BAB 2
DASAR TEORI

2.1 KAJIAN PUSTAKA

Pada penelitian sebelumnya Penelitian [11] membahas tentang implementasi dan analisis keamanan jaringan IDS menggunakan *Suricata* pada *web server*. Fokus utama penelitian ini adalah kemampuan *Suricata* dalam mendeteksi dan mencegah serangan intrusi pada *web server* sesuai aturan yang telah dibuat. Penelitian ini tidak hanya memberikan wawasan mengenai efektivitas *Suricata* sebagai IDS namun juga menyajikan hasil mendalam mengenai langkah implementasi dan analisis keamanan yang diterapkan. Menjelajahi implementasinya, ia mempelajari secara detail bagaimana *Suricata* dapat mengatasi serangan yang dapat membahayakan *server web*. Aturan yang dibuat sangat penting untuk proses deteksi dan penelitian yang menjelaskan bagaimana *Suricata* dapat beradaptasi terhadap berbagai jenis ancaman. Selain itu, hasil penelitian ini memberikan wawasan mengenai respons *Suricata* terhadap serangan, termasuk langkah diagnostik dan perencanaan tindakan yang diambil untuk memulihkan ancaman keamanan yang teridentifikasi.

Selain itu, penelitian [12] Membahas secara komprehensif mengenai pentingnya keamanan jaringan komputer dalam era teknologi informasi saat ini menjadi suatu hal yang tidak dapat diabaikan. Penelitian sebelumnya telah menyoroti betapa krusialnya keamanan jaringan komputer dalam menjaga integritas dan keaslian data, serta memastikan layanan yang tersedia bagi para pengguna. Dengan manusia yang semakin bergantung pada sistem informasi, terdapat peningkatan statistik insiden keamanan jaringan yang menuntut solusi yang efektif. Salah satu pendekatan yang diambil adalah melalui simulasi. Simulasi menjadi instrumen yang berharga karena mampu mensimulasikan situasi keamanan jaringan nyata dengan cara yang lebih teramati, mempercepat proses deteksi dan penanganan gangguan, serta pemulihan sistem atau layanan yang terkena dampaknya. Metode penelitian yang digunakan dalam jurnal ini mengikuti

serangkaian langkah analisis sistem keamanan yang diinginkan. Langkah-langkah ini mencakup persiapan perangkat keras dan perangkat lunak sebagai pendukung implementasi IDS dalam rangka memperkuat keamanan jaringan komputer. Selain itu, pada tahap pemasangan IDS dengan menggunakan *Snort*, diperlukan paket-paket khusus untuk memastikan instalasi berjalan tanpa hambatan. Proses instalasi IDS memerlukan perintah khusus di dalam lingkungan sistem operasi *Linux* agar paket-paket yang diperlukan dapat terinstal dengan benar, memastikan IDS berfungsi secara optimal dalam mendeteksi dan menanggapi ancaman keamanan jaringan. Hasil dari penelitian ini menunjukkan bahwa pemanfaatan IDS dalam simulasi dapat efektif membantu menangani ancaman keamanan jaringan dengan cepat dan optimal. IDS berperan penting dalam mendeteksi serangan serta aktivitas mencurigakan pada jaringan, sambil terus memonitor kejadian melalui analisis *log* sistem. Dengan demikian, IDS mampu meningkatkan tingkat keamanan jaringan komputer dengan memberikan respons yang cepat terhadap ancaman yang muncul, sehingga proses mitigasi gangguan dapat dilaksanakan secara efisien. Kesimpulannya, jurnal ini memberikan gambaran menyeluruh tentang urgensi keamanan jaringan komputer, peranan IDS sebagai solusi, metode penelitian yang digunakan, serta hasil simulasi yang relevan dalam konteks keamanan jaringan dan komputer.

Lebih lanjut penelitian [13] yang dijelaskan oleh penelitian memberikan informasi rinci tentang implementasi sistem deteksi intrusi IDS menggunakan *Snort* sebagai pilar utama sistem keamanan. Keunikan penelitian ini terletak pada integrasi *WhatsApp* dan *Telegram* sebagai media notifikasi yang menekankan pada aspek komunikasi dan respon cepat. Pendekatan *Network Development Life Cycle* (NDLC) yang disoroti dalam artikel menunjukkan bahwa penelitian ini tidak hanya berfokus pada aspek teknis saja namun juga menggambarkan upaya pengembangan sistem yang terencana dan terstruktur. Penelitian ini mengeksplorasi bagaimana *Intrusion Detection System Snort*, dalam konteks penerapan *Intrusion Detection System* umum, dapat secara efektif mengidentifikasi dan mendeteksi berbagai serangan.

