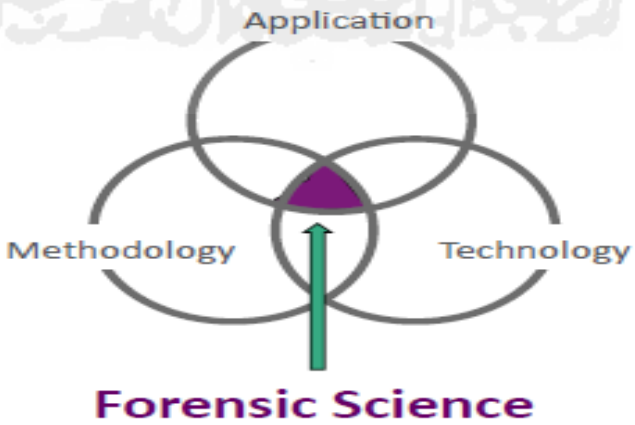
Bab 2 Landasan Teori

2.1 Forensik dan Forensik Jaringan

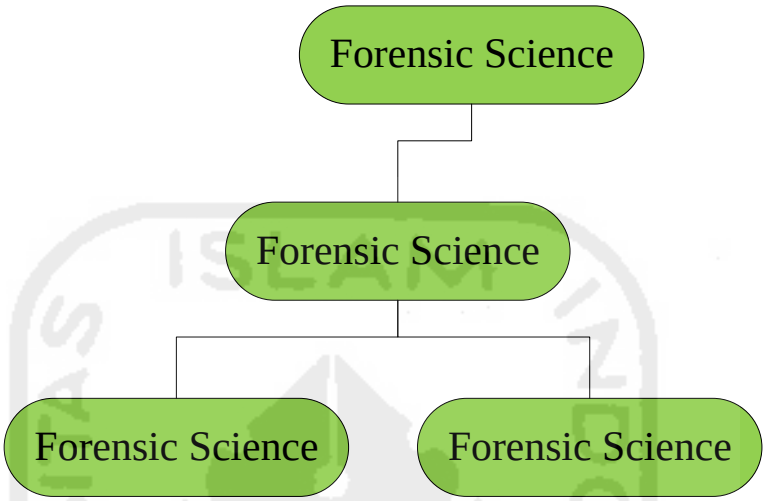
Forensik adalah suatu proses ilmiah atau suatu usaha ilmiah yang didasari ilmu pengetahuan dalam mengumpulkan, mengnalisa dan menghadirkan bukti dalam suatu persidangan di pengadilan untuk membantu pengungkapan suatu kejahatan melalui pengungkapan bukti-bukti yang sah menurut undangn-undang dan peraturan yang berlaku. Digital forensik merupakan aplikasi bidang ilmu pengetahuan dan teknologi komputer untuk kepentingan pembuktian hukum (pro justice), yang dalam hal ini adalah untuk membuktikan kejahatan berteknologi tinggi atau computer crime secara ilmiah (scientific) sehingga bisa mendapatkan bukti-bukti digital yang dapat digunakan untuk menjerat pelaku kejahatan tersebut.

Menurut Franke, 2010 (Franke, n.d.) yang terlihat dari Gambar 2.1 menunjukan bahwa metode forensik dari pendekatan untuk melakukan tugas seperti: (1) Menyelidiki TKP, (2) Mengumpulkan jenis data-data dan menganalisis jenis bukti yang telah ditemukan, (3) Mengidentifikasi, mengelompokkan, mengukur, memisahkan antara pelaku, objek dan proses penanganan, (4) Membangun hubungan, mengasosiasi, dan merekontruksi ulang kejadian, (5) Menggunakan barang bukti di pengadilan.



Gambar 2. 1 Forensic Science

Forensic jaringan merupakan turunan dari forensic digital yang merupakan salah satu ilmu forensic seperti istilah pada bidang kedokteran. *Forensic science* ini mempunyai beberapa cabang turunan ilmu forensik yang dikembangkan menjadi digital forensik, computer forensik dan network forensik. Berikut Gambar 2.2 yang menjelaskan turunan ilmu forensik :



Gambar 2. 2 Turunan Ilmu Forensik

Penelitian T.Charles dan M. Pollock (2015) menyebutkan bahwa digital forensic merupakan metode ilmiah untuk melestarikan, mengoleksi, validasi, identifikasi, analisis, interpretasi, dokumentasi, dan presentasi digital untuk tujuan menfasilitasi atau merekontruksi peristiwa ditemukan tindak kriminal, atau membatu untuk mengantisipasi tindakan yang tidak sah atau terbukti mengganggu proses perencanaan operasi. (Charles & Pollock, 2015)

Penelitian Aleksandar V, Heis S, Mellisa I (2014) mendefinisikan *digital forensic* sebagai penggunaan ilmiah yang diturunkan dengan bukti metode identifikasi, pengumpulan, transportasi, penyimpanan, analisis, diartikan, dipresentasikan dan didistribusikan kembali dari bukti digital yang berasal dari sumbernya. Dengan mendapatkan otorisasi untuk semua kegiatan yang berinteraksi langsung dengan penyelidikan, melestarikan barang bukti, melacak barang bukti atau melakukan rekontruksi peristiwa ditemukannya insiden.(Server & Aktivitas, 2013).

2.2 Model Proses *Forensic*

Model proses *forensic* dijelaskan dalam empat komponenen tahapan dalam menanganinya berupa:

1. Tahap Pengoleksian (*Collection*)

   Yaitu pada tahap ini yang dilakukan meneliti dan mencari bukti-bukti, pengenaln terhadap bukti-bukti penyusupan, dan pengumpulan bukti. Sistem IDS *snort* digunakan untuk mendeteksi serangan. Pada *snort* terdapat aturan yang mengekstrak ciri dari paket yang melewati jaringan, sehingga jika ada paket ang mencurigakan dan sesuai dengan aturan lalu lintas mengirimkan pesan *alert* dan menyimpannya sebagai *log*.

2. Tahap Pemeriksaan (*Examination*)

   Adalah tahap pencarian informasi yang tersembunyi dan mengungkapkan dokumentasi yang relevan. Pemeriksaan dilakukan pada *file log* yang telah diambil menggunakan IDS *snort*. Setelah log tersimpan sebagai *alert*, maka *log* diteliti dan diperiksa.

3. Tahap Analsis (*Analysis*)

   Dari tahap pemeriksaan terlihat hasil untuk nilai pembuktian pada kasus yang ada. Tahap ini digunakan untuk menjawab pertanyaan forensik, yaitu serangan **apa** yang terjadi, IP **siapa** yang melakukan serangan, **kapan** serangan itu terjadi, **dimana** serangan itu terjadi, **bagaimana** serangan tersebut bisa terjadi, dan **mengapa** itu terjadi.

4. Tahap Pelaporan (*Reporting*)

   Penulisan laporan mengenai proses pemeriksaan dan data yang diperoleh dari semua penyelidikan, untuk membuat laporan tentang serangan yang terjadi pada jaringan dari hasil analisis bukti *log* dan setelah itu dilakukan rekontruksi aliran data dari kejadian tersebut dengan tidak merusak *file log* tersebut.

Empat komponen dalam *Digital Forensic* dari bukti digital selanjutnya akan membahas hal yang dangat penting yaitu *Chain of Custody*. *Chain of Custody* merupakan proses untuk merekam kronologi pengamanan, penahanan, pengendalian, dan pemindahan barang bukti fisik atau elektronik. *Chain of Custody* dituliskandalan dalam sebuah dokumen yang berfungsi untuk menjelaskan kronologi penanganan barang bukti tersebut, sehingga diharapkan tidak menimbulkan keraguan pada saat proses pengadilan. Ketika barang bukti akan digunakan dalam proses pengadilan, makan diperlukan penanganan yang sangat hati-hati untuk mencegah terjadinya kontaminasi atau perubahan dari barang bukti tersebut. Ide dibalik *Chain of Custody* ini adalah untuk menegaskan bahwa barang bukti tersebut memang benar-benar terkait dengan tindak kejahatan, bukan semata barang bukti yang ditanamamkan di tempat kejahatan, hanya untuk membuat seseorang tampak bersalah.

Pihak yang berwenang harus selalu memiliki akses terhadap barang bukti, mendokumentasikannya, dan meyerahkannya kepada pihak yang bertanggung jawab terhadap

evidence room (tempat pengamanan di mana barang bukti disimpan). Dokumen *Chain of Custody* tidak memiliki format yang standart atau baku, namun harus berisi informasi mengenai:

a. Barang bukti yang dikumpulkan.

b. Identitas semua penanggung jawab barang bukti.

c. Durasi penyimpanan barang bukti.

d. Pemindahan barang bukti (termasuk di dalamnya adalah tanda tangan pihak yang terlibat dalam proses pemindahan barang bukti.

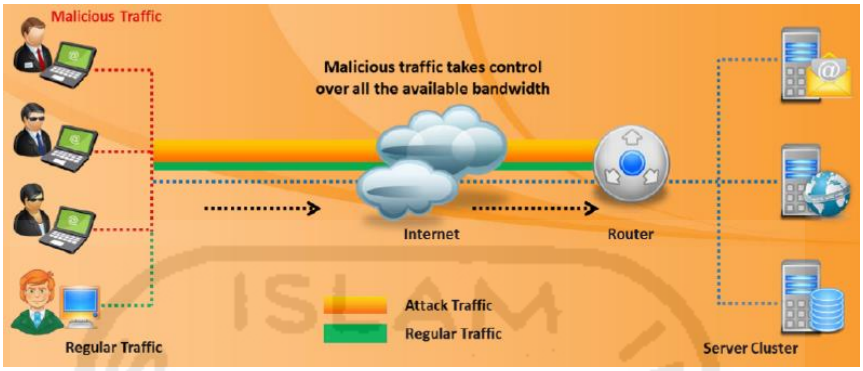
2.3 Komponen Jaringan

Beberapa jenis peralatan memurut Volonino dan Anzaldua (2008) untuk memahami bagaimana system forensic bekerja pada jaringan ang telah dilakukan untuk tingkat besar, adalah:

1. Router: sebuah computer husus yang bertujuan memindahkan data yang melintasi dua jaringan IP address ang berbeda. Router bekerja pada lapisan tiga dalam model OSI.
2. *Switch*: computer jaringan yang menggunakan *Media Access Control* (MAC) identifikasi dari sebuah host pada jaringan untuk memindahkan data dalam jaringan. *Switch* bekerja pada lapisan tiga dalam model OSI yang merupakan penghubung jaringan multiport untuk menembatani segmen jaringan.
3. Hub: merupakan bagian utama dari jaringan yang berfungsi untuk mengirimkan data yang diterima pada semua port. Perangkat ini bekerja pada layer sua karena tidak ada skema pengalamatan pada lapisan kedua. Sekarang hub jarang digunakan karena cenderung meningkatkan volume traffic dan memeperlambat jaringan sedangkan switch jauh lebih efisien dalam memindahkan data.
4. *Network Interface Card* (NIC): sebuah peangkat yang terdapat MAC (*Media Access Control*) yaitu aamat computer yang unik untuk mnegidentifikasi host atau computer. NIC adalah penghubung antara jaringan dan host.
5. Host; setipa perangkat komputasi yang terpasang ke jaringan memiliki alamat IP dan alamat MAC. Computer adalah sebuah host yang memiliki alamat IP dan alamat MAC, juga laptop, PDA, WAP, router, switch, maupun perangkat mobile seperti smartphone, ipod juga telah memiliki alamat IP dan MAC.
6. Media: sebuah bagian dari jaringan yang dapat berbentuk kabel tembaga, kabel serat optic atau gelombang radio. Memungkinkan untuk menghubungkan perangkat ke jaringan dan media yang berbeda juga proptokol yang berbeda untuk membantu menciptakan rentang waktu dan data ang dapat mengaitkan tersangka.

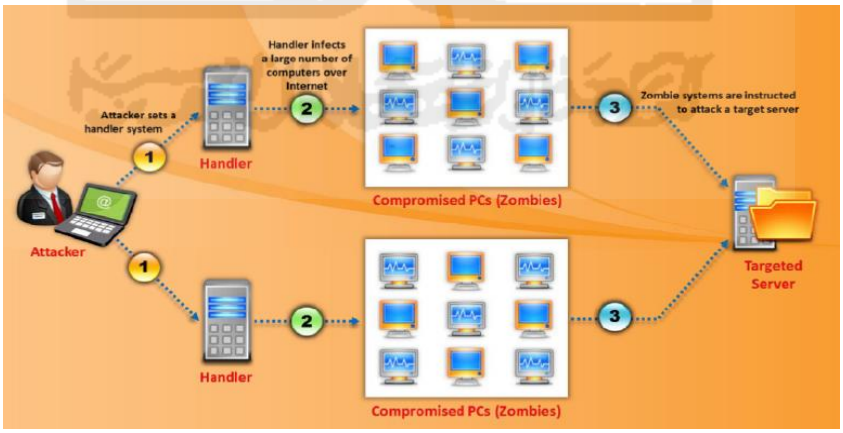
2.4 Serangan *Flooding*

Serangan *Flooding* pada modul *Certified Ethical Hacker* (CEH) terdiri dari DoS (*Denial of Service)* dan Ddos *(Distributed Denial of Service),* DoS merupakan serangan yang ditunjukkan untuk mengacaukan atau menhentikan sebuah layanan. Skema serangan DoS dapat dilihat pada Gambar 2.3.



Gambar 2. 3 Serangan DoS

Sedangkan Ddos *(Distributed Denial of Service)* adalah serangan yang dilakukan dengan banyak computer secara bersama-sama. DoS dan DdoS yang sebenarnya jenis serangan yang sama terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (resource) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut. Berikut skema serangan DDoS dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Serangan DDoS

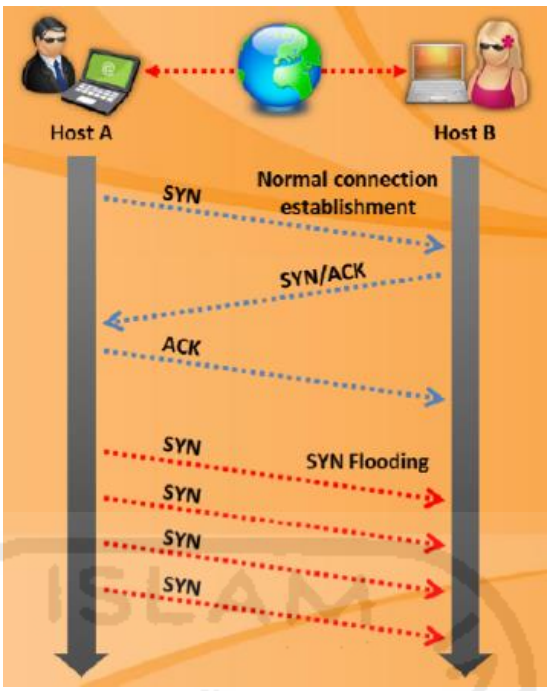
Serangan *DoS (Denial-of-Service attacks)* dan DdoS (*Distributed-Denial-of-Service*) akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut:

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

Bentuk serangan *Denial of Service* awal adalah

1. *SYN Flooding Attack*, yang pertama kali muncul pada tahun 1996 dan mengeksploitasi terhadap kelemahan yang terdapat di dalam protokol *Transmission Control Protocol (TCP)*. Serangan-serangan lainnya akhirnya dikembangkan untuk mengeksploitasi kelemahan yang terdapat di dalam sistem operasi, layanan jaringan atau aplikasi untuk menjadikan sistem, layanan jaringan, atau aplikasi tersebut tidak dapat melayani pengguna, atau bahkan mengalami crash. Beberapa tool yang digunakan untuk melakukan serangan DoS pun banyak dikembangkan setelah itu (bahkan beberapa tool dapat diperoleh secara bebas), termasuk di antaranya *Bonk, LAND, Smurf, Snork, WinNuke,* dan *Teardrop.*

Meskipun demikian, serangan terhadap TCP merupakan serangan DoS yang sering dilakukan. Hal ini disebabkan karena jenis serangan lainnya (seperti halnya memenuhi ruangan hard disk dalam sistem, mengunci salah seorang akun pengguna yang valid, atau memodifikasi tabel routing dalam sebuah router) membutuhkan penetrasi jaringan terlebih dahulu, yang kemungkinan penetrasinya kecil, apalagi jika sistem jaringan tersebut telah diperkuat. Contoh serangan paket SYN pada Gambar 2.5.

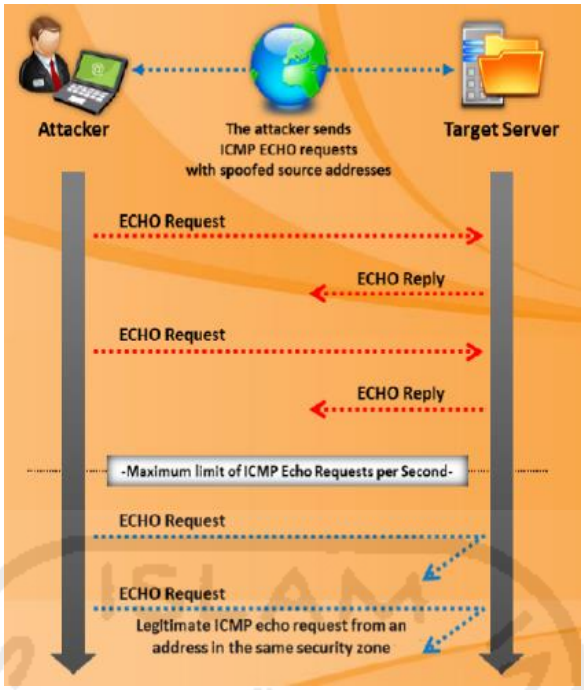


Gambar 2. 5 SYN Flooding

Dalam aturan ini, computer sumber akan mengirimkan paket SYN yang dibalas dengan paket SYN/ACK dan dibahas lagi dengan paket ACK. Sampai tiga kali berhubungan yang menjadi aturan baku, namun hacker menemukan cara untuk mengacaukan system operasi dengan megacaukankan aturan tiga respon tersebut

1. Serangan paket ICMP

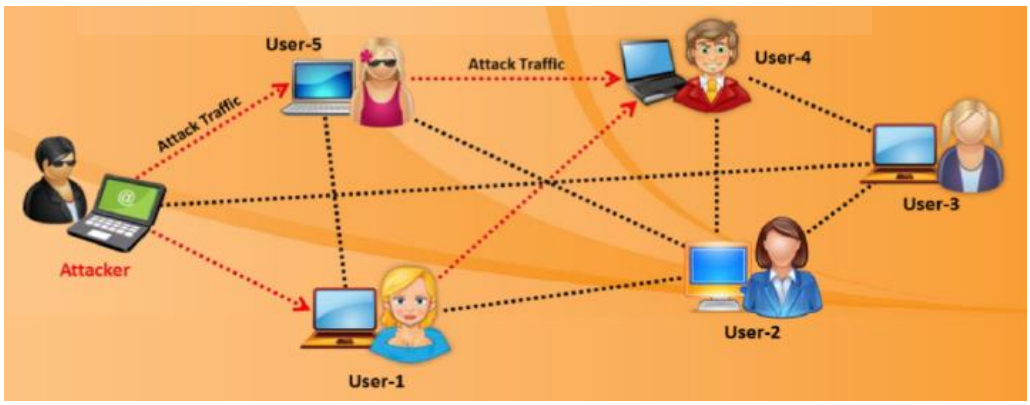
IMCP atau Internet Control Message Protokol adalah protocol sederhana yang umumnya digunakan untuk melacak keberadaan device seperti perintah PING. ICM juga digunakan oleh fungsi tracert yang digunakan untuk melacak jalur yang dilalui oleh paket data dari sumber menuju tujuan. Karena didukung oleh hampir semua operasi yang ada, paket ICMP juga digunakan oleh hacker untuk melakukan penerangan yang sesuai dengan karakteristiknya. Contoh serangan ICMP pada Gambar 2.6.



Gambar 2. 6 Serangan ICMP

2. Serangan *Peer-to-Peer*

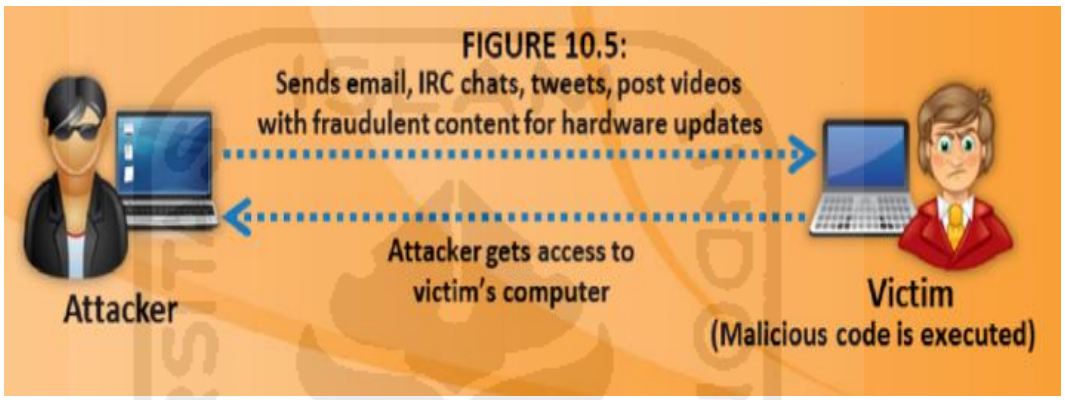
Jaringan *peer-to-peer* sangatlah popular dan digunakan oleh banyak sekali pengguna dan hacker yang kreatif pernah memanfaatkan jaringan ii untuk melakukan serangan DDoS. Serangan ini memungkinkan untuk terjadi karena adanya kelemahan pada jaringan ini pada waktu itu. *Hacker* megeksplorasi kelemahan pada direct connect (DC) yang digunakan oleh jaringan *peer-to-peer* untuk melakukan koneksi langsung. Berkat eksploitasi yang dilakukan,pengguna jaringan ini secra tidak sengaja membuat koneksi ke computer korban secara bersama-sama. Karena jumlah pengguna yang banyak, dengan koneksi yang banyak pula, secara otomatis serangan DDoS akan terjadi. Contoh serangan *peer-to-peer* pada Gambar 2.7.



Gambar 2. 7 Serangan Peer-to-peer

3. Permanen DoS

Serangan permanen DoS tidak membutuhkan banyak computer dan menimbulkan dampak yang sangat besar dan bahkan biasanya akan menyebabkan *downtime* yang cukup lama. Tujuan *hacker* adalah merusak secara permanen alat yang digunakan oleh korban. Untuk melakukan perusakan terhadap alat yang digunakan, salah satu cara yang digunakan oleh *hacker* adalah menipu korban agar agar mengupdate *firmware* alatnya dengan *firmware* palsu yang telah disiapkan oleh *hacker*. Contoh Gambar 2.8 yang menggambarkan serangan permanen DoS.



Gambar 2. 8 Permanen DoS

2.5 Intrusion Detection System (IDS)

Intrusion Detection System (IDS) adalah sebuah aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS dapat melakukan inspeksi terhadap lalu lintas inbound dan outbound dalam sebuah sistem atau jaringan, melakukan analisis dan mencari bukti dari percobaan intrusi (penyusupan). OSI model dan sensor jaringan pasif yang secara khusus diposisikan pada *choke point* pada jaringan metode dari lapisan OSI.

Kebanyakan produk IDS merupakan sistem yang bersifat pasif, mengingat tugasnya hanyalah mendeteksi intrusi yang terjadi dan memberikan peringatan kepada administrator jaringan bahwa mungkin ada serangan atau gangguan terhadap jaringan. Akhir-akhir ini, beberapa vendor juga mengembangkan IDS yang bersifat aktif yang dapat melakukan beberapa tugas untuk melindungi host atau jaringan dari serangan ketika terdeteksi, seperti halnya menutup beberapa port atau memblokir beberapa alamat IP Address.

Ada dua jenis IDS, yakni:

1. *Network-based Intrusion Detection System* (NIDS): Semua lalu lintas yang mengalir ke sebuah jaringan akan dianalisis untuk mencari apakah ada percobaan serangan atau

penyusupan ke dalam sistem jaringan. NIDS umumnya terletak di dalam segmen jaringan penting di mana server berada atau terdapat pada "pintu masuk" jaringan. Kelemahan NIDS adalah bahwa NIDS agak rumit diimplementasikan dalam sebuah jaringan yang menggunakan switch Ethernet, meskipsun beberapa vendor switch Ethernet sekarang telah menerapkan fungsi IDS di dalam switch buatannya untuk memonitor port atau koneksi.

2. *Host-based Intrusion Detection System* (HIDS): Aktivitas sebuah host jaringan individual akan dIpantau apakah terjadi sebuah percobaan serangan atau penyusupan ke dalamnya atau tidak. HIDS seringnya diletakkan pada server-server kritis di jaringan, seperti halnya firewall, web server, atau server yang terkoneksi ke Internet.

Ada beberapa cara bagaimana IDS bekerja. Cara yang paling populer adalah dengan menggunakan pendeteksian berbasis *signature* (seperti halnya yang dilakukan oleh beberapa antivirus), yang melibatkan pencocokan lalu lintas jaringan dengan basis data yang berisi cara-cara serangan dan penyusupan yang sering dilakukan oleh penyerang. Sama seperti halnya antivirus, jenis ini membutuhkan pembaruan terhadap basis data signature IDS yang bersangkutan.

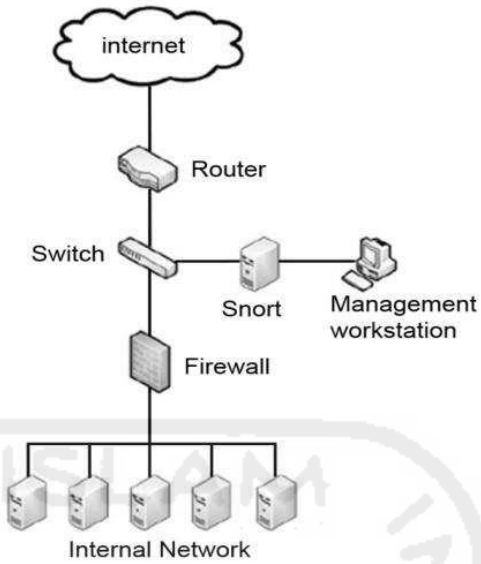
Metode selanjutnya adalah dengan mendeteksi adanya anomali, yang disebut sebagai Anomaly-based IDS. Jenis ini melibatkan pola lalu lintas yang mungkin merupakan sebuah serangan yang sedang dilakukan oleh penyerang. Umumnya, dilakukan dengan menggunakan teknik statistik untuk membandingkan lalu lintas yang sedang dIpantau dengan lalu lintas normal yang biasa terjadi. Metode ini menawarkan kelebihan dibandingkan signature-based IDS, yakni ia dapat mendeteksi bentuk serangan yang baru dan belum terdapat di dalam basis data signature IDS. Kelemahannya, adalah jenis ini sering mengeluarkan pesan false positive. Sehingga tugas administrator menjadi lebih rumit, dengan harus memilah-milah mana yang merupakan serangan yang sebenarnya dari banyaknya laporan false positive yang muncul.

Teknik lainnya yang digunakan adalah dengan memantau berkas-berkas sistem operasi, yakni dengan cara melihat apakah ada percobaan untuk mengubah beberapa berkas sistem operasi, utamanya berkas log. Teknik ini seringnya diimplementasikan di dalam HIDS, selain tentunya melakukan pemindaian terhadap log sistem untuk memantau apakah terjadi kejadian yang tidak biasa.

2.6 Snort

Snort salah satu produk open source yang secara defacto menjadi standar IDS (*Intrusion Detection System*) di industri. Snort merupakan salah satu software untuk mendeteksi instrupsi pada sistem, mampu menganalisis secara real-time traffic dan logging IP Address, mampu menganalisis port dan mendeteksi segala macam intrusion atau serangan dari luar seperti *buffter*

overflows, stealth scan, CGI attacks, SMP probes, OS fingerprinting. Skema topologi Snort dapat di lihat pada Gambar 2.9.



Gambar 2. 9 Simple Snort Network Topology

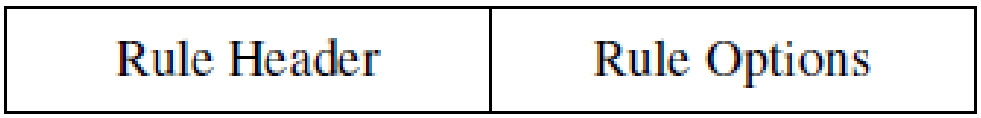
2.6.1 Komponen Snort

Menurut penelitian komponen IDS Snort dibagi menjadi beberapa bagian, berupa:

1. *Packet Dekoder:* packet dekoder mengambil paket yang sesuai dengan paket ang di tangkap dalam bentuk struktur data dan melakukan identifikasi protocol, decode IP dan kemudian TCP atau UDP yang dapat disesuaikan sesuai yang dibutuhkan.
2. *Prepocessor:* komponen atau plug-in yang dapat digunakan dengan snort untuk mengatur atau memodifikasi paket data dalam melakukan beberapa operasi untuk mengetahui apakah paket sedang digunakan oleh penyusup atau tidak.
3. *Rules Files:* merupakan suatu file teks yang berisidaftar aturan yang sintakna sudah diketahui.
4. *Detection Engine:* merupakan detection plug-in untuk mengenali paket serangan atau bukan.
5. *Output Plugins:* merupakan suatu modul yang mengatur format dari keluaran untuk alert dan file logs yang bisa diakses dan menyimpan output yang dihasilkan oleh logging dari system alert dari log.

2.6.2 Aturan Snort

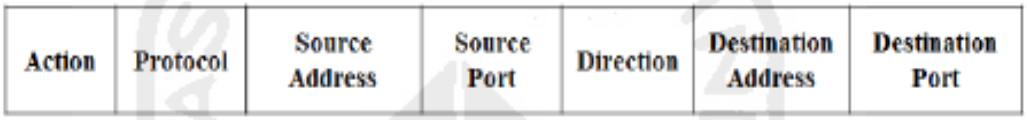
Aturan snort ada 2 struktur yang terlihat dalam Gambar 2.10 dalam penelitian (Khamphakdee, 2014) yaitu:



Gambar 2. 10 Struktur Rule

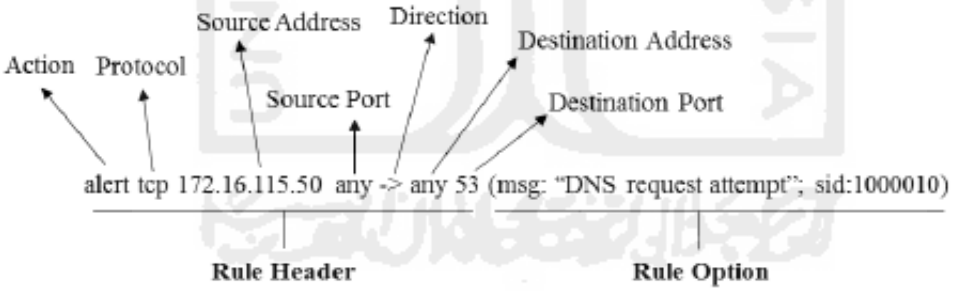
Keterangan:

a. *Rule header* : merupakan bagian rule dimana aksi-aksi *rule* diidentifikasikan *alert, log, active, dynamic,* dan lain-lainya yang termsuk diantara aksi-aksi penting yang digunakan dalam dalam desain rule Snort pada Gambar 2.11.



Gambar 2. 11 Snort IDS Rule Header Structure

b. *Rule options*: merupakan bagian rule dimana pesan-pesan peringtana (alert) di identifikasi. Contohnya:



Gambar 2. 12 Snort IDS Example

Gambar 2.12 menunjukkan contoh aturan IDS snort yang mendengus aturan yang menunjukkan kewaspadaan. Jika protokol tcp, dengan sumber alamat IP nomor 172.16.115.50 terdeteksi dari port dikirim ke alamat IP tujuan dan destination port number 53 (DNS). Selain itu juga menampilkan pesan permintaan DNS dengan jumlah aturan 1000010.

2.6.3 Kepala Aturan (rule header)

Kepala aturan mengandung informasi yang menetapkan siapa, dimana, paket beserta apa yang harus dilakukan pada saat sebuah kejadian cocok dengan atribut pada aturan. Ada Lima tindakan yang tersedia pada snort berupa:

a. *Alert*: menbuat sebuah pesan peringatan (alert) menggunakan metode alert terpilih kemudian me-log paket yang dimaksut.

b. *Log*: intruksi untuk me-log paket

c. *Pass*: mengabaikan paket.

d. *Activate*: memberi pesan peringatan lalu mengaktifkan aturan dinamis lainnya.

2.6.4 *Class Type*

Digunakan untuk mengkategorikan sebuah rule sebagai pendeteksi sebuah serangan yang menjadi bagiandari jenis serangan yang lebih umum. Snort menediakan pembagian *rules* serangan yang digunakan oleh satu set *rule* yang yang diberikan. Penetapan klasifikasi ini terbagi dalam 3 prioritas tinggi, sedang, rendah. Lihat tambel 2.1 Prioritas klasifikasi serangan.

Tabel 2. 1 Prioritas klasifikasi serangan

Classtype	Deskripsi	Tingkat
Attempted-admin	Mencoba mendapatkan hak administrasi	Tinggi
Attempted-user	Mencoba mendapatkan hak user	Tinggi
Kickass-porn	Pornografi	Tinggi
Policy-violation	Serangan privasi perusahaan	Tinggi
Shellcode-detect	Kode executable terdeteksi	Tinggi
Successful-admin	Sukses untuk mendapatkan hak administrator	Tinggi
Successful-user	Sukses mendapatkan hak user	Tinggi
Trojan-activity	Trojan jaringan terdeteksi	Tinggi
Unsuccessful-user	Tidak sukses mendapatkan hak user	Tinggi
web-aplication-attack	Serangan aplikasi web	Tinggi
attempte-dos	Percobaan Denial 0f Service (DoS)	Sedang
attempted-recon	Percobaan penyadapan informasi	Sedang
bad-unknown	Trafik yang jelek atau rusak	Sedang
default-login-attemp	Mencoba login dengan default username dan password	Sedang

Tabel 2. 2 Con't Prioritas klasifikasi serangan

Classtype	**Deskripsi**	**Tingkat**
denial-of-service	Deteksi atas sebush serangan Denial of Service (DoS)	Sedang
misc-attack	Serangan lain-lain	Sedang
non-standard-protocol	Deteksi atas protocol atau event non-standar	Sedang
rpc-portmap-decode	Decode pada RPC query	Sedang
succesfull-dos	Serangan Denial of Service (DoS)	Sedang
succesfull-recon-langescale	Sabotase informasi besar-besaran	Sedang
succesfulli-recon-limited	Penadapan informasi	Sedang
suspicious--filename-detect	Nama file yang mencurigakan terdeteksi	Sedang
suspicious-login	Sebuah usaha login menggunakan username yang mencurigakan	Sedang
system-call-detect	Sebuah system call terdeteksi	Sedang
unusual-client-port-connection	Klien yang menggunakan port yan tidak biasa	Sedang
web-aplication-activity	Akses ke sebuah aplikasi web yang rentan	Sedang
icmp-event	Event umum ICMP	Rendah
misc-activity	Aktivitas mencurigakan	Rendah
Network scan	Terdeteksi scan jaringan	Rendah
Not-suspicious	Trafik yang mencurigakan	Rendah
Protokol-command-decode	Decode pada perintah protocol terdeteksi	Rendah
String-detect	Sebuah string mencurigakan terdeteksi	Rendah
Unknown	Trafik yang tidak diketahui	Rendah
Tcp-connection	Sebuah koneksi TCP terdeteksi	Rendah

2.6.5 Jenis Opsi *Rule*

Opsi rule yang terdapat didalam rule option dapat dipisahkan dengan karakter semi colom (;) yang menggambarkan kemudahan penggunaan kekuatan dan fleksibilitas yang berupa kategori:

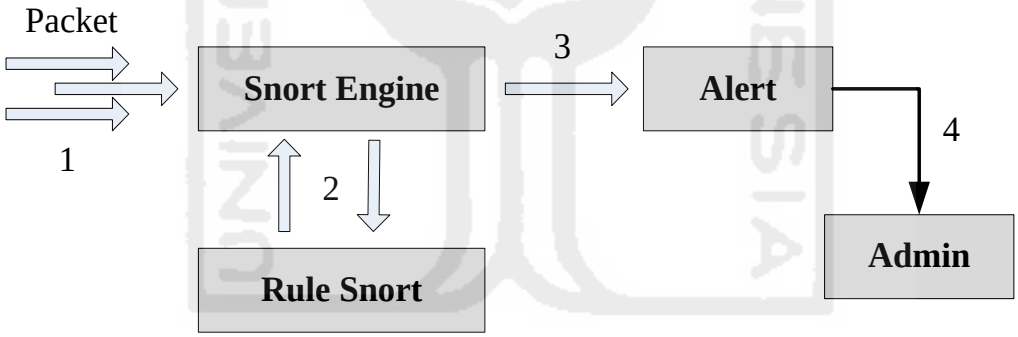
1. General: opsi yang menyediakan informasi tentang rule.
2. Payload: opsi ini mencari data dalam *payload* paket.
3. Non payload: opsi ini mencari data non-payload
4. Post detection: opsi ini merupakan pemicu aturan tertentu yang berjalan setelah aturan tersebut diaktifkan.

2.6.6 Membaca Aturan Snort

Aturan snort yaitu kumpulan aturan perilaku snort berupa tahapan aturan seperti mengidentifikasi karakteristik dari trafik yang dicurigai, menulis rule berdasarkan karakteristik, mengimplementasi aturan, mengecek trafik yang dicurigai, mengubah aturan sesuai dengan pengetesan, mengetes dan mengecek hasilnya.

Sedangkan dilihat dari kemampuan mendeteksi penyusupan pada jaringan, IDS dibagi menjadi: *knowledge-based* atau *misuse detectiondan behavior based* atau *anomaly based*. *Knowledge-based* dapat mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkan dengan database aturan IDS Snort (berisi catatan serangan). Sedangkan *behavior based* (anomaly) dapat mendeteksi adanya penusupan dengan mengamati adanya kejanggalan-kejanggalan pada sistem, atau adanya penyimpangan-penyimpangan dari kondisi normal.

Sedangkan dilihat dari kemampuan mendeteksi penyusupan pada jaringan, IDS dibagi menjadi dua yakni: *host based* dan *network based*. *Host based* mampu mendeteksi hanya pada host tempat implementasi IDS, sedangakn *network based* IDS mampu mendeteksi seluruh host yang berada satu jaringan dengan host implementasi IDS tersebut. Berikut Gambar 2.13 yang menjelaskan *block* diagram sistem pencegahan penyusupan.



Gambar 2. 13 Block Diagram

Keterangan:

1. Packet
2. Snort engine berfungsi untuk membaca paket data dan membandingkannya dengan aturan basis data, jika paket data diibaratkan sebagai penyusup/serangan, maka snort engine akan menghasilkan alert (berbentuk file log).
3. Rule snort menyediakan aturan berupa jenis pola serangan. Rule ini berupa file text yang disusun dengan aturan tertentu. Setelah paket data melintasi jaringan maka akan di deteksi oleh aturan snort.
4. Alert bagian ini merupakan pencatatan serangan pada sebuah file log. Apabila paket yang melintasi jaringan sesuai dengan pola ang ada maka akan muncul tanda peringatan.