BAB 2
DASAR TEORI

2.1 KAJIAN PUSTAKA

Penelitian yang dilakukan oleh Dian Novianto, Yohanes Setiawan Japriadi, dan Lukas Tommy yang membahas mengenai Implementasi Keamanan Akses Terhadap *Website* Menggunakan WireGuard VPN di *Routerboard* Mikrotik. Pada penelitian ini, melakukan implementasi WireGuard VPN menggunakan metode *remote access*. Hasil dari penelitian ini disimpulkan bahwa, Keunggulan dari VPN WireGuard terletak pada kemudahan implementasinya serta kemampuan yang baik dalam melakukan enkripsi menggunakan kunci yang ada pada kedua sisi (*peer*). Selain itu, mekanisme kombinasi antara authentikasi dan VPN WireGuard telah terbukti efektif dalam menyembunyikan isi komunikasi di protokol HTTP dalam sistem [2].

Penelitian yang dilakukan oleh Muhammad Iqbal dan Imam Riadi membahas mengenai *Analysis of Security Virtual Private Network* (VPN) *Using* OpenVPN. Penelitian ini memaparkan upaya pembuatan *Virtual Private Network* dengan menggunakan rancangan jaringan OpenVPN di laboratorium penelitian Informatika Universitas Ahmad Dahlan dan seberapa besar tingkat keamanannya. Hasil dari implementasi *Virtual Private Network* menggunakan OpenVPN adalah percobaan tersebut memberikan hasil yang positif, hal ini dibuktikan dengan adanya *sniffing* data yang tidak dapat mendeteksi *user*name dan *password* yang dikirimkan. Hasil pengukuran *Quality of Service* mengalami penurunan kualitas jaringan dengan parameter *delay* naik dari 51,4 ms menjadi 463,4 ms, *packet loss* naik dari 7,8% menjadi 20,2%, *throughput* turun dari 82,8% menjadi 71,6%, dan *bandwidth* turun dari 64786,6 bit/s menjadi 55589 bit/s, hal ini dikarenakan proses enkripsi dan enkapsulasi yang memakan waktu [7].

Penelitian yang dilakukan oleh Dina Olivia dan Zulhendra membahas mengenai Penelitian ini menganalisis *Quality of Service* (QoS) pada Jaringan *Virtual Private Network* (VPN) yang menggunakan protokol IPSec, dengan mengambil studi kasus di SMK Negeri 3 Pariaman. Pendekatan yang digunakan

dalam penelitian ini adalah tatacara kuantitatif dengan jenis penelitian deskriptif, yang melibatkan analisis statistik. Untuk mengukur parameter QoS seperti *delay*, *packet loss*, *throughput*, *jitter*, dan *bandwidth*, digunakan aplikasi axence nettols. Hasil pengukuran tersebut kemudian dibandingkan dengan standar TIPHON. Hasil analisis data menunjukkan beberapa temuan penting sebagai berikut: Pertama, kualitas layanan jaringan VPN di SMK Negeri 3 Pariaman termasuk dalam kategori yang berkualitas baik, karena nilai rata-rata parameter QoS menunjukkan hasil yang memuaskan. Kedua, nilai pengukuran *delay* dengan standar TIPHON menunjukkan kualitas yang sangat bagus, dengan rata-rata 34,7 ms pada pagi hari dan 32,9 ms pada sore hari. Ketiga, nilai pengukuran *throughput* dengan standar TIPHON juga menunjukkan kualitas yang bagus, dengan rata-rata 59% pada pagi hari dan 75% pada sore hari. Keempat, nilai pengukuran *packet loss* dengan standar TIPHON menunjukkan kualitas yang sangat bagus, dengan rata-rata 2,5% pada pagi hari dan 1,6% pada sore hari. Terakhir, nilai pengukuran *jitter* dengan standar TIPHON juga menunjukkan kualitas yang sangat bagus, dengan rata-rata 0,007 ms pada pagi hari dan sore hari [8].

Penelitian yang dilakukan oleh Raka Nugraha Pangestu, Herlina Harahap, dan Rismayanti membahas mengenai Implemetasi Keamanan Jaringan Berbasis VPN dan Anti-DDoS dalam Melindungi Server Linux dari Serangan Hammer. Penelitian ini mencakup pengujian keamanan pada server Linux Mint tanpa menggunakan VPN dan Anti-DDoS, kemudian mengimplementasikan langkah-langkah keamanan jaringan untuk melindungi server Linux Mint dari serangan Hammer dengan memanfaatkan VPN dan Anti-DDoS. Selain itu, penelitian ini juga melakukan pengujian serangan DDoS dan VPN pada platform Linux Mint. Hasil dari pengujian menunjukkan bahwa penerapan Anti-DDoS dan VPN pada sistem Linux Mint berhasil dilaksanakan. Melalui metode ini, Anti-DDoS dan VPN telah terbukti memiliki kemampuan untuk mencegah serangan DDoS pada server Linux dengan cara memblokir akses dari IP yang tidak dikenal dan mengalihkan serangan ke virtual IP yang disediakan oleh VPN. Dengan langkah-langkah ini, keamanan dan ketahanan server Linux Mint terhadap serangan Hammer yang berpotensi merugikan dapat meningkat secara signifikan [9].

Penelitian Arief Indriarto Haris, Budhi Riyanto, Ferry Surachman, dan Ardio Adi Ramadhan membahas tentang Analisis Pengamanan Jaringan Menggunakan *Router* Mikrotik dari Serangan DoS dan Pengaruhnya Terhadap Performansi. Penelitian ini menggunakan pendekatan metodologi PPDIOO dengan tujuan menganalisis tingkat keamanan jaringan dari serangan DoS menggunakan *router* MikroTik dan mengevaluasi efektivitas fitur-fitur keamanan bawaannya. Pengujian serangan DoS dilakukan pada enam kondisi *router*, dan pada setiap kondisi dilakukan pengukuran terhadap lima indikator. Hasil penelitian menunjukkan bahwa penerapan *firewall* raw pada kondisi 6 adalah yang paling efektif dan efisien dibandingkan dengan kondisi lainnya. Penggunaan CPU berhasil diturunkan hingga 20%, dan ping *response time* kembali normal, serta proses deteksi dan blokir serangan berjalan secara otomatis. Meskipun demikian, secara keseluruhan, pengamanan menggunakan fitur-fitur keamanan bawaan dinilai tidak efektif dalam menghadapi serangan DoS. Hal ini dapat dilihat dari tingginya konsumsi CPU yang belum mencapai level normal, dan traffic DoS yang tidak dapat sepenuhnya dihilangkan, hanya latensi saja yang dapat dinormalisasi [10].

2.2 *VIRTUAL PRIVATE NETWORK* (VPN)

VPN adalah suatu metode yang aman untuk mengakses jaringan lokal (*local area network*) dari jarak jauh dengan menggunakan internet atau jaringan umum lainnya sebagai medium untuk melakukan transmisi data paket secara pribadi [11]. VPN memberikan manfaat berupa penghematan biaya dan kemampuan untuk mentransfer data atau mengendalikan komputer di rumah atau kantor dari lokasi manapun. VPN juga dapat menjadi alternatif teknologi yang murah untuk menghubungkan jaringan lokal yang luas karena menggunakan jaringan publik yang sudah ada, menghindari kebutuhan untuk membangun jaringan pribadi. Namun, kelemahan dari VPN adalah tergantung pada jaringan publik (internet) yang tidak dapat diprediksi. Karena VPN menggunakan koneksi dari pihak lain, pengguna tidak memiliki kendali penuh terhadap jaringan tersebut [12].

2.2.1 Fungsi VPN

Teknologi VPN menawarkan berbagai fungsi utama untuk penggunaannya, di antaranya adalah sebagai berikut [13]:

1. *Confidentially* (Kerahasiaan)

   VPN adalah teknologi yang menggunakan jaringan internet atau jaringan publik yang memiliki potensi rawan terhadap pencurian informasi atau data. Oleh karena itu, VPN menggunakan metode enkripsi untuk mengacak data yang melewati jaringan. Dengan penggunaan metode enkripsi ini, keamanan data menjadi lebih terjamin dari pencurian data. Meskipun ada kemungkinan bagi pihak-pihak tertentu untuk menyadap data yang melewati jaringan internet atau jalur VPN itu sendiri, namun data yang disadap tersebut tidak dapat dibaca karena telah diacak sebelumnya. Fungsi *confidentially* dalam VPN bertujuan untuk memastikan bahwa data yang ditransmisikan hanya dapat diakses oleh orang yang memiliki hak akses yang sah. Dengan demikian, data tersebut tetap terlindungi dari akses yang tidak sah.

2. Data *Intergrity* (Keutuhan data)

   Teknologi VPN memiliki kemampuan untuk menjaga keutuhan informasi atau data sejak data dikirim hingga sampai ke tujuan akhir. Dengan demikian, data yang sedang dalam perjalanan dapat terhindar dari berbagai gangguan seperti hilangnya data, kerusakan data, atau manipulasi data oleh pihak-pihak yang tidak bertanggung jawab.

3. *Origin Authentication* (Autentikasi sumber)

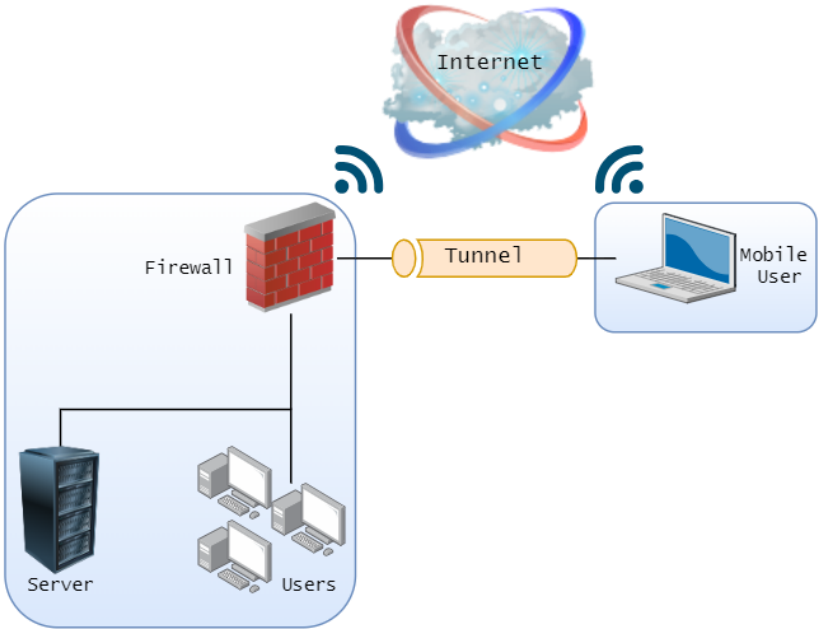
   VPN memiliki kemampuan untuk melakukan autentikasi terhadap sumber pengiriman data sebelum data diterima. Dengan demikian, VPN dapat memeriksa dan mengakses informasi tentang sumber data yang masuk. Jika proses autentikasi berhasil, alamat sumber data akan disetujui. Hal ini memastikan bahwa VPN dapat menjamin semua data yang dikirimkan dan diterima berasal dari sumber yang sah dan benar-benar seharusnya. Dengan adanya autentikasi ini, tidak ada informasi atau data yang dapat dikirimkan oleh pihak lain atau data yang dipalsukan yang dapat lolos dan diterima oleh VPN.

2.2.2 Jenis-jenis VPN

Seiring berjalannya waktu, teknologi VPN berkembang menjadi beberapa jenis, yaitu [14]:

1. *Remote Access* VPN

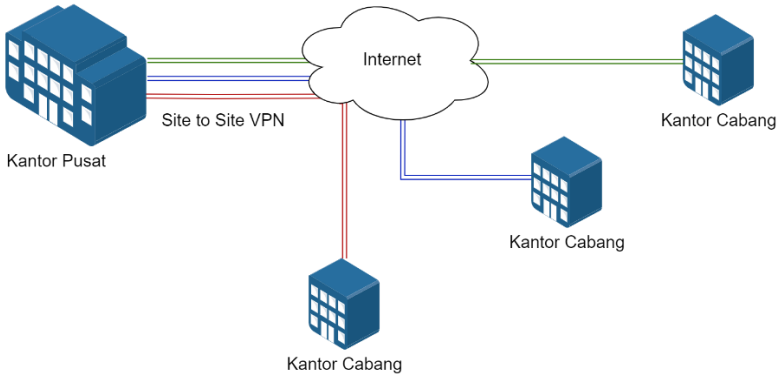
Tipe VPN ini memungkinkan pegawai yang sedang bekerja di luar kantor atau di luar negara untuk mengakses jaringan LAN di kantor pusat melalui internet. Keunggulan ini terutama bermanfaat untuk menerima *email* yang tersedia di jaringan LAN kantor pusat. Selain itu, manfaat serupa juga berlaku untuk kantor cabang yang tidak memiliki koneksi langsung ke kantor pusat. Kantor cabang tersebut dapat melakukan koneksi lokal melalui layanan *dial-up* ke sebuah ISP dan selanjutnya terhubung ke kantor pusat, sebagaimana diilustrasikan dalam gambar 2.1.



Gambar 2.1 *Remote Access* VPN

2. *Site-to-Site* VPN

Site-to-Site VPN merupakan solusi yang memungkinkan perluasan jaringan pribadi (*private network*) melalui jaringan internet atau layanan publik lainnya dengan cara yang aman, seperti terlihat pada gambar 2.2. Dikenal juga dengan sebutan LAN to LAN VPN, *Site-to-Site* VPN berfungsi sebagai alternatif dari infrastruktur WAN yang biasanya menghubungkan kantor cabang, kantor pusat, atau mitra bisnis dengan seluruh jaringan yang ada di perusahaan.



Gambar 2.2 *Site-to-Site* VPN

Site-to-Site VPN juga dibedahkan menjadi dua jenis yaitu:

a. Internet VPN

   VPN internet digunakan untuk menghubungkan kantor pusat dengan kantor cabang atau lokasi kantor yang berjauhan melalui infrastruktur publik.

b. *Extranet* VPN

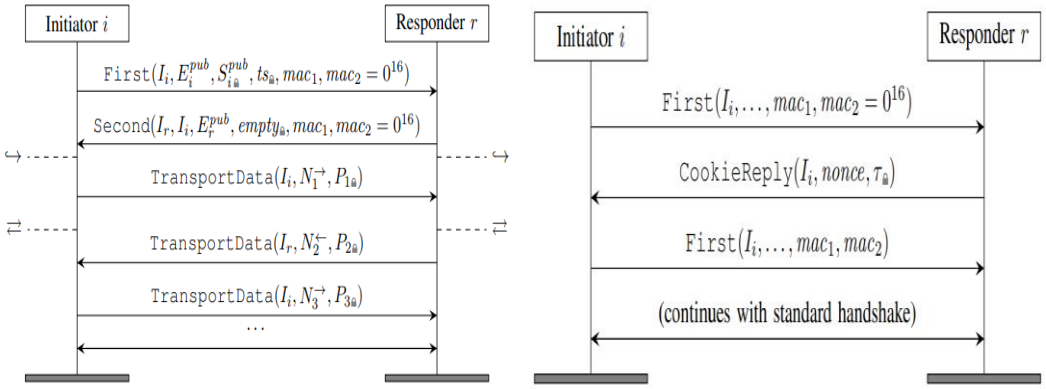
   Extranet VPN adalah bentuk perluasan dari internet suatu perusahaan yang memungkinkan integrasi pengguna dari luar perusahaan, seperti pemasok, penjual, pelanggan, dan mitra bisnis lainnya. Melalui *Extranet* VPN, kedua perusahaan dapat saling bertukar informasi dengan cepat dan mudah, sambil tetap menjaga keamanan jaringan internet dengan menggunakan *firewall*.

2.3 WIREGUARD

WireGuard adalah solusi *Virtual Private Network*s (VPN) baru, yang beroperasi pada lapisan 3, dirancang oleh Jason A. Donenfeld. WireGuard menggunakan kriptografi canggih seperti ChaCha20Poly1305 untuk enkripsi yang diautentikasi, BLAKE2s untuk *hashing* dan HKDF untuk derivasi kunci. Protokol pertukaran kunci yang digunakan dalam WireGuard didasarkan pada protokol IKpsk2 dari *Noise Protocol Framework* yang dibakukan berbagai protokol pertukaran kunci kriptografi dua pihak. Protokol ini memiliki banyak properti bagus, seperti penawaran *forward secrecy*, menghindari *replay attacks* dan peniruan *keycompromise*. WireGuard mengusulkan konsep penting yang disebut *Cryptokey Routing*, yang mengasosiasikan kunci publik dan *remote endpoint* jarak

jauh dengan daftar terowongan yang diizinkan dan alamat IP yang diizinkan di dalam terowongan. Setiap antarmuka jaringan memiliki kunci pribadi dan daftar rekan. Setiap rekan memiliki kunci publik yang digunakan untuk saling otentikasi antara *peers*. Ini bekerja dengan menambahkan antarmuka jaringan *virtual* yang disebut wg0 (atau wg1, wg2, dll). Kemudian pengguna mengonfigurasinya dengan miliknya kunci pribadi sendiri dan kunci publik rekan untuk mengirim paket terowongan aman yang disediakan wg0 [15].

WireGuard membuat *instance* salah satu protokol, yang disebut IKpsk2 dan memperluasnya untuk memberikan jaminan lebih lanjut yang dibutuhkan oleh VPN [5].



Gambar 2.3 Protokol *Message* WireGuard (kiri), Mekanisme *Cookie Under Load* (kanan) [5]

Dalam gambar 2.3 kiri, merupakan penggambaran dari pertukaran pesan pada protokol utama WireGuard. Protokol dimulai ketika *i* mengirim *handshake* pertama ke *r*, yang mencakup komponen-komponen berikut:

- I_i: pengidentifikasi sesi baru, yang di-*generate* oleh inisiator (*i*)
- E_i^{pub}: kunci publik sesaat yang baru, dihasilkan oleh i
- $S_{\text{🔒}}^{pub}$: kunci publik statis milik *i*, dienkripsi untuk *r*
- $ts_{\text{🔒}}$: *timestamp*, dienkripsi dengan kunci yang hanya dapat dihitung oleh *i* dan *r*
- mac_1, mac_2: kode otentikasi *message*

Kemudian *r* membalas dengan mengirimkan *handshake* kedua yang berisi:

- I_i: pengidentifikasi sesi *i*
- I_r: pengidentifikasi sesi baru, yang di-*generate* oleh *responder* (*r*)
- E_r^{pub}: kunci publik sesaat yang baru, dihasilkan oleh i
- $empty_{\text{🔓}}$: *bytestring* kosong yang dienkripsi dengan kunci yang hanya dapat dihitung oleh *i* dan *r*
- mac_1, mac_2: kode otentikasi *message*

Kedua pesan memiliki muatan yang dienkripsi berperan sebagai alat autentikasi: dengan menghitung kunci enkripsi yang cocok, setiap pihak membuktikan bahwa mereka mengetahui kunci privat yang sesuai dengan kunci publik statisnya. Untuk pesan kedua, kunci enkripsi juga membutuhkan pengetahuan tentang psk opsional yang memberikan jaminan autentikasi tambahan. Dua kunci sementara menambahkan materi kunci sesi khusus baru yang dapat digunakan untuk menghitung (meneruskan) kunci sesi rahasia yang hanya diketahui oleh pihak i dan r [5].

Gambar 2.3 kanan adalah mekanisme *cookie under load* untuk perlindungan lebih aktif terhadap DoS. WireGuard menggabungkan protokol berbasis *cookie* untuk melindungi aktif terhadap serangan *Denial of Service* (DoS). Protokol ini memungkinkan *host* untuk menolak pesan jabat tangan pertama saat mencurigai serangan DoS. Sebagai gantinya, *host* mengirimkan *cookie* segar khusus kepada inisiator yang dihitung menggunakan kunci rahasia, alamat IP inisiator, dan *port* sumber. *Cookie* ini dienkripsi oleh responder dan dikirim kembali ke inisiator. Inisiator mendekripsi *cookie* tersebut dan mengirim pesan jabat tangan pertama lagi dengan bidang mac2 yang berisi MAC dari pesan sebelumnya menggunakan *cookie* sebagai kunci MAC. Responder memverifikasi MAC tersebut dan melanjutkan jabat tangan standar. Protokol *cookie* ini melindungi dari serangan *brute-force* dengan membutuhkan kemampuan membaca pesan di jalur jaringan dan pengetahuan kunci statis inisiator. Mekanisme MAC ini tidak memengaruhi keamanan protokol saluran aman dan perlu dipelajari dampaknya terhadap jaminan penyembunyian identitas IKpsk2 karena menggunakan kunci publik statis dari kedua *host* [5].

2.3.1 Tujuan Keamanan WireGuard

Dalam pengembangannya, WireGuard memiliki beberapa tujuan diantaranya [5]:

- **Ketepatan (*Correctness*)**

  Jika *initiator* dan *responder* jujur menyelesaikan WireGuard *handshake* dan pesannya tidak diubah oleh musuh, maka data trans*port*asi kunci ($T^{\rightarrow}$, $T^{\leftarrow}$) dan transkip *hash* H7 adalah sama pada kedua *host*.

- **Kerahasiaan (*Secrecy*)**

  Jika pesan data *transport* P dikirim melalui terowongan antara dua *host* yang jujur, maka pesan ini dirahasiakan dari musuh. Selain itu, kunci lalu lintas untuk terowongan ini juga bersifat rahasia.

- ***Forward Secrecy***

  Kerahasiaan untuk suatu sesi berlaku meskipun kedua kunci privat statis dan *pre-shared key* (PSK) diketahui oleh musuh, tetapi hanya setelah sesi selesai dan semua kunci lalu lintas dan kunci rantainya dihapus oleh kedua belah pihak. Kerahasiaan juga berlaku meskipun kunci statis dan sesaat dikompromikan selama kunci yang dibagikan sebelumnya tidak dikompromikan.

- ***Mutual Authentication***

  Jika seorang inisiator yang jujur (resp. *responder*) menyelesaikan jabat tangan (seolah-olah) dengan rekan yang jujur, maka rekan tersebut pasti telah berpartisipasi dalam *handshake* ini. Selain itu, jika *host* A menerima pesan teks biasa melalui terowongan WireGuard yang mengklaim berasal dari *host* B, maka B harus (sengaja) mengirim pesan ini ke A.

- **Perlawanan terhadap *Key Compromise Impersonation (KCI)***

  Penerima pesan dapat mengautentikasi pengirim pesan meskipun kunci statis penerima disusupi.

- **Perlawanan terhadap *Mis-binding Identity***

  Jika dua pihak yang jujur memperoleh kunci lalu lintas yang sama di beberapa sesi WireGuard, maka mereka menyetujui identitas satu sama lain, meskipun salah satu atau keduanya telah berinteraksi dengan pihak yang

tidak jujur atau pihak yang jujur dengan kunci yang disusupi. Properti ini juga disebut resistensi terhadap serangan *keyshare* yang tidak diketahui.

- **Perlawanan terhadap *Replay***

  Setiap pesan protokol yang dikirim dapat diterima paling banyak satu kali oleh penerima

- ***Session Uniqueness***

  Ada paling banyak satu sesi inisiator jujur dan paling banyak satu sesi responden jujur untuk kunci lalu lintas tertentu. Demikian pula, ada paling banyak satu sesi inisiator yang jujur dan paling banyak satu sesi responden yang jujur untuk pesan *handshake* yang diberikan.

- ***Channel Binding***

  Dua sesi yang memiliki transkrip *hash* H7 sesi terakhir yang sama berbagi tampilan yang sama dan kunci sesi yang sama.

- ***Identity Hiding***

  Hanya dengan melihat pesan yang dikirimkan melalui jaringan, musuh pasif tidak dapat menyimpulkan kunci statis yang terlibat dalam sebuah sesi. (Namun, identitas ini bukan *forward secrecy*: Jika kunci statis responden disusupi, musuh nantinya dapat mendekripsi kunci publik statis initiator yang ditransmisikan dalam pesan pertama.)

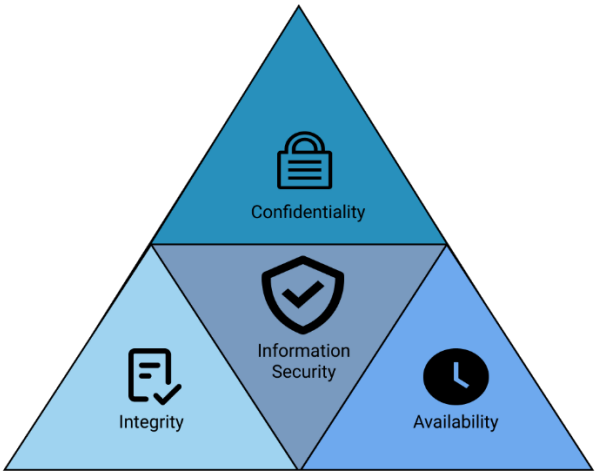
- **DoS *Resistance***

  Musuh tidak dapat memiliki pesan yang diterima oleh penerima yang dimuat tanpa terlebih dahulu melakukan *round trip* dengan penerima itu. Dalam praktiknya, ini berarti musuh harus berada di alamat yang diklaim.

2.4 KEAMANAN JARINGAN

Menjaga keamanan jaringan komputer sebagai bagian dari sistem sangat penting untuk memastikan integritas dan validitas data, serta memberikan layanan yang tersedia bagi pengguna. Keamanan jaringan komputer harus dilindungi dari berbagai serangan, upaya penyusupan, atau pemindaian oleh pihak yang tidak berhak. Dalam konteks keamanan jaringan, terdapat konsep CIA (*Confidentiality*, *Integrity*, dan *Availability*) yang merupakan prinsip dasar dalam menentukan tingkat keamanan jaringan dan informasi diilustrasikan dengan gambar 2.4. Selain

itu, terdapat juga Perkerian *Hexad* yang mengacu pada *Confidentiality*, *Posession or Control*, *Integrity*, *Authenticity*, *Availability*, dan *Utility* sebagai prinsip lain dalam menjaga keamanan jaringan dan informasi. [16].



Gambar 2.4 CIA *Triad* [17]

A. *Confidentiality* (rahasia)

Confidentiality (kerahasiaan) berarti memastikan informasi tetap rahasia dengan membatasi hak akses individu, sering kali melalui proses enkripsi. Contoh data yang harus dijaga kerahasiaannya mencakup informasi pribadi seperti nama, tanggal lahir, riwayat penyakit, nomor kartu kredit, nama ibu kandung, dan lainnya, serta data milik organisasi atau perusahaan.

B. *Integrity* (keaslian)

Integrity (keaslian) adalah prinsip yang menjamin bahwa data atau informasi yang dimiliki tetap otentik dan tidak mengalami perubahan tanpa persetujuan pemilik informasi. *Integrity* mengacu pada tingkat kepercayaan terhadap integritas suatu informasi. Dalam konteks *integrity*, terdapat dua mekanisme keamanan, yaitu mekanisme preventif dan mekanisme detektif. Mekanisme preventif digunakan sebagai kontrol akses untuk mencegah terjadinya modifikasi data. Sementara itu, mekanisme detektif digunakan untuk mendeteksi modifikasi yang telah dilakukan oleh pihak lain.