Analisis mendalam terhadap jenis serangan yang berhasil diatasi oleh *Snort* merupakan aspek penting yang berkontribusi pada pemahaman umum tentang

keandalan dan tanggung jawab sistem ini dalam melindungi jaringan komputer. Integrasi *WhatsApp* dan *Telegram* sebagai media notifikasi membawa dimensi baru dalam lanskap keamanan. Pemberitahuan *real-time* melalui platform komunikasi populer ini memberikan lapisan keamanan tambahan dengan memungkinkan respons cepat terhadap ancaman yang terdeteksi. Keterlibatan media sosial dalam proses pemberitaan menunjukkan kemampuan penelitian ini untuk beradaptasi dengan tren media kontemporer, memastikan bahwa informasi keselamatan dapat disebarluaskan dengan cepat dan ditindaklanjuti oleh para pemangku kepentingan.

Mendalami keamanan siber, penelitian [14] menyoroti implementasi dan analisis *Snort* dan *Suricata* sebagai IDS dan *Intrusion Prevention System* (IPS) untuk mencegah DOS dan *Distributed Denial of Service* (DDoS) . Penelitian ini menjadi relevan dalam menghadapi ancaman yang semakin kompleks terhadap ketersediaan layanan dalam sistem. Selama fase implementasi, diskusikan bagaimana mengintegrasikan *Snort* dan Suricata sebagai solusi keamanan yang efektif. Analisis mendalam terhadap kinerja deteksi serangan mereka, terutama melalui pengujian menggunakan alat *Hping*3 dan *Slowloris*, menunjukkan ketangguhan mereka dalam menghadapi berbagai skenario ancaman. Keberhasilan dalam mencegah serangan DoS dan DDoS memberikan tolok ukur penting untuk mengevaluasi efektivitas IDS dan IPS sebagai lapisan pertahanan utama.

Penelitian [15] Membahas tentang keamanan jaringan internet dalam lingkup pekerjaan adalah suatu hal yang esensial untuk memastikan bahwa para pengguna dapat mengakses koneksi internet yang terjamin keamanan dan kestabilannya. Keamanan jaringan komputer merupakan aspek krusial dalam penggunaan sistem, namun seringkali kelemahan dalam sistem jaringan komputer diabaikan. Oleh karena itu, penerapan IDS dalam keamanan jaringan dapat membantu dalam mendeteksi kegiatan yang mencurigakan dan menghadapi potensi ancaman yang mungkin timbul. Metode penelitian ini menggabungkan pendekatan observasional dan studi kepustakaan. Observasi digunakan untuk mengamati dan membandingkan berbagai desain serta sistem keamanan jaringan yang ada, sementara studi kepustakaan dilakukan untuk mengumpulkan data dan referensi terkait dengan keamanan jaringan. Analisis keamanan jaringan dilakukan melalui pendekatan kuantitatif untuk menilai respons sistem IDS dalam mendeteksi dan

menanggulangi ancaman *brute force*. Hasil penelitian menunjukkan bahwa menerapkan sistem keamanan *brute force* pada *router mikrotik* menghasilkan respons yang efektif terhadap serangan tersebut. Sistem keamanan IDS mampu secara cepat mendeteksi dan memblokir alamat *Internet Protocol* (IP) yang mencurigakan yang melakukan serangan *brute force*, dalam waktu kurang dari empat detik. Selain itu, penerapan ARP *list* juga terbukti efektif dalam membatasi akses *client* terhadap router *mikrotik*, sehingga alamat IP yang tidak dikenal atau tidak terdaftar dalam ARP *list* tidak dapat terhubung ke jaringan. Dengan demikian, hasil penelitian menunjukkan bahwa menggunakan *Firewall* IDS dan menerapkan ARP *list* dapat meningkatkan keamanan jaringan komputer dari serangan *brute force* dan membatasi akses yang tidak sah ke *router mikrotik.*

Pada Penelitian ini bertujuan untuk mengetahui kinerja dari IDS Snort untuk mendeteksi serangan DoS dengan metode serangan *flood attack*, pada jaringan virtual. Berbeda dengan penelitian sebelumnya hanya berfokus pada IDS *Suricata* dengan *Dirbuster*, pada penelitian ini menggunakan SDN *Floodlight*, dimana *controller* yang mengatur pada jaringan virtual, penjelasan dari penelitian terdahulu dapat dilihat pada tabel 2.1

Tabel 2. 1 Penelitian Terdahulu

Peneliti, Tahun	Metode	Sistem	Hasil
Stephani, E., Nova, F., & Asri, E. (2020)	*Diagnosis, Action Planning, Intervention (Action Taking), Evaluation (Assessment), dan Reflection (Learning)*.	IDS Suricata pada *web server*	Implementasi *Suricata* dengan *firewall* OPNsense dapat mendeteksi dan mencegah anomali pada web server dari penyusup.
Barany Fachri (2020)	Pengujian sistem IDS dengan serangan-serangan seperti Nmap (*Scanning Port*) dan DOS (*Denial Of Service*).	IDS *Snort*	*Snort* berhasil menganalisis serangan yang terdeteksi dengan mode IDS, memberikan *alert* sesuai dengan ancaman yang terjadi, dan menampilkan penyalahgunaan jaringan yang terdeteksi.
Purnama, T. (2023).	*Network Development Life Cycle* (NDLC) yang mencakup tahapan *Analysis, Design, Simulation, Prototype, Implementation*, Monitoring, dan *Management*.	IDS *Snort, WhatsApp, Telegram,* NDLC	IDS Snort dapat mengidentifikasi dan mendeteksi berbagai serangan, seperti ping attack (ICMP Traffic), Port Scanning, serta DOS/DDOS UDP Flooding yang ditujukan ke server.
Santoso, D., Noertjahyana, A.,	Pengujian dengan *Hping3* dan *Slowloris*	IDS *Snort* dan Suricata sebagai	Dalam serangan HTTP *Flood*, *Suricata* menghilangkan

Peneliti, Tahun	Metode	Sistem	Hasil
& Andjarwirawan, J. (2022).		IDS dan IPS pada *linux*	serangan lebih cepat daripada *Snort*.
S.A. Puntadheva, Rr.Y.R. Kusumaningsih, J. Triyono (2023)	observasi dan studi kepustakaan.	*Firewall Intrusion Detection System* (IDS)	firewall IDS efektif dalam mendeteksi dan mengatasi serangan brute force terhadap router mikrotik. Saat terjadi serangan brute force, firewall IDS langsung memblokir IP yang dicurigai melakukan serangan jika melebihi batas percobaan login yang telah ditentukan.

2.2 Keamanan Jaringan

Keamanan siber merupakan landasan penting untuk melindungi informasi dan sistem komputer dari berbagai ancaman yang dapat menimbulkan kerugian. Konsep utamanya menyangkut aspek privasi, yang berupaya mencegah akses tidak sah terhadap informasi sensitif dengan menggunakan teknik enkripsi. *Confidentiality* memastikan bahwa informasi hanya dapat diakses oleh orang yang berwenang dan dilindungi dari akses yang tidak sah. Konsep ini sangat penting dalam menjaga privasi dan keamanan data sensitif. Berbagai metode seperti enkripsi, kontrol akses, dan otentikasi digunakan untuk menjaga kerahasiaan.

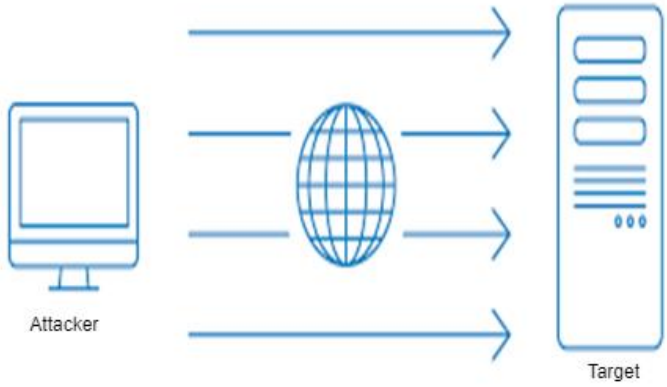
Gambar 2. 1 Keamanan Jaringan

Integrity memastikan bahwa informasi tetap akurat, konsisten, dan tidak diubah tanpa izin selama penyimpanan atau transmisi. Integritas data sangat penting untuk memastikan bahwa informasi yang digunakan dalam pengambilan keputusan adalah benar dan dapat dipercaya. Teknik seperti checksum, hash, dan tanda tangan digital sering digunakan untuk memverifikasi integritas data. *Availability*

memastikan bahwa sistem dan data yang dibutuhkan selalu dapat diakses oleh pengguna yang berwenang ketika dibutuhkan. Ketersediaan mencakup berbagai langkah seperti perencanaan pemulihan bencana, pencadangan data, dan pemeliharaan infrastruktur untuk mencegah *downtime* atau gangguan. Dari penjelasan ini dapat dilihat pada Gambar 2.1 [16].

2.3 Serangan *Denial of Service* (DoS)

DoS adalah jenis serangan yang bertujuan untuk membuat suatu situs web atau layanan online menjadi tidak tersedia bagi pengguna yang sah. Prinsip dasar dari serangan ini adalah dengan mengirimkan sejumlah besar permintaan palsu ke sistem target. Akibatnya, sistem tersebut menjadi terlalu sibuk untuk menangani permintaan yang sebenarnya, sehingga tidak dapat melayani pengguna yang sah. Salah satu cara umum untuk melancarkan serangan DoS adalah dengan membanjiri sistem target dengan data atau permintaan palsu, yang membuatnya kelebihan beban dan tidak dapat berfungsi secara normal. Serangan DoS juga dapat dimulai dengan memanfaatkan kelemahan dalam perangkat lunak atau infrastruktur jaringan yang digunakan oleh sistem target.



Gambar 2. 2 DoS

Dalam serangan yang lebih kompleks, penyerang sering menggunakan jaringan botnet yang terdiri dari komputer yang terinfeksi, untuk melancarkan serangan secara bersamaan dari berbagai lokasi. Dampak dari serangan DoS bisa sangat merugikan, tidak hanya menyebabkan gangguan layanan online, tetapi juga mengakibatkan kerugian finansial dan reputasi bagi organisasi yang menjadi korban.. Dari penjelasan ini dapat dilihat pada Gambar 2.2[17].

2.3.1 Jenis – Jenis Serangan DoS

Jenis serangan DoS mencakup serangan banjir lalu lintas, serangan yang mengeksploitasi kelemahan sistem, dan serangan yang mengeksploitasi sumber daya terbatas, seperti *bandwidth* atau kapasitas penyimpanan. Dalam beberapa kasus, serangan DoS dapat berkembang menjadi serangan penolakan layanan terdistribusi DDoS, yang mana serangan tersebut berasal dari berbagai sumber yang tersebar luas, sehingga membuat remediasi menjadi lebih sulit. Penting untuk dipahami bahwa serangan DoS tidak hanya menimbulkan ancaman bagi bisnis atau organisasi besar, namun juga dapat memengaruhi individu atau situs web kecil. Oleh karena itu, perlindungan terhadap serangan DoS sangat penting untuk menjaga ketersediaan layanan dan kelangsungan operasional suatu entitas di era digital. Menerapkan strategi keamanan yang kuat, memantau lalu lintas jaringan secara aktif, dan memperbarui sistem secara berkala merupakan langkah penting dalam memerangi ancaman serangan penolakan layanan [18].

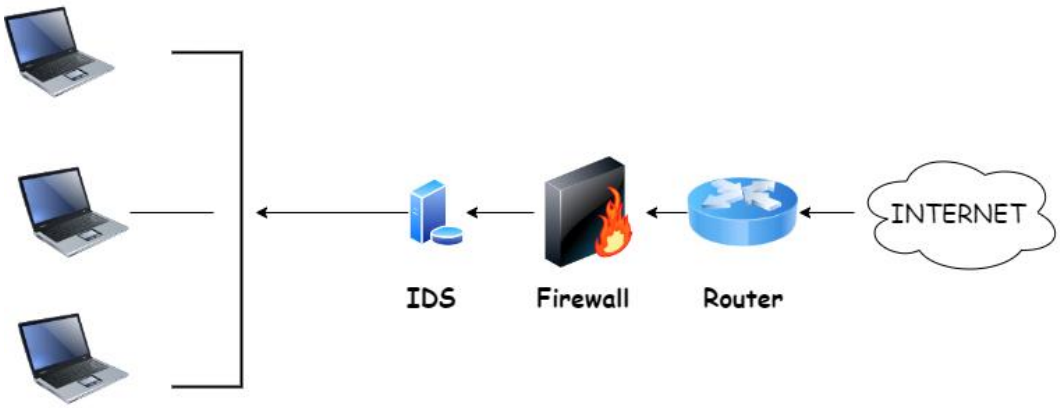
2.3.2 *Hping3*

Hping3 adalah sebuah alat baris perintah yang digunakan untuk menghasilkan dan mengirim paket jaringan, serta menganalisis *respons* dari perangkat jaringan lainnya. Dikembangkan oleh Salvatore Sanfilippo, *Hping3* sering digunakan dalam pengujian penetrasi dan audit keamanan jaringan karena kemampuannya untuk memodifikasi dan mengirim berbagai jenis paket. Alat ini juga memungkinkan pengguna untuk menyesuaikan hampir semua aspek dari paket yang dikirim, dan tujuan, *flag Transmission Control Protocol* (TCP), ukuran *payload*, dan waktu antar paket. *Hping3* memiliki kemampuan untuk melakukan *traceroute*, *flood*, dan deteksi *firewall*, serta mendukung *spoofing Internet Protocol* (IP) , yang dapat digunakan untuk menguji kebijakan keamanan jaringan. [19].

2.4 Intrusion Detection System (IDS)

Sistem deteksi intrusi (IDS) adalah sistem keamanan yang dirancang untuk mengidentifikasi dan memberikan peringatan tentang aktivitas mencurigakan atau tidak biasa dalam jaringan atau sistem komputer. Fungsi utama IDS adalah memantau lalu lintas jaringan, aktivitas pengguna, dan kejadian sistem untuk

mendeteksi pola yang mungkin mengindikasikan serangan atau pelanggaran keamanan. IDS merupakan bagian penting dari perlindungan keamanan siber karena memiliki kemampuan untuk mengidentifikasi ancaman yang tidak dapat diatasi oleh mekanisme keamanan konvensional, seperti *firewall*. Pentingnya IDS dalam deteksi ancaman terletak pada kemampuannya untuk memberikan peringatan dini terhadap aktivitas mencurigakan, memungkinkan organisasi merespons dan menyelesaikan serangan sebelum menyebabkan kerusakan yang signifikan. IDS dapat dibagi menjadi dua jenis utama, IDS berbasis tanda tangan, yang menggunakan database tanda tangan serangan yang diketahui untuk mengenali pola serangan yang terjadi sebelumnya, dan IDS berbasis perilaku, yang menganalisis aktivitas yang tidak diketahui, normal atau tidak normal. bertukar surat. model biasa. Dari penjelasan ini dapat dilihat pada gambar 2.3[20] .



Gambar 2. 3 IDS

2.4.1 Jenis – Jenis IDS

Sistem deteksi intrusi berbasis jaringan *Network – based Intrusion Detection System* (NBIDS) adalah sistem deteksi intrusi yang berfokus pada pemantauan dan analisis lalu lintas jaringan untuk mengidentifikasi pola atau tanda serangan. NBIDS ditempatkan pada titik-titik kunci dalam jaringan, seperti gateway atau switch, sehingga dapat memantau seluruh lalu lintas yang melewati titik tersebut. Keunggulan utama NBIDS adalah kemampuannya dalam mendeteksi serangan pada beberapa server, sehingga memberikan gambaran keamanan jaringan yang komprehensif. Namun keterbatasannya terletak pada ketidakmampuannya memantau aktivitas di tingkat server secara detail. Di sisi lain, Sistem Deteksi

Intrusi Berbasis *Host. Host – Based Intrusion Detection System* (HBIDS) fokus pada pendeteksian intrusi pada tingkat host individu atau sistem operasi. HBIDS ditempatkan langsung di server yang ingin di pantau, sehingga memungkinkan pemantauan aktivitas internal dan eksternal di server tersebut.

Manfaat HBIDS mencakup kemampuan untuk mendeteksi serangan terhadap host tertentu serta memberikan informasi lebih rinci tentang aktivitas tersebut. Namun keterbatasannya terletak pada sulitnya memantau seluruh jaringan, terutama jaringan yang besar dan kompleks. Kedua jenis IDS ini berperan penting dalam rantai perlindungan keamanan informasi. Kombinasi NBIDS dan HBIDS dapat memberikan pandangan komprehensif mengenai ancaman keamanan, memungkinkan organisasi mendeteksi dan merespons berbagai jenis serangan. Sementara NBIDS berfokus pada lalu lintas jaringan yang melewati titik-titik kritis, HBIDS menawarkan keuntungan dalam mendeteksi ancaman spesifik yang terkait dengan host. Oleh karena itu, penggunaan kedua jenis *Intrusion Detection System* ini secara bersamaan dapat meningkatkan kemampuan organisasi dalam mendeteksi dan merespons berbagai ancaman keamanan[21].

2.4.2 Waktu Respon

Waktu Respons mengacu pada durasi yang diperlukan oleh sistem IDS *Snort* untuk mendeteksi dan merespon ancaman setelah serangan simulasi dilancarkan. Pengukuran waktu respons penting untuk mengevaluasi kecepatan dan efektivitas IDS dalam merespons insiden keamanan. IDS *Snort* diatur untuk memonitor lalu lintas jaringan dan mendeteksi paket berbahaya. Waktu mulai simulasi serangan dan waktu deteksi pertama oleh *Snort* dicatat. Selisih antara waktu mulai serangan dan waktu Snort mendeteksi ancaman dan mengeluarkan peringatan dihitung dalam satuan detik (s) atau milidetik (ms). Proses tersebut diulang dengan berbagai ukuran data dan jumlah paket untuk mendapatkan variasi waktu respon di kondisi berbeda, untuk rumus waktu Untuk menghitung waktu respons, menggunakan log entri yang merekam waktu permintaan dan waktu respons. Rumus dasar untuk menghitung waktu respons adalah selisih antara waktu ketika permintaan diterima (*timestamp* permintaan) dan waktu ketika respons dikirim (*timestamp* respons). [22].

(Waktu Respons = $timestamp$ respons − $timestamp$ permintaan)

2.4.3 Akurasi IDS *Snort* Dalam Mendeteksi Intrusi Lalu Lintas Jaringan

Akurasi deteksi mengevaluasi kemampuan IDS Snort dalam membedakan antara lalu lintas normal dan berbahaya. Akurasi ini diukur dengan menghitung tingkat true positives (jumlah serangan yang terdeteksi) dan true negatives adalah lalu lintas normal yang benar-benar tidak terdeteksi sebagai ancaman. Pertama jaringan diuji dengan lalu lintas normal dan berbahaya secara bergantian, di mana lalu lintas terdiri dari data biasa tanpa aktivitas serangan dan lalu lintas berbahaya terdiri dari paket-paket yang mensimulasikan serangan DoS. Kedua, *Snort* diatur untuk memonitor dan mencatat setiap aktivitas yang dianggap mencurigakan, dan hasil deteksi dikumpulkan untuk dianalisis [23].

2.4.4 Beban Kerja

Beban kerja pada memori terjadi ketika perangkat kehabisan *Random Access Memory* (RAM) yang cukup untuk menyimpan dan memproses data yang masuk, yang dapat mengakibatkan peningkatan waktu latensi, packet loss, dan bahkan *crash* pada perangkat. Ketika RAM penuh, sistem operasi akan mulai menggunakan penyimpanan disk yang lebih lambat untuk *swap*, yang menyebabkan kinerja keseluruhan sistem menurun drastis. Hal ini terutama menjadi masalah dalam aplikasi yang membutuhkan akses cepat ke data, seperti basis data atau aplikasi *real-time*. Beban kerja pada *Central Processing Unit* (CPU) terjadi ketika terlalu banyak proses yang harus dijalankan secara bersamaan atau ketika aplikasi memerlukan daya komputasi yang tinggi.

Kondisi ini dapat menyebabkan penurunan throughput jaringan dan peningkatan waktu pemrosesan karena CPU tidak mampu menangani semua tugas secara efisien. Ketika CPU bekerja pada kapasitas maksimal, setiap tambahan proses atau permintaan akan semakin memperlambat kinerja keseluruhan sistem, yang sering kali terlihat sebagai lag atau freeze pada aplikasi. sementara itu, beban kerja jaringan terjadi ketika terlalu banyak data mencoba melewati jalur jaringan yang kapasitasnya terbatas, mengakibatkan penurunan kinerja dan peningkatan waktu tunggu. Fenomena ini dapat disebabkan oleh berbagai faktor, termasuk

peningkatan jumlah pengguna yang mengakses jaringan secara bersamaan, aplikasi yang memerlukan *bandwidth* tinggi seperti *video streaming* atau *video conferencing*, serta serangan siber seperti DoS [24].

2.4.5 Metode IDS

Sistem deteksi intrusi IDS adalah sistem yang dirancang untuk mendeteksi aktivitas mencurigakan atau tidak diinginkan dalam jaringan atau sistem komputer. IDS bekerja dengan memantau lalu lintas jaringan atau aktivitas sistem untuk mengidentifikasi pola yang mencurigakan, yang dapat berupa pola yang tidak biasa atau pola yang terkait dengan serangan yang diketahui sebelumnya. Ada dua pendekatan utama dalam metodologi IDS. analisis berbasis tanda tangan dan analisis anomali. Metode berbasis tanda tangan menghubungkan pola lalu lintas dengan serangan yang diketahui, sementara analisis anomali mencari pola yang tidak biasa dalam lalu lintas atau aktivitas sistem. Berbagai algoritma dan teknik digunakan dalam IDS untuk mendeteksi serangan, mulai dari algoritma berbasis aturan hingga pembelajaran mesin dan kecerdasan buatan. Setelah serangan terdeteksi, IDS dapat mengeluarkan respons yang sesuai, seperti memblokir alamat IP yang mencurigakan atau mengirimkan peringatan ke administrator jaringan. Evaluasi kinerja juga penting dalam pengembangan IDS untuk memastikan bahwa sistem dapat mengidentifikasi serangan secara akurat dan efektif tanpa menimbulkan terlalu banyak alarm palsu. Oleh karena itu, IDS berperan penting dalam melindungi jaringan dan sistem komputer dari berbagai ancaman keamanan siber [25].

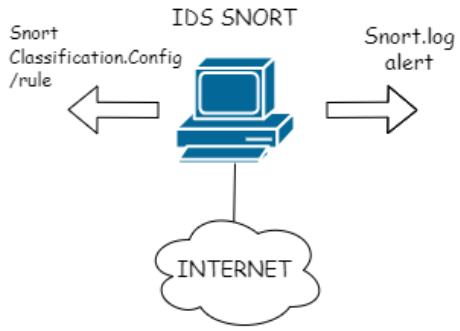
2.5 Pengenalan *Snort*

Snort, pertama kali dikembangkan oleh Martin Roesch pada tahun 1998, adalah salah satu sistem deteksi intrusi (IDS) *open source* yang paling terkenal dan banyak digunakan di dunia. Sejak awal, *Snort* telah mengalami evolusi yang signifikan dan menjadi alat yang sangat efektif untuk mendeteksi ancaman keamanan di lingkungan jaringan. Snort bukan hanya salah satu proyek IDS *open source* yang paling matang, namun terus berkembang dengan kontribusi dari komunitas global, menjadikannya pilihan utama dalam upaya keamanan siber.

Arsitektur dasar *Snort* terdiri dari beberapa komponen utama. Praprosesor berfungsi memproses dan menyiapkan lalu lintas jaringan sebelum deteksi dimulai. Detektor *snort*, menggunakan aturan dan tanda tangan, membantu mengidentifikasi pola lalu lintas yang konsisten dengan ancaman yang telah ditentukan sebelumnya. Komponen perekam bertugas mencatat informasi terkait aktivitas mencurigakan atau terdeteksi sebagai serangan [26].

