BAB 2
LANDASAN TEORI

Pada bab ini akan membahas tentang teori dasar yang melandasi permasalahan dan penyelesaian yang diangkat dalam skripsi ini. Dasar teori yang diberikan meliputi keamanan jaringan, konsep dari IDS, SNORT, dan BASE.

2.1. Jaringan Komputer

Sejarah jaringan komputer berawal dari *time-sharing networks*, yaitu 'rangkaian' terminal yang terhubung dengan komputer sentral yang disebut *mainframe*. Contoh *time-sharing networks* adalah *IBM's System Network Architecture (SNA)* dan *Digital Network Architecture* [**4**].

Tujuan dari jaringan komputer adalah untuk melakukan komunikasi data, *sharing* data maupun pemakaian *resource* bersama seperti printer dan media penyimpanan sekunder. Komunikasi data sendiri memiliki tujuan yang lebih khusus [**5**], yaitu:

1. Memungkinkan pengiriman data dalam jumlah besar efisien, ekonomis, dan tanpa kesalahan dari suatu tempat ke tempat yang lain.
2. Memungkinkan penggunaan sistem komputer dan peralatan pendukung dari jarak jauh (*remote*).
3. Memungkinkan penggunaan komputer secara terpusat maupun secara tersebar sehingga mendukung manajemen dalam hal kontrol, baik desentralisasi ataupun sentralisasi.
4. Mempermudah kemungkinan pengelolaan dan pengaturan data yang ada dalam berbagai macam sistem komputer.
5. Mengurangi waktu untuk pengelolaan data.
6. Mendapatkan data langsung dari sumbernya.
7. Mempercepat penyebarluasan informasi.
8. Komunikasi data berkaitan dengan pertukaran data diantara dua perangkat yang terhubung secara langsung yang memungkinkan adanya pertukaran data antar kedua pihak.

2.2.Jenis Jaringa Komputer

Ada 3 macam jenis jaringan, yaitu:

1. Local Area Network (LAN)

   LAN adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantran di sebuah gedung atau sebuah sekolah dan tidak jauh dari sekitar 1 km persegi.

2. Metropolitas Area Network (MAN)

   MAN meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu provinsi. Dalam hal ini jaringan menghubungkan beberapa buah jaringan-jaringan kecil ke dalam lingkungan area yang lebih besar, sebagai conth yaitu jaringan bank dimana beberapa kantor cabang sebuah bank di dalam sebuah kota besar dihubungkan antara satu dengan lainnya.

3. Wide Area Network (WAN)

   WAN adalah jaringan yang lingkupnya sudah menggunakan media satelit atau kabel bawah laut, sebagai contoh keseluruhan jaringan bank yang ada di Indonesia atau yang ada di negara-negara lain.

2.1.1. Arsitektur Jaringan Komputer

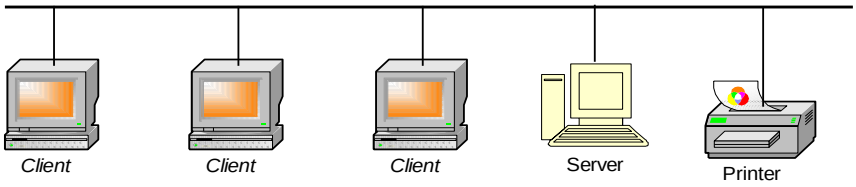
Arsitektur sebuah jaringan komputer dibedakan menjadi arsitektur fisik dan arsitektur logik. Arsitektur fisik berkaitan dengan susunan fisik sebuah jