Flood



Gambar 3. 11 *SYN Flood*
(Sumber: www.dlpng.com)

SYN Flood adalah serangan DoS di mana penyerang mengirim paket TCP SYN dalam jumlah besar ke *server*. Setelah itu *server* mencoba untuk menanggapi paket SYN ini dengan mengirimkan paket ACK-SYN ke alamat pengirim, yang mana alamat penyerang tidak akan membalas paket ACK-SYN dengan paket ACK, melainkan penyerang akan terus mengirimkan paket SYN, sehingga menyebabkan adanya antrian koneksi untuk beberapa waktu. Ketika jumlah koneksi yang menunggu melebihi batas antrian, semua permintaan SYN berikutnya dibatalkan dan hal ini mengarah kepada penolakan layanan dari *server* terhadap permintaan dari pengguna. Tujuan serangan ini adalah membuat sistem menjadi tidak responsif terhadap permintaan pengguna yang sah. Gambar 3.11 adalah gambaran cara kerja serangan ini.

3.13. IDS

Intrusion Detection System (IDS) merupakan penghambat semua serangan yang akan mengganggu sebuah jaringan. IDS memberikan peringatan kepada administrator *server* saat terjadi sebuah aktivitas tertentu yang tidak diinginkan administrator sebagai penanggung jawab sebuah sistem. Selain memberikan peringatan, IDS juga mampu melacak jenis aktivitas yang merugikan sebuah sistem. Suatu IDS akan melakukan pengamatan terhadap paket-paket yang melewati jaringan dan berusaha menemukan apakah terdapat paket-paket yang berisi aktivitas mencurigakan sekaligus melakukan tindak lanjut pencegahan (Ariyus, 2005).

Menurut cara beroperasinya, IDS dikategorikan menjadi dua jenis yaitu, *Host-based Intrusion Detection System* (HIDS), dan *Network-based Intrusion Detection System* (NIDS).

3.13.1. *Host-based Intrusion Detection System* (HIDS)

HIDS adalah jenis IDS yang menganalisis aktivitas hanya pada host yang bersangkutan secara individual apakah terdapat percobaan penyerangan atau pengusupan ke dalam jaringan dan melakukan pengawasan terhadap paket-paket yang berasal dari dalam maupun luar hanya pada satu alat saja dan kemudian memberikan peringatan terhadap sistem atau administrator jaringan (Gemilang, 2018).

3.13.2. *Network-based Intrusion Detection System* (NIDS)

NIDS adalah jenis IDS yang melakukan pemantauan terhadap serangan serta *traffic* pada seluruh bagian jaringan. NIDS menangkap paket data yang bergerak di suatu media jaringan (kabel, nirkabel) dan mencocokkan paket data tersebut dengan *signature-signature* yang terletak di *database*. Bila paket data cocok dengan *signature* dari penyerang maka peringatan (*alert*) akan dibuat dan paket data akan disimpan ke sebuah file atau *database* (Ariyus, 2005).

3.13.3. Proses Utama *Intrusion Detection System*

Anisya menjelaskan (Dikutip dalam Stiawan, 2005) bahwa IDS memiliki tiga komponen fungsi fundamental yang merupakan proses utama dalam IDS :

1. Pengambilan data (*information sources*). Komponen ini merupakan fungsi untuk melakukan pengambilan data dari berbagai sumber yang ada pada sistem yang diamati.
2. Analisis. Bagian ini melakukan organisasi terhadap data yang diperoleh, mengambil kesimpulan terhadap pelanggaran baik yang sedang terjadi maupun yang telah terjadi.
3. Respon. Komponen ini melakukan beberapa aksi pada sistem setelah pelanggaran yang terjadi telah terdeksi. Respon ini dapat dikelompokkan menjadi dua, yaitu respon aktif dan respon pasif. Respon aktif dalam hal ini berarti melakukan beberapa aksi secara otomatis untuk mengintervensi

sistem yang ada. Sedangkan pasif adalah memberikan *report* pada administrator yang akan melakukan respon terhadap sistem

3.14. Suricata

Suricata adalah aplikasi pendeteksi ancaman jaringan yang berbasis *open source*. Suricata mampu mendeteksi intrusi (IDS) secara *real time*, pencegahan intrusi (IPS), dan pemantauan keamanan jaringan (NSM) dan pemrosesan pcap *offline*. Suricata dimiliki oleh sebuah komunitas non-profit, yaitu Open Information Security Foundation (OISF). Suricata dikembangkan oleh OISF dan vendor pendukungnya.

Suricata bekerja dengan cara memeriksa lalu lintas jaringan menggunakan *rules and signature language*. Suricata mendukung Lua *scripting* untuk mendeteksi ancaman yang kompleks. Dengan format *input* dan *output* standar seperti YAML dan JSON yang memungkinkan kemudahan integrasi dengan *tool* seperti Splunk, Logstash / Elasticsearch, Kibana. Logo suricata dapat dilihat pada gambar 3.12.

Gambar 3. 12 Logo Suricata
(Sumber: www.suricata-ids.org)

3.14.1. Fitur Suricata

Dikutip dari (Tentang Suricata, 2017), fitur – fitur yang ada pada suricata adalah sebagai berikut :

1. IDS IPS Suricata mengimplementasikan *signature language* yang lengkap untuk mencocokkan dengan ancaman yang dikenal dan perilaku berbahaya. Suricata juga mendeteksi banyak anomali pada lalu lintas. Suricata mampu menggunakan *ruleset* dari *Emerging Threats Suricata* dan VRT *ruleset*.

2. *High Performance*. Suricata mampu melakukan inspeksi lalu lintas multi-gigabit. *Engine* pada suricata dibangun secara *multi threading*, modern, basis kode yang bersih dan *scalable*. Suricata secara eksperimental mampu menggunakan GPU *Acceleration* untuk tugas yang intensif.
3. *Automatic Protocol Detection*. Suricata secara otomatis mendeteksi protokol seperti HTTP pada sembarang *port* dan mengaplikasikan pendeteksian yang diperlukan dan *logging logic*. Ini sangat membantu untuk menemukan *malware* dan CnC *channels*.
4. NSM : IDS Suricata mampu melakukan log HTTP *request*, mencatat dan menyimpan sertifikat TLS, ekstrak file yang mengalir dan menyimpan ke disk. Mendukung pcap *capture* secara penuh untuk memudahkan analisa. Ini membuat Suricata sebuah mesin yang *powerfull* bagi *Network Security Monitoring* (NSM).
5. *Lua Scripting*. Analisa dan fungsionalitas yang lebih maju tersedia untuk mendeteksi sesuatu tidak memungkinkan antara *ruleset syntax*.
6. *Industry Standard outputs* pada suricata versi 2.0 diperkenalkan “Eve”, semua JSON even dan alert output. Ini memperbolehkan integrasi yang mudah dengan *logstash* dan *tool* yang mirip.

3.14.2. *Rule* Suricata

Rule pada suricata merupakan file dengan ekstensi *.rules* yang berisi *signature rule* yang berfungsi dalam proses pencocokan jenis ancaman serangan yang terjadi. Salah satu contoh *signature rule* adalah :

```
alert icmp any any -> $HOME_NET any (msg:"Koneksi icmp terjadi"; sid:1000001;)
```

Pada contoh di atas dapat dilihat bahwa s*ignature rule* terdiri dari beberapa bagian, yaitu :

a) *Action*

Menentukan apa yang akan dilakukan jika terdapat kecocokan pada *signature rule.* Ada 4 tipe *action* yakni :

1. *Pass* = melewati paket yang cocok.

2. *Drop* = membuang atau menghentikan paket yang masuk tanpa memberi pesan kepada pengirim paket. Belaku pada mode IPS.
3. *Reject* = membuang atau menghentikan paket yang masuk dengan memberi pesan kepada pengirim paket. Berlaku pada mode IPS.
4. *Alert* = memberikan peringatan mengenai paket yang mana hanya bisa dibaca oleh administrator sistem.

b) *Header*

Mendefinisikan asal dan tujuan protokol, port, dan alamat IP pada paket.

c) *Rule Options*

Mendefinisikan aturan-aturan lain secara spesifik.

3.15. API

API adalah singkatan dari *Application Programming Interface* dan memungkinkan *developer* untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan (Sandi, 2017). Maksudnya adalah seperangkat antarmuka (bisa berbentuk fungsi, *method* atau URL *endpoint*) yang dapat digunakan untuk mengembangkan aplikasi, baik dalam satu platform maupun lintas platform.

Tujuan dari API adalah untuk mempercepat pembuatan suatu aplikasi karena *programmer* tidak perlu menulis kode dari nol. API juga disediakan oleh sebuah platform untuk dapat mengakses fitur dari platform tersebut. Contoh dari API yang lintas platform adalah API Twitter dan Facebook, yang memungkinkan kita untuk dapat mengakses data pengguna platform tersebut pada aplikasi kita. Contoh lain seperti API Bot Telegram dan LINE yang memungkinkan aplikasi kita untuk dapat mengirim dan membaca *chat* dari pengguna platform tersebut secara otomatis.

3.16. Telegram

Dari (Wikipedia, 2015) dijelaskan bahwa telegram adalah sebuah aplikasi layanan pengirim pesan instan *multiplatform* berbasis *cloud* yang gratis dan nirlaba.

Telegram tersedia untuk berbagai perangkat telepon seluler seperti Android, iOS, Windows Phone, dan Ubuntu Touch. Selain itu telegram juga mendukung sistem operasi Windows, OS X, dan Linux. Dengan menggunakan telegram, para pengguna dapat mengirim pesan dan bertukar foto, video, stiker, audio, dan tipe berkas lainnya. Telegram juga memungkinkan pengguna untuk mengirim pesan secara rahasia karena telegram memiliki fitur enkripsi *end-to-end* sebagai keamanan tambahan.

3.16.1. API Bot Telegram

Bot API pada telegram memungkinkan dengan mudah membuat program yang menggunakan pesan telegram untuk sebuah antarmuka. API ini menghubungkan bot atau program yang dibuat ke sistem telegram.

Akun dari telegram bot ini adalah akun khusus yang tidak memerlukan nomor telepon tambahan untuk mengaturnya. Akun ini berfungsi sebagai antarmuka untuk kode yang dijalankan di suatu tempat di *server*.

Dalam penggunaannya, tidak perlu mengetahui tentang cara kerja protokol enkripsi MTProto yang digunakan oleh telegram. *Server* perantara telegram akan menangani semua enkripsi dan komunikasi dengan API Telegram. Komunikasi *server* adalah ini melalui antarmuka HTTPS sederhana yang menawarkan versi sederhana dari API Telegram.


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

BAB IV
INSTALASI DAN KONFIGURASI *INTRUSION DETECTION SYSTEM* (IDS) MENGGUNAKAN SURICATA DENGAN NOTIFIKASI TELEGRAM PADA *SERVER* DINAS TENAGA KERJA BANDUNG DI PT BENTANG INSPIRA TEKNOLOGI

4.1. Tahap Perencanaan

Tahap perencanaan ini adalah sebagai acuan dalam pengerjaan melakukan instalasi dan konfigurasi IDS di *server* Disnaker Bandung. Tahap ini berisi kebutuhan alat bahan yang diperlukan dan skenario dari proses instalasi hingga pengujian, diantaranya sebagai berikut.

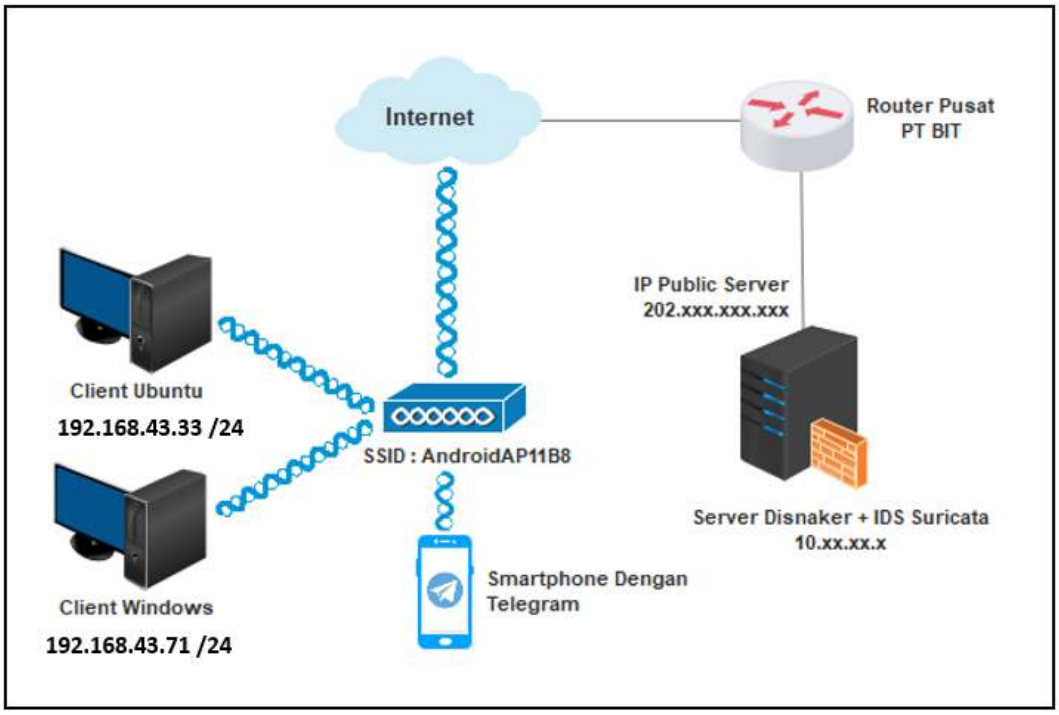
Instalasi dan konfigurasi *Intrusion Detection System* (IDS) atau sistem pendeteksi gangguan pada *server* Dinas Tenaga Kerja Bandung di PT BIT sebagai solusi agar keamanan jaringan pada *server* tersebut lebih terjaga dan terpantau. Administrator *server* dapat langsung mengetahui jika terjadi serangan karena sistem IDS yang diterapkan akan langsung mengirimkan notifikasi melalui telegram.

Instalasi pada *server* dilakukan secara *remote* dari laptop penulis menggunakan aplikasi PuTTY. *Server* Disnaker sendiri menggunakan sistem operasi CentOS 7 dan telah menyediakan *domain* disnaker.bandung.go.id dan juga *port* untuk melakukan instalasi. Mesin yang digunakan penulis sebanyak dua mesin, yakni untuk melakukan *remote server* dan sebagai penyerang untuk pengujian.

Aplikasi IDS yang digunakan adalah Suricata versi 3.1. Penulis hanya akan menerapkan *rule* atau aturan DoS *SYN Flood* pada IDS sebagai bahan pengujian instalasi IDS ini. Jika terjadi serangan DoS *SYN Flood* maka IDS suricata akan memberikan *alert* atau peringatan yang mana ini akan tersimpan di file log suricata. Log ini akan menjadi acuan detail bagi administrator mengenai serangan yang terjadi seperti alamat IP asal, waktu, dan lain-lain. Untuk sistem notifikasi serangan ke telegram penulis membuat bot khusus suricata di telegram yang nantinya akan menerima notifikasi dari *server* dan mengirimkannya ke grup telegram yang

beranggotakan bot dan administrator. Pada sisi *server*, penulis membuat skrip *shell* untuk sistem pengiriman notifikasi serangan ke telegram yang cara kerjanya adalah mengambil data dari file log suricata.

4.2. Topologi Perencanaan



Gambar 4. 1 Topologi Instalasi *Intrusion Detection System*

Berdasarkan topologi pada gambar agar dapat terhubung dengan *server* Disnaker Kota Bandung di PT Bentang Inspira Teknologi, mesin *client* Windows dan Ubuntu yang dimiliki penulis yang masing-masing memiliki fungsi untuk *remote server* dan sebagai penyerang, terhubung secara *wireless* ke *access point* (dalam hal ini *handphone* penulis) yang juga langsung terhubung ke internet. Sedangkan pada sisi *server* Disnaker hanya terhubung ke *router* pusat sebelum terkoneksi dengan jaringan *public* yakni internet. *Handphone* penulis yang telah terinstal telegram pun terkoneksi ke jaringan internet agar dapat menerima notifikasi dari *server* Disnaker Kota Bandung. *Server* Disnaker ini sendiri telah terinstal *Web Server* sebagai layanannya.

4.3. Data Teknis

Data teknis di bawah ini menjelaskan mengenai alat dan bahan serta spesifikasi sistem yang digunakan dalam proses pengerjaan instalasi dan konfigurasi IDS suricata pada *server* Disnaker Bandung, seperti yang terlihat pada tabel 4.1.

Tabel 4. 1 Data Teknis

Nama	**Spesifikasi**	**Penjelasan**
Server Disnaker Kota Bandung	1) OS : Centos 7 2) CPU : Intel(R) Xeon(R) CPU E5620 @ 2.40GHz 3) Memori : 4 GB 4) HDD : 1.8 TB	*Server* dengan layanan utamanya Web *Server*, tempat diterapkannya IDS Suricata.
PC Client (*remote server*)	1) OS : Windows 10 2) CPU : Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz 3) Memori : 4 GB 4) HDD : 300 GB	Mesin yang digunakan untuk melakukan instalasi IDS di *server* Disnaker dengan cara remote *server*.
PC Client (*attacker*)	1) OS : Ubuntu 16.04 2) CPU : Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz 3) Memori : 512 MB 4) HDD : 10 GB	Mesin yang digunakan untuk melakukan serangan DoS SYN Flood sebagai bahan pengujian.
Paket Instalasi Suricata	Paket instalasi tersedia di www.openinfosecfoundation.org versi 3.1, dengan mode Host-based Intrusion Detection System (HIDS).	Aplikasi IDS sebagai pendeteksi adanya serangan.

Telegram	Versi 7.2.1	Aplikasi telekomunikasi sebagai media penyampaian notifikasi serangan suricata kepada administrator.
Aplikasi PuTTY	Versi release 0.70	Aplikasi *remote connection* dengan tipe koneksi SSH untuk mengakses *server* Disnaker Kota Bandung.
Aplikasi Hping3	Versi 3.0.0	Aplikasi yang dapat mengrimkan paket jaringan dan digunakan sebagai aplikasi penguji yang melakukan serangan DoS *SYN Flood* pada *server* Disnaker

4.4. Langkah Kerja

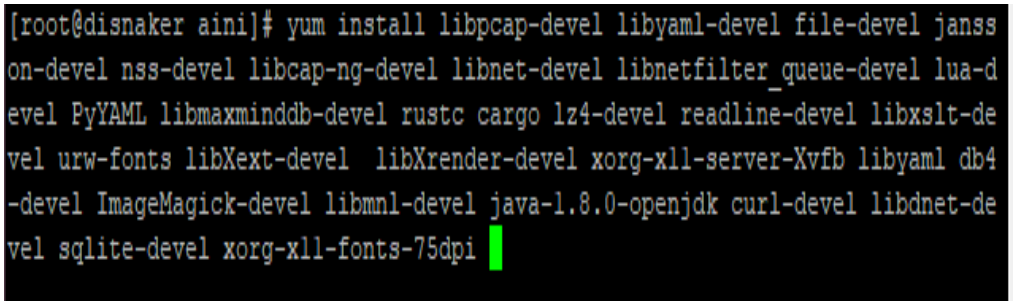
Semua proses instalasi penulis lakukan melalui akses *remote server* Disnaker Bandung menggunakan aplikasi PuTTY. Untuk mengakses *server* penulis menggunakan jenis koneksi SSH dengan alamat domain *server* yakni disnaker.bandung.go.id. Adapun langkah-langkah yang dikerjakan oleh penulis, meliputi :

1. Instalasi Suricata.
2. Konfigurasi Suricata.
3. Pengaturan Telegram dan Pembuatan *Shell Script* Pengiriman *Alert* Telegram.

4.4.1. Instalasi Suricata

Langkah-langkah yang dilakukan dalam instalasi suricata adalah sebagai berikut :

1. Melakukan instalasi paket-paket pendukung dalam membangun IDS menggunakan perintah *yum install* seperti yang ditunjukkan pada gambar 4.2.



Gambar 4. 2 Instalasi Paket Pendukung

Adapun beberapa paket pendukung yang dibutuhkan namun sudah terinstall pada *server* yaitu : *gcc pcre-devel zlib-devel tar make wget git openssl-devel libxml2-devel libX11-devel fontconfig-devel unzip gdbm-devel libffi-devel ethtool ImageMagick curl libcurl libcurl-devel git gcc-c++ apr-devel apr-util-devel patch readline zlib flex bzip2 autoconf automake libtool bison xorg-x11-fonts-Type1 libX11 libXext libXrender libjpeg mariadb-devel postgresql-devel.*

2. Mengunduh aplikasi suricata versi 3.1. Penulis menggunakan *tool wget* untuk mengambil berkas dari *website* www.openinfosecfoundation.org seperti pada gambar 4.3.

```
[root@disnaker download]# wget http://www.openinfosecfoundation.org/download/s
uricata-3.1.tar.gz
--2020-11-30 11:15:06--  http://www.openinfosecfoundation.org/download/suricat
a-3.1.tar.gz
Resolving www.openinfosecfoundation.org (www.openinfosecfoundation.org)... 52.
14.249.179, 2600:1f16:db2:4f00:da9d:37d6:e8b9:9802
Connecting to www.openinfosecfoundation.org (www.openinfosecfoundation.org)|52
.14.249.179|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://openinfosecfoundation.org/download/suricata-3.1.tar.gz [foll
owing]
--2020-11-30 11:15:07--  https://openinfosecfoundation.org/download/suricata-3
.1.tar.gz
Resolving openinfosecfoundation.org (openinfosecfoundation.org)... 52.14.249.1
79, 2600:1f16:db2:4f00:da9d:37d6:e8b9:9802
Connecting to openinfosecfoundation.org (openinfosecfoundation.org)|52.14.249.
179|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3327181 (3.2M) [application/x-gzip]
Saving to: 'suricata-3.1.tar.gz'

100%[==================================>] 3,327,181    917KB/s   in 3.5s

2020-11-30 11:15:11 (917 KB/s) - 'suricata-3.1.tar.gz' saved [3327181/3327181]
```

Gambar 4. 3 Download Suricata 3.1

3. Mengekstrak file hasil *download* yang berekstensi .tar.gz dengan perintah *tar –xvzf* seperti pada gambar 4.4.

```
[root@disnaker download]# tar -xzvf suricata-3.1.tar.gz
```

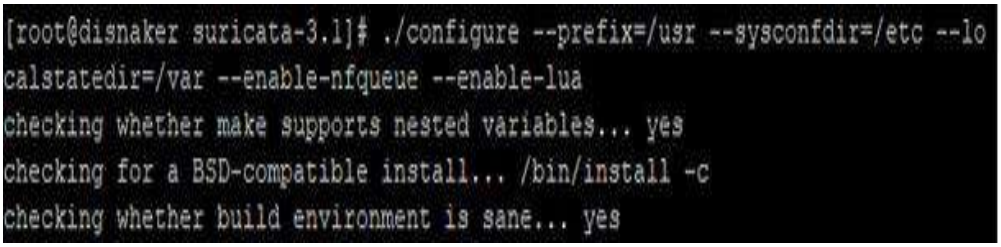
Gambar 4. 4 Ekstrak File *Download* Suricata

Perintah *tar* dengan opsi *–xvzf* itu berarti :

a. x : melakukan proses ekstrak arsip.
b. v : memperlihatkan file yang diarsipkan dalam proses pengarsipan.
c. z : menyaring arsip melalui gzip.
d. f : hasil ekstrak asip yang akan dibuat menjadi sebuah file dalam folder.

4. Mengarah ke folder hasil ekstrak suricara lalu melakukan proses instalasi suricata dengan urutan perintah sebagai berikut :
   1) *./configure --prefix=/usr --sysconfdir=/etc --localstatedir=/var --enable-nfqueue --enable-lua*

      Penjelasan : Perintah *./configure* bertugas untuk mempersiapkan *environment* untuk membangun sistem sebelum proses instalasi. Perintah ini juga bertanggung jawab untuk mengecek kembali apakah paket-paket pendukung dan *library* sudah terinstall dengan baik. Ditunjukkan pada gambar 4.5.



Gambar 4. 5 *./configure* Suricata

   2) *make*

      Penjelasan : Perintah *make* bertugas untuk membangun sistem aplikasi. Ditunjukkan pada gambar 4.6.

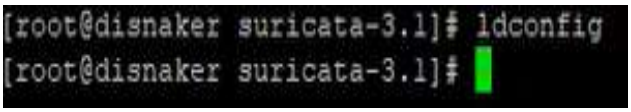
Gambar 4. 6 *make* Suricata

3) *make install*

Penjelasan : Perintah *make install* bertugas untuk melakukan instalasi yang meliputi menyalin program dan *library*-nya ke lokasi yang benar pada *server*. Ditunjukkan pada gambar 4.7.

Gambar 4. 7 *make install* Suricata

4) *Ldconfig* : Perintah ini akan membuat *link* dan *cache* yang diperlukan. Seperti yang ditunjukkan pada gambar 4.8.



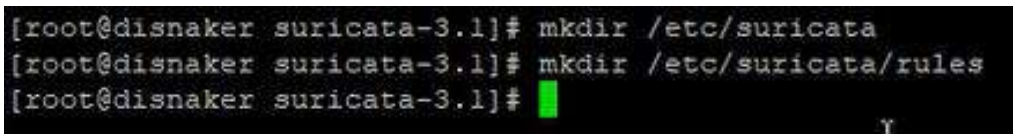
Gambar 4. 8 *ldconfig* Suricata

4.4.2. Konfigurasi Suricata

Selanjutnya melakukan langkah konfigurasi pada suricata sebagai berikut :

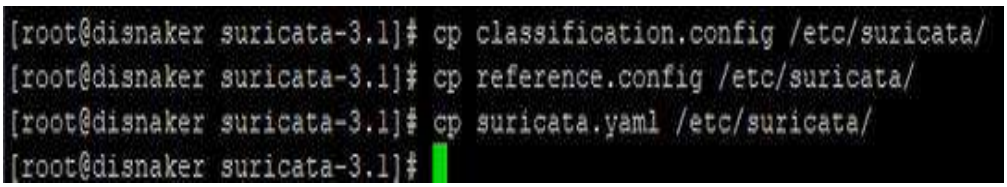
1. Membuat beberapa direktori yang dibutuhkan yakni direktori */var/log/suricata* untuk menyimpan log suricata, */etc/suricata* untuk menyimpan file konfigurasi suricata, dan */etc/suricata/rules* untuk menyimpan *rules* atau aturan dalam mendeteksi serangan. Dapat dilihat pada gambar 4.9 dan gambar 4.10.

Gambar 4. 9 Pembuatan File /var/log/suricata



Gambar 4. 10 Pembuatan File /etc/suricata dan /etc/suricata/rules

2. Menyalin beberapa file konfigurasi yang ada di dalam direktori **/suricata-3.1* yaitu file *classification.config*, *reference.config*, dan *suricata.yaml* ke direktori */etc/suricata* yang telah dibuat seperti yang diperlihatkan pada gambar 4.11.

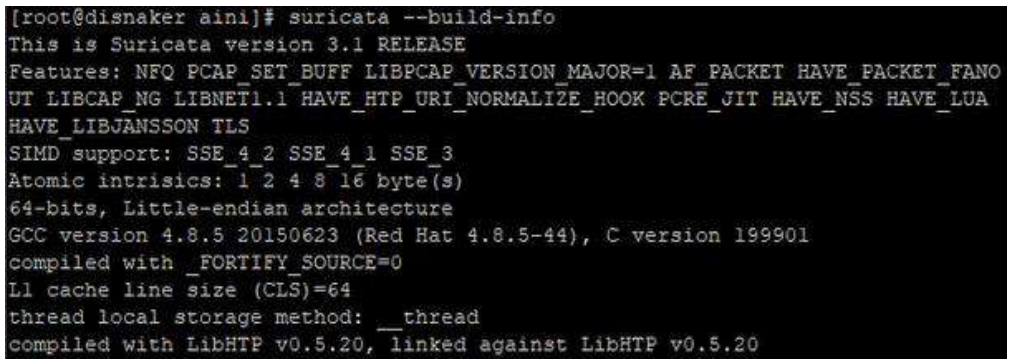


Gambar 4. 11 Menyalin File Konfigurasi ke /etc/suricata

3. Melakukan konfigurasi pada file */etc/suricata/suricata.yaml*, yaitu mengganti nilai dari *HOME_NET* menjadi ip privat dari *server* Disnaker Bandung dengan prefix /32 agar suricata hanya melindungi ip *server* ini saja. Hal ini bisa dilihat pada gambar dan gambar 4.12.

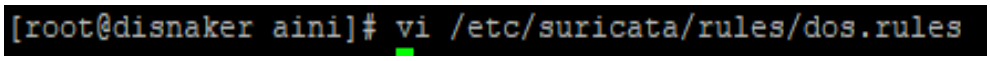
Gambar 4. 12 Konfigurasi HOME_NET

4. Untuk memastikan bahwa suricata sudah terinstal penulis menggunakan perintah *suricata --build-info* untuk melihat versi suricata yang terinstal. Ditunjukkan pada gambar 4.13.

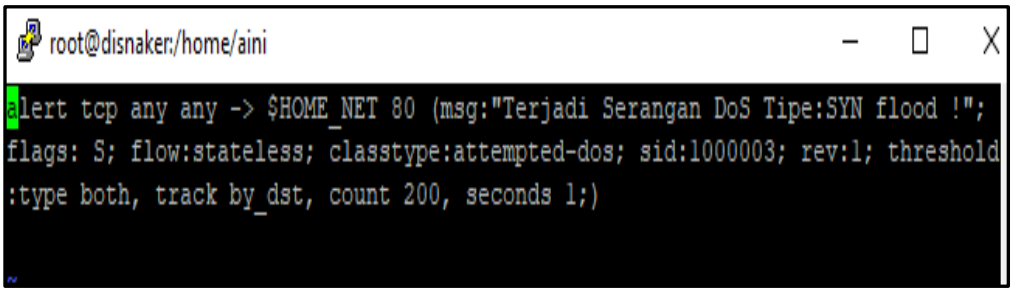


Gambar 4. 13 Pengecekkan Versi Suricata

5. Selanjutnya menambahkan *rule* dengan *signature* yang dibuat untuk mendeteksi serangan DoS *syn flood* pada file *dos.rules* dengan perintah "vi /etc/suricata/rules/dos.rules"yang disimpan di direktori */etc/suricata/rules* seperti pada gambar 4.14 dan gambar 4.15 berikut.



Gambar 4. 14 Pembuatan *Rule File* Untuk DoS



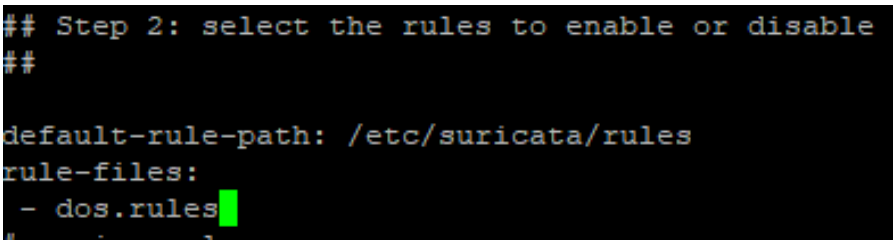
Gambar 4. 15 Isi Dari *Rule File* DoS

Penjelasan *signature rule* dos pada gambar 4.17 :

1) **alert tcp any any -> $HOME_NET 80** : memberikan *alert* atau notifikasi ketika ada paket dengan protokol tcp dari *source* mana saja dan port mana saja menuju $HOME_NET atau *server* itu sendiri melalui port 80.
2) **msg:"Terjadi Serangan DoS! Tipe:SYN Flood"** : pesan informasi *rule* yang disampaikan oleh suricata.

3) **flags: S** : *flag* pada paket tcp adalah SYN.
4) **flow: stateless** : *flow* atau aliran paket tcp tidak tergantung pada *state* apapun.
5) **classtype:attempted-dos** : ini adalah opsi yang menunjukkan *classtype* dari serangan yang akan menentukan tingkat kerentanan serangannya. Tingkat kerentanan pada *classtype* ini telah ditentukan pada file konfigurasi *classification.config*.
6) **sid:1000003** : *signature id* merupakan id dari *rule* ini. Penulis mengatur pada angka 1000003 karena 1000000 – 1999999 adalah alokasi untuk *custom rule* pada suricata.
7) **rev:1** : sid selalu disertai dengan rev yakni sebagai versi dari *rule signature*. Jika *signature rule* diubah maka jumlah rev akan bertambah.
8) **threshold: type both, track by_dst, count 200, seconds 1** : *threshold* artinya ambang batas yang digunakan *rule* sebagai acuan frekuensi peringatan aturan. *type both* merupakan salah satu tipe dari *threshold* yang akan memberikan satu peringatan per interval waktu. *track by_dst* artinya *rule* melacak berdasarkan tujuan dari paket tcp. *count 200* dan *seconds 1* artinya melacak adanya 200 serangan atau paket yang masuk dalam 1 detik.

6. Pada file *suricata.yaml* penulis menambahkan file *rule dos.rules* dan di baris *rule-files:* seperti yang ditunjukkan pada gambar 4.16.



Gambar 4. 16 Menambah Rule File Pada File suricata.yaml

Pada gambar terdapat baris *default-rule-path* yang mengarah ke direktori penyimpanan *rule* yakni */etc/suricata/rules*. Pada baris *rule-files* penulis menambakan *rule file dos.rules.*

7. Selanjutnya menjalankan suricata dengan perintah *suricata –D –c /etc/suricata/suricata.yaml –i enp1s0f0.* Dapat dilihat bahwa tidak ada *output error* seperti pada gambar 4.17 maka suricata sudah berhasil dijalankan.

```
[root@disnaker aini]# suricata -D -c /etc/suricata/suricata.yaml -i enp1s0f0
17/12/2020 -- 16:00:38 - <Notice> - This is Suricata version 3.1 RELEASE
```

Gambar 4. 17 Menjalankan Suricata

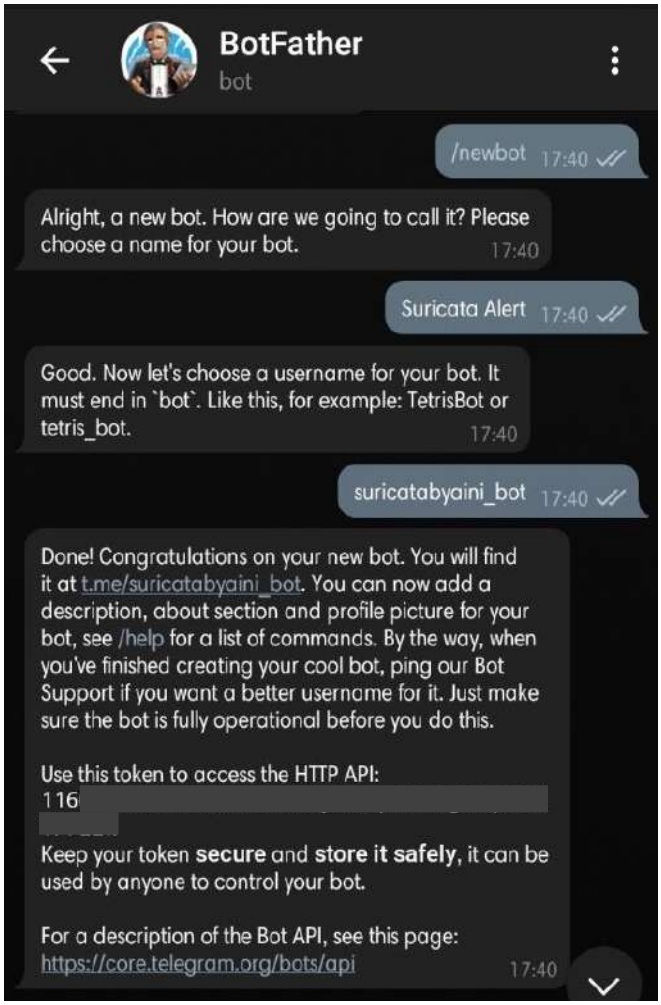
Maksud dari perintah pada gambar yang digunakan untuk menjalankan suricata adalah :

1) Opsi –D : berarti menjalankan suricata sebagai *daemon* yang berjalan di latar belakang.
2) Opsi –c : mengacu pada file konfigurasi suricata.yaml.
3) Opsi –i : berarti *interface* dimana suricata berjalan, yakni enp1s0f0.

4.4.3. Pengaturan Telegram dan Pembuatan Shell Script Pengiriman Notifikasi Suricata

Penulis telah menginstalkan aplikasi telegram dan telah memiliki akun telegram. Langkah-langkah yang dilakukan dalam konfigurasi telegram sebagai media notifikasi suricata adalah sebagai berikut :

1. Langkah pertama seperti pada gambar 4.18, pada aplikasi telegram penulis membuat bot untuk notifikasi suricata menggunakan *BotFather* yakni fitur bot telegram untuk membuat dan mengatur bot yang dibuat oleh pengguna telegram. Untuk membuat bot baru pada botfather penulis menggunakan perintah */newbot*.



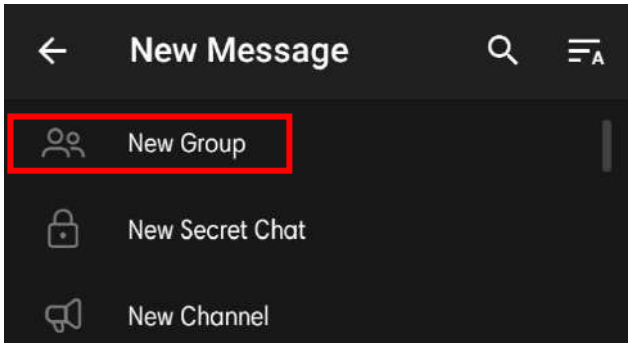
Gambar 4. 18 Pembuatan Bot Telegram

Penulis memberikan nama bot yakni Suricata Alert dan *username* untuk bot baru suricatabyaini_bot seperti pada gambar 4.18. Setelah itu telegram akan memberikan token untuk mengakses HTTP API.

2. Setelah membuat bot telegram, penulis membuat grup telegram agar notifikasi dari bot suricata dapat diterima oleh pihak – pihak selain dari pembuat bot. Jika tidak dibuat grup, maka yang dapat menerima notifikasi bot suricata hanyalah pembuat bot. Pertama – tama penulis memilih *icon* pena yang ada pada ujung bawa kiri telegram seperti pada gambar 4.19.

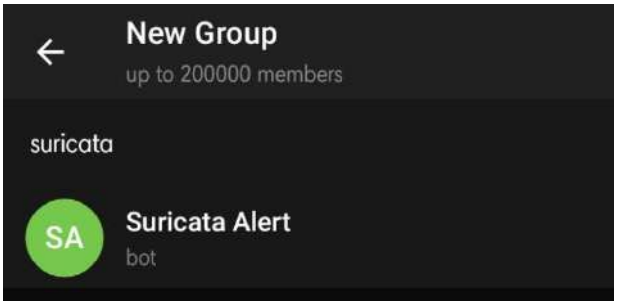
Gambar 4. 19 Membuat Grup Baru

3. Lalu memilih *New Group* untuk membuat grup baru seperti pada gambar 4.20.



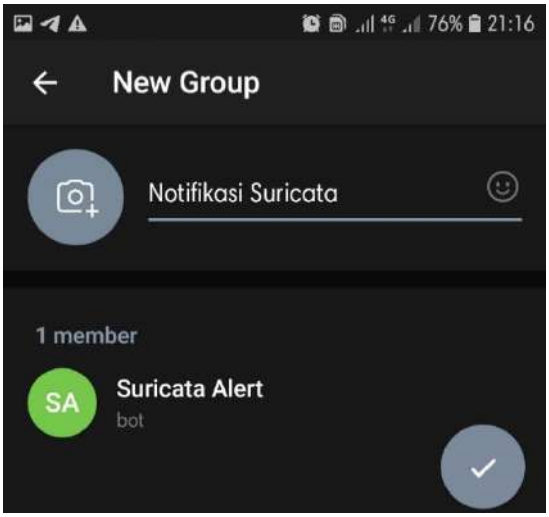
Gambar 4. 20 Memilih Grup Baru

4. Menambahkan anggota grup. Penulis menambahkan bot suricata Suricata Alert ke dalam grup, seperti pada gambar 4.21.



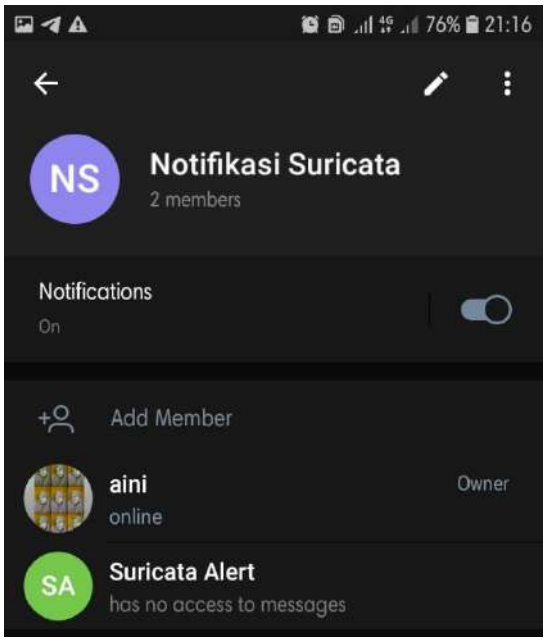
Gambar 4. 21 Menambahkan Bot Suricata ke Grup

5. Selanjutnya mengatur profil grup seperti nama dan foto. Penulis hanya mengatur nama grup dengan nama Notifikasi Suricata.



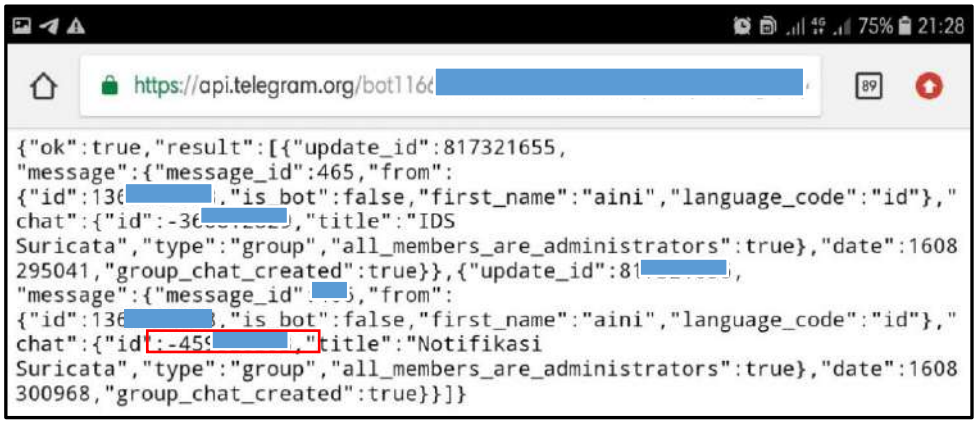
Gambar 4. 22 Memberi Nama Grup

6. Setelah selesai maka tampilan grupnya seperti yang ditunjukkan pada gambar 4.23.



Gambar 4. 23 Tampilan Grup

7. Langkah selanjutnya penulis membutuhkan *chat ID* dari bot suricata yang akan digunakan untuk mengirim notifikasi ke grup Notifikasi Suricata yang tadi telah dibuat. Untuk mendapatkan *chat ID* bot suricata, penulis menggunakan url https://api.telegram.org/bot$token/getupdates pada *web browser* seperti pada gambar 4.24. $token diganti nilainya menjadi token telegram yang sudah didapat pada tahap 1.



