BAB 2
KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1 KAJIAN PUSTAKA

Alfred dan Joko Christian Chandra [4] pada penelitiannya di tahun 2018 yang berjudul "Pemanfaatan *Firewall* Pada Jaringan Komputer SMK FADILAH" membahas mengenai pembatasan konten yang di akses pada situs yang berbau isu sara dan pornografi khususnya situs jejaring sosial dan situs *streaming* dengan menggunakan *firewall filter rules* pada mikrotik. *Firewall Filter Rules* merupakan fitur yang dapat membatasi penggunaan internet untuk mengakses situs-situs negatif dan media sosial pada saat proses belajar mengajar berlangsung. Pemblokiran berdasarkan pada alamat dari suatu situs jejaring sosial dan *streaming*. Dengan menggunakan fitur tersebut dilakukan tahap pengujian yang diimplementasikan langsung di SMK Fadilah dan memperoleh hasil yaitu tidak ada lagi siswa-siswi yang membuka situs jejaring sosial dan situs *streaming* pada saat belajar di Lab sebagaimana yang tercantum pada Tabel 2.1.

Roni Reza Abdullah dan Ade Nurhayati [5] pada penelitiannya di tahun 2019 yang berjudul "Monito*ring* Sistem Keamanan Jaringan Berbasis Telegram Bot Pada *Local Area Network*" membahas mengenai keamanan jaringan khususnya keamanan server dari serangan menuju ICMP, TCP, dan UDP. Menggunakan metode *Intrusion Prevention System* (IPS) yang merupakan metode keamanan jaringan yang dapat mendeteksi adanya serangan atau aktivitas mencurigakan pada sebuah sistem/jaringan. Dengan menggunakan metode *Intrusion Prevention System* (IPS), maka admin akan mengetahui apa yang dilakukan oleh *attacker* dalam sebuah server dengan menerima notifikasi yang disampaikan melalui bot telegram. Hasil yang diperoleh berupa tingkat perbandingan akurasi waktu penerimaan ketika terjadi serangan dengan beberapa jenis serangan seperti DDOS, *Port Scan*, dan *Brute Force* sebagaimana yang tercantum pada Tabel 2.1.

Alfi Nurfauzi, Esron Rikardo Nainggolan, Siti Nur Khasanah, dan Ahmad Setiadi [6] pada penelitiannya di tahun 2018 yang berjudul "Implementasi *Firewall*

Filtering Web dan Manajemen *Bandwith* Menggunakan Mikrotik" membahas mengenai penyaringan situs-situs negatif menggunakan *Web Filtering* serta memajemen *bandwidth* menggunakan teknik *Simple Queue* yang diterapkan di SMK Multi Media Mandiri. Metode *Web Filtering* merupakan metode yang dapat memblokir penggunaan internet untuk mengakses situs-situs negatif dan media sosial, kemudian teknik *Simple Queue* merupakan teknik untuk memanajemen *bandwidth* dengan cara menentukan besaran *bandwidth* yang diterima oleh masing-masing *user* agar pembagian *bandwidth* sesuai dengan porsinya. Dengan menggunakan metode *web filtering* serta teknik *Simple Queue* tersebut dilakukan tahap pengujian yang diimplementasikan langsung di SMK Multi Media Mandiri memperoleh hasil yaitu *user* tidak dapat lagi mengakses *website* dengan bebas, karena beberapa situs terlarang sudah di *setting blocked access* dan manajemen *bandwidth* sudah berhasil dikonfigurasi sesuai dengan kebutuhan yaitu masing-masing mendapatkan 128 *Kb* sebagaimana yang tercantum pada Tabel 2.1.

Penelitian Sumardi Jayanto, Ahmad Tantoni, dan Hasyim Asyari [7] pada tahun 2021 yang berjudul "Implementasi Keamanan Jaringan dengan *Packet Filtering* Berbasis Mikrotik Untuk Internet Positif Di SMKN 1 Praya" meneliti tentang penggunaan *Packet Filtering* yang diterapkan di sekolah menengah. Metode *packet filtering* merupakan metode yang dapat membatasi penggunaan internet untuk media sosial dan *game online* pada jam pembelajaran berlangsung agar tidak mengganggu kegiatan belajar mengajar, kemudian internet akan normal kembali ketika pembelajaran di sekolah selesai (pulang sekolah). Pemblokiran dilakukan berdasarkan alamat IP dari media sosial dan *game online*. Dengan menggunakan metode *packet filtering* tersebut dilakukan tahap pengujian yang diimplementasikan langsung di SMKN 1 Praya pada pukul 09.00 yang merupakan jam pembelajaran berlangsung dan memperoleh hasil sesuai dengan yang diharapkan yaitu tidak dapat mengakses *game online* dan media sosial yang berarti proses pemblokiran atau pembatasan menggunakan metode *Packet Filtering* tersebut berhasil dan efektif sebagaimana yang tercantum pada Tabel 2.1.

Penelitian Wahyat dan Agus Teddyana [8] pada tahun 2021 yang berjudul "*Monitoring* Jaringan Internet Menggunakan Notifikasi *Bot* API Telegram" meneliti tentang *monitoring* sistem koneksi jaringan komputer dengan

menggunakan Bot API Telegram dan *Netwatch* Mikrotik untuk mendapatkan informasi berupa pesan notifikasi Telegram secara *realtime* kepada *administrator system*. Bot API Telegram akan mengirimkan pesan notifikasi ketika terjadi kendala koneksi internet *down* dan internet *up*. Pada penelitian ini dilakukan pengujian waktu respon notifikasi Bot Telegram yang dilakukan sebanyak 5 kali yang kemudian diambil nilai rata-ratanya ketika terjadi kendala koneksi internet (*down/up*). Penerapan dari penelitian ini memperoleh hasil sesuai dengan yang diharapkan yaitu Mikrotik RB 941 dengan Bot API Telegram akan melakukan pendeteksian bila terjadi kendala pada sistem jaringan internet dengan mengirimkan pesan notifikasi melalui bot telegram ke grup aplikasi Telegram dengan status koneksi *down* atau koneksi internet *up* sebagaimana yang tercantum pada Tabel 2.1.

Pada penelitian ini yang berjudul "Implementasi Dan Analisis Pemblokiran Akses Situs Dan *Game Online* Dengan Menggunakan *Packet Filtering* Pada Mikrotik Untuk *Parenting*" meneliti tentang penerapan *packet filtering* untuk membatasi akses internet. Pembatasan akses internet dilakukan berdasarkan alamat IP dan nomor *port*. Keterkaitan penelitian sebelumnya dengan penelitian ini terletak pada metode yang digunakan yaitu metode *Packet Filtering* berdasarkan alamat IP. Namun penelitian ini juga menggunakan metode *packet filtering* berdasarkan nomor *port* serta menggunakan aplikasi Telegram sebagai penerima notifikasi yang akan diterima oleh *administrator system* ketika terjadi pelanggaran pengaksesan pada beberapa situs seperti judi *online*, media sosial, dan *game online*.

Tabel 2. 1 Kajian Penelitian Sebelumnya

Nama	Judul	Tahun	Hasil
Alfred dan Joko Christian Chandra	Pemanfaatan *Firewall* Pada Jaringan Komputer SMK FADILAH	2018	Melakukan pemblokiran berdasarkan alamat dari suatu situs. Fokus pemblokiran pada situs jejaring sosial dan *streaming*.
Roni Reza Abdullah, dan Ade Nurhayati	*Monitoring* Sistem Keamanan Jaringan Berbasis Telegram Bot Pada *Local Area Network*	2019	Menggunakan metode *Intrusion Prevention System* (IPS), maka admin akan mengetahui apa yang dilakukan oleh *attacker* dalam sebuah *server* dengan menerima notifikasi yang disampaikan melalui bot Telegram.

Nama	Judul	Tahun	Hasil
Alfi Nurfauzi, Esron Rikardo Nainggolan, Siti Nur Khasanah, dan Ahmad Setiadi	Implementasi *Firewall Filtering Web* dan Manajemen *Bandwith* Menggunakan Mikrotik	2018	Melakukan pemblokiran dan manajemen *bandwidth.* Pemblokiran *berdasarkan* alamat IP. Fokus pemblokiran pada situs-situs yang mengandung pornografi.
Sumardi Jayanto, Ahmad Tantoni, Hasyim Asyari	Implementasi Keamanan Jaringan dengan *Packet Filtering* Berbasis Mikrotik Untuk Internet Positif Di SMKN 1 Praya	2021	Pemblokiran berdasarkan alamat IP. Fokus pemblokiran pada media sosial dan *game online* pada jam tertentu
Wahyat dan Agus Teddyana	*Monitoring* Jaringan Internet Menggunakan Notifikasi Bot API Telegram	2021	*Netwatch* pada Mikrotik dengan Bot API Telegram akan melakukan pendeteksian bila terjadi kendala pada sistem jaringan internet dengan mengirimkan pesan notifikasi melalui bot telegram ke grup aplikasi telegram dengan status koneksi *down* atau koneksi internet *up*.

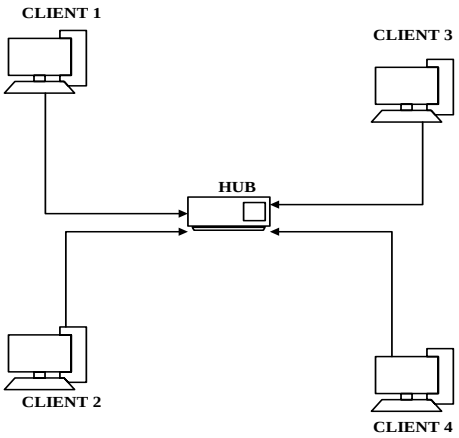
2.2 LANDASAN TEORI

2.2.1 Jaringan Komputer

2.2.1.1 Jenis-jenis Jaringan Komputer

Jaringan komputer adalah kumpulan dari beberapa komputer yang tersebar yang dapat dihubungkan bersama atau terhubung satu sama lain untuk menjalankan fungsinya. Misalnya, jika dua komputer dapat berbagi data, bertukar informasi, menjalankan aplikasi yang sama. Maka mereka dapat dikatakan terhubung satu sama lain. Kabel tembaga, kabel koaksial, kabel *twisted pair*, kabel serat optik, dan berbagai teknologi nirkabel semuanya dapat digunakan untuk menghubungkan jaringan komputer. Contoh dari jaringan komputer yaitu *Local Area Network* (LAN), *Metropolitan Area Network* (MAN), dan *Wide Area Network* (WAN) [9].

a. *Local Area Network* (LAN)



Gambar 2.1 *Local Area Network*

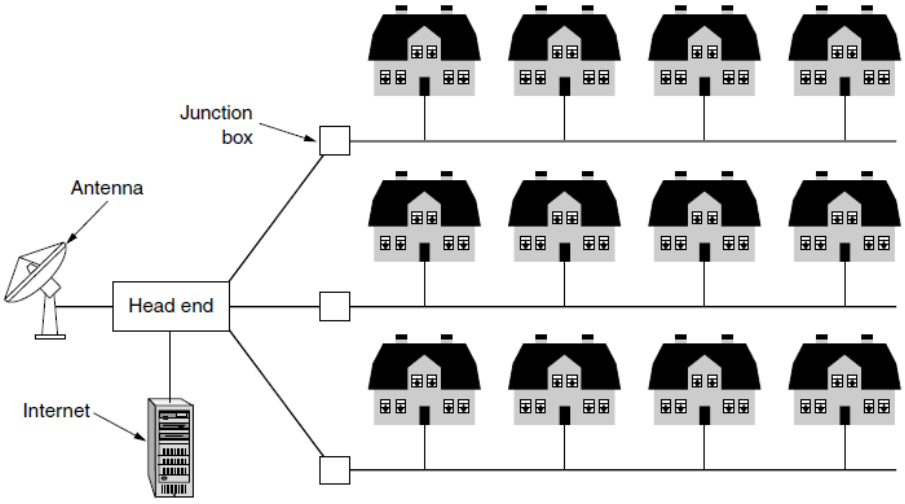
Jaringan milik pribadi yang disebut *Local Area Network* (LAN) bekerja di dalam dan dekat dengan satu struktur, seperti rumah, kantor, atau pabrik. LAN sering digunakan untuk menghubungkan PC dan *gadget* konsumen lainnya sehingga mereka dapat berbagi sumber daya (seperti printer) dan berkomunikasi. Jaringan perusahaan adalah LAN yang digunakan oleh bisnis. Contoh dari LAN dapat dilihat pada Gambar 2.1 yang mana beberapa pc dapat terhubung dengan *hub* yang berfungsi untuk menghubungkan beberapa PC yang masih dalam jaringan yang sama sehingga dapat bertukar informasi antara satu sama lain.

b. *Metropolitan Area Network* (MAN)

Sebuah kota dicakup oleh *Metropolitan Area Network* (MAN). Jaringan televisi kabel yang dapat diakses di banyak kota adalah contoh MAN yang paling terkenal. Teknik ini berevolusi dari sistem antena lingkungan sebelumnya yang diterapkan di daerah dengan penerimaan televisi *free-to-air* di bawah standar. Sistem awal seperti itu melibatkan pemasangan antena besar di bukit terdekat, yang kemudian memantulkan sinyal ke rumah pelanggan.

Awalnya dibuat menggunakan pendekatan *ad hoc* lokal. Perusahaan kemudian mulai beroperasi dan memenangkan kontrak dari otoritas regional untuk menghubungkan seluruh kota. Fase berikutnya adalah konten televisi, bahkan mungkin stasiun kabel saja. Saluran ini sering

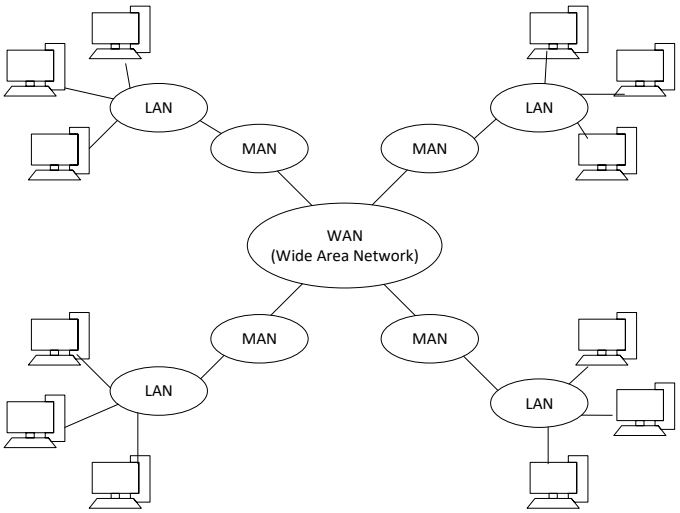
memiliki fokus yang sempit, seperti semua olahraga, semua masakan, semua berkebun, dll. Namun, dari saat mereka dikembangkan hingga akhir 1990-an, mereka hanya ditujukan untuk penerimaan televisi. Operator jaringan kabel mulai memahami bahwa mereka dapat menawarkan layanan Internet dua arah di bagian spektrum yang kurang dimanfaatkan dengan perbaikan sistem sederhana ketika Internet mulai menarik khalayak luas. Pada saat itu, sistem TV kabel mulai berkembang menjadi jaringan wilayah metropolitan bukan hanya sarana penyiaran televisi. MAN perkiraan awal mungkin menyerupai sistem pada Gambar 2.2 [10].



Gambar 2.2 *Metropolitan Area Network* [10].

c. *Wide Area Network* (WAN)

Wide Area Network (WAN) adalah istilah yang digunakan untuk menggambarkan jaringan komputer yang menjangkau area yang luas, seperti jaringan komputer antar kota, wilayah, atau bahkan negara. WAN juga dapat digunakan untuk menggambarkan jaringan komputer yang membutuhkan *router* dan saluran komunikasi terbuka. Ilustrasi dari jaringan WAN ini adalah Internet. Jaringan WAN ini juga sering disebut sebagai pengelompokan (kumpulan) berbagai LAN dan MAN yang terlihat seperti pada Gambar 2.3 [11].



Gambar 2.3 *Wide Area Network*

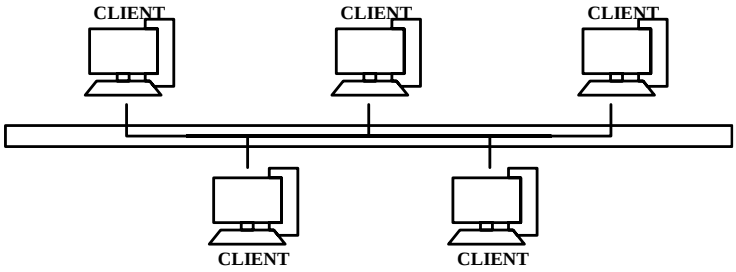
2.2.1.2 Topologi Jaringan Komputer

Ilustrasi perencanaan hubungan antar komputer dalam LAN yang sering menggunakan kabel sebagai media komunikasi, bersama dengan konektor, ethernet, dan perangkat pendukung lainnya, dikenal sebagai topologi jaringan atau arsitektur jaringan. Berikut beberapa jenis topologi jaringan yang sering digunakan pada jaringan lokal area :

a. Topologi *Bus*

   Topologi *Bus* adalah kabel tunggal yang ditutup di kedua ujungnya dan memiliki simpul atau *node* yang berjarak di sepanjang kabel. Dengan menggunakan arsitektur jaringan ini, sinyal hanya dibawa dalam satu arah melintasi kawat, yang memungkinkan terjadinya tabrakan.

   Manfaat dari topologi ini adalah murah karena menggunakan sedikit media transmisi (kabel) seperti yang terlihat pada Gambar 2.4, kabel yang digunakan pun umumnya tersedia di pasaran, dan koneksi langsung antara masing-masing komputer. Sedangkan untuk kerugiannya yaitu ketika jalur di waktu yang sama dipakai oleh lebih dari satu pasang atau biasa disebut *hang*/crosstalk.