C. *Availability* (ketersediaan)

Availability (ketersediaan) berkaitan dengan ketersediaan informasi pada saat dibutuhkan. Ini berarti informasi harus selalu dapat diakses oleh pengguna dan tersedia dengan cepat. Salah satu jenis serangan yang dapat

mengancam ketersediaan ini adalah *Distributed Denial of Service* (DDoS). Dalam serangan ini, sumber daya atau server diisi dengan permintaan yang berlebihan atau tidak terduga, menyebabkan server tidak dapat melayani permintaan lain atau bahkan menjadi tidak berfungsi (*down*).

2.5 ANCAMAN PADA JARINGAN KOMPUTER

Ada beberapa bentuk ancaman pada jaringan komputer yang sering ditemui diantaranya adalah [18]:

1. *Brute Force and Dictionary*

   Serangan ini dilakukan dengan upaya masuk ke dalam jaringan dengan menyerang *database password* atau *login prompt* yang aktif untuk mencari *password* akun pengguna. Serangan ini dilakukan secara sistematis dengan mencoba berbagai kombinasi angka, huruf, atau simbol.

2. *Denial of Services* (DoS)

   Denial of Service (DoS) merupakan salah satu ancaman keamanan jaringan yang menyebabkan layanan jaringan menjadi tidak berfungsi. Serangan ini menyebabkan jaringan tidak dapat diakses dan sistem menjadi tidak responsif terhadap lalu lintas atau permintaan layanan terhadap objek dan sumber daya jaringan.

3. *Spoofing*

   Istilah spoofing sering digunakan untuk merujuk pada tindakan memalsukan atau menyisipkan informasi palsu yang menyesatkan dalam *header email* atau *netnews*. Pemalsuan *header* ini bertujuan untuk menipu penerima atau jaringan aplikasi mengenai sumber asli pesan.

4. Serangan *Man-in-the-middle*

   Man-in-the-middle terjadi ketika pengguna yang merusak dapat berada di antara dua titik link komunikasi. Para penyerang ini beroperasi tanpa terlihat oleh kedua belah pihak yang berkomunikasi dan mampu mengubah isi dan arah lalu lintas data. Dengan cara ini, penyerang bisa mencuri kredensial *logon* atau bahkan memodifikasi isi pesan yang dikirimkan antara kedua titik komunikasi tersebut.

5. *Spamming*

   Spamming adalah salah satu bentuk penyalahgunaan teknologi *email* yang paling umum dan sering ditemui. Pengguna sering kali menerima *email spam* yang berisi pesan-pesan komersial yang tidak diinginkan. Meskipun tingkat ancamannya dianggap rendah dibandingkan dengan *email worm* atau *phishing*, pengguna sering kali mendapati produk hasil *spamming* dalam fasilitas *email* mereka.

6. *Sniffer*

   Sniffer (*snooping attack*) adalah aktivitas yang dilakukan oleh pengguna yang bermaksud mendapatkan informasi tentang jaringan atau lalu lintas data yang melewatinya. *Sniffer* biasanya berupa program yang dapat menangkap paket data yang lewat di media jaringan dan menduplikasi isinya ke dalam *file*. Serangan *Sniffer* sering kali difokuskan pada koneksi awal antara klien dan server untuk mencuri *logon credential* dan *password*.

7. *Crackers*

   Cracker adalah istilah untuk individu yang masuk ke dalam sistem orang lain dan memiliki sifat yang lebih destruktif. Mereka biasanya beroperasi dalam jaringan komputer dan mencoba untuk melewati *password* atau lisensi program komputer. Tindakan mereka melawan keamanan komputer dengan sengaja, seperti merubah tampilan halaman web (*deface*) milik orang lain, bahkan bisa mencuri data atau menghapus data milik orang lain. *Cracker* sering melakukan tindakan ini untuk keuntungan pribadi, tujuan jahat, atau sebagai respon terhadap tantangan. Beberapa proses pembobolan dilakukan untuk menunjukkan kelemahan dalam sistem keamanan.