Gambar 4. 24 Melihat *Chat Id* Bot Telegram

Maka didapatlah *chat id* telegram. *Chat id* yang digunakan adalah *chat id* yang penulis tandai dengan warna merah. *Chat id* tersebut adalah *chat id* yang digunakan bot untuk menyampaikan pesan ke grup Notifikasi Suricata.

8. Setelah pengaturan pada telegram selesai, dilakukan pembuatan *shell script* untuk *server* agar dapat memberikan notifikasi ke telegram mengenai adanya serangan yang dideteksi suricata. *Shell script* yang dibuat diberi nama *bot_tele.sh*, dapat dilihat pada gambar 4.25. Penulis mengambil data deteksi serangan suricata dari file log suricata *fast.log*. Setelahnya penulis menyimpan *shell script* ini.

root@disnaker:/home/aini/telegram_conf

```
#!/bin/bash

initCount=0
logs=/var/log/suricata/fast.log
msg_caption=/tmp/telegram_msg_caption
chat_id="-459920393"
token="1166541519:AAFXahF0HXFyERLqxK0DSgC2q45RhYPrLEk"

function sendAlert
{
        curl -s -F chat_id=$chat_id -F text="$notifikasi" https://api.telegram.org/bot$token/sendMessage #> /dev/null 2&>1
}

while true
do
    lastCount=$(wc -c $logs | awk '{print $1}')
    if(($(($lastCount)) > $initCount));
       then
        msg=$(tail -n 1 $logs) #GetLastLineLog
        echo $msg > $msg_caption #set Caption / Pesan
        #echo -e "Ada Serangan\n\nServer Time : $(date +"%d %b %Y %T")\n\n"$msg > $msg_caption #set Caption / Pesan
        caption=$(<$msg_caption) #set Caption
        waktu=$(echo $caption | cut -d "." -f1 )
        pesan=$(echo $caption | cut -c 47-1000 |cut -d "!" -f1 )
        ip_asal=$(echo $caption | cut -d "}" -f2 | cut -d ":" -f1 )
        port_asal=$(echo $caption | cut -d "}" -f2 | cut -d ":" -f2 | cut -d "-" -f1)
        ip_tujuan=$(echo $caption | cut -d "}" -f2 | cut -d ":" -f2| cut -d ">" -f2)
        port_tujuan=$(echo $caption | cut -d "}" -f2 | cut -d ":" -f3 | cut -d " " -f1)
        notifikasi=$(echo -e $pesan "\n\nWaktu :" $waktu "\nIP Asal :" $ip_asal "\nMelalui Port" $port_asal "\nIP Tujuan: " $ip_tujuan "\nMenuju Port" $port_tujuan)
        sendAlert #Panggil Fungsi di function
        echo "Alert Terkirim"
        initCount=$lastCount
        rm -f $msg_caption
        sleep 1
    fi

done
```

Gambar 4. 25 *Shell Script* Telegram

Secara garis besar, cara kerja dari skrip pada gambar 4.25 adalah jika pada file log suricata yakni *fast.log* terdeteksi adanya penambahan baris baru yang mana itu menandakan adanya serangan baru yang terjadi, maka akan diambil baris terakhir dari file tersebut dan selanjutnya dikirimkan ke telegram. Penjelasan lebih lanjut dari gambar 4.25 disajikan dalam bentuk tabel berikut :

Tabel 4. 2 Penjelasan Skrip Telegram

Baris ke	**Penjelasan**
1	Menunjukkan bahwa file ini adalah sekumpulan perintah yang akan diterjemahkan oleh penerjemah perintah menggunakan *Bash Shell*.
3	Variabel $initCount ini dijadikan indikasi adanya baris baru yang masuk ke file /var/log/suricata/fast.log. Diatur 0 sebagai permulaan.
4	Variable berisi file /var/log/suricata/fast.log.
5	Variabel berisi file telegram_msg_caption yang bersifat sementara yang disimpan di /tmp. Digunakan sebagai penyimpan sementara data serangan yang diambil dari fast.log.
6	Variabel $chat_id berisi nilai dari *chat id* grup telegram notfikasi suricata.
7	Variabel $token berisi nilai dari token bot suricata pada telegram.
9	Membuat *function* bernama SendAlert. Fungsinya adalah untuk mengirimkan pesan ke telegram melalui API Telegram.
10 dan 12	Tanda kurung kurawal "{}" sebagai penanda blok *function.*
11	Isi dari *function* SendAlert. Variabel $chat_id dan $token digunakan sebagai identifikasi ke pengguna telegram mana notifikasi ini akan dikirimkan. Variabel $notifikasi adalah pesan yang dikirimkan (penjelasan ada di baris 29).
14	*while* adalah perintah perulangan. *true* adalah kondisi benar (dalam hal ini perintah sebelumnya). Jadi artinya "selama kondisi perintah sebelumnya benar".
15	*do* artinya "lakukan".

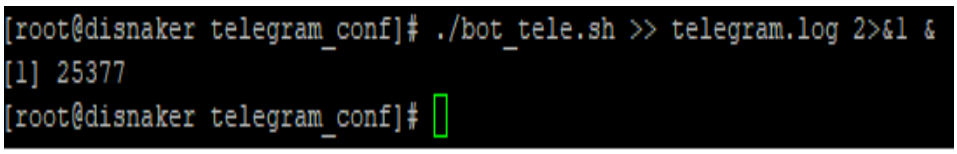
16	Perintah “wc –c /var/log/suricata/fast.log | awk ‘{print $1}’” adalah untuk mengambil banyaknya *bytes* pada file fast.log. Hasil dari perintah ini yaitu hasil banyaknya *bytes* dari file fast.log dimasukkan ke dalam variabel $lastCount.
17	*if* adalah perintah pernyataan kondisional. Arti dari baris ini adalah “jika $lastCount lebih besar nilainya dari $initCount”.
18	*then* adalah perintah yang berarti “maka lakukan (baris selanjutnya)”.
19	Pengambilan baris terakhir dari file fast.log yang hasilnya dimasukkan ke dalam variabel $msg.
20	*echo* menampilkan variabel $msg (baris terakhir fast.log) dan selanjutnya dimasukkan ke $msg_caption (file sementara di /tmp).
22	Hasil dari variabel $msg_caption dimasukkan ke variabel $caption.
23	Mengambil data waktu dari baris data serangan yang telah diambil ke dalam variabel $waktu.
24	Mengambil data pesan dari baris data serangan yang telah diambil ke dalam variabel $pesan.
25	Mengambil data IP asal atau IP penyerang dari baris data serangan yang telah diambil ke dalam variabel $ip_asal.
26	Mengambil data *port* yang digunakan penyerang dari baris data serangan yang telah diambil ke dalam variabel $port_asal.
27	Mengambil data IP tujuan (*server* Disnaker) dari baris data serangan yang telah diambil ke dalam variabel $ip_tujuan.
28	Mengambil data *port* yang dituju penyerang dari baris data serangan yang telah diambil ke dalam variabel $port_tujuan.
29	Menyusun kalimat pesan notifikasi yang akan dikirimkan ke telegram ke dalam variabel $notifikasi.

30	Menjalankan *function* sendAlert.
31	Baris perintah ini penulis gunakan sebagai indicator bahwa blok *if* berjalan dengan baik.
32	Mengganti nilai $initCount menjadi sama dengan nilai $lastCount yang terakhir kali.
33	Menghapus file sementara yaitu /tmp/telegram_msg_caption.
34	Memberikan *delay* satu detik pada eksekusi perintah.
35	*fi* adalah penutup dari blok kondisi *if.*
37	*done* adalah penutup dari blok perulangan *while.*

9. Membuat file telegram.log sebagai file untuk menyimpan *output* ketika file *shell script* dijalankan. Ditunjukkan pada gambar 4.26.