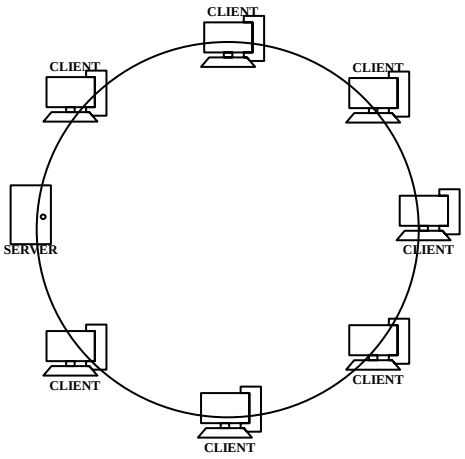


Gambar 2. 4 Topologi *Bus*

b. Topologi *Ring*

Topologi yang memiliki simpul dan berbentuk seperti lingkaran tertutup seperti pada Gambar 2.5 disebut Topologi *Ring*. Untuk mencegah kemungkinan pergerakan data yang sangat cepat, sinyal mengalir dalam dua arah. Lingkaran koneksi mengelilingi setiap komputer (seperti *bus* tetapi ujung *bus* disambung). Untuk memastikan bahwa data yang dikirimkan mencapai komputer yang dituju, diberikan alamat tujuan.

Manfaat menggunakan arsitektur jaringan ini adalah bahwa kesalahan transmisi dapat dikurangi dengan menggunakan koneksi *point to point*, dan kegagalan koneksi yang disebabkan oleh gangguan media dapat diatasi menggunakan saluran lain yang masih aktif. Sedangkan untuk kerugiannya yaitu ketika mengirim data melalui banyak komputer, transfer datanya akan menjadi lambat.

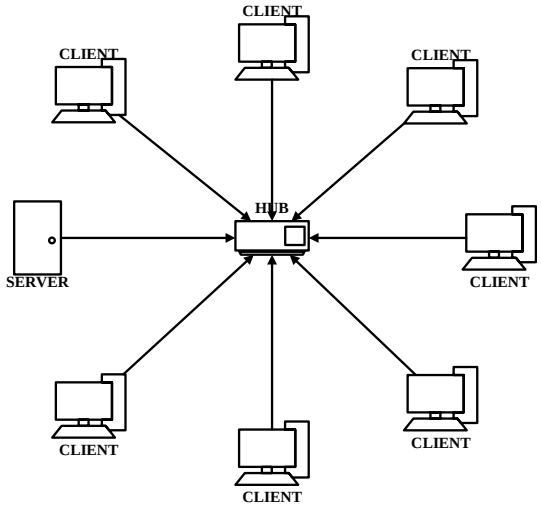


Gambar 2. 5 Topologi *Ring*

c. Topologi *Star*

Karakteristik yang membedakan topologi jaringan ini adalah bahwa *node* (stasiun) berkomunikasi satu sama lain secara langsung melalui *hub* atau *switch* seperti yang terlihat pada Gambar 2.6, dengan lalu lintas data yang mengalir dari *node* ke *hub* dan kemudian dirutekan ke *node* tujuan (stasiun). Jaringan lain tidak akan terputus jika salah satu segmen kabel putus.

Manfaat menggunakan topologi ini termasuk akses cepat ke stasiun lain (*client* atau *server*), kemampuan untuk menambahkan workstation baru selama *port* pada *node* pusat (*hub* / *switch*) tersedia, kemampuan untuk mengatur *hub* / *switch* secara seri (bertingkat) untuk meningkatkan jumlah stasiun yang terhubung ke jaringan, dan kemampuan untuk menambahkan lebih banyak pengguna daripada topologi *bus* atau *ring*. Jika terjadi tabrakan karena lalu lintas data yang tinggi, semua komunikasi akan tertunda. Jika *hub* atau *switch* menentukan tidak ada jalur alternatif yang digunakan *node* lain, koneksi akan berlanjut secara *random*.

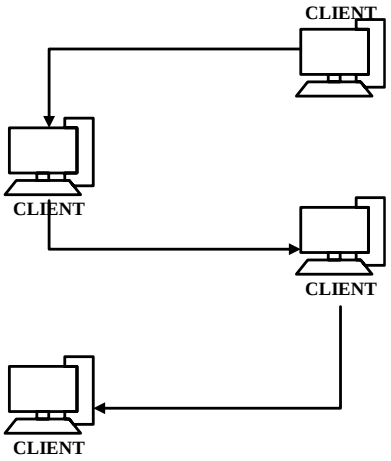


Gambar 2. 6 Topologi *Star*

d. Topologi *Daisy-Chain* (Linear)

Setiap *node* dalam topologi ini terhubung ke dua *node* lain oleh segmen kabel, namun segmen tersebut membuat saluran daripada lingkaran

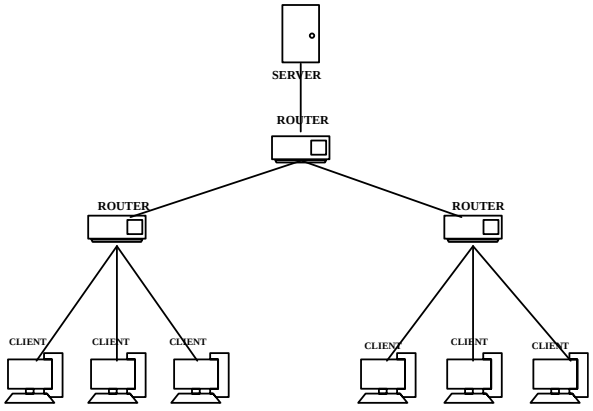
lengkap, berbeda dengan topologi *Bus* dan topologi *Ring*, di mana koneksi dibuat melalui lingkaran kabel yang lengkap. Ini mirip dengan terhubung secara seri antar komputer seperti pada Gambar 2.7. Untuk keuntungan menggunakan topologi ini adalah instalasi dan pemeliharaan topologi ini tergolong murah. Sedangkan untuk kerugiannya yaitu tidak dapat menyesuaikan dengan zaman atau dapat dikatakan kurang dapat diandalkan.



Gambar 2. 7 Topologi *Daisy-Chain*

e. Topologi *Tree*

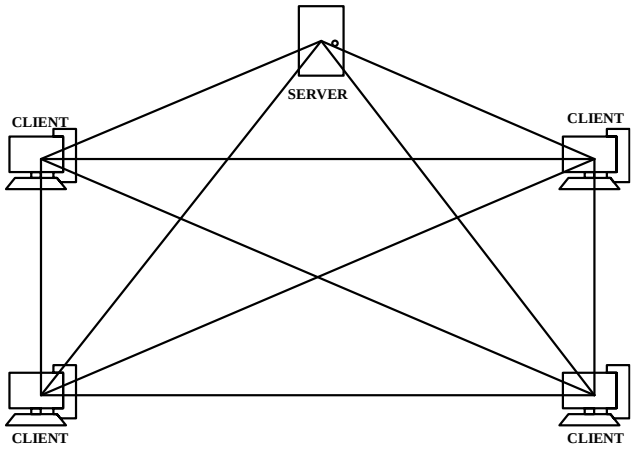
Lokasi setiap stasiun berbeda-beda. Jaringan sangat bergantung pada stasiun di tempat yang lebih tinggi karena mereka mengontrol stasiun di bawahnya (*hierachical topology*) seperti yang terlihat pada Gambar 2.8, dan posisi stasiun yang sama disebut sebagai *peer topology*.



Gambar 2. 8 Topologi *Tree*

f. Topologi *Mesh*

Koneksi antarsentral sepenuhnya digunakan dalam struktur jaringan ini. Meningkatnya jumlah pusat yang terpasang meningkatkan kompleksitas jaringan secara linier. Tidak hanya kurang ekonomis tetapi juga agak mahal untuk dioperasikan.



Gambar 2. 9 Topologi *Mesh*

Karena sistemnya yang kompleks, topologi mesh ini adalah teknologi unik yang tidak dapat diproduksi dengan kabel, tetapi teknologi topologi nirkabel ini sangat mungkin terwujud (karena sudah pasti tidak akan ada kabel yang berputar-putar). Biasanya, perangkat radio (air point) yang bertindak sebagai repeater untuk memperkuat sinyal dan mengontrol arah koneksi data ditempatkan di tengah (area) antara komputer kosong untuk memperkuat sinyal transmisi data yang ditransmisikan [10]. Contoh Gambaran dari topologi ini yaitu seperti pada Gambar 2.9.

2.2.2 *Internetworking*

2.2.2.1 Internet

Internet adalah sistem yang digunakan dalam skala global untuk komunikasi, pemasaran, berbagi data dan informasi, dan tujuan lainnya. Advanced Research Project Agency (ARPA) membantu membawa Internet ke Amerika Serikat pada 1970-an, tetapi baru pada awal 1990-an

masyarakat umum mulai menggunakannya. Melalui surat elektronik, newsgroup, dan siaran audio dan video, Internet memungkinkan interaksi manusia dan memungkinkan individu untuk berkolaborasi di banyak tempat yang berbeda dan secara real-time. Pembaruan harian yang dapat diselesaikan dengan cepat dan mudah juga dapat diakses, membuka pintu untuk perluasan pasar. Banyak "e-business" yang melakukan sebagian besar penjualan dan layanan mereka secara online dan menjangkau audiens yang lebih luas melalui visibilitas yang lebih tinggi dan lebih baik.

Istilah "internet" berasal dari konsep "internetworking," yang menghubungkan banyak jaringan. Jaringan lokal terhubung ke komputer, dan internet menghubungkan jaringan lokal bersama-sama sehingga semua komputer kita dapat berkomunikasi satu sama lain [12].

2.2.2.2 *Domain Name System* (DNS)

Domain Name System adalah sistem yang melacak nama *host* dan *domain* dalam bentuk *database* terdistribusi. Setiap *domain* akan menerima *email* yang didata oleh setiap *server* transmisi surat *(mail exchange server)* dan setiap nama *host* akan disediakan alamat IP oleh DNS. Sederhananya DNS adalah sistem yang menerjemahkan nama domain menjadi alamat IP agar bisa dipahami oleh komputer saat kita mengakses sebuah situs menggunakan nama *domain*.

Pengalamatan dan penjaluran *(routing)* merupakan tugas dari perangkat keras komputer dan jaringan bekerja dengan menggunakan alamat IP. Penunjukkan sumber (URL) dan alamat surel lebih banyak dilakukan oleh manusia. Misalnya, pengguna akan diarahkan ke alamat IP 124.81.92.144 (IPv4) dan 2001:e00:d:10:3:140::83 (IPv6) ketika mereka memasukkan www.indosat.net.id dengan *browser web*. Cara kerja DNS dapat dibandingkan dengan direktori telepon *online*.

2.2.2.3 Protokol Jaringan

Protokol adalah seperangkat aturan yang menentukan berbagai tugas yang harus diselesaikan oleh pihak pengirim/pemancar dan sisi penerima

agar komunikasi terjadi sebagaimana mestinya, termasuk mentransfer pesan, data, informasi, dan tugas lainnya. Protokol ini juga memungkinkan dua atau lebih komputer untuk berbicara satu sama lain dalam bahasa yang sama. Sederhananya, protokol adalah media yang digunakan untuk menghubungkan pengirim dan penerima. Protokol dapat diterapkan pada perangkat keras dan perangkat lunak. Jadi hampir semua komunikasi yang terjadi pada jaringan komputer pasti melibatkan protokol.

Protokol memiliki beberapa fungsi, antara lain:

a. Fragmentasi dan *Reassembly,* Ketika pihak pengirim mengirimkan informasi, sebelumnya informasi tersebut dibagi menjadi beberapa paket data, yang kemudian akan digabungkan kembali oleh pihak penerima menjadi satu paket berita setelah diterima.
b. *Encaptulation, encaptulation* melayani tujuan menyelesaikan berita yang dikirim dengan alamat, kode koreksi, dan detail lainnya.
c. *Connection Control*, dari pengirim/pemancar dan penerima dibangun hubungan komunikasi oleh *connection control*.
d. *Flow Control*, dari pengirim/pemancar ke penerima perjalanan datanya diatur oleh *flow control*.
e. *Error Control*, pada saat pengiriman data dilakukan, *error control* akan mengontrol terjadinya kesalahan yang terjadi.
f. *Transmission Service*, Layanan transmisi menjalankan fungsi layanan komunikasi data dengan memperhatikan prioritas, keamanan, dan perlindungan data.

2.2.2.4 OSI *Layer*

OSI *Layer* (*Open Systems Interconnection*) adalah sebuah model referensi yang digunakan untuk memahami dan merancang protokol jaringan dan komunikasi antar komputer. Model ini terdiri dari tujuh lapisan yang mewakili fungsi-fungsi yang berbeda dalam proses komunikasi. Setiap lapisan memiliki tanggung jawab khusus, tetapi juga bekerja sama dengan lapisan-lapisan lain untuk memungkinkan komunikasi yang efisien antar

perangkat. Pada tabel 2.2 berikut adalah tujuh lapisan OSI *Layer* beserta tanggung jawab utama masing-masing [13].

Tabel 2. 2 OSI 7 *Layer* Model

OSI *LAYER*		
Layer	**Fungsi**	**Contoh**
Application	Menunjang Aplikasi untuk berkomunikasi melalui jaringan	SMTP
Presentation	Memformat data sehingga dapat dikenali oleh penerima	JPG, GIF, HTTPS, SSL, TLS
Session	Membentuk koneksi, kemudian memutuskannya ketika seluruh data telah terkirim	NetBIOS, PPTP
Transport	Mengatur flow control, acknowledgement dan mengirim ulang data jika diperlukan	TCP, UDP
Network	Menambahkan Alamat jaringan pada paket	Router, Layer 3 Switch
Data Link	Menambahkan MAC Address pada paket	Switch
Physical	Mengirimkan data melalui media transmisi	Hub, NIC, Kabel

2.2.2.5 Alamat IP

Alamat IP merupakan protokol jaringan yang biasa disebut TCP/IP karena secara umum protokol tersebut dijalankan bersama protokol TCP. Mengintegrasi jaringan internet di dunia merupakan konsekuensi dari penerapan internet protokol. Jika ingin berkomunikasi dan terhubung dengan internet, maka seluruh host di seluruh dunia harus memiliki alamat IP yang mana berfungsi sebagai pengenal host pada jaringan. Alamat IP yang dipakai oleh host yang berbeda tidak boleh sama, dengan kata lain alamat IP harus bersifat unik. Lembaga sentral internet yang dikenal dengan IANA merupakan lembaga yang mengkoordinasi penggunaan alamat IP di seluruh dunia.

Alamat IP terdiri dari 2 macam, yaitu IP versi 4 (IPv4) dan IP versi 6 (IPv6). Menurut Kominfo (Kementrian Komunikasi dan Informatika), berikut perbedaan antara IPv4 dengan IPv6 :

a. Fitur

   Jumlah alamat pada IPv4 yaitu menggunakan 32-bit sedangkan pada IPv6 menggunakan 128-bit.

b. Routing

   Ketika IPv4 mengelola tabel *routing* yang besar akan menurun performanya sedangkan IPv6 jauh lebih efisien.

c. Mobilitas

   Kemampuan *roaming* dari IPv4 saat beralih satu jaringan ke jaringan yang lain terbatas terhadap mobilitas sedangkan IPv6 tetap terjaga kelangsungan sambungan karena memenuhi kebutuhan mobilitas yang tinggi.

d. Keamanan

   IPsec sama-sama digunakan sebagai pengaman jaringan baik pada IPv4 dan IPv6, yang membedakan yaitu *IPsec* merupakan fitur tambahan pada standar IPv4 sedangkan pada IPv6 merupakan fitur wajib pada standar implementasi IPv6.

e. Ukuran *Header*

   Ukuran *header* dasar dari IPv4 yaitu 20 oktet sedangkan pada IPv6 ukuran tetap 40 oktet.

Bit *network* dan bit *host* merupakan bagian dari alamat IP yang dipisahkan. Kedua bagian tersebut mempunyai perannya masing-masing. Identifikasi suatu *network* dari *network* yang lain merupakan peran dari bit *network* sedangkan identifikasi *host* dalam suatu *network* merupakan peran dari bit *host*. Bit *network* yang sama akan dimiliki oleh seluruh *host* yang tersambung dalam jaringan yang sama. Alamat IP dibagi menjadi 3 kelas alamat utama dalam TCP/IP yaitu:

a. Kelas A (0-127)

   Kelas A memiliki jumlah *network* 128 dengan jumlah *host* per *network* 16.777.216. Bit pertama dari kelas A yaitu 0. Bit *network* terletak pada

8-bit pertama dan bit *host* terletak pada 24-bit selanjutnya (setelah bit *network*).

b. Kelas B (128-191)

   Kelas B memiliki jumlah *network* 16.384 dengan jumlah *host* per *network* 65.536. Bit pertama dari kelas A yaitu 10. Bit *network* terletak pada 16-bit pertama dan bit *host* terletak pada 16-bit selanjutnya (setelah bit *network*).

c. Kelas C (192-223)

   Kelas C memiliki jumlah *network* 2.097.152 dengan jumlah *host* per *network* 254. Bit pertama dari kelas A yaitu 110. Bit *network* terletak pada 24-bit pertama dan bit *host* terletak pada 8-bit selanjutnya (setelah bit *network*) [14].

2.2.2.6 Subnet Mask

Subnet mask adalah sebuah teknik khusus yang bertujuan untuk memecahkan maupun membagi jaringan pada komputer sehingga bisa jadi *subnetwork-subnetwork* yang memiliki ukuran lebih kecil lagi. (Dalam literatur Internet, *subnet* juga disebut sebagai jaringan IP atau hanya jaringan.) Alamat IP 223.1.1.0/24 memberikan *subnet* ini alamat, dan notasi /24 ("slash-24") menunjukkan *subnet mask*, yang menentukan alamat *subnet* sebagai 24 bit paling kiri dari nilai 32-bit. Tiga antarmuka host (223.1.1.1, 223.1.1.2, dan 223.1.1.3) dan satu antarmuka *router* (223.1.1.4) membentuk subnet 223.1.1.0/24 sebagai hasilnya. Setiap host baru yang terhubung ke jaringan 223.1.1.0/24 harus memiliki alamat dalam format 223.1.1.xxx.

Strategi penugasan alamat Internet dikenal sebagai *Classless Interdomain Routing* (CIDR) [RFC 463]. CIDR menggeneralisasi gagasan pengalamatan *subnet*. Alamat IP 32-bit dibagi menjadi dua bagian, seperti pengalamatan *subnet*, dan memiliki bentuk desimal putus-putus yang sama a.b.c.d/x, di mana x adalah jumlah bit di bagian pertama alamat. Komponen jaringan dari alamat IP, juga dikenal sebagai *prefix* (atau awalan jaringan) dari alamat itu, terdiri dari bit x yang paling penting dan memiliki bentuk a.b.c.d/x. Blok alamat yang berdekatan, atau rentang alamat dengan awalan

serupa, biasanya ditetapkan ke organisasi (lihat fitur Prinsip dalam Praktik). Dalam skenario ini, semua alamat IP perangkat organisasi akan memiliki awalan yang sama. Karena *subnet* dengan alamat *subnet* 8-, 16-, dan 24-bit masing-masing disebut sebagai jaringan kelas A, B, dan C, bagian jaringan dari alamat IP dibatasi hingga 8, 16, atau 24 bit. Metode pengalamatan ini dikenal sebagai *classful addressing*. Persyaratan bahwa bagian *subnet* dari alamat IP harus tepat 1, 2, atau 3 *byte* panjang ternyata bermasalah untuk mendukung jumlah organisasi yang berkembang pesat dengan *subnet* kecil dan menengah. *Subnet* kelas C (/24) hanya dapat menampung hingga 2 − 2 = 254 *host* (dua dari 2 = 256 alamat dicadangkan untuk penggunaan khusus) terlalu kecil untuk banyak organisasi. Namun, *subnet* kelas B (/ 16), yang mendukung hingga 65.634 *host*, terlalu besar. Di bawah pengalamatan *classful*, sebuah organisasi dengan, katakanlah, 2.000 *host* biasanya dialokasikan alamat *subnet* kelas B (/ 16). Hal ini menyebabkan penipisan cepat ruang alamat kelas B dan pemanfaatan ruang alamat yang ditetapkan yang buruk. Misalnya, organisasi yang menggunakan alamat kelas B untuk 2.000 *host*-nya dialokasikan cukup dari ruang alamat hingga 65.534 antarmuka meninggalkan lebih dari 63.000 alamat yang tidak dapat digunakan oleh organisasi lain[15]. Pada Tabel 2.2 berikut merupakan *subnetting* alamat IP kelas C.

Tabel 2. 3 *Subnetting* Alamat IP Kelas C

Prefix	/24	/25	/26	/27	/28	/29	/30	/31	/32
Jumlah IP	256	128	64	32	16	8	4	2	1
Subnetmask	0	128	192	224	240	248	252	254	255

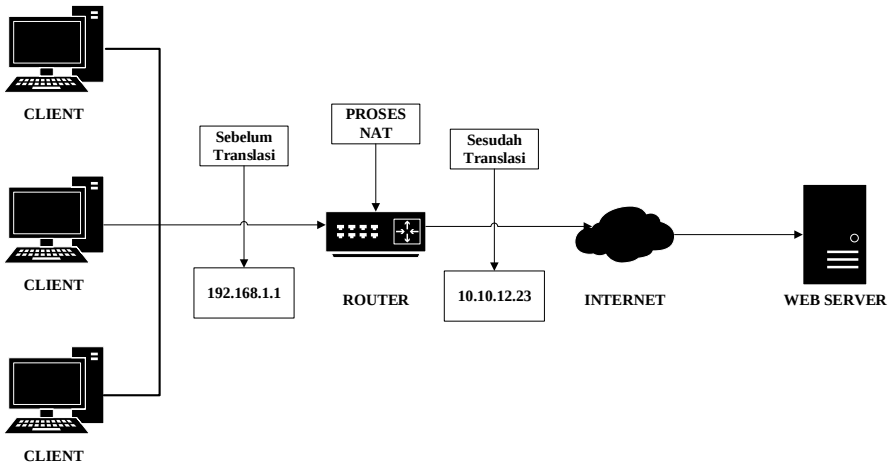
2.2.2.7 Network Address Translation (NAT)

Masalah kehabisan alamat IP bukanlah masalah teoritis yang mungkin terjadi di beberapa titik di masa depan yang jauh. Itu terjadi di sini dan saat ini. Solusi jangka panjangnya adalah seluruh Internet bermigrasi ke IPv6, yang memiliki alamat 128-bit. Transisi ini perlahan-lahan terjadi, tetapi akan bertahun-tahun sebelum prosesnya selesai. Untuk bertahan sementara itu, perbaikan cepat diperlukan. Perbaikan cepat yang banyak

digunakan saat ini datang dalam bentuk NAT (*Network Address Translation*), yang dijelaskan dalam RFC 3022.

Ide dasar di balik NAT adalah agar ISP menetapkan setiap rumah atau bisnis satu alamat IP (atau paling banyak, sejumlah kecil dari mereka) untuk lalu lintas Internet. Dalam jaringan pelanggan, setiap komputer mendapatkan alamat IP unik, yang digunakan untuk merutekan lalu lintas intramural. Namun, tepat sebelum paket keluar dari jaringan pelanggan dan pergi ke ISP, terjemahan alamat dari alamat IP internal yang unik ke alamat IP publik bersama terjadi. Terjemahan ini menggunakan tiga rentang alamat IP yang telah dinyatakan sebagai pribadi. Jaringan dapat menggunakannya secara internal sesuai keinginan. Satu-satunya aturan adalah bahwa tidak ada paket yang berisi alamat ini dapat muncul di Internet itu sendiri. Berikut merupakan rentang dari alamat IP:

a. 10.0.0.0 – 10.255.255.255/8 (16.777.216 Hosts)
b. 172.16.0.0 – 172.31.255.255/12 (1.048.576 Hosts)
c. 192.168.0.0 – 192.168.255.255/16 (65.536 Hosts)



Gambar 2. 10 NAT

Pengoperasian NAT ditunjukkan pada Gambar 2.10. Di dalam tempat pelanggan, setiap mesin memiliki alamat unik dari formulir 10.x.y.z. Namun, sebelum paket meninggalkan tempat pelanggan, ia melewati kotak. Kotak NAT sering digabungkan dalam satu perangkat dengan *firewall*, yang

memberikan keamanan dengan hati-hati mengontrol apa yang masuk ke jaringan pelanggan dan apa yang keluar darinya [10].

2.2.2.8 Port

Selain menggunakan protokol IP, dalam komunikasi jaringan komputer juga menggunakan protokol TCP/UDP yang lebih dikenal sebagai protokol jaringan TCP/IP karena kedua protokol tersebut bekerja bersama-sama sesuai dengan *layer* masing-masing. Dalam protokol jaringan TCP/IP, untuk mengidentifikasi aplikasi dan layanan yang menggunakan koneksi diperlukan yang namanya *port* karena TCP dan UDP menggunakan sistem *port* untuk hal koneksi. Oleh karena itu di dalam *header* paketnya, protokol TCP/UDP akan menambahkan *port* asal dan tujuan.

Komputer dapat mengaktifkan banyak sesi koneksi dengan komputer lain dan program perangkat lunak di jaringan berkat perangkat yang disebut *port*. Komputer atau program yang saling berkomunikasi dalam jaringan dengan menggunakan gabungan dari *port* dan alamat IP, hal tersebut dikenal dengan *socket address*. Nomor *port*, pengidentifikasi 16-bit (dua *byte*) untuk *port*, dibagi menjadi *port* TCP dan UDP tergantung pada jenis protokol *transport* yang digunakan. Jumlah total maksimum *port* untuk setiap protokol *transport* yang digunakan adalah 65.536 karena memiliki angka 16-bit. Menurut nomor *port*, ada tiga jenis *port* UDP dan TCP:

a. *Well-known Port* : yang pada awalnya hanya mengizinkan bilangan bulat dari 0 hingga 255 tetapi kemudian menambahkan dukungan untuk nilai hingga 1023. Nomor *port* terkenal diberikan oleh *Internet Assigned Numbers Authority* (IANA) untuk menunjukkan layanan jaringan yang sama setiap saat. Beberapa *port* dalam rentang nomor *port* dicadangkan untuk digunakan oleh layanan yang akan datang tetapi belum ditetapkan.
b. *Registered Port* : Sistem operasi dan aplikasi yang dibuat oleh pemasok komputer atau jaringan yang berbeda didukung melalui *port* terdaftar. Karena *port* terdaftar diketahui dan didaftarkan oleh IANA tetapi tidak ditetapkan secara permanen, vendor lain diizinkan untuk menggunakan

nomor *port* yang sama. Beberapa dari mereka telah didefinisikan secara dinamis, dengan kisaran dari 1.024 hingga 49.151.

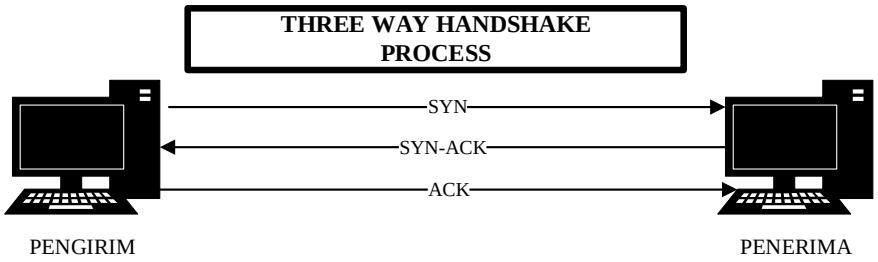
c. *Dynamically Assigned Port* : Ketika pengguna masuk, permintaan ditangani pada *Dynamically Assigned Port* oleh sistem operasi atau aplikasi. *Port* dapat ditambahkan atau dihapus sesuai kebutuhan dan ukurannya berkisar dari 1.024 hingga 65.536 [14].

2.2.2.9 *Transmission Control Protocol* (TCP)

Pada lapisan *transport*, *Transmission Control Protocol* (TCP) adalah protokol berorientasi koneksi yang dapat diandalkan. Spesifikasi TCP RFC 793. Ketika protokol lapisan aplikasi menuntut layanan transportasi data yang dapat diandalkan yang tidak ditawarkan oleh protokol lapisan aplikasi itu sendiri, TCP sering digunakan. Contoh protokol berbasis TCP adalah HTTP dan FTP. TCP memiliki sejumlah ciri, termasuk:

a. Berorientasi pada sambungan (*connection-oriented*), dua proses yang berjalan pada lapisan aplikasi harus terlebih dahulu bernegosiasi untuk membuat sesi koneksi agar data dapat dikirim antara dua *host*. Mekanisme penghentian koneksi TCP digunakan untuk mengakhiri koneksi TCP.
b. *Full Duplex*, koneksi antara dua *host* TCP untuk setiap *host* terdiri dari dua jalur, jalur masuk dan jalur keluar.
c. *Reliable*, nomor urut paket akan digunakan untuk mengurutkan data yang diberikan ke koneksi TCP, yang kemudian akan menunggu paket *positive acknowledgment* dari tujuan. Segmen TCP (unit data protokol dalam protokol TCP) akan dikirim ulang jika penerima tidak mengirim paket *acknowledgment* apa pun. Singkatnya ketika paket sudah diterima sesuai dengan nomor urut yang benar maka penerima akan mengirimkan pemberitahuan kepada pengirim seperti yang diilustrasikan pada Gambar 2.11.
d. *Byte Stream*, aliran *byte* yang berdekatan adalah bagaimana TCP menafsirkan data yang dikirim dan diterima melalui dua jalur masuk dan keluar TCP.

e. TCP mengimplementasikan layanan *flow control* milik pengirim yang terus memantau dan mengatur jumlah data yang dikirim dalam satu waktu untuk mencegah terlalu banyak data dikirim sekaligus, yang akhirnya menyebabkan jaringan *internetwork* IP macet [16].
f. TCP sering digunakan dalam aplikasi yang membutuhkan keandalan tinggi dan kecepatan transfer data yang cukup cepat. *Browsing* dan transfer data merupakan contoh aplikasi yang biasa menggunakan protokol TCP [17].



Gambar 2. 11 Cara Kerja TCP

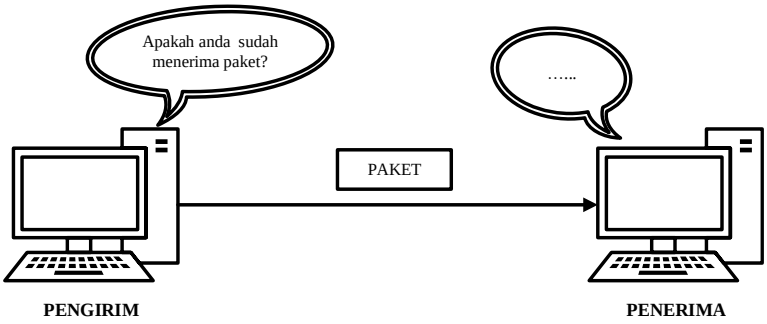
2.2.2.10 *User Datagram Protocol* (UDP)

Salah satu protokol lapisan transport TCP/IP yang dikenal sebagai *User Datagram Protocol* (UDP) memungkinkan komunikasi tanpa koneksi dan tidak stabil antara host-host pada jaringan TCP/IP. Aplikasi dapat mengirimkan data ke aplikasi lain di host yang berbeda di jaringan menggunakan UDP tanpa terlebih dahulu membuat koneksi dengan host tersebut. Transfer data yang berhasil dan transmisi data duplikat tidak dijamin oleh UDP. Berikut beberapa tugas yang sering UDP jalankan:

a. Protokol yang "*ring*an": Beberapa protokol lapisan aplikasi menuntut penggunaan protokol *ring*an yang dapat menjalankan operasi tertentu dengan bertukar pesan satu sama lain untuk menghemat memori dan memproses sumber daya.
b. Layanan transfer data yang andal yang disediakan oleh protokol lapisan aplikasi: Jika protokol lapisan aplikasi menawarkan layanan transfer data yang andal, fungsionalitas keandalan TCP tidak lagi diperlukan.
c. Transmisi *Broadcast*: Transmisi siaran juga dimungkinkan dengan UDP karena UDP adalah protokol yang tidak memerlukan pembuatan

koneksi dengan *host* tertentu terlebih dahulu. Paket data dapat dikirim dari protokol lapisan aplikasi ke berbagai lokasi menggunakan alamat *multicast* atau *broadcast*. Sebaliknya, protokol TCP hanya dapat mengirim transmisi satu ke satu [16].

Aplikasi seperti *game*, *streaming*, dan VOIP yang membutuhkan transfer data yang sangat cepat sering kali menggunakan UDP [17]. Kebalikan dari TCP yang menyelesaikan pekerjaan dengan menggunakan nomor urut pada paket datanya, sedangkan UDP tidak seperti itu. UDP tidak menyelesaikan pekerjaannya tanpa memerlukan nomor urut sehingga UDP tidak mengoreksi kesalahan pada saat pengiriman serta tidak peduli apabila terdapat paket yang hilang seperti yang diilustrasikan pada Gambar 2.12.



Gambar 2. 12 Cara Kerja UDP

2.2.2.11 *Dynamic Host Configuration Protocol (DHCP)*

Protokol yang secara otomatis melakukan penyebaran alamat IP ke perangkat lain biasa disebut sebagai DHCP atau *Dynamic Host Configuration Protocol*. Agar setiap komputer mendapatkan alamat IP maka harus melakukan konfigurasi secara manual jika tidak menggunakan DHCP sehingga akan merepotkan dan tidak efisien waktu. Sedangkan dengan menggunakan DHCP, alamat IP secara otomatis akan dibagikan ke banyak perangkat dalam satu waktu sehingga lebih efisien dan praktis.

Terdapat istilah *server* dan *client* dalam sebuah jaringan. Perangkat yang berfungsinya untuk menyimpan, mengelola, dan mengatur segala data dan aktivitas pada suatu jaringan disebut sebagai *server*. Sedangkan perangkat yang meminta layanan dari *server* disebut sebagai *client*. Pada

DHCP terdapat dua istilah yaitu DHCP *Server* dan DHCP *Client*, berikut perbedaannya:

a. DHCP *Server*: Mengatur dan memberikan alamat IP khusus yang dinamis secara otomatis kepada komputer *client* sehingga komputer *client* atau admin jaringan tidak perlu melakukan konfigurasi sendiri secara manual. Hal tersebut merupakan salah satu keuntungan dari DHCP *Server*.
b. DHCP *Client*: perangkat seperti komputer atau *handphone* yang menerima alamat IP dari DHCP *server* disebut sebagai DHCP *client* [18].

2.2.3 Perangkat *Routing*

2.2.3.1. *Router*

Perangkat yang biasanya berbentuk kotak ini yang terlihat seperti pada Gambar 2.13, berfungsi sebagai koneksi jaringan untuk perangkat akhir seperti PC. Perangkat ini dapat memiliki 4 *port* atau hingga 52 *port* seperti yang terlihat pada Gambar 2.14, dan beberapa jenis *switch* bahkan dapat ditumpuk untuk menyediakan sejumlah besar *port* bila diperlukan.

Gambar 2. 13 *Router*

Router adalah perangkat keras dengan fungsi mengambil paket, menganalisisnya, dan mengirimkannya ke jaringan lain. Selain itu, perangkat dapat digunakan untuk menghalangi, mengonversi paket ke antarmuka jaringan lain, dan melakukan tugas terkait jaringan lainnya [19].

Gambar 2. 14 *Port* pada *Router*

Dalam jaringan komputer, *router* biasanya merupakan alat yang beroperasi pada lapisan jaringan lapisan OSI. Tabel *routing*, yang merupakan tabel yang berisi alamat jaringan yang diperlukan untuk melayani tujuan paket data yang akan diteruskan pada jaringan, hadir di *router* ini. Memanfaatkan suatu bentuk sistem operasi, seperti *Windows*, *Unix*, *Linux,* atau jenis sistem operasi lain pada komputer PC kita hanya membutuhkan penambahan dua *Network Interface Card* (NIC) untuk membuat *router*. Satu lagi NIC dapat ditambahkan jika komputer kita sudah memilikinya, seperti PC yang ada di dalamnya. Dengan menggunakan *router* ini, kita dapat membangun jaringan LAN dengan beberapa kelas, seperti kelas B dan kelas C atau lainnya [20].