2.6 PARAMETER *QUALITY OF SERVICE*

Quality of Service (QoS) adalah kemampuan suatu jaringan untuk menyediakan layanan yang optimal dengan mengalokasikan *bandwidth*, mengatasi *jitter*, dan mengurangi *delay*. Parameter QoS mencakup *delay*, *jitter*, *packet loss*, *throughput*, MOS (*Mean Opinion Score*), *echo cancellation*, dan PDD (*Post-Dial Delay*). Kualitas QoS sangat bergantung pada kualitas jaringan yang digunakan. Beberapa faktor yang dapat mempengaruhi kualitas QoS meliputi redaman, distorsi,

dan *noise*. Tujuan dari QoS adalah untuk meningkatkan produktivitas pengguna akhir dengan memastikan bahwa aplikasi berbasis jaringan berjalan dengan kinerja yang handal [19].

Terdapat 4 (empat) parameter dalam *quality of services*, diantaranya:

1. *Packet Loss*

   Packet Loss merupakan parameter yang menunjukan kondisi dimana jumlah total paket yang hilang selama pengiriman *upload download bandiwth* berjalan. Penyebab terjadinya *packet loss* antara lain [20]:

   A. Kongesti akibat jalur sudah penuh
   B. Kongesti akibat antrian berlebih
   C. *Node* yang melebihi kapasitas *buffer* yang tersedia
   D. Memori yang terbatas pada setiap *host*
   E. Kontrol dari *network* administrator untuk membuang paket ketika paket sudah tidak lagi dibutuhkan

   Menurut standarisasi TIPHON, suatu *packet loss* dikatakan baik maupun buruk berdasarkan penilaian pada tabel 2.1.

Tabel 2.1 Kategori *Packet Loss* [21]

Kategori	*Packet Loss*
Sempurna	0%
Bagus	3%
Sedang	15%
Buruk	25%

Dengan perhitungan [20]:

$$\textbf{\textit{Packet Loss}} = \frac{\textbf{\textit{Packet Send}} - \textbf{\textit{Packet Received}}}{\textbf{\textit{Packet Sent}}} \times \mathbf{100}\% \qquad \textbf{(2.1)}$$

2. *Delay*

   Delay adalah waktu yang diperlukan pada saat paket dikirimkan dari pengirim ke penerima. Nilai *delay* sangat berpengaruh terhadap kualitas jaringan. *Delay* terbagi menjadi tiga jenis, antara lain:

   a. *Delay propagation* yaitu waktu yang dibutuhkan oleh sebuah sinyal untuk bergerak dalam media komunikasi seperti fiber, kabel maupun gelombang wireless.

b. *Delay transmission* adalah waktu yang dibutuhkan suatu system untuk melewatkan sebuah paket data.
c. *Delay queue* yaitu waktu yang dibutuhkan pada suatu paket sebelum paket tersebut sampai ke tujuan.

Tabel 2.2 Kategori *Delay* [21]

Kategori *Delay*	***Delay***
Sangat Bagus	< 150 ms
Bagus	150 s/d 300 ms
Sedang	300 s/d 450 ms
Buruk	> 450 ms

Tabel 2.2 merupakan standarisasi nilai dari THIPON. Nilai *delay* dapat dihitung menggunakan persamaan [20]:

$$\boldsymbol{Delay} = (\textbf{Total Paket} - \mathbf{1}) \frac{\boldsymbol{Number\ of\ bits\ to\ transmit}}{\boldsymbol{transmission\ Rate}} \qquad \textbf{(2.2)}$$

3. *Throughput*

Thtoughput adalah jumlah total keseluruhan data yang dikirimkan dari pengirim kepada penerima. Nilai *throughput* dapat dihitung menggunakan persamaan [20]:

$$\boldsymbol{Throughput} = \sum_{i}^{n} \frac{\boldsymbol{packet\ size}}{\boldsymbol{finish\ time - start\ time}} \qquad \textbf{(2.3)}$$

4. *Jitter*

Jitter adalah nilai variasi dari waktu kedatangan sebuah paket. Perbedaan dengan *delay* adalah, *jitter* merupakan nilai variasi dari *delay* dimana tidak berbeda jauh antara *low jitter* dan *high jitter*. Jaringan penyedia layanan harus menyediakan *bandwidth* yang cukup dan kemungkinan latensi untuk memastikan koneksi jaringan yang berkualitas, itu dapat membantu mengurangi *jitter*. Menurut standarisasi TIPHON, *jitter* dikategorikan seperti pada tabel 2.3 [20].

Tabel 2.3 Kategori *Jitter* [21]

Kategori *Jitter*	***Peak Jitter***
Sangat Bagus	0 m/s

Kategori *Jitter*	***Peak Jitter***
Bagus	0 s/d 75 m/s
Sedang	76 s/d 125 m/s
Buruk	> 225 m/s

Nilai *jitter* dapat dihitung menggunakan persamaan:

$$\boldsymbol{Jitter = \frac{Total\ variasi\ delay}{paket\ data\ diterima - 1}} \qquad \textbf{(2.4)}$$