Gambar 4. 26 Pembuatan File telegram.log

10. Selanjutnya adalah menjalankan *shell script* telegram yang telah dibuat dengan perintah *./bot_tele.sh >> telegram.log 2>&1 &* yang berarti jalankan skrip bot_tele.sh dan *output* dari skrip dimasukkan ke file telegram.log. Tanda & di akhir menunjukkan bahwa *shell script* dijalankan di latar belakang. Ditunjukkan pada gambar 4.27.



Gambar 4. 27 Menjalankan Skrip Telegram

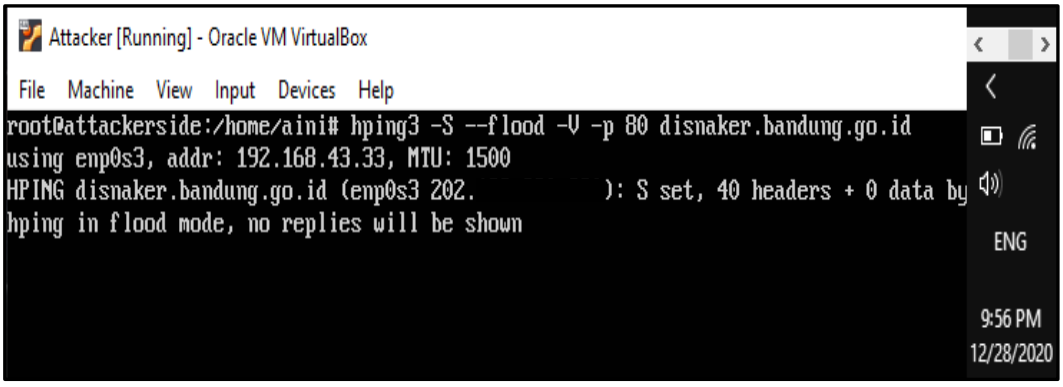
4.5. Pengujian

Dalam melakukan pengujian penulis melakukan tahapan kerja meliputi :

1. Melakukan penyerangan DoS *SYN Flood* melalui *client* Ubuntu.
2. Pengecekkan notifikasi pada telegram .
3. Pengecekkan log suricata .

4.5.1. Melakukan *SYN Flood Attack*

Dalam melakukan serangan *SYN Flood* kepada *server* Disnaker Kota Bandung, penulis menggunakan *client* dengan sistem operasi ubuntu. *Tool* yang penulis gunakan dalam melakukan serangan *SYN Flood* adalah hping3 yang mana *tool* ini sudah terinstal pada *client* ubuntu. Penulis menggunakan perintah seperti pada gambar 4.28 untuk melakukan serangan.



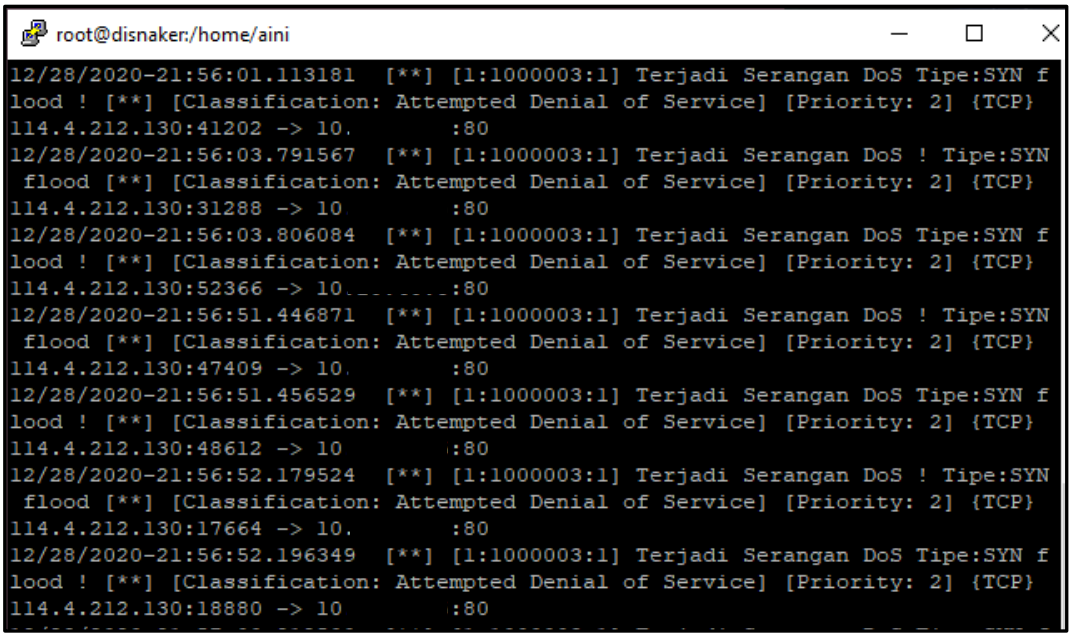
Gambar 4. 28 Melakukan Pengujian DoS *SYN Flood*

Penjelasan dari gambar :

1) hping3 = nama *tool*
2) –S = mengirimkan paket SYN
3) –flood = mengirimkan paket terus-menerus atau *flood mode*
4) –V = menampilkan *output* ke layar
5) –p 80 = tujuan *port* yaitu *port* 80
6) disnaker.bandung.go.id = domain tujuan (*server* disnaker)

4.5.2. Pengecekkan Pada Log Suricata

Pengecekkan pada log suricata dilakukan untuk melihat *detail* data mengenai serangan yang terjadi. Log suricata terdapat pada file /var/log/suricata/fast.log. Penulis menggunakan perintah *vi /var/log/suricata/fast.log* dan hasilnya seperti pada gambar 4.29. Penulis menuju ke baris paling akhir dengan mengetikkan "ctrl + G" untuk melihat notifikasi terakhir yang baru masuk.

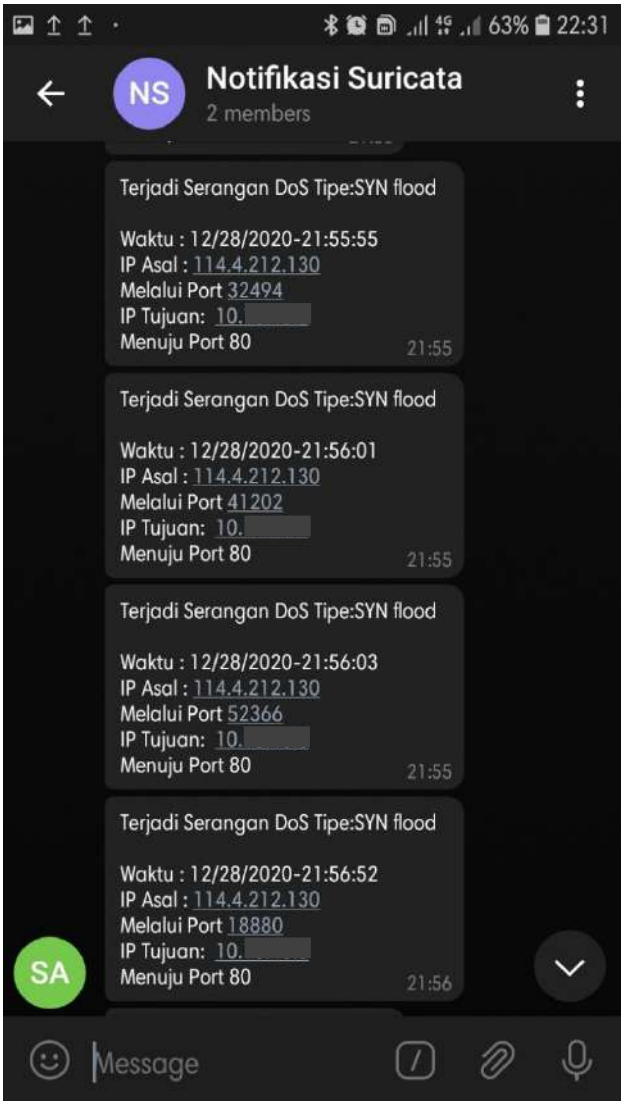


Gambar 4. 29 Laporan Serangan Pada File Log Suricata

Pada gambar 4.29 bisa dilihat terdapat tanggal dan waktu kejadian, yakni tanggal 28 Desember 2020 pukul 12:56, data sid *rule*, yakni 1000003, *classification*, yakni *Attempted Denial of Service*, protokol TCP, alamat ip publik penyerang, *port* yang digunakan penyerang, ip tujuan berupa ip privat dari *server*, dan port tujuan ke *server* yaitu *port* 80. Terdapat juga *Priority: 2* yang menunjukkan tingkat bahaya serangan yang mana ini ditentukan oleh *classification* di file konfigurasai *classification.config.*

4.5.3. Pengecekkan Notifikasi Telegram

Pada saat penulis menjalankan serangan DoS *SYN Flood*, maka saat itu juga notifikasi telegram masuk. Hal tersebut dapat dilihat pada gambar 4.30.



