BAB 2
DASAR TEORI

2.1 KAJIAN PUSTAKA

Beberapa penelitian telah dilakukan untuk mengatasi masalah keamanan jaringan dan deteksi serangan. Penelitian pertama [8] fokus pada klasifikasi data *log* IDS dengan menggunakan *data mining* dan algoritma C4.5. Penelitian ini berhasil menghasilkan 66 aturan untuk menentukan serangan dengan tingkat akurasi mencapai 96,371%. Penelitian kedua [9] memperkenalkan Snort sebagai solusi keamanan jaringan dan menganalisis pengaruh serangan DDoS terhadap kinerja CPU. Hasil pengujian menunjukkan bahwa serangan DDoS dapat memaksimalkan kinerja CPU hingga 40% dan setelah serangan berakhir, kinerja CPU kembali normal sebesar 1%.

Penelitian selanjutnya [10] melibatkan penggunaan *firewall* dan IDS Suricata pada *tools OPNSense* untuk melindungi *server* dari serangan. Metode penyerangan yang diuji meliputi *port scanning* dan serangan DoS. Dalam pengujian, *port* 80 berhasil terdeteksi terbuka pada *server* Ubuntu, dan serangan terhadap situs *web* tertentu berhasil diblokir. Penelitian lainnya [11] menggabungkan mekanisme IDS dan IPS dengan menggunakan Snort sebagai perangkat deteksi serangan dan *Telegram*-API untuk notifikasi dan pengendalian serangan. Metode serangan yang diuji meliputi DoS SYN-*Flood* dan Nmap *scanning*. Hasil pengujian menunjukkan selisih waktu antara *log* Snort dan notifikasi *Telegram*.

Penelitian berikutnya [12] membandingkan performa IDS Snort dan Suricata dalam mendeteksi serangan TCP SYN *Flood*. Hasil pengujian menunjukkan bahwa Snort memiliki akurasi deteksi, kecepatan deteksi, dan efektivitas deteksi yang lebih tinggi daripada Suricata. Namun, Suricata lebih hemat dalam penggunaan sumber daya sistem. Penelitian terakhir [13] juga menganalisis kinerja Snort dan Suricata sebagai IDS dalam mendeteksi serangan SYN *Flood* pada *Web Server Apache*. Hasil pengujian menunjukkan bahwa Snort lebih banyak mendeteksi serangan, lebih baik dalam penggunaan CPU, dan memiliki fitur

informasi serangan yang lebih lengkap. Namun, Suricata lebih unggul dalam efektivitas deteksi serangan dari paket yang tidak terdeteksi dan penggunaan RAM.

Kemudian jika mengacu pada perbedaan penelitian ini dengan penelitian sebelumnya, bahwa pada penelitian sebelumnya (penelitian 8), hanya menggunakan mekanisme IDS. Namun, penelitian ini menggunakan IDS dan *Firewall* secara bersamaan sebagai upaya pencegahan serangan *cyber*. Selain itu, penelitian 8 melakukan klasifikasi *log* Snort dengan menggunakan *platform rapidminer*, sedangkan penelitian ini menggunakan *platform orange*. Selanjutnya, dalam penelitian 9 digunakan metode *port scanning* dan ICMP *Flood*, sedangkan dalam penelitian ini digunakan metode SYN TCP *Flood* dan UDP *Flood*. Pada penelitian 10, digunakan *tools* serangan dan *Firewall* seperti LOIC dan *OPNSense*, namun dalam penelitian ini digunakan Hping3 dan *pfSense*. Penelitian 11 mensimulasikan serangan DoS, sedangkan penelitian ini mensimulasikan serangan DDoS. Selain itu, dalam penelitian 11, pemblokiran IP dilakukan secara manual melalui *firewall rules bot telegram*, sedangkan dalam penelitian ini pemblokiran IP dilakukan secara otomatis melalui *pfSense*. Penelitian 12 berfokus pada analisis kinerja IDS Snort dan Suricata, sedangkan penelitian ini terdapat proses deteksi dan pencegahan menggunakan IDS Snort dan *Firewall pfSense*, yang kemudian diikuti dengan analisis kinerja IDS Snort. Pada penelitian 13, menggunakan parameter jumlah serangan terdeteksi oleh sistem IDS, efektivitas deteksi, serta penggunaan sumber daya RAM dan CPU. Sedangkan dalam penelitian ini, menggunakan parameter akurasi deteksi, efektivitas deteksi, kecepatan deteksi, dan penggunaan sumber daya CPU.

Pada Tabel 2.1 merupakan penjelesan singkat mengenai penelitian sebelumnya yang relevan.

Tabel 2.1 Kajian Penelitian Sebelumnya

Author	Objective	Network Configuration	Result
Thifal Baraas, Akbar Juliansyah, Ahmad Ashril Rizal pada Tahun 2019	Klasifikasi Data Log Intrusion Detection Sistem (Ids) Dengan Decision Tree C4.5	IDS	Data log IDS dapat diklasifikasikan dengan menggunakan algoritma C4.5 2. Algoritma C4.5 menghasilkan 66 rule untuk menentukan serangan dengan tingkat akurasi 96.371%
Winrou Wesley Purba dan Rissal Efendi pada Tahun 2020	Perancangan dan analisis pada sistem keamanan jaringan komputer menggunakan Snort	IDS pada server	Jika Snort mendeteksi serangan kinerja CPU 40%, sedangkan jika serangan telah diberhentikan kinerja CPU 1%.
Reza Rizky Adha, Mochammad Fahru Rizal dan Setia Juli Irzal Ismail pada Tahun 2021	Membangun sistem keamanan jaringan berbasis firewall dan IDS menggunakan Tools OPNSense	IDS, Web filtering dan Firewall pada server	OPNSense tidak mampu menerjemahkan semua IP dari youtube karena terlalu banyak dan membutuhkan waktu yang sangat lama. Dengan membuat rules firewall pada interface LAN berhasil melakukan block pada client yang mencoba mengakses ketiga situs web dan menampilkan pemberitahuan dalam bentuk log bahwa web tersebut di block.
Daniel Desma Mahendra dan Fransiska Sisilia Mukti pada Tahun 2022	Menerapkan sistem deteksi dan kontrol serangan DoS pada Server berbasis Snort dan Telegram-API	IDS dan IPS, serta Message Notification	Notif telegram pengujian nmap scanning telegram dengan delay rata – rata sebanyak 8,6 detik. Pada metode syn-flood log snort dengan delay rata – rata sebanyak 5,8 detik.
Emir Risyad, Mahendra Data, Eko Sakti Pramukantoro pada Tahun 2018	Membandingkan Performa IDS Snort dan Suricata dalam mendeteksi serangan TCP SYN Flood	IDS	Performa IDS Snort dan IDS Suricata dalam mendeteksi serangan TCP SYN Flood mendapatkan hasil positif. IDS Snort memiliki reliabilitas lebih baik daripada IDS Suricata dalam mengukur akurasi deteksi, kecepatan deteksi dan efektivitas deteksi.

