BAB II
DASAR TEORI

2.1 Kajian Pustaka

Pada tahun 2020, Mohammad Affandi dan Sigit Setyowibowo melaksanakan sebuah penelitian berjudul “Implementasi Snort Sebagai Alat Pendeteksi Intrusi Menggunakan Linux”. Penelitian tersebut fokus pada pengamanan *webserver* dengan menggunakan *Damn Vulnerable Web Application* (DVWA) sebagai *platform* pengujian untuk serangan SQL *Injection*. Dengan demikian, DVWA berfungsi sebagai sarana yang sah bagi para pengembang *web* untuk menguji alat dan meningkatkan pemahaman terhadap keamanan *webserver*. Dalam penelitian ini, berhasil dilakukan pengujian pada IDS Snort yang mampu mendeteksi serangan SQL *Injection* dan akses ke *database*. Snort juga mampu memberikan peringatan atas serangan keamanan, sehingga secara keseluruhan dapat meningkatkan tingkat keamanan jaringan [8].

Pada tahun 2023, Andhika Kurniawan dan Lukman Medriavin Silalahi Siti melakukan penelitian dengan judul "Analisis Keamanan Jaringan Menggunakan *Intrusion Prevention System* (IPS) Dengan Metode *Traffic Behavior*". Penelitian ini menggambarkan implementasi *rules* pada IDS/IPS Suricata 6.0.4 sebagai pendeteksi dan pencegah ancaman dari penyusupan *Port Scanning*, DDoS, dan *Bruteforce* dengan menggunakan metode *traffic behavior*. Penerapan metode perilaku lalu lintas (*traffic behavior*) bertujuan untuk memantau lalu lintas jaringan, mendeteksi aktivitas yang mencurigakan, serta melakukan tindakan pencegahan awal terhadap intrusi atau ancaman terhadap sistem jaringan komputer. Berdasarkan pengujian dan analisis, Suricata berhasil menghasilkan 4 *alert* pemblokiran pada serangan *Port Scanning*, 41 *alert* pemblokiran selama 1 menit pada serangan DDoS, dan 9 *alert* pemblokiran pada serangan *Bruteforce* [9].

Pada tahun 2022, Faula Tanang Anugrah, Syariful Ikhwan, dan Jafaruddin Gusti Amri Ginting melaksanakan penelitian berjudul "Implementasi *Intrusion Prevention System* (IPS) Menggunakan Suricata Untuk Serangan SQL *Injection*". Penelitian ini membahas penggunaan *tool* Suricata untuk melindungi *webserver* dari serangan SQL *Injection*. Serangan tersebut dilakukan dengan menyisipkan

perintah (*query*) SQL berbahaya, yang bertujuan untuk mengakses *database* dan informasi penting lainnya. Suricata akan mendeteksi setiap serangan SQL *Injection* dengan memeriksa aturan tanda tangan (*signature rules*) apakah ada kecocokan atau tidak. Aturan yang terbukti efektif dalam menghadapi serangan SQL *Injection* adalah aturan yang menggunakan beberapa kode ASCII sebagai kata kunci. Pengujian dilakukan dengan mengamati parameter *response time* pada Suricata sebanyak 30 kali, dan Suricata menghasilkan rata-rata waktu tanggapan sebesar 4,2 ms untuk setiap paket [6].

Darryl Santoso, Agustinus Noertjahyana, dan Justinus Andjarwirawan telah melakukan penelitian dengan judul "Implementasi dan Analisa Snort dan Suricata Sebagai IDS dan IPS Untuk Mencegah Serangan DoS dan DDoS". Penelitian ini dilakukan pada tahun 2022 dan bertujuan untuk membandingkan penerapan IDS/IPS menggunakan Snort dan Suricata sebagai *rule base* dalam menghadapi serangan ICMP *flood*. Pengujian dilakukan berdasarkan penggunaan CPU dan waktu yang diperlukan untuk mengatasi serangan tersebut. Hasil penelitian menunjukkan bahwa Suricata mengalami penggunaan CPU yang lebih tinggi dibandingkan dengan Snort, namun Suricata mampu menangani serangan tersebut dalam waktu 11 detik, sementara Snort membutuhkan waktu 37 detik [5].

Pada tahun 2021, Dio Agung Alberiante dan Fatoni melakukan penelitian dengan judul "Pemanfaatan Telegram Dan SMS Sebagai Notifikasi Serangan Untuk Jaringan Di PT.SP2J Menggunakan *Tool Intrusion Detection Sistem*". Penelitian ini secara umum membahas tentang *monitoring* lalu lintas jaringan secara *realtime* dengan memanfaatkan Telegram dan SMS untuk memberikan notifikasi kepada *administrator* jaringan. Sistem ini didesain untuk mendeteksi ancaman serangan yang masuk ke server dan mencatatnya dalam *log* yang disimpan dalam *database*. Dengan menggunakan *script* PHP bernama *Swatchdog*, data *log* tersebut diekstraksi dan dikirimkan kepada *administrator* jaringan melalui Telegram. Pengujian dilakukan untuk mengevaluasi efektivitas pendeteksian dan penanganan keamanan Suricata. Hasil pengujian menunjukkan bahwa Suricata berhasil mendeteksi serangan *Port Scanning* dengan tingkat keamanan 100%, dan *administrator* berhasil menangani 90% dari serangan tersebut. Demikian pula, Suricata berhasil mendeteksi serangan *Bruteforce* dengan tingkat keamanan 100%,

dan *administrator* berhasil menangani 90% dari serangan tersebut. Namun, untuk serangan DoS, Suricata mampu mendeteksi 100% serangan, tetapi *administrator* hanya dapat menangani 20% dari serangan tersebut [10].

Penelitian ini bergantung pada hasil penelitian yang telah dijelaskan dalam subbab 2.1 Kajian Pustaka, yang kemudian disajikan dalam Tabel 2.1 di bawah ini.

Tabel 2.1 Kajian Pustaka Penelitian Terdahulu

No.	Jurnal	Tahun	Keterangan
1	Mohammad Affandi, Sigit Setyowibowo, "Implementasi Snort Sebagai Alat Pendeteksi Intrusi Menggunakan Linus"	2020	Melakukan pengujian keandalan *rule* Snort sebagai sistem keamanan *webserver* dalam menghadapi serangan SQL *Injection* dengan menggunakan sarana pengujian yang sah yaitu *Damn Vulnerable Web Application* (DVWA).
2	Andhika Kurniawan, Lukman Medriavin Silalahi, "Analisis Keamanan Jaringan Menggunakan *Intrusion Prevention System* (IPS) Dengan Metode *Traffic behavior*"	2023	Menerapkan Suricata sebagai sistem *Intrusion Detection System* (IDS) dan *Intrusion Prevention System* (IPS) dengan menggunakan metode *traffic behavior* untuk menghadapi ancaman jaringan seperti *Port Scanning*, DDoS, dan *Bruteforce*.
3	Faula Tanang Anugrah, Syariful Ikhwan, Jafaruddin Gusti Amri Ginting, "Implementasi *Intrusion Prevention System* (IPS)	2022	Suricata berfungsi sebagai *Intrusion Prevention System* (IPS) yang dibangun dengan memanfaatkan *Firewall* IPTables untuk melindungi

No.	Jurnal	Tahun	Keterangan
	Menggunakan Suricata Untuk Serangan SQL *Injection*"		*webserver* dari serangan SQL *Injection*.
4	Darryl Santoso, Agustinus Noertjahyana, Justinus Andjarwirawan, "Implementasi dan Analisa Snort dan Suricata Sebagai IDS dan IPS Untuk Mencegah Serangan DoS dan DDoS"	2022	Menyusun perbandingan antara penerapan IDS/IPS menggunakan Snort dan Suricata sebagai *rule base* untuk menghadapi serangan ICMP *flood* berdasarkan penggunaan CPU.
5	Dio Agung Alberiante, Fatoni, "Pemanfaatan Telegram Dan SMS Sebagai Notifikasi Serangan Untuk Jaringan Di PT.SP2J Menggunakan *Tool Intrusion Detection Sistem*"	2021	Melakukan pemantauan jaringan komputer dengan menggabungkan IDS Suricata, serta mengintegrasikan layanan Telegram dan SMS. Dengan memanfaatkan *topchdog* dan *Log* Suricata, sistem ini akan memberikan laporan berupa notifikasi terkait ancaman jaringan yang diterima oleh server.

2.2 Dasar Teori

2.2.1 Jaringan Komputer

Sebuah rangkaian unit komputer yang dirancang untuk berbagi sumber daya, berkomunikasi, dan mengakses informasi melalui media transmisi kabel dan nirkabel dikenal sebagai jaringan komputer. Untuk memudahkan pemahaman tentang jaringan komputer, para ahli telah mengelompokkan jenisnya berdasarkan cakupan area, termasuklah: [11]:

1. *Personal Area Network* (PAN)

   PAN, yang merupakan singkatan dari *Personal Area Network*, adalah jenis jaringan komputer yang menghubungkan dua atau lebih sistem komputer dalam jarak yang tidak terlalu jauh, biasanya sekitar 4 sampai 6 meter. Salah satu

contohnya adalah ketika mengaplikasikan PAN untuk menghubungkan printer dengan *smartphone* melalui teknologi *bluetooth*.

2. *Local Area Network* (LAN)

   Jaringan LAN memiliki cakupan yang relatif pendek, menghubungkan perangkat dalam jarak yang terbatas. Biasanya, jenis jaringan ini dipakai di area-area seperti gedung sekolah, kantor, dan rumah dengan instalasi baik melalui kabel atau teknologi nirkabel. Kecepatan transfer data pada jaringan LAN berkisar antara 10 hingga 100 Mbps (*Megabit per second*).

3. *Metropolitan Area Network* (MAN)

   Jaringan komputer ini sebenarnya merupakan kombinasi dari beberapa jaringan LAN yang ada di suatu kota, dengan jarak mencapai 10 hingga 50 kilometer. Jaringan MAN sering digunakan untuk menghubungkan gedung-gedung perusahaan atau sekolah. Kecepatan transfer data yang dimiliki oleh jaringan MAN mencapai 150 Mbps.

4. *Wide Area Network* (WAN)

   Jaringan komputer WAN mencakup area sangat luas, menghubungkan lokasi yang berjarak pulau, negara, bahkan antar benua dengan jarak 100 hingga 1000 kilometer. Didesain untuk efisien mengatasi jarak yang besar, WAN memungkinkan komunikasi dan pertukaran data antara lokasi yang sangat terpisah.

2.2.2 Keamanan Jaringan

Keamanan jaringan memiliki peran penting dalam melindungi aset-aset yang berharga dari ancaman dan serangan yang terus berkembang dalam lingkungan digital yang semakin kompleks. Dalam upaya menjaga keamanan, keandalan, dan integritas jaringan, perlu dilakukan penerapan prinsip dasar keamanan informasi yang dikenal sebagai CIA *Triad*. Prinsip ini bertujuan untuk memberikan panduan bagi individu dalam mengembangkan aplikasi, prosedur, atau kebijakan yang terkait dengan keamanan informasi, sehingga dapat menghindari berbagai macam serangan yang ada.

Pertama, kerahasiaan (*Confidentiality*) menjadi aspek penting dalam menjaga keamanan jaringan. Prinsip kerahasiaan ini menjamin bahwa informasi

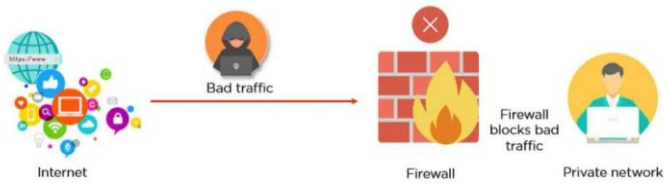
dan data penting hanya dapat diakses oleh individu yang berwenang atau memiliki hak akses. Langkah-langkah keamanan, seperti enkripsi data dan sistem pengaturan akses, diimplementasikan untuk mencegah akses oleh pihak yang tidak berwenang, sehingga informasi penting tetap terjaga dari eksploitasi oleh pihak yang tidak bertanggung jawab.

Kedua, integritas (*Integrity*) menjadi faktor krusial untuk memastikan bahwa data dan informasi dalam jaringan tetap utuh dan tidak mengalami manipulasi oleh pihak yang tidak sah. Prinsip integritas ini menjamin bahwa data yang dikirim dan diterima melalui jaringan tidak mengalami perubahan yang tidak sah selama proses transmisi. Dengan menerapkan teknologi hash dan tanda tangan digital, sistem jaringan dapat memverifikasi integritas data dan memastikan bahwa data yang diterima benar-benar berasal dari sumber yang sah dan tidak mengalami perubahan.

Ketiga, ketersediaan (*Availability*) menjadi hal yang krusial dalam menghadapi ancaman jaringan. Prinsip ketersediaan ini menjamin bahwa informasi dan layanan jaringan selalu tersedia dan dapat diakses oleh pengguna yang berwenang ketika dibutuhkan. Dalam upaya memastikan ketersediaan ini, langkah-langkah *redundancy* dan peningkatan kapasitas server diimplementasikan agar jaringan dapat terus beroperasi bahkan ketika terjadi serangan atau gangguan [12].

2.2.3 *Firewall*

Firewall merupakan salah satu alat yang digunakan untuk menjaga keamanan jaringan komputer. Tugas utamanya adalah mengatur dan mengawasi lalu lintas data yang masuk dan keluar dari jaringan berdasarkan sejumlah aturan yang telah ditetapkan seperti yang ditujukkan Gambar 2.1. Prinsip kerjanya adalah memberikan izin kepada paket data yang dianggap sah dan diperbolehkan untuk melewati jaringan. Di sisi lain, paket data yang dianggap berbahaya atau tidak diinginkan akan ditolak, mencegahnya untuk melewati jaringan tersebut.

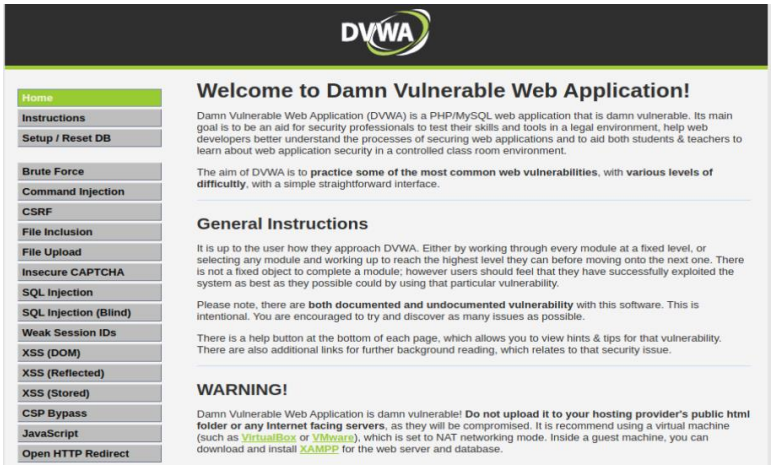


Gambar 2.1 Alur Kerja *Firewall* [13]

Dengan demikian, *Firewall* berperan sebagai penghalang yang efektif dalam melindungi jaringan dari potensi ancaman dan memastikan keamanan data yang terdapat di dalamnya. Penting untuk diingat bahwa *Firewall* tidak hanya melibatkan perlindungan dari ancaman luar, tetapi juga mampu mengontrol lalu lintas yang keluar dari jaringan. Hal ini membantu dalam mengendalikan akses ke data sensitif dan menjaga integritas informasi yang dikirimkan dan diterima oleh jaringan [13].

2.2.4 DVWA

Damn Vulnerable Web Application (DVWA) merupakan sebuah aplikasi yang sengaja dibuat dengan kerentanan yang dapat dieksplorasi, dirancang khusus untuk tujuan uji penetrasi terhadap keamanan. Aplikasi ini menggunakan layanan *webserver* Apache dan berjalan melalui protokol HTTP. Sasaran utamanya adalah menjadi alat yang berharga bagi individu yang sedang memulai perjalanan dalam dunia keamanan siber, maupun bagi para profesional yang ingin mengasah dan menguji kembali keterampilan mereka dalam sebuah lingkungan yang sah secara hukum. Selain memberikan manfaat bagi pengembang *web* dalam memahami lebih lanjut tentang proses keamanan aplikasi *web*, DVWA juga berperan penting dalam mendukung pemahaman konsep ini di lingkungan pendidikan. Aplikasi ini dapat digunakan oleh para pendidik sebagai alat bantu untuk membantu siswa dan mahasiswa memahami konsep keamanan aplikasi *web* dalam suasana yang terkontrol di dalam ruang kelas. Tampilan halaman DVWA divisualisasikan pada Gambar 2.2



Gambar 2.2 Halaman Utama DVWA

DVWA menawarkan kesempatan dalam serangan *Bruteforce*, di mana pengguna dapat menguji keamanan autentikasi dengan mencoba berbagai kata sandi, serta serangan SQL *Injection*, di mana pengguna dapat memasukkan kode berbahaya melalui input untuk mengakses atau memanipulasi data basis data [14].

2.2.5 *Intrusion Detection System* (IDS)

IDS adalah suatu sistem yang memiliki kemampuan untuk mendeteksi kejanggalan atau serangan yang terjadi pada jaringan komputer, baik itu jaringan lokal maupun jaringan internet. Dalam hal terjadi aktivitas yang mencurigakan, sistem ini secara *realtime* memberikan peringatan kepada *administrator* jaringan yang bertanggung jawab. Dengan kata lain, IDS menggunakan sensor yang dapat dipercaya untuk memastikan deteksi yang efektif dalam melindungi jaringan komputer. Jenis IDS dapat diklasifikasikan sebagai berikut [15]:

1. *Network-based Intrusion Detection System* (NIDS)
   Jenis NIDS memiliki fungsi untuk memantau anomali di seluruh jaringan dan mendeteksi aktivitas mencurigakan pada semua *host* yang ada dalam jaringan tersebut.
2. *Host-based Intrusion Detection System* (HIDS)
   HIDS berfokus pada pemantauan aktivitas pada *host* jaringan individual, dengan tujuan mendeteksi upaya serangan atau penyusupan ke *host* tersebut.

2.2.6 *Intrusion Prevention System* (IPS)

IPS adalah sebuah sistem yang berperan dalam pemantauan lalu lintas jaringan, mendeteksi aktivitas mencurigakan, serta mengambil langkah-langkah pencegahan awal terhadap upaya penyusupan atau aktivitas yang dapat mengganggu kinerja jaringan. Secara dasar, IPS merupakan perkembangan dari metode IDS, dengan kemampuan untuk mengambil tindakan berdasarkan data pengklasifikasian dari serangan tanpa harus menunggu tindakan dari *administrator* jaringan. Hal ini memungkinkan IPS untuk mengenali kriteria-kriteria serangan yang umumnya terjadi dan bertindak seperti *Firewall* dengan mengizinkan atau memblokir paket data yang masuk [16]. Dalam penggolongannya, IPS dibagi menjadi dua jenis, yakni:

1. *Network-based Intrusion Prevention System* (NIPS)

   Jenis NIPS mampu memantau dan memberikan perlindungan pencegahan intrusi dalam satu jaringan secara menyeluruh. NIPS didesain untuk menganalisis, mendeteksi, dan melaporkan seluruh lalu lintas jaringan, sehingga aktivitas mencurigakan dapat segera terdeteksi dan diblokir.

2. *Host-based Intrusion Prevention System* (HIPS)

   Berbeda dengan NIPS, HIPS hanya fokus pada pemantauan dan pencegahan gangguan jaringan pada satu *host*. Dari perspektif keamanan, HIPS lebih mungkin untuk mencegah ancaman terhadap *host*, namun, dari sudut pandang kinerja, hal ini dapat memiliki dampak negatif karena memerlukan penggunaan sumber daya yang lebih besar.

2.2.7 IPTables

IPTables adalah sebuah alat yang berfungsi sebagai *Firewall* untuk melakukan pengaturan filter terhadap lalu lintas data di dalam jaringan komputer. Dengan fungsi utam*any*a sebagai pengawas lalu lintas data masuk dan keluar, IPTables memiliki keunggulan dalam pembuatan aturan-aturan yang detail untuk mengatur perizinan atau pemblokiran data dalam jaringan. Aturan-aturan ini menjadi panduan penting dalam menjaga keamanan jaringan dan menciptakan lapisan pertahanan awal yang kuat.

IPTables memberikan fleksibilitas bagi *administrator* untuk menetapkan parameter lalu lintas data yang diinginkan. Pengaturan aturan dapat meliputi pembatasan jumlah data yang diizinkan dalam periode waktu tertentu, jenis paket atau datagram yang diijinkan untuk berkomunikasi dalam dan keluar jaringan, serta pengaturan sumber dan tujuan data yang boleh atau tidak boleh berkomunikasi. Selain itu, alat ini juga memungkinkan *administrator* untuk mengelola akses ke layanan dan aplikasi tertentu melalui pengaturan *port*. Kelebihan ataupun keunggulan IPTables juga terletak pada kemampuannya dalam menghadapi berbagai protokol jaringan seperti *Transmission Control Protocol* (TCP), *User Data Protocol* (UDP), dan *Internet Control Message Protocol* (ICMP) sehingga dapat memberikan tingkat keamanan yang jauh lebih tinggi dan sesuai dengan kebutuhan khusus di lingkungan jaringan [13].

2.2.8 Suricata

Suricata merupakan salah satu aplikasi IDS dan IPS yang sangat berfungsi. Sebagai IDS, Suricata memiliki kemampuan untuk memantau seluruh lalu lintas data dalam jaringan dan memberi notifikasi kepada *administrator* jaringan ketika mendeteksi ancaman potensial. Apabila dikonfigurasi sebagai IPS, Suricata dapat melakukan pemantauan lalu lintas data dan bertindak secara proaktif dengan mencegah lalu lintas berbahaya agar tidak masuk ke dalam jaringan, dan memberi tahu *administrator* tentang upaya pemblokiran tersebut.

Suricata adalah perangkat lunak *open source* yang dikembangkan oleh *Open Information Security Foundation* (OISF). Organisasi ini telah memastikan bahwa Suricata memiliki performa yang sangat baik, terutama dalam kemampuan *multi-threaded*, sehingga dapat memanfaatkan banyak *core* pada saat bersamaan. Fitur ini memungkinkan Suricata untuk menangani banyak kejadian secara bersamaan tanpa mengganggu kinerja sistem atau mempengaruhi permintaan lain. Kemampuan *multi-threading* juga memungkinkan Suricata untuk mendistribusikan beban kerja secara merata di seluruh CPU, sehingga meningkatkan kinerja keseluruhan dalam menganalisis lalu lintas jaringan. Hal ini memungkinkan Suricata untuk memproses lalu lintas data dalam jumlah besar tanpa mengorbankan jumlah peraturan (*rules*) yang dapat digunakan. [17].

2.2.9 *Bruteforce*

Serangan *Bruteforce* adalah metode yang digunakan untuk mencoba meretas *password* dengan cara memaksa dengan mencoba semua kemungkinan kombinasi karakter dan panjang *password* tertentu. Namun, proses meretas *password* dengan metode ini memerlukan waktu yang cukup lama tergantung pada panjang dan kompleksitas kombinasi karakter yang ingin dipecahkan. Tujuan dari serangan *Bruteforce* adalah untuk mendapatkan akses ilegal ke *host* atau data yang terenkripsi. Oleh karena itu, sangat penting untuk menghindari penggunaan *password* yang lemah, seperti kata-kata umum, kombinasi karakter yang pendek, atau informasi pribadi seperti nama atau nomor telepon, karena hal ini sangat tidak aman. Proses serangan *Bruteforce* dapat memakan waktu berbulan-bulan atau bahkan bertahun-tahun, tergantung pada tingkat kompleksitas kode yang

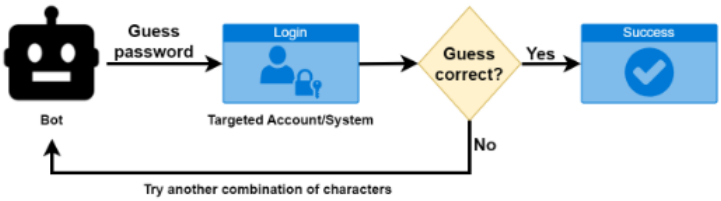
digunakan. Serangan *Bruteforce* memiliki berbagai metode yang dibagi menjadi beberapa bagian, antara lain [18] :

1. *Simple Attack*

   Serangan ini mencoba menebak *password* secara logis dan sistematis tanpa menggunakan bantuan *software* atau alat apapun. Biasanya, *simple attack* hanya digunakan untuk meretas *password* atau PIN yang sederhana.

2. *Dictionary Attack*s

   Serangan *dictionary* menggunakan kamus atau *wordlist* yang berisi koleksi *password* yang akan digunakan dalam upaya meretas keamanan. Penyerang mencoba semua kemungkinan kombinasi karakter secara berurutan untuk menemukan kata sandi yang benar seperti visualisasi gambar 2.3. Ini berarti mencoba semua kemungkinan dari satu karakter, dua karakter, tiga karakter, dan seterusnya hingga kata sandi ditemukan. Berikut gambar alur dari *dictionary attack*.



Gambar 2.3 Alur *Dictionary Attack* [18]

3. *Hybrid Attack*s

   Serangan *Hybrid Bruteforce* adalah kombinasi dari pendekatan serangan kamus (*dictionary attack*) dan serangan *Bruteforce* sederhana (*simple Bruteforce attack*). Pada jenis serangan ini, pendekatan awal dilakukan dengan menggunakan koleksi kata sandi dari kamus, kemudian dilanjutkan dengan mencoba variasi lebih lanjut dengan menambahkan angka atau mengubah ukuran huruf (huruf besar dan huruf kecil).

4. *Reverse Attack*s:

   Strategi serangan ini membalikkan pendekatan dengan memulai dari *password* yang sudah diketahui. Penyerang mencoba mencocokkan *password* tersebut dengan jutaan nama pengguna yang ada. Banyak pelaku kejahatan memanfaatkan *password* yang bocor dan tersedia secara *online* melalui pelanggaran data.

5. *Credential Stuffing*

   Jika seorang penyerang berhasil mendapatkan kombinasi nama pengguna dan kata sandi yang berhasil digunakan di satu situs *web*, mereka kemungkinan akan mencoba kombinasi tersebut di banyak situs lainnya.

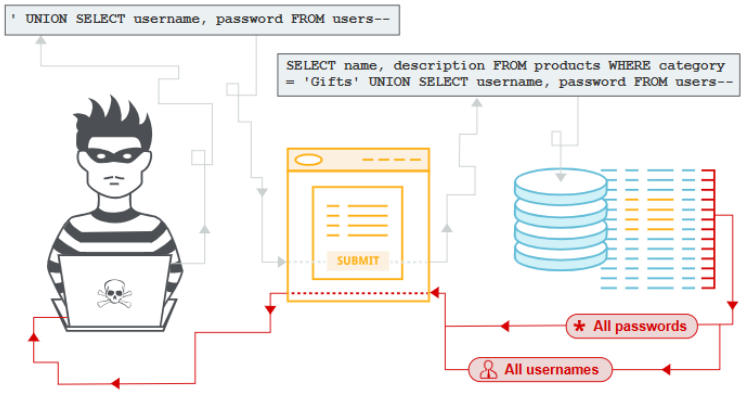
Meskipun terdapat beragam metode *Bruteforce* yang tersedia, dalam penelitian ini, peneliti memilih untuk menggunakan metode serangan kamus (*dictionary attack*), sehingga memerlukan sebuah daftar kata (*wordlist*).

2.2.10 Hydra

Hydra adalah sebuah perangkat lunak yang dikembangkan oleh organisasi yang dikenal sebagai "*The Hacker's Choice*" (THC). Fungsinya adalah untuk menguji kelemahan pada *password* dengan menggunakan metode *Bruteforce* dan serangan kamus (*dictionary attack*) pada satu atau beberapa *host* remote yang menjalankan berbagai layanan yang berbeda. Tujuan utama dari pengembangan Hydra adalah untuk membuktikan seberapa mudahnya mengakses *password* yang lemah atau sederhana. Perangkat lunak ini dirancang untuk mendukung berbagai layanan dan protokol, termasuk FTP, HTTP, HTTPS, MS-SQL, MySQL, dan banyak lagi. Dengan berbagai fitur yang dimiliki, Hydra dapat menjadi alat yang berpotensi kuat dalam menguji keamanan *password* dan meningkatkan kesadaran tentang pentingnya menggunakan *password* yang kuat dan aman [19].