Gambar 4. 30 Notifikasi Suricata ke Telegram

Seperti yang ada pada gambar 4.30 notifikasi yang dikirimkan menunjukkan informasi dari kejadian DoS *SYN Flood*, yaitu waktu terjadinya serangan, ip public dari mesin yang melakukan serangan beserta *port* yang digunakan, juga ip privat dari *server* Disnaker Kota Bandung beserta *port* yang dituju yakni port 80.


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

BAB V
PENUTUP

5.1. Kesimpulan

Berdasarkan pembahasan mengenai Instalasi dan Konfigurasi *Intrusion Detection System* (IDS) Menggunakan Suricata Dengan Notifikasi Telegram Pada *Server* Dinas Tenaga Kerja Bandung (Disnaker), penulis dapat menarik beberapa kesimpulan, yaitu *Intrusion Detection System* (IDS) dengan aplikasi suricata pada *server* Disnaker Kota Bandung di PT Bentang Inspira Teknologi sebagai pendeteksi adanya aktivitas mencurigakan yang berpotensi sebagai serangan telah berhasil terapkan. IDS Suricata dapat mendeteksi adanya ancaman jaringan sesuai dengan *rule* serangan yang diterapkan, yakni DoS *SYN Flood*. Selain itu, sistem IDS yang dibuat telah berhasil mengirimkan notifikasi serangan ke telegram. Administrator dapat mengetahui adanya serangan melalui telegram melalui grup notifikasi yang beranggotakan bot suricata dan administrator. Penulis belum menambahkan administrator dari *server* Disnaker pada grup tersebut.

5.2. Saran

Setelah melaksanakan kegiatan Praktik Kerja Industri dan pengerjaan instalasi konfigurasi IDS (*Intrusion Detection System*) menggunakan suricata pada *server* Disnaker Bandung di PT Bentang Inspira Teknologi, penulis memiliki beberapa saran untuk para pembaca yang berkeinginan untuk melakukan instalasi dan konfigurasi IDS menggunakan suricata, sebagai berikut :

1. Dalam melakukan instalasi *intrusion detection system* (IDS) menggunakan aplikasi suricata yang perlu diperhatikan adalah paket-paket pendukung yang harus terinstal terlebih dahulu sebelum melakukan instalasi suricata.

2. Usahakan untuk membaca *manual book* dari *website* resmi suricata untuk mengetahui cara kerja hingga perintah yang digunakan untuk instalasi dan menjalankan suricata.
3. Jika suricata telah terinstal namun tidak dapat mendeteksi serangan, maka yang perlu diperhatikan dan diperiksa kembali adalah *rule* serangan yang dibuat. Kemungkinan aturan yang dibuat tidak sesuai indikatornya dengan serangan yang terjadi atau mungkin file *rule* yang dibuat belum ditambahkan ke file konfigurasi suricata.yaml.
4. Setelah instalasi IDS berhasil, sebaiknya serangan yang dapat dideteksi ditambahkan lagi agar fungsi IDS menjadi lebih bermanfaat. Penambahan *rule* serangan secara manual dapat dilakukan dengan memanfaatkan *search engine* atau referensi buku yang ada. Atau bisa juga menggunakan *ruleset* yang telah tersedia menggunakan *tool oinkmaster* atau *tool* lainnya dalam penambahannya.


SEKOLAH MENENGAH KEJURUAN NEGERI 1 CIMAHI
KOTA CIMAHI

DAFTAR PUSTAKA

Apa itu Server. (2020). Diakses pada 12 Desember 2020, dari https://www.dicoding.com/blog/apa-itu-*server*/

Ariyus, D. (2005). *COMPUTER SECURITY.* Yogyakarta: C.V ANDI OFFSET.

Cara Menggunakan PuTTY. (2020). Diakses pada 22 Desember 2020, dari https://gegeriyadi.com/cara-menggunakan-putty/

Daryanto. (2010). *Teknik Jaringan Komputer*. Bandung: ALFABETA.

Eril. (2020). *Mengenal Lebih Lengkap Sistem Keamanan Jaringan Komputer*. Diakses pada 5 Desember 2020, dari https://gudangssl.id/sistem-keamanan-jaringan-komputer

Gemilang, R. (2018). *Pengertian IDS Security, Jenis, Dan Cara Kerjanya*. Diakses pada 5 Desember 2020, dari https://www.immersa-lab.com/pengertian-ids-jenis-dan-cara-kerjanya.htm

Ish, J. (2016). *CentOS Installation*. Diakses pada 5 Desember 2020, dari https://redmine.openinfosecfoundation.org/projects/suricata/wiki/CentOS_Installation

ismitggwp. (2018). *TCP 3 Way Handshake*. Diakses pada 10 Desember 2020, dari https://bolosbelajar.wordpress.com/2018/06/11/tcp-3-way-handshake/

Kanigoro, B. (2019). *Port (Jaringan Komputer)*. Diakses pada 22 Desember 2020, dari https://socs.binus.ac.id/2019/11/06/port-jaringan-komputer/

Keyan, K. (2017). *Get Server Notification on Telegram App*. Diakses pada 14 Oktober 2020, dari https://www.assistanz.com/get-server-notification-telegram-app/

Kho, D. (2020). *Pengertian Media Transmisi dan Jenis-jenis Media Transmisi*. Diakses pada 27 November 2020, dari https://teknikelektronika.com/pengertian-media-transmisi-jenis-jenis-media-transmisi/

Klasifikasi Jaringan Komputer. (2016). Diakses pada 27 November 2020, dari http://kuhitung123.blogspot.com/2016/12/klasifikasi-jaringan-komputer-terbagi.html?m=1

Mufrizal, R. (2016). *Belajar Shell Script*. Diakses pada 1 Januari 2021, dari https://rizkimufrizal.github.io/belajar-shell-script/

Mutaqin, A. F. (2016). *Rancang Bangun Sistem Monitoring Keamanan Jaringan Prodi Teknik Informatika Melalui SMS Alert dengan Snort . Naskah Publikasi Ilmiah* tidak diterbitkan. Pontianak: Universitas Tanjungpura .

Nuryanto, A. (2015). *ANALISIS DAN IMPLEMENTASI SURICATA, SNORBY, DAN BARNYARD2. Naskah Publikasi Ilmiah* tidak diterbitkan . Sukoharjo: UNIVERSITAS MUHAMMADIYAH SURAKARTA.

Onnocenter. (2017). *Suricata: Instalasi Snorby & barnyard2*. Diakses pada 7 November 2020, dari http://onnocenter.or.id/wiki/index.php/Suricata:_Instalasi_Snorby_%26_barnyard2

Pengertian CentOS Linux. (2019). Diakses pada 5 Desember 2020 https://klikgss.com/2019/04/02/pengertian-centos-linux/

Pengertian Komunikasi Data. (2020). Diakses pada 17 Desember 2020, dari https://jagad.id/pengertian-komunikasi-data/

Rachman, O., & Yugianto, G. G. (2008). *TCP/IP Dalam Dunia Informatika dan Telekomunikasi.* Bandung: Informatika Bandung.

Sandi, A. (2017). *Mengenal Apa itu Web API*. Diakses pada 1 Januari 2021, dari https://www.codepolitan.com/mengenal-apa-itu-web-api-5a0c2855799c8

Sukmaaji, A., & Rianto. (2008). *Jaringan Komputer Konsep Dasar Pengembangan Jaringan & Keamanan Jaringan.* Jogjakarta: C.V ANDI OFFSET.

Suricata Rule Thresholding. (2016, Agustus 5). Diakses pada 19 Desember 2020, dari https://selcuks61.blogspot.com/2016/08/suricata-rule-thresholding.html?m=1

Syahputra, A. (2002). *Jaringan Berbasis Linux.* Yogyakarta: C.V ANDI OFFSET.

Tentang Suricata. (2017). Diakses pada 6 Desember 2020, dari https://text-id.123dok.com/document/rz3jn1j7y-tentang-suricata-fitur-suricata.html

Tiyas. (2020, Agustus 26). *IP Address* . Diakses pada 5 Desember 2020, dari https://www.yuksinau.id/ip-address/#3_Daya_Tampung

Wikipedia. (2015). *Telegram (aplikasi).* Diakses pada 6 Desember 2020, dari https://id.wikipedia.org/wiki/Telegram_(aplikasi)

Zakaria. (2019, September 2). *Pengertian Sistem Operasi (OS) Beserta Fungsi dan Contoh Sistem Operasi*. Diakses pada 5 Desember 2020, dari https://www.nesabamedia.com/pengertian-dan-fungsi-sistem-operasi/amp/