Author	Objective	Network Configuration	Result
			Dalam hal penggunaan sumber daya, IDS Suricata lebih unggul daripada IDS Snort.
Lukman, Melati Suci pada Tahun 2020	Membandingkan kinerja Snort dan Suricata sebagai IDS Dalam Mendeteksi Serangan Syn Flood pada Web Server Apache	IDS	Hasil pengujian dapat disimpulkan bahwa IDS Snort lebih unggul dalam pendeteksian serangan, penggunaan resource CPU, dan fitur informasi data serangan, sedangkan Suricata lebih unggul dalam efektivitas serangan dari data uncaptured paket dan penggunaan RAM.
Jamaludin Nur Indramukti	Menerapkan sistem IDS dan Firewall pada server dengan notifikasi Email	IDS, Firewall, Email Notification	Penelitian ini mengahasilkan nilai kinerja dari sistem deteksi Snort melalui akurasi deteksi, kecepatan deteksi, efektivitas deteksi, dan penggunaan sumber daya.

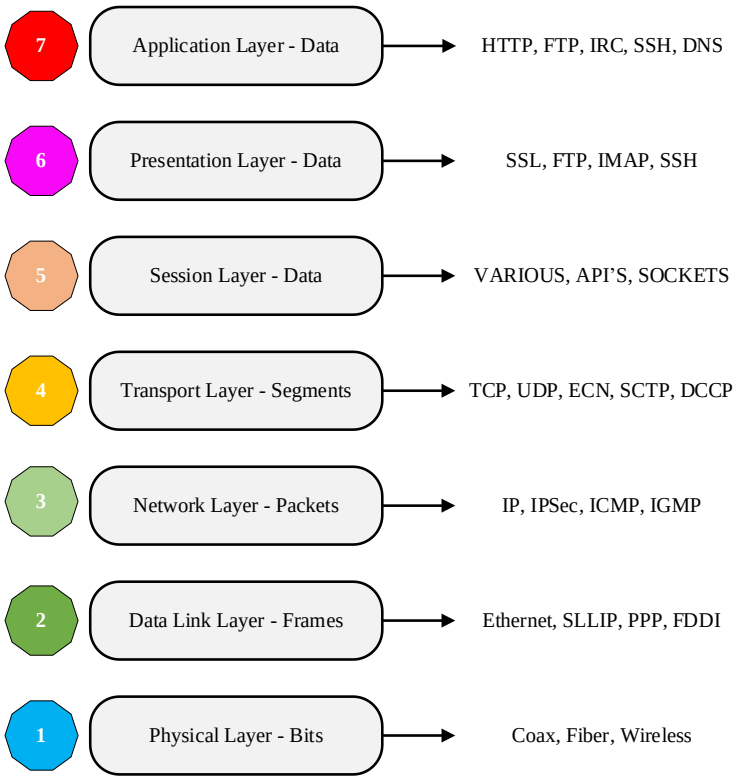
2.2 TEORI PENDUKUNG

2.2.1 Konsep Jaringan Komputer

Jaringan komputer adalah jaringan telekomunikasi yang memungkinkan komputer saling berkomunikasi dengan pertukaran data. Tujuan dari jaringan komputer adalah untuk memungkinkan setiap bagian dari jaringan untuk memberikan dan menerima layanan demi mencapai tujuan tertentu. Pihak yang meminta atau menerima layanan disebut klien (*client*), sedangkan pihak yang memberikan atau mengirim layanan disebut peladen (*server*). Sistem *client-server* merupakan desain yang digunakan pada hampir semua aplikasi jaringan komputer. Dalam desain ini, dua komputer yang masing-masing dilengkapi dengan kartu jaringan dihubungkan melalui kabel atau nirkabel sebagai media transmisi data. Dengan menggunakan perangkat lunak sistem operasi jaringan, maka jaringan komputer sederhana dapat terbentuk. Namun, jika ingin membuat jaringan komputer yang lebih luas, diperlukan peralatan tambahan seperti *hub*, *bridge*, *switch*, *router*, dan *gateway* sebagai alat interkoneksi[14].

2.2.2 Konsep *Server*

Jaringan komputer memiliki peran yang sangat penting dalam teknologi informasi. Oleh karena itu, diperlukan sistem keamanan *server* pada jaringan komputer untuk melindungi *server* dari ancaman yang disengaja atau tidak disengaja. Jaringan komputer terdiri dari *server* dan klien. *Server* biasanya dikendalikan oleh seorang *administrator* yang melakukan pemantauan langsung atau secara *remote*. Seorang *administrator* yang mengendalikan *server* harus memiliki keahlian dan kompetensi yang memadai di bidangnya. Dalam dunia *cyber* saat ini, banyak pelaku kejahatan yang sengaja masuk ke dalam sistem dan merusak konfigurasi *server*, baik untuk kesenangan pribadi maupun keuntungan finansial. Beberapa jenis serangan mendasar pada jaringan yang sering terjadi adalah IP *Spoofing/Session Hijacking*, *Packet Sniffer*, DDoS, *Man-in-the-Middle*, dan *Back Door*. Oleh karena itu, keamanan pada *server* sangatlah penting. Keamanan *server* melibatkan berbagai lapisan penunjang sistem keamanan atau *layer* OSI[15][16]. Seperti pada Gambar 2.1 menunjukkan OSI *Layer*.