Berdasarkan alamat asal dan alamat tujuan, *router* akan mencari cara paling efisien untuk menyampaikan pesan. *Router* mengetahui alamat *IP* dari semua perangkat yang terhubung ke jaringan lokalnya, serta alamat *router* dan jembatan lainnya. Dengan mengamati sisi jaringan mana yang paling sibuk, *router* juga dapat menentukan seluruh jaringan dan menarik data dari sisi sibuk itu sampai sisi itu jelas [21].

2.2.3.2. Mikrotik *Router*board 951Ui-2HND

Sebuah perusahaan Latvia bernama Mikrotik menjual perangkat keras untuk jaringan komputer. Produk unggulan perusahaan adalah Mikrotik *Router* OS. Sistem operasi berbasis *Linux* ini dirancang untuk mampu mengendalikan jaringan komputer skala kecil, menengah, dan besar. Kemampuan produk router ini untuk berfungsi sebagai *gateway*

hotspot captative hanyalah salah satu dari banyak manfaatnya. Fitur ini memungkinkan Mikrotik untuk mengarahkan pengguna yang terhubung ke jaringan *hotspot* ke alamat *web* tertentu. Dari segi pemasaran, Mikrotik ditawarkan dalam berbagai format, seperti *routerboard*, perangkat nirkabel, dan sistem operasi yang mungkin dapat diinstal langsung pada PC [22].

Perangkat jaringan komputer yang disebut Mikrotik terdiri dari perangkat keras dan perangkat lunak dan dapat digunakan sebagai *router*, serta untuk penya*ring*an, *switching*, dan keperluan lainnya. Perangkat keras Mikrotik tersedia sebagai *Router Board* atau sebagai *Router* PC yang diinstal pada komputer (telah dibangun langsung dari perusahaan Mikrotik) [23]. Mikrotik biasanya terdiri dari 5 *port Ethernet* seperti yang terlihat pada Gambar 2.15.

Gambar 2. 15 *Router* Mikrotik

PC biasa dapat menjadi *router* dengan menggunakan sistem operasi dan *software* yang bernama Mikrotik. Hanya dengan *board* yang dilengkapi dengan OS Mikrotik, Mikrotik dapat dibedakan sebagai sistem operasi (OS) dan *board* yang tidak memerlukan komputer. Fungsi khusus untuk jaringan IP dan jaringan nirkabel termasuk dalam OS Mikrotik.

Berbagai jenis dari mikrotik:

a. *Router* Mikrotik. Dalam bentuk perangkat lunak yang diinstal melalui CD untuk komputer.
b. *Built in Hardware* Mikrotik. Berupa *hardware* yang dibundel di dalam *router* board dan berisi OS Mikrotik. Papan *router* Mikrotik memiliki lisensi bawaan [24].

Sedangkan untuk fitur-fitur dari Mikrotik adalah sebagai berikut :

a. *Address List*: Mengelompokkan alamat IP berdasarkan nama.
b. *Asynchronous*: Mendukung *dial-on-demand*, kumpulan modem hingga 128 *port, dial-in/dial-out* melalui serial PPP, otentikasi menggunakan Radius, CHAP, PAP, MSCHAPv1 dan MSCHAPv2, CHAP, dan PAP.
c. *Bonding*: Mendukung penggabungan beberapa antarmuka *Ethernet* menjadi satu pipa melalui koneksi cepat.
d. *Bridge*: Mendukung beberapa antarmuka *bridge*, menjembatani *firewalling*, dan fungsi *bridge spinning tree*.
e. *Data Rate Management*: *Burst*, PCQ, RED, SFQ, antrian FIFO, CIR, MIR, dan batas *peer to peer* semuanya digunakan dalam *QoS* berbasis HTB.
f. DHCP: mendukung DHCP *leases* baik statis maupun dinamis, serta DHCP per antarmuka, DHCP *Relay*, DHCP *Client*, dan beberapa jaringan DHCP.
g. *Firewall* dan NAT: Koneksi *peer-to-peer*, NAT sumber, dan NAT tujuan didukung. Opsi pem-*filter*-an termasuk MAC, alamat IP, rentang port, protokol IP, dan pilihan pengaturan untuk protokol ICMP, bendera TCP, dan MSS.
h. *Hotspot* : *Gateway hotspot* berbasis RADIUS untuk autentikasi. Batasi kecepatan data, SSL, dan HTTPS didukung.
i. ISDN : Dukungan *dial-in*/*dial-out* ISDN. Radius mendukung autentikasi PAP, CHAP, MSCHAPv1, dan MSCHAPv2. mendukung protokol garis untuk bundel 128K, Cisco HDLC, x751, x75ui, dan x75bui.
j. *Monitoring*/*Accounting*: Laporan, log, dan statistik grafik untuk lalu lintas IP yang dapat diakses melalui HTTP [25].

2.2.3.3. Firewall

Firewall adalah perangkat lunak yang memisahkan lalu lintas jaringan yang aman dari yang tidak aman dengan memblokir komunikasi yang dianggap berbahaya. *Firewall* pada dasarnya diatur pada *router* yang berfungsi sebagai *gateway* antara jaringan lokal dan jaringan Internet.

Firewall dapat berupa perangkat keras atau perangkat lunak apa pun karena merupakan sistem keamanan. Berikut jenis-jenis *firewall* :

a. *Firewall* berbasis perangkat keras

   Firewall berbasis perangkat keras adalah bagian dari piranti keras yang merupakan bagian dari sistem jaringan, seperti *router*. Agar berfungsi dengan baik, *firewall* jenis ini perlu dikonfigurasi. *Firewall* harus mengetahui paket utama, sumber, dan tujuan agar berfungsi. Sistem akan membandingkan data secara *internal* menggunakan aturan yang telah ditetapkan sebelumnya. Kemudian menentukan data mana yang harus di-*drop* atau diteruskan ke lokasi yang diinginkan.

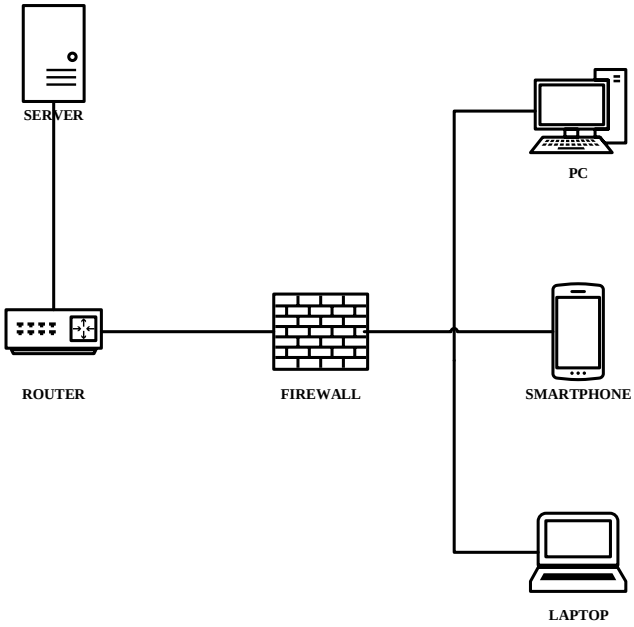
b. *Firewall* berbasis perangkat lunak

   Untuk pengguna internet rumahan, *firewall* berbasis perangkat lunak menjadi opsi lain mengenai keamanan jaringan. Biasanya, *firewall* ini dikembangkan sebagai program mandiri atau sebagai fungsi tambahan dari *anti-virus*. *Firewall* semacam ini melindungi dari *virus Trojan* dan *Worm* serta mengamankan lalu lintas masuk (*Inbound*) dan keluar (*Outbound*).

Cara kerja dari *firewall* yaitu data (paket) yang menghubungkan jaringan di internet disaring oleh *firewall*. Selain itu juga dapat mengizinkan atau melarang komputer mengakses paket. *Firewall* memiliki beberapa fungsi yaitu :

a. Melindungi informasi dari peretas dan pengguna tidak sah
b. *Firewall* bertindak sebagai barikade yang mencegah akses tidak sah ke data *internal* seperti yang diilustrasikan pada Gambar 2.16. Peretas dan orang luar tidak dapat mengakses data pengguna karena *firewall*. Dengan kata lain, jika komputer tidak memiliki *firewall* yang terpasang, data akan rentan terhadap pencurian.
c. *Block* Konten dan pesan yang tak diinginkan
d. Satu situs *web* atau konten dapat diblokir menggunakan *firewall*. Pengguna dapat secara manual menentukan jenis konten yang dilarang diakses melalui komputer.
e. Memantau *Bandwidth*

f. *Firewall* berguna untuk lebih dari sekadar menjaga keamanan jaringan komputer. Selain itu, *firewall* dapat digunakan untuk mengatur dan memantau penggunaan *bandwidth*. Misalnya pengguna dapat menetapkan batasan untuk musik, gambar, dan konten hiburan. Kemudian berikan konten yang lebih penting untuk bisnis prioritas yang lebih tinggi untuk *bandwidth*.
g. Mengakses layanan *Virtual Private Network* (VPN)
h. Koneksi *Virtual Private Network* (VPN) juga dapat difasilitasi oleh *firewall*. Pengguna memiliki akses ke jaringan *internal* mereka menggunakan layanan ini. Pengguna kini dapat mengakses situs *web* atau konten yang sebelumnya dilarang oleh pihak ketiga dengan menggunakan layanan yang sama. Produktivitas, kerja tim, dan pertukaran data semuanya dapat memperoleh manfaat dari VPN [26].