2.5.1 Cara Kerja *Snort*

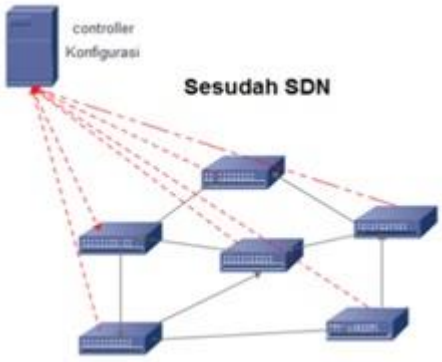
Snort menggunakan dua metode utama untuk mendeteksi serangan, termasuk deteksi berbasis tanda tangan dan deteksi berbasis anomali. Deteksi berbasis tanda tangan melibatkan perbandingan lalu lintas jaringan dengan database tanda tangan yang telah ditentukan sebelumnya. *Snort* akan mengeluarkan peringatan atau respons jika ditemukan kecocokan antara pola lalu lintas dan tanda tangan yang diidentifikasi. Deteksi berbasis anomali, di sisi lain, melibatkan pemantauan aktivitas jaringan dan mengidentifikasi perilaku yang tidak wajar atau tidak wajar. *Snort* akan mengeluarkan peringatan jika mendeteksi aktivitas yang tidak sesuai dengan perilaku normal. Melalui pendekatan gabungan ini, *Snort* memberikan fleksibilitas dan akurasi tingkat tinggi dalam deteksi ancaman. Keberhasilan *Snort* tidak hanya terletak pada kemampuan pendeteksiannya yang andal, namun juga pada kemampuannya untuk berintegrasi dengan perangkat lunak keamanan lainnya dan memberikan laporan informatif kepada administrator keamanan. Sebagai proyek yang sedang berjalan, *Snort* terus meningkatkan kemampuannya terhadap ancaman keamanan yang terus berkembang, menjadikannya solusi IDS yang andal dan efektif di era jaringan yang dinamis ini, Dari penjelasan ini dapat di lihat pada Gambar 2.4 [27].



Gambar 2. 4 IDS *Snort*

2.6 *Software Defined Network* (SDN)

SDN adalah paradigma manajemen jaringan baru yang telah mengalami pertumbuhan pesat sejak diperkenalkan. SDN mendefinisikan ulang manajemen dan kontrol jaringan dengan memisahkan lapisan kontrol dari lapisan transfer data. Hal ini memungkinkan pengguna untuk mengelola dan mengendalikan infrastruktur jaringan secara terpusat melalui perangkat lunak, memberikan fleksibilitas yang belum pernah terjadi sebelumnya,



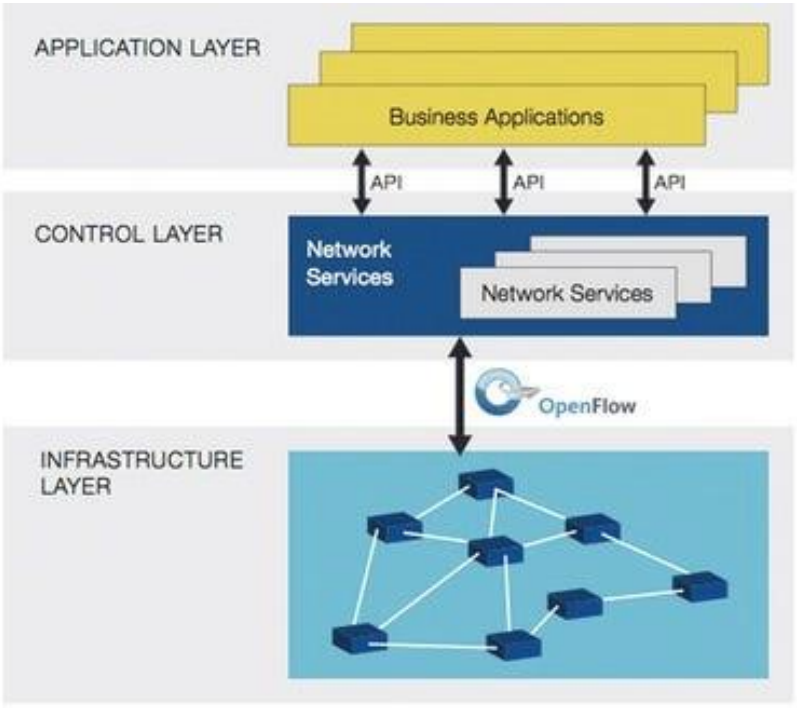
Gambar 2. 5 SDN [28]

Pengembangan SDN dimulai sebagai respon terhadap keterbatasan model tradisional yang mengandalkan perangkat keras untuk menjalankan fungsi kontrol. Dengan memindahkan kontrol jaringan ke lapisan perangkat lunak, SDN memberikan kemampuan untuk menskalakan dan mengelola sumber daya jaringan dengan lebih efektif. Hal ini menyebabkan perubahan mendasar dalam cara memahami dan mendekati manajemen jaringan, dari penjelasan ini dapat di lihat pada gambar 2.5 [29].

2.6.1 Komponen Software SDN

Komponen-komponen kunci dalam SDN melibatkan *Controller*, *Forwarding Plane*, serta *Southbound* dan *Northbound APIs*, membentuk fondasi yang mendukung kontrol terpusat dan fleksibilitas dalam manajemen jaringan. *Controller*, yang dapat diibaratkan sebagai otak SDN, berfungsi sebagai pusat pengendalian yang mengelola lapisan kontrol (*control plane*). Dengan mengolah instruksi dan kebijakan jaringan, *Controller* memberikan kemampuan untuk mengatur dan mengontrol aliran lalu lintas di seluruh jaringan SDN. Di samping itu, *Forwarding Plane*, atau *data plane*, berperan dalam mengimplementasikan

instruksi dari *Controller*. Dengan menangani proses pengalihan paket data berdasarkan kebijakan yang telah ditetapkan, *Forwarding Plane* memastikan pengiriman paket sesuai dengan keputusan yang dibuat oleh *Controller*. Dari penjelasan ini dapat di lihat pada gambar 2.6[30] .