Gambar 2.1 OSI *Layer*

2.2.3 Jenis-jenis Jaringan Komputer

Beberapa jenis jaringan komputer yang umum ditemui dapat dikelompokkan berdasarkan cakupan area, yaitu LAN, MAN, WAN[17].

1) LAN (*Local Area Network*)

   LAN adalah sebuah konsep yang menghubungkan perangkat jaringan dalam jarak yang relatif pendek, seperti gedung sekolah, kantor, atau rumah. Jaringan LAN biasanya menggunakan konektivitas tertentu, seperti *Ethernet* dan *Token Ring*. Selain itu, ada juga jaringan LAN yang menggunakan teknologi nirkabel atau *wireless* seperti Wi-Fi, yang dikenal sebagai *Wireless Local Area Network* (WLAN).

2) MAN (*Metropolitan Area Network*)

   MAN merupakan suatu konsep yang menghubungkan perangkat jaringan antar kota. Jika jarak antar perangkat jaringan sudah terlalu jauh untuk dihubungkan dengan LAN, maka jaringan MAN akan digunakan. Jaringan MAN menggunakan perangkat khusus dan memerlukan operator telekomunikasi sebagai penghubung antar jaringan komputer.

3) WAN (*Wide Area Network*)

   WAN adalah konsep jaringan komputer yang menghubungkan perangkat dalam wilayah yang sangat luas, bahkan melintasi negara atau benua. Dalam membangun jaringan WAN, digunakan peralatan yang lebih canggih dan kompleks dibandingkan dengan jaringan MAN atau LAN. Contohnya adalah *fiber optic*, yang biasanya ditanam di dalam tanah atau di bawah laut. Jaringan WAN memerlukan infrastruktur dan peralatan khusus serta operator telekomunikasi yang mampu menghubungkan jaringan dari berbagai lokasi dengan teknologi yang handal[18][19].

2.2.4 Jenis-jenis Topologi Jaringan

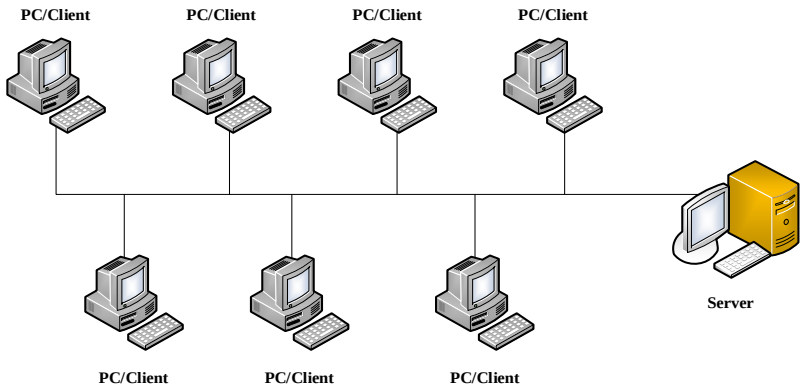
Topologi jaringan mengacu pada cara bagaimana komputer dan perangkat jaringan lainnya diatur secara fisik dan logis pada sebuah jaringan. Topologi jaringan merupakan istilah yang digunakan secara luas oleh para ahli jaringan dan merujuk pada desain dasar sebuah jaringan[20]. Pemilihan satu topologi akan dapat mempengaruhi:

1. Jenis peralatan yang diperlukan jaringan
2. Kemampuan dari peralatan
3. Pertumbuhan jaringan
4. Cara jaringan diatur

Topologi jaringan dapat dikategorikan dalam beberapa jenis atau bentuk, yaitu Topologi *Bus*, Topologi *Star*, Topologi *Ring*, Topologi *Tree*, dan Topologi *Mesh*.

1) Topologi *Bus*

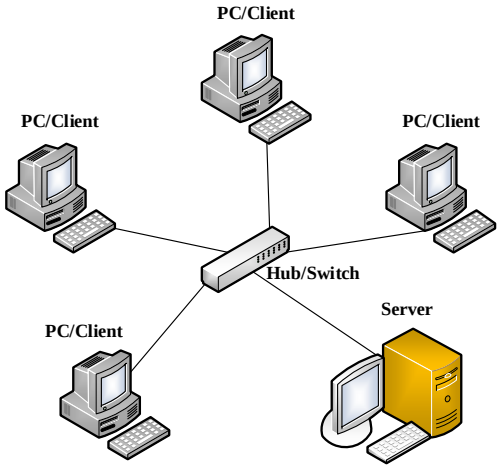
Topologi BUS merupakan jenis topologi jaringan yang menggunakan satu jalur kabel tunggal sebagai media transmisi dan menjadi pusat bagi seluruh perangkat yang terhubung. Setiap perangkat dihubungkan ke kabel utama melalui konektor BNC, dan diakhiri dengan terminator. Seperti pada Gambar 2.2 menunjukkan bentuk dari topologi *Bus*.