Gambar 2.16 Cara Kerja *Firewall*

2.2.3.4. *Packet Filtering*

Suatu bentuk teknologi keamanan yang dikenal sebagai *firewall packet filtering* digunakan untuk mengontrol paket mana yang diizinkan masuk ke sistem atau jaringan dan paket mana yang dilarang. Pem-*filter*-an paket sering digunakan untuk membatasi lalu lintas yang dicurigai berasal dari

alamat IP, nomor *port* TCP / UDP, dan jenis protokol aplikasi yang mencurigakan. Bagian *header* paket data, yang memberikan rincian penting tentang paket, adalah apa yang diperiksa.

a. Protokol, *byte* membentuk data yang disimpan di *header* ini. Informasi mengenai protokol terkandung dalam *byte* kesembilan.
b. Alamat IP sumber paket data 32-*byte* dikenal sebagai alamat IP sumber.
c. Alamat IP tujuan paket, yang berukuran 32 *byte*, adalah alamat IP yang dikirim.
d. *Port* dari mana paket data pengirim berasal dikenal sebagai *port* sumber (TCP atau UDP). Meskipun *port* TCP berbeda dan cukup jauh dari *port* UDP, *port* terhubung ke salah satu ujung koneksi TCP atau UDP. *Port* dengan angka di bawah 1024 dibalik karena sudah didefinisikan secara khusus, tetapi *port* dengan angka lebih dari atau sama dengan 1024 lebih sering disebut sebagai *port ephermal*. Konfigurasi nomor pengalamatan ini disediakan sesuai dengan preferensi vendor.
e. *Port* Tujuan, berfungsi sebagai paket data komputer yang menerima saluran masuk komputer.
f. Ketika paket data ditransfer melalui jaringan, status koneksi menunjukkan apakah itu adalah paket awal sesi. Ketika sebuah paket adalah paket awal, *header* TCP diatur ke "*false*" atau "0," dan *host* dicegah untuk membuat koneksi dengan menolak atau membuang paket dengan bit yang ditetapkan.

Cara *packet filtering* bekerja adalah *header* paket data diperiksa dengan membandingkannya dengan kebijakan yang telah ditentukan oleh *administrator* jaringan. Paket data yang tiba akan di *drop* jika salah satu aturan ini tidak sesuai dengan aturan yang sudah ditentukan [27].

2.2.4 Sistem Notifikasi

2.2.4.1. Telegram

Aplikasi pesan instan yang dikembangkan oleh perusahaan asal Rusia yang dikenal dengan nama Telegram. Aplikasi pesan instan tersebut digunakan oleh pengguna untuk saling berkirim pesan baik itu pesan teks,

suara, video, maupun dokumen. Aplikasi ini tersedia untuk berbagai macam sistem operasi seperti *android*, *ios*, *windows*, *linux*, dan lain-lain.

Telegram memiliki sebuah *Application Programming Interface* (API). API tersebut tersedia dan tak terbatas untuk umum. Berbeda dengan layanan pesan instan lainnya seperti *Whatsapp* dan *LINE*. *Whatsapp* tidak menawarkan API untuk publik, namun *LINE* menawarkan API dengan versi uji coba atau terbatas [28].

2.2.4.2. *Application Programming Interface* (API)

Application Programming Interface (API) memiliki fungsi yaitu dapat menghubungkan satu aplikasi dengan aplikasi lain (baik itu dari satu *platform* yang sama maupun *platform* yang berbeda). API adalah antarmuka yang memungkinkan pengguna untuk meningkatkan fungsionalitas yang telah disediakan oleh mekanisme ekstensi dalam berbagai cara dan ke berbagai tingkatan. Cara API membuat interaksi antara berbagai perantara perangkat lunak adalah dengan mendefinisikan jenis panggilan atau permintaan yang dapat dibuat, cara membuatnya, format data yang harus digunakan, dan aturan yang harus dipatuhi. Analogi API seperti pelayan di restoran, yang tugasnya menghubungkan pelanggan dan koki. Pelanggan memesan makanan sesuai dengan menu, yang disampaikan pelayan kepada juru masak, yang kemudian membawanya ke pelanggan.

Terdapat 4 jenis API yang sesuai dengan hak aksesnya berdasarkan pada penggunaannya yaitu:

a. *Public* API: sering dikenal sebagai *Open* API karena jenis API ini yang dapat digunakan oleh siapa saja dalam lintas *platfrom*. Selain itu, pengguna cukup mendaftar dan memanfaatkan langsung pada aplikasi pengguna yang mana hal tersebut menjadikan API jenis ini paling mudah digunakan. Contohnya API untuk *google maps* dan API untuk data covid Indonesia.
b. *Private* API: dalam pengembangan aplikasi diperlukan API jenis ini karena *private* API didesain untuk keperluan internal. Misal API dari yang digunakan untuk mengakses *frontend* dari sebuah *website* atau

aplikasi untuk pengembangan aplikasi *mobile*. Maka dari itu, jenis API ini disebut *private* API karena API jenis ini penggunaanya tidak diperuntukkan untuk umum, kebalikan dari API *public*.

c. *Partner* API: sama halnya dengan API *public*, dalam penggunaannya *Partner* API juga dapat diperuntukkan untuk umum namun pengguna harus memiliki izin terlebih dahulu kepada penyedia API. Setelah memiliki izin, baru pengguna dapat menggunakan aplikasi tersebut sesuai dengan perjanjian. Contohnya yaitu API Pinterest.
d. *Composite* API: dalam sekali akses dengan menggunakan API jenis ini, pengguna dapat berbagai jenis data sehingga sangat menghemat waktu bagi pengguna karena API jenis ini dapat menyimpan data dari berbagai *server* atau *hosting* dalam satu tempat [29].

2.2.4.3. Waktu Respon Notifikasi

Waktu Respon Notifikasi merupakan lama waktu yang dibutuhkan oleh sistem untuk menyampaikan informasi kepada *administrator*. Waktu respon notifikasi biasa diujikan dengan tujuan untuk mengetahui seberapa responsif sistem notifikasi bekerja dengan cara melakukan beberapa kali pengujian, kemudian akan diambil nilai rata-ratanya [30].

2.2.5 *Filter Rules*

Filter Rules adalah *firewall* yang menangani paket data yang ditujukan pada salah satu *interface router* dengan menentukkan apakah paket data tersebut dapat masuk ke dalam sistem mikrotik atau tidak. Pada *rules* ini terdapat parameter utama yang memiliki kegunaan untuk menentukan jenis trafik yang akan dikelola oleh fitur *firewall* yang dikenal dengan nama "*Chain*". *Filter rules* dapat diaktifkan dan dinonaktifkan oleh admin sesuai dengan keinginan.

2.2.5.1. *Chain*

Chain merupakan parameter utama pada *filter rules*, berfungsi menentukan jenis trafik yang akan dikelola oleh *firewall*. *Chain*

dianalogikan sebagai tempat admin mencegat sebuah trafik kemudian melakukan kebijakan sesuai kebutuhan. *Chain* sendiri memiliki 3 macam dalam *filter rules*, yaitu *forward*, *input*, dan *output*. Dari ketiga macam *chain* tersebut memiliki fungsi masing-masing yaitu sebagai berikut :

a. *Forward*: *Chain forward* digunakan untuk memproses lalu lintas paket data yang hanya berjalan melalui *router*. transfer data di jaringan lokal dan jaringan publik, misalnya. Misalnya, ketika orang menggunakan internet untuk *browsing*.
b. *Input*: Paket data yang berasal dari jaringan lokal atau jaringan publik akan ditangani oleh *router* melalui antarmukanya dan akan memiliki alamat IP router sebagai tujuan alamat IP-nya. *Chain* ini menangani pemrosesan aliran data.
c. *Output*: *Chain output* memproses lalu lintas data yang melewati salah satu antarmuka *router* dan merupakan kebalikan dari *chain input*. [31].

2.2.5.2. *Action*

Action merupakan langkah untuk mengeksekusi pada paket data dengan kriteria tertentu. Dengan kata lain *action* adalah jenis eksekusi. Salah satu *action* yang terdapat pada *filter rules* yaitu *drop*. *Drop* mempunyai fungsi yaitu menolak paket secara diam-diam (tidak melakukan pesan penolakan ICMP), sederhananya paket data yang masuk tidak diijinkan masuk jika memilih *action drop* [32].