2.2.11 SQL *Injection*

Serangan SQL *Injection* adalah jenis serangan yang dilakukan oleh para penyerang dengan tujuan mengeksploitasi celah keamanan pada *database* suatu *website*. Serangan ini memanfaatkan kelemahan dalam *input data* yang dikirimkan ke aplikasi, yang kemudian memungkinkan penyerang untuk memanipulasi *Structured Query Language* (SQL) *query* yang dijalankan oleh aplikasi tersebut pada *database*. Dengan keahlian ini, penyerang dapat mengubah sintaks SQL, mempengaruhi kekuatan dan fleksibilitas *database*, dan bahkan memanipulasi fungsi sistem operasi yang terhubung dengan *database*. Selain berdampak pada aplikasi *web*, serangan SQL *Injection* juga dapat mempengaruhi program lain yang menggunakan pernyataan SQL.



Gambar 2.4 Serangan SQL *Injection* [20]

Gambar 2.4 merupakan tingkat bahaya dari SQL *Injection* sangat besar karena ketika penyerang berhasil mendapatkan akses ke *database* sistem, mereka dapat melakukan pencurian data atau melakukan manipulasi terhadap informasi yang ada dalam *database* tersebut. Oleh karena itu, penting bagi para pengembang dan *administrator web* untuk meningkatkan keamanan aplikasi dan *database* guna melindungi data dan menghindari celah-celah yang dapat dimanfaatkan oleh serangan SQL *Injection* [21].

2.2.12 SQLMap

SQLMap adalah sebuah perangkat lunak sumber terbuka yang khusus dirancang untuk mendeteksi dan mengeksploitasi kerentanan pada aplikasi *web* yang menggunakan *database* SQL. Perangkat ini memiliki beberapa fungsi penting dalam melakukan serangan dan pengujian keamanan:

1. Deteksi Kerentanan

   SQLMap dapat melakukan pemindaian pada aplikasi *web* untuk mencari kerentanan terkait SQL *Injection*.

2. Eksploitasi Kerentanan

   Setelah mendeteksi kerentanan, SQLMap memanfaatkannya dengan melakukan serangan SQL *Injection* pada aplikasi *web*. Ini memungkinkan penyerang untuk menjalankan perintah pada *database* SQL yang mendasari dan memanipulasi data serta struktur *database*.

3. Pengambilalihan Server *Database*

   SQLMap memiliki kemampuan untuk mengambil alih kontrol atas server *database*. Penyerang dapat mengakses data, menjalankan perintah pada sistem

operasi yang menjalankan server *database*, dan bahkan mengakses sistem *file* dari server tersebut.

4. Pemetaan Struktur *Database*

   SQLMap dapat digunakan untuk mengambil detail tentang struktur *database* yang digunakan oleh aplikasi *web* target. Ini memberikan informasi berharga kepada penyerang, memungkinkan mereka untuk mengeksplorasi dan memahami lebih lanjut tentang data yang ada dalam *database* target.

5. Manipulasi Data

   Dengan SQLMap, penyerang dapat melihat, mengubah, atau bahkan menghapus data dalam *database* target. Hal ini memberikan kemampuan untuk merusak integritas data atau bahkan mencuri informasi sensitif.

Secara keseluruhan, SQLMap merupakan alat yang kuat untuk menguji dan menguji keamanan aplikasi *web* yang menggunakan *database* SQL. Namun, penting untuk selalu menggunakan SQLMap dengan izin pemilik sistem dan dalam konteks pengujian keamanan yang sah. [22].

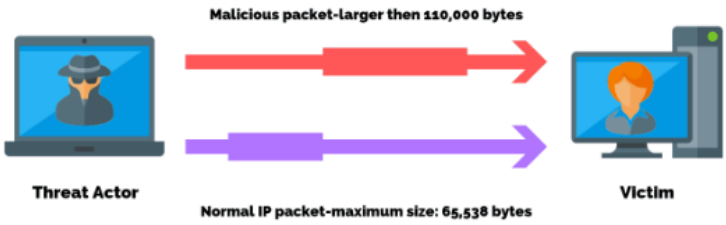
2.2.13 *Denial of Service* (DoS)

Serangan DoS (*Denial of Service*) bertujuan untuk menghambat akses pengguna terhadap layanan dari sistem yang menjadi target. Upaya ini dapat dilakukan dengan cara membanjiri jaringan dengan lalu lintas data yang sangat besar atau dengan sengaja menghabiskan sumber daya terbatas yang dimiliki oleh sistem tersebut. Serangan DoS beroperasi secara satu lawan satu, di mana hanya diperlukan satu komputer atau *host* yang memiliki kekuatan yang cukup, baik dari segi perangkat keras, sistem operasi, maupun aplikasi, untuk membanjiri lalu lintas ke *host* korban, sehingga menghalangi pengguna yang sah dan berhak untuk mengakses server.

Sebagai contoh, serangan DoS dapat terjadi pada *webserver*, yang bertugas menyajikan informasi dalam bentuk halaman HTML kepada pengunjung *web*. Dalam kondisi normal, pengunjung dapat meminta sumber daya dari *webserver* dan menampilkan konten di peramban mereka. Namun, jika *webserver* terkena serangan DoS, para pengunjung tidak dapat menikmati layanan dari *webserver* karena ketersediaan sumber daya terbatas telah habis terkuras oleh

serangan. Serangan DoS dapat menyebabkan gangguan kinerja sistem, menurunkan kualitas layanan, dan bahkan menyebabkan sistem gagal berfungsi sepenuhnya. Oleh karena itu, penting untuk menerapkan langkah-langkah pencegahan dan mitigasi yang tepat guna melindungi sistem dari serangan DoS.

Untuk mencapai tujuan serangan DoS dalam membuat komputer target menjadi tidak responsif dan pada akhirnya menyebabkan kegagalan sistem, para penyerang membutuhkan sumber daya yang signifikan. Beberapa sumber daya yang terkuras dalam serangan DoS meliputi *bandwidth*, RAM, dan *disk*. Para penyerang berusaha memanfaatkan sebanyak mungkin sumber daya ini untuk memberatkan komputer target dan mengganggu kinerja sistem secara keseluruhan. Dengan menggunakan taktik ini, para penyerang berharap dapat mengakibatkan komputer tersebut menjadi tidak berfungsi atau merespons dengan lambat. Karenanya, penting untuk mengimplementasikan mekanisme perlindungan dan pengawasan yang efektif agar sistem terlindungi dari serangan DoS.



Gambar 2.5 Serangan ICMP *Flood* [23]

Ada beberapa jenis serangan DoS, seperti *Ping of Death* (ICMP *flood*), *Teardrop*, SYN *attack*, *Land attack*, *Smurf attack*, dan UDP *flood*. Dalam penelitian ini, fokus akan ditujukan pada serangan *Internet Control Message Protocol* (ICMP) *Flood*. Serangan ini menggunakan utilitas ping yang ada dalam sistem operasi komputer. *Ping* digunakan untuk menguji waktu yang diperlukan untuk mengirimkan data tertentu dari satu komputer ke komputer lainnya. Meskipun panjang maksimum data menurut protokol TCP/IP adalah 65.538 *byte*, serangan ini mengirimkan data yang melebihi batas tersebut, sekitar 110.000 *byte*, hal ini divisualisasikan seperti Gambar 2.5 [23].

2.2.14 Hping3

Hping3 adalah sebuah alat utilitas jaringan yang memiliki kemampuan untuk mengirimkan paket ICMP/UDP/TCP yang dapat disesuaikan, dan

menampilkan balasan dari target, mirip dengan perintah ping yang mendapatkan balasan ICMP. Alat ini mampu mengatasi fragmentasi dan mengontrol isi serta ukuran paket sesuai kebutuhan. Selain itu, hping3 juga dapat digunakan untuk mentransfer *file* melalui protokol yang didukung. Dengan menggunakan hping3, pengguna dapat melakukan berbagai fungsi, termasuk pengujian aturan *Firewall*, pemindaian *port* dengan teknik *spoofed*, pengujian kinerja jaringan dengan berbagai protokol, identifikasi sistem operasi pada target yang berjauhan, melakukan audit pada tumpukan protokol TCP/IP, dan masih banyak lagi. Alat ini memberikan fleksibilitas dalam melakukan berbagai jenis uji coba dan analisis pada jaringan serta sistem yang menjadi targetnya [24].

2.2.15 Httperf

Httperf merupakan sebuah alat yang digunakan untuk melakukan pengukuran kinerja *webserver*. Alat ini dirancang dan dikembangkan oleh David Mosbeger dari HP Labs. Httperf menawarkan fitur yang sangat fleksibel dalam pembuatan beban kerja yang dapat disesuaikan dengan berbagai variabel yang diberikan. Tujuan utama dari Httperf adalah untuk menghasilkan sejumlah permintaan HTTP GET yang telah ditentukan sebelumnya, sehingga memungkinkan evaluasi nilai *response time* dari server. Dengan menggunakan Httperf, pengguna dapat menganalisis performa *webserver* dengan lebih efektif dan mendapatkan wawasan yang berguna mengenai kecepatan dan responsivitas server tersebut. Alat ini memberikan kemampuan bagi pengguna untuk secara akurat mengukur dan memahami bagaimana *webserver* menangani beban kerja tertentu, sehingga dapat digunakan sebagai referensi untuk melakukan perbaikan dan peningkatan performa jika diperlukan [25].