Gambar 2.2 Topologi *Bus*

2) Topologi *Star*

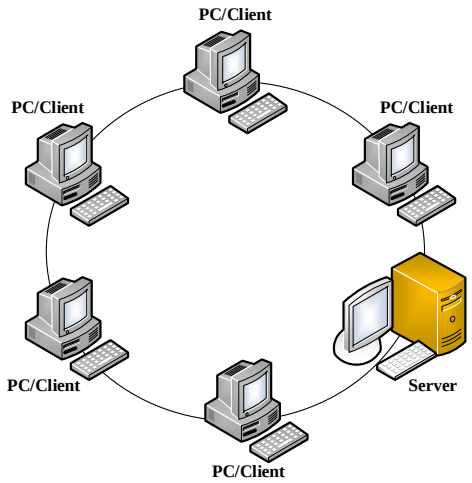
Topologi STAR adalah jenis topologi yang memiliki sebuah pusat penghubung (*hub* atau *switch*) yang menghubungkan setiap komputer ke satu titik sentral. Pusat penghubung tersebut berfungsi untuk menghubungkan setiap komputer yang terhubung ke jaringan dan juga menghubungkan komputer ke *server file*. Seperti pada Gambar 2.3 menunjukkan bentuk dari topologi *Star*.



Gambar 2.3 Topologi *Star*

3) Topologi *Ring*

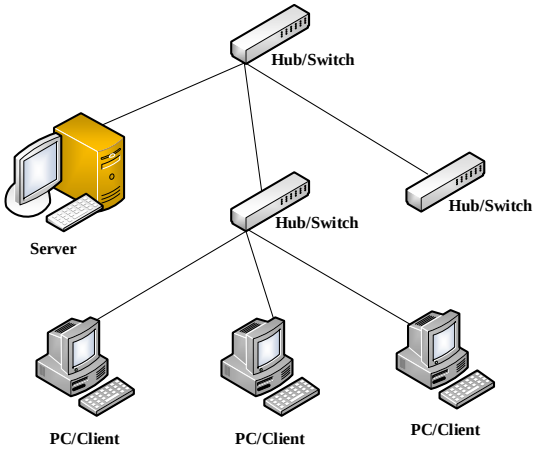
Topologi RING adalah jenis topologi jaringan yang membentuk suatu cincin atau *loop*. Setiap komputer dihubungkan secara seri dengan komputer lainnya hingga membentuk sebuah cincin, di mana komputer pertama dan terakhir saling terhubung. Data akan bergerak melalui kabel dari satu komputer ke komputer berikutnya hingga kembali ke komputer awal, membentuk suatu jalur tertutup atau *loop*. Karena sifatnya yang tertutup, jika terjadi kerusakan pada salah satu kabel maka seluruh jaringan akan terganggu. Seperti pada Gambar 2.4 menunjukkan bentuk dari topologi *Ring*.



Gambar 2.4 Topologi *Ring*

4) Topologi *Tree*

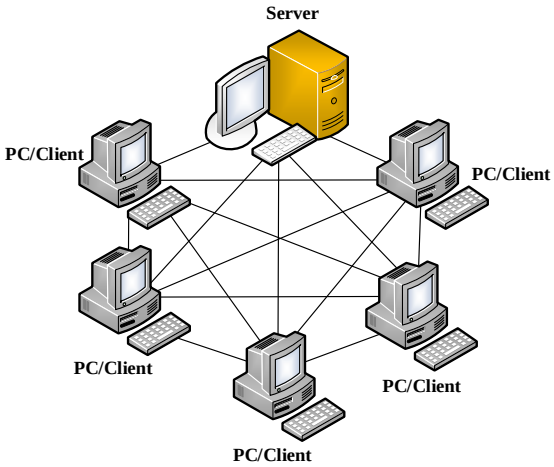
Topologi TREE merupakan sebuah jenis topologi jaringan yang berstruktur bertingkat atau hierarkis. Pada topologi ini, setiap cabang dari jaringan terhubung ke sebuah *Hub* atau *Switch* yang menjadi pusat dari setiap cabang tersebut. Dengan demikian, setiap komputer pada jaringan dapat terhubung dengan *file server* melalui *hub* atau *switch* tersebut. Topologi *Tree* ini memungkinkan untuk menghubungkan beberapa topologi *Star* menjadi satu kesatuan yang lebih besar. Seperti pada Gambar 2.5 menunjukkan bentuk dari topologi *Tree*.



Gambar 2.5 Topologi *Tree*

5) Topologi *Mesh*

Topologi MESH adalah suatu bentuk jaringan di mana setiap *node* atau komputer terhubung langsung dengan *node* atau komputer lainnya. Dalam topologi ini, setiap *node* memiliki jalur yang unik dan tidak teratur untuk berkomunikasi dengan *node* lainnya. Karena setiap *node* terhubung langsung dengan *node* tujuan, maka *transfer* data dapat dilakukan dengan cepat tanpa harus melalui *node* lainnya. Topologi *Mesh* biasanya digunakan pada jaringan yang memerlukan keamanan tinggi dan keandalan yang tinggi, karena jika satu *node* mengalami masalah, komunikasi masih dapat berlangsung melalui jalur lainnya. Seperti pada Gambar 2.6 yang menunjukkan bentuk dari topologi *Mesh*.

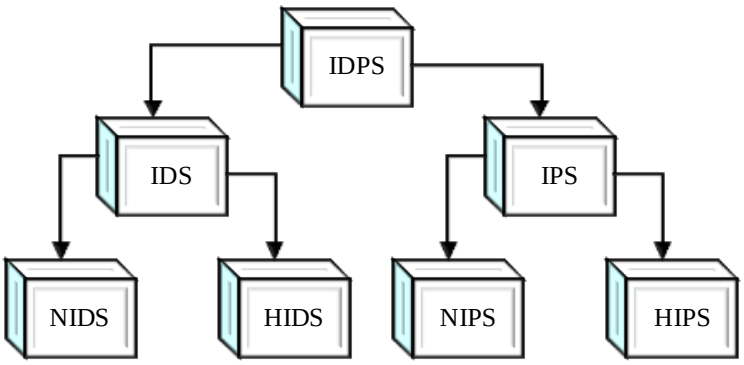


Gambar 2.6 Topologi *Mesh*[21]

2.2.5 Konsep IDS dan IPS

Sistem deteksi intrusi (IDS) merupakan sebuah perangkat lunak aplikasi yang memantau aktivitas jaringan atau sistem untuk mendeteksi upaya intrusi dan menghasilkan peringatan serta mencatat informasi mengenai upaya intrusi tersebut untuk *administrator*. Sementara itu, sistem pencegahan intrusi (IPS) merupakan jenis IDS dengan kemampuan tambahan untuk memblokir intrusi dengan cara menjatuhkan paket berbahaya, memblokir alamat IP jahat, atau mengatur ulang koneksi. IPS mampu bertindak lebih cepat terhadap ancaman daripada IDS yang hanya mengambil salinan lalu lintas jaringan dan membuat peringatan untuk dievaluasi oleh *administrator*. Kedua jenis sistem ini secara bersama-sama disebut sebagai sistem deteksi dan pencegahan intrusi (IDPS)[22].

Pada Gambar 2.7 menunjukkan pembagian fungsi IDPS. Terdapat dua tipe IDS, yaitu NIDS dan HIDS. IPS juga memiliki dua tipe, yaitu NIPS dan HIPS.

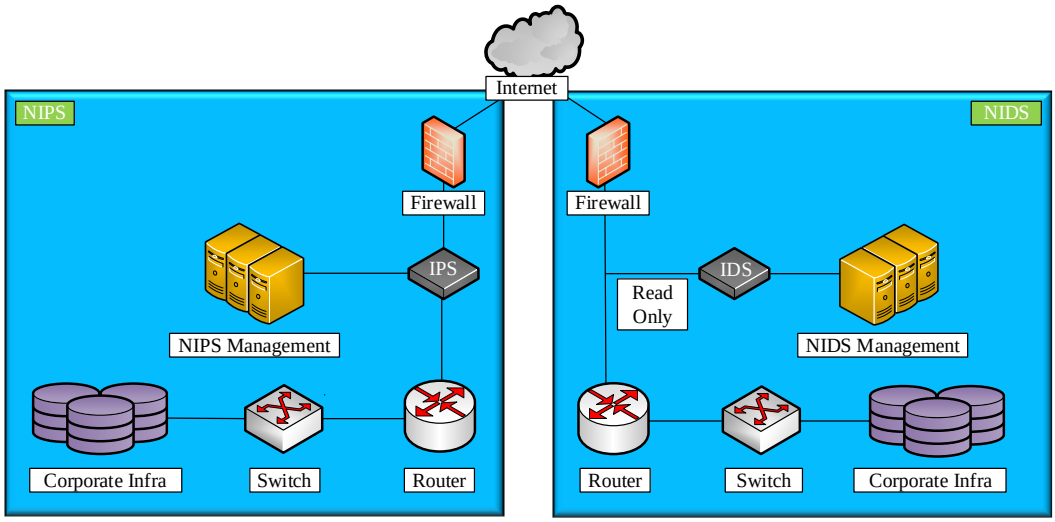


Gambar 2.7 Hierarki IDPS

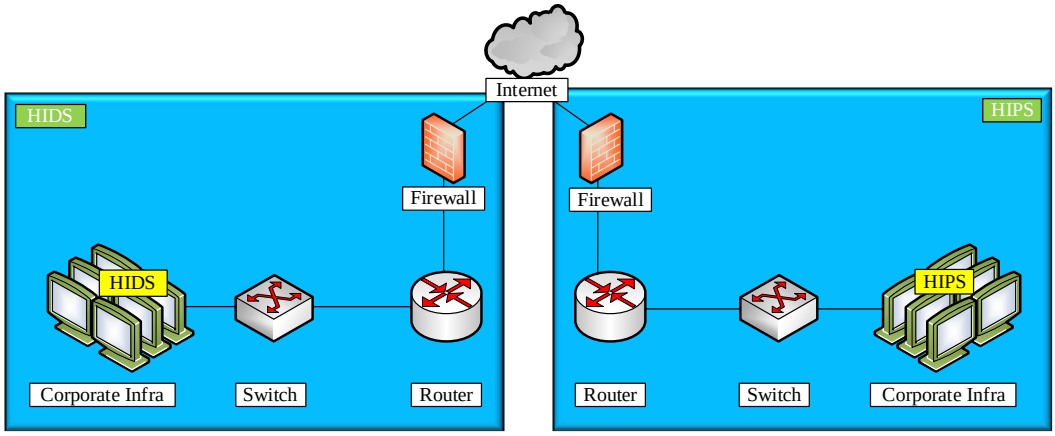
2.2.6 Tipe IDS dan IPS

Sistem Deteksi Intrusi Berbasis Jaringan (NIDS) beroperasi pada titik masuk jaringan atau sebagai pengawal pintu gerbang, di mana mereka mencari lalu lintas masuk yang berpotensi menjadi ancaman dan mengirimkan peringatan mengenai potensi ancaman kepada *administrator*. IDS membandingkan lalu lintas masuk dengan *database* tanda tangan serangan yang dikenal, ketika terdeteksi suatu peristiwa yang dikenal, peringatan dikeluarkan yang merinci insiden tersebut. IDS bertanggung jawab untuk memantau lalu lintas jaringan, namun tidak dapat mencegah lalu lintas masuk ke jaringan. Sedangkan HIDS berfungsi sama dengan NIDS yaitu menghasilkan peringatan potensi ancaman, tetapi diletakkan pada *host* yang berdiri sendiri dan hanya melakukan pengawasan pada paket-paket yang berasal dari dalam dan luar *server* yang terhubung. HIDS kemudian memberikan peringatan pada *administrator* tentang kegiatan yang mencurigakan yang terdeteksi oleh sistem. Oleh karena itu, baik IDS maupun HIDS dikenal sebagai Sistem Deteksi Intrusi, tetapi tidak mampu melakukan tindakan pencegahan secara langsung.

Network based Intrusion Prevention System (NIPS) berfungsi seperti penjaga keamanan dengan memindai lalu lintas masuk dan mencegah lalu lintas yang mencurigakan dan berbahaya agar tidak dapat masuk ke dalam jaringan. NIPS memeriksa lalu lintas masuk dengan *database* tanda serangan yang diketahui dan jika terdeteksi adanya ancaman, NIPS akan memblokir atau menolak lalu lintas tersebut. Dibandingkan dengan *firewall* yang umumnya hanya melakukan evaluasi pada tingkat permukaan, NIPS dapat melakukan evaluasi pada tingkat yang lebih dalam sebelum mengizinkan lalu lintas melalui *port*. *Host based Intrusion Prevention System* (HIPS) memiliki kemiripan dengan HIDS, di mana HIPS juga dipasang pada sistem individual. Namun, karena HIPS sebenarnya merupakan sistem dari IPS, selain dapat mengidentifikasi ancaman dunia maya, HIPS juga dapat mengambil tindakan untuk memulihkan ancaman tersebut. Contohnya, jika suatu program melampaui izinnya, maka HIPS dapat memblokir program tersebut agar tidak melakukan tindakan yang tidak disetujui[23][24]. Seperti pada Gambar 2.8 merupakan infrastruktur dari (a) NIDS/NIPS dan (b) HIDS/HIPS.



(a)



(b)

Gambar 2.8 Infrastruktur (a) NIDS/NIPS dan (b) HIDS/HIPS

2.2.7 Konsep Snort

Snort adalah sebuah sistem deteksi intrusi berbasis *open-source* yang pertama kali dirilis pada tahun 1998 oleh *Martin Roesch* dari *Sourcefire*, kemudian diambil alih oleh *Cisco* pada tahun 2013. Snort menggunakan metode *signature detection* yang didasarkan pada aturan-aturan (*rules*) yang ditulis dalam bentuk teks, dan mengidentifikasi serta memberikan *respons* terhadap *event* yang terdeteksi. Snort adalah sebuah sistem deteksi intrusi *singlethreading* yang dapat mengelola *rules* yang sangat besar dan kompleks. Snort telah mengalami perkembangan dari sekadar sistem deteksi intrusi (IDS) atau analisis protokol, menjadi sebuah sistem deteksi dan pencegahan intrusi (IDPS) yang lebih luas fungsinya dan sangat bermanfaat dalam merespons serangan terhadap *host* pada

jaringan. Snort dapat dioperasikan melalui *command line* dan juga menggunakan *Berkeley Packet Filter* sebagai opsional. Selain mudah diimplementasikan, *rules* Snort sangat efektif dalam mendeteksi jenis paket yang mencurigakan dan tidak diinginkan[5][25].

Snort memiliki fitur-fitur yang berguna dalam menjaga keamanan sistem jaringan seperti memberikan peringatan atau *alert* dan juga merekam setiap sesi yang sedang berjalan. Fitur-fitur ini memungkinkan Snort untuk mendeteksi ancaman pada sistem jaringan dan sangat bermanfaat bagi penggunanya[26]. Snort dapat dioperasikan dengan 3 mode, yaitu:

1. ***Packet sniffer***, Dalam mode operasi ini, Snort dapat memantau semua paket yang sedang berlangsung di jaringan tempat Snort ditempatkan. Snort dapat mendeteksi *sniffing* dan memberikan respon secara *real-time.*
2. ***Packet logger***, pada mode ini Snort mencatat setiap paket yang berjalan dan mengubahnya ke dalam bentuk *file.*
3. ***Network intrusion detection and prevention***, Dalam mode ini, sistem akan melakukan pengawasan terhadap lalu lintas jaringan dan melakukan analisis terhadap lalu lintas tersebut berdasarkan aturan-aturan yang telah ditentukan oleh analis keamanan[27].

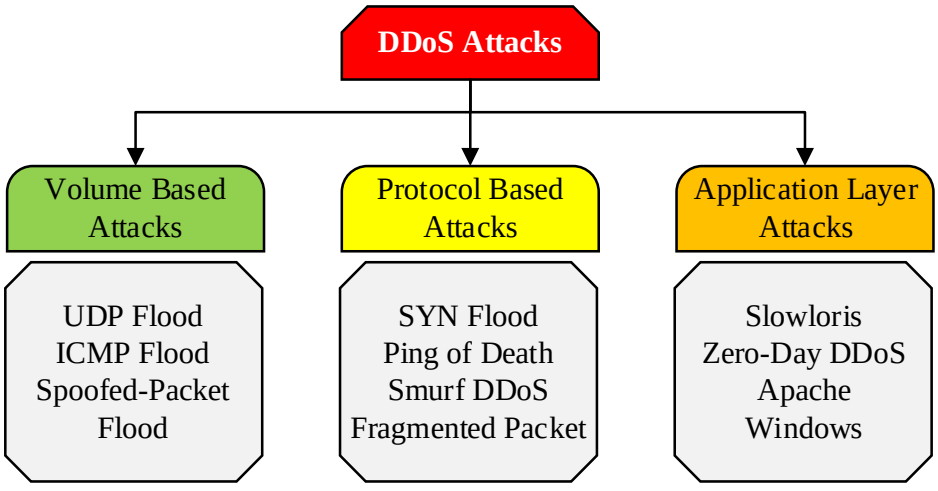
Snort memiliki tujuan utama untuk melakukan analisis terhadap paket yang masuk dan keluar dari jaringan, dan menolak paket-paket yang tidak sesuai dengan aturan tanda tangan. Selain itu, Snort juga dapat menghasilkan laporan yang meliputi informasi tentang *packet drops*, *packet analyses,* paket yang diterima, dan peringatan lainnya, termasuk serangan dan gangguan pada jaringan[28].

1. *Packet Decoder*: Tugas *decoder* paket adalah untuk menangkap paket yang melewati jaringan dari antarmuka jaringan yang berbeda dan mempersiapkan *preprocessing* paket.
2. *Preprocessor*: Penataan dan modifikasi paket berlangsung dalam fase *preprocessing* sebelum dikirim untuk dianalisis ke mesin deteksi.
3. *Detection Engine*: Fungsi mesin ini adalah untuk mengidentifikasi intrusi berdasarkan definisi serangan yang telah ditentukan sebelumnya. Paket dibandingkan dengan aturan tanda tangan untuk kecocokan jika ditemukan, tindakan yang sesuai disarankan untuk membuang atau menjatuhkan paket.

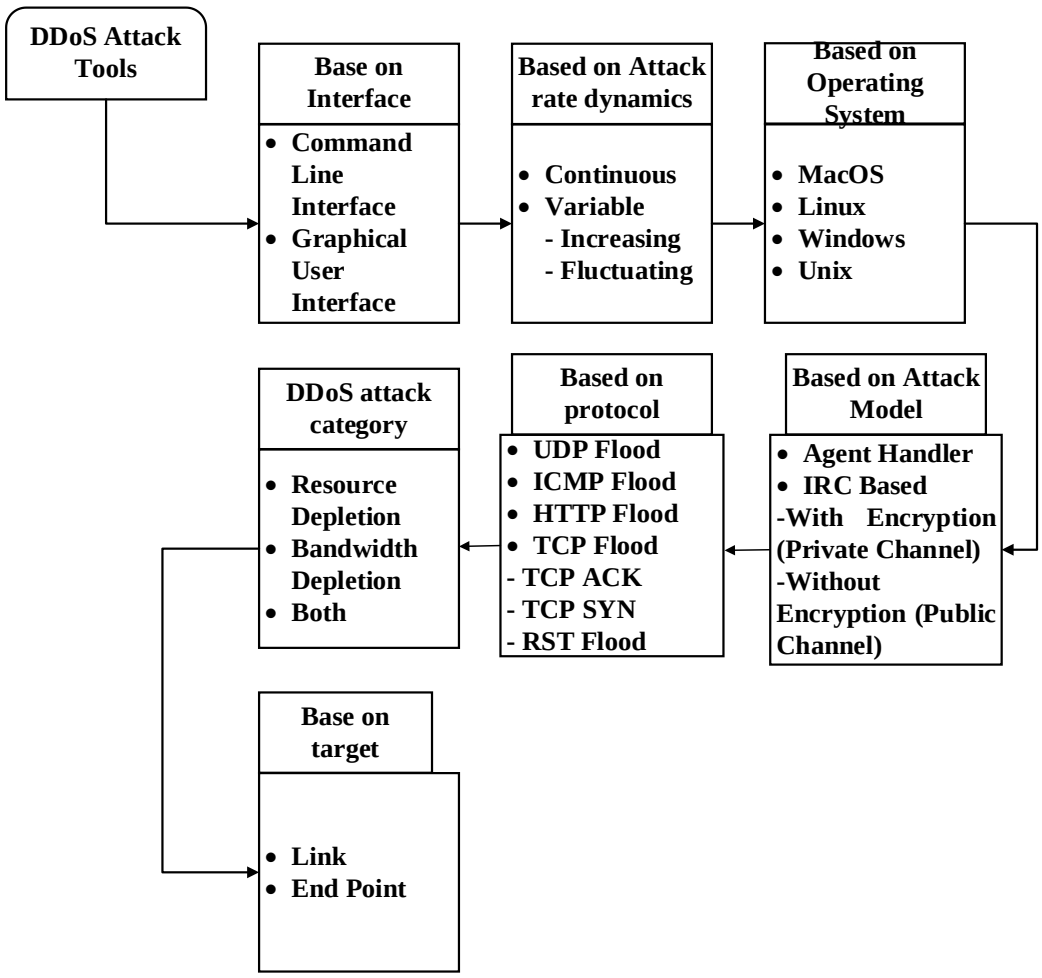
4. *Log and Alert System*: Catatan *log* dihasilkan berdasarkan hasil mesin deteksi dalam pola *file* teks atau *format* TCP-*dump*.

2.2.8 Konsep DDoS

Serangan DDoS mengacu pada *triad* CIA (*Confidentiality*, *Integrity*, dan *Availability*) yang merupakan prinsip dasar dari manajemen risiko dan keamanan informasi. Khususnya, serangan DDoS menargetkan ketersediaan (*Availability*) dari *triad* CIA, meskipun bisa juga mempengaruhi integritas dan kerahasiaan informasi jika disertai dengan jenis serangan lainnya. Ketersediaan bukanlah sesuatu yang mutlak, melainkan terdiri dari berbagai level. Sasaran dari serangan DDoS adalah membuat layanan menjadi tidak tersedia secara penuh, mengurangi kinerja layanan, atau bahkan menghentikan sepenuhnya layanan yang diserang[29][30]. Dalam hal ini, kesuksesan serangan DDoS tidak selalu berarti sumber daya target tidak tersedia sama sekali, tetapi dapat mengurangi kinerja sistem target secara signifikan. Serangan dapat dilakukan oleh satu sumber atau perangkat, disebut sebagai *Denial of Service* (DoS), atau dari beberapa sumber yang dikenal sebagai serangan *Distributed Denial of Service* (DDoS). Seperti pada Gambar 2.9 dan Gambar 2.10 merupakan klasifikasi serangan DDoS dan pengelompokkan (*Taxonomy*) alat serangan DDoS.



Gambar 2.9 Klasifikasi Serangan DDoS[31]



Gambar 2.10 ***Taxonomy of DDoS Attack Tools***[32]

Karena tujuan penyerangan yang berbeda, metode kemunculannya berbeda, yang terbagi dalam tiga klasifikasi umum[33]:

1. *Volumetric attacks*

   Serangan ini sering disebut sebagai "*surge*" dan bertujuan untuk mengalirkan kapasitas transmisi data jaringan dengan *volume data* yang sangat tinggi yang tampaknya sah. Hal ini menyebabkan respons sangat lambat atau bahkan tidak ada respons sama sekali pada sistem target. Mayoritas serangan DDoS adalah serangan jenis ini dan sering dilakukan menggunakan *botnet* yang terdiri dari perangkat IoT yang tidak aman.

2. *Protocol attacks*

   Ini dikenal sebagai serangan *'TCP state-exhaustion'* yang bertujuan untuk menghabiskan kemampuan pemrosesan infrastruktur jaringan seperti *load balancer*, *firewall*, *server*, dan komponen lainnya. Serangan ini terjadi

dengan mengeksploitasi semua koneksi bersamaan untuk komunikasi protokol lapisan 3 dan lapisan 4 dengan mengirimkan permintaan yang mencurigakan. Akibatnya, serangan ini dapat menurunkan kualitas layanan untuk permintaan yang sah.

3. *Application attacks*

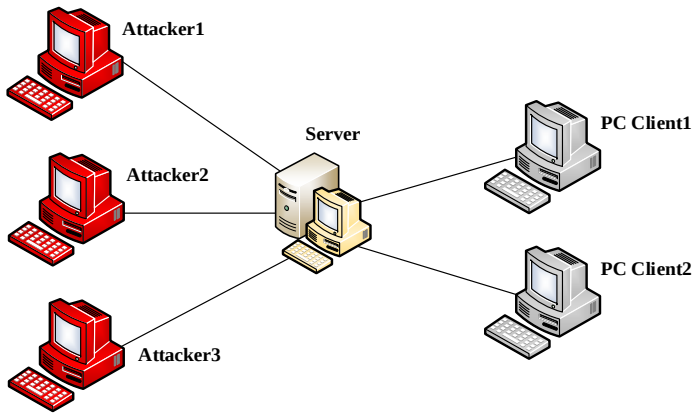
   Serangan ini juga dikenal sebagai '*Application Layer attacks'* dan bertujuan untuk membanjiri *server* atau aplikasi target dengan permintaan yang menghabiskan koneksi. Serangan ini lebih sulit dideteksi karena hanya membutuhkan sejumlah kecil mesin untuk meluncurkan serangan dengan kecepatan yang rendah, sehingga aktivitas yang muncul tampak lebih nyata.

Tabel 2.2 Tipe Serangan DDoS [34]

DDoS Attack	DDoS characteristics and type			
	Infrastructure	Application	Direct	Reflection
UDP Flood	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	Melumpuhkan aplikasi atau layanan di server	Mengirim lalu lintas serangan langsung ke alamat IP sasaran	Mengirim permintaan UDP palsu dengan alamat IP sasaran palsu
TCP Flood	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	-	Mengirim lalu lintas serangan langsung ke alamat IP sasaran	-
HTTP Flood	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	Melumpuhkan aplikasi atau layanan di server	Mengirim lalu lintas serangan langsung ke alamat IP sasaran	-
ICMP Flood	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	-	Mengirim lalu lintas serangan langsung ke alamat IP sasaran	-

DDoS Attack	DDoS characteristics and type			
	Infrastructure	Application	Direct	Reflection
XML Flood	-	Melumpuhkan aplikasi atau layanan di server	Mengirim lalu lintas serangan langsung ke alamat IP sasaran	-
Ping of death	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	Melumpuhkan aplikasi atau layanan di server	-	-
Smurf	Melumpuhkan perangkat jaringan seperti router, firewall, atau server DNS	-	-	Mengirim permintaan UDP palsu dengan alamat IP sasaran palsu

Seperti pada Tabel 2.2 merupakan tipe dan karakteristik dari serangan DDoS. Beberapa serangan DDoS yang paling umum adalah ICMP *Flood*, SYN *Flood*, DNS *amplification attacks*, *Smurf Attack*, dan *Fraggle Attack*. Sebagian besar serangan ini bertujuan untuk menghabiskan sumber daya jaringan atau *server*[35]. Kemudian pada Gambar 2.11 merupakan topologi dari serangan DDoS.



Gambar 2.11 Topologi DDoS *Attack*[36]

2.2.9 Konsep *Firewall*

Berbagai bentuk serangan bahkan ancaman baik secara langsung maupun tidak langsung akan memberikan dampak pada aktifitas yang terjadi pada jaringan internet tersebut, sehingga untuk mengamankan aktivitas jaringan *internet* dari ancaman langsung maupun tidak langsung, diperlukan perlindungan dari berbagai jenis serangan. Salah satu cara yang bisa dilakukan adalah dengan menggunakan *firewall*, yaitu suatu mode keamanan yang dapat memberikan perlindungan terhadap kemungkinan terjadinya serangan pada jaringan. *Firewall* adalah sebuah konsep keamanan pada sistem operasi yang berfungsi untuk melindungi sistem dari serangan yang tidak diinginkan. Sistem operasi dalam jaringan komputer adalah *platform* pengaturan sumber daya yang memberikan keamanan dan perlindungan untuk jaringan, serta mengontrol akses pengguna untuk terhubung ke sumber daya jaringan. Sementara itu, *Firewall* dikonfigurasi untuk mencegah akses yang tidak diinginkan ke jaringan, baik dari dalam maupun luar[37]. Pada penelitian ini menggunakan *pfSense* sebagai *firewall* untuk melakukan pencegahan dari serangan DDoS dengan mekanisme pemblokiran IP.

2.2.10 Konsep *Email Notification*

Pada penelitian ini, *pfSense* menyediakan fitur *Email Notification* yang berfungsi untuk memberikan pesan peringatan (*alert*) kepada *administrator* ketika ada klien yang berhasil mengakses salah satu *port* dan melakukan serangan terhadap *server*. Fitur ini akan mengirimkan pemberitahuan melalui *email* menggunakan koneksi SMTP yang terhubung ke *server Email*. Untuk menggunakan fitur ini, *server* harus dikonfigurasi untuk mengizinkan *relay* dari *firewall* atau menerima koneksi SMTP yang sudah diotentikasi[38].