Gambar 2. 6 Komponen SDN [31]

2.7 Floodlight *Controller* SDN

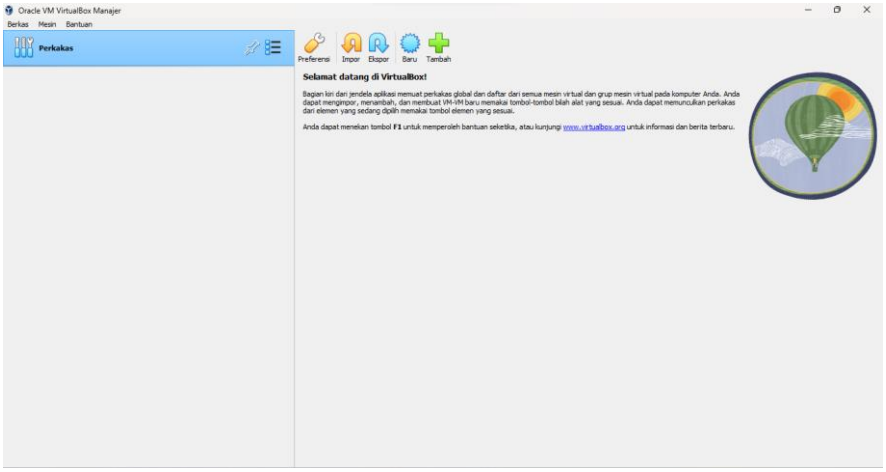
Floodlight, sebagai salah satu pengontrol SDN terkemuka, menandai tonggak penting dalam evolusi jaringan yang ditentukan perangkat lunak SDN. Kisah *Floodlight* dimulai pada tahun 2011, ketika proyek ini pertama kali diperkenalkan oleh Big Switch Networks sebagai proyek sumber terbuka. Sejak saat itu, *Floodlight* telah menjadi salah satu pengontrol SDN yang paling dikenal dan tepercaya di komunitas teknologi. Keunggulan utama *Floodlight* adalah fleksibilitas dan kemudahan penggunaannya. *Floodlight* dirancang sebagai platform terbuka dan dapat disesuaikan yang memungkinkan pengembang dan administrator jaringan berintegrasi dengan mudah dengan berbagai perangkat keras jaringan dan aplikasi SDN. Dukungan untuk berbagai bahasa pemrograman dan arsitektur terbuka memberikan kebebasan kepada pengguna untuk membuat kebijakan jaringan yang sesuai dengan kebutuhan spesifik [32].

2.7.1 Komponen dan Fitur Utama *Floodlight*

Floodlight, sebagai pengontrol terkemuka, menawarkan serangkaian komponen dan fitur yang menjadi landasan keandalannya dalam manajemen jaringan terpusat. Salah satu komponen utama arsitektur *Floodlight* adalah modul kontrol, yang merupakan jantung sistem, yang mengelola kontrol dan logika pengambilan keputusan untuk memandu perangkat keras jaringan. Fitur ini memberikan kontrol terpusat yang memungkinkan kebijakan jaringan diatur secara efisien dan konsisten. *Floodlight* juga menyertakan modul yang bertanggung jawab atas manajemen arus, memastikan distribusi lalu lintas yang adil sesuai dengan kebijakan yang telah ditentukan. Modul ini mencakup kemampuan pemantauan lalu lintas secara terperinci, memungkinkan pengguna untuk memahami dan mengoptimalkan penggunaan sumber daya jaringan [33]

2.8 VirtualBox

VirtualBox merupakan virtualisasi yang dikembangkan oleh *Oracle Corporation*, yang memungkinkan pengguna untuk menjalankan berbagai sistem operasi secara simultan pada satu komputer tanpa perlu *restart* atau instalasi fisik. Sebagai perangkat lunak yang mendukung multipatform, *VirtualBox* bisa berjalan di atas sistem operasi *host* seperti *Windows*, *Linux*, *MacOS*, dan *Solaris*. Fitur kunci dari *VirtualBox* termasuk portabilitas dengan dukungan format *Open Virtualization Format* (OVF) untuk memudahkan migrasi mesin virtual, kemampuan menyimpan snapshots dari kondisi mesin virtual pada waktu tertentu, serta dukungan luas untuk perangkat keras virtual.



Gambar 2. 7 *VirtualBox*

Fitur mode *Seamless* dari *VirtualBox* memungkinkan integrasi yang mulus antara aplikasi sistem operasi tamu dengan desktop sistem operasi *host*. Kelebihan utama *VirtualBox* meliputi statusnya sebagai perangkat lunak gratis dan *open source* di bawah lisensi GPL, dukungan untuk beragam sistem operasi tamu, antarmuka pengguna yang intuitif, dan konfigurasi yang fleksibel. Dari penjelasan ini dapat dilihat pada gambar 2.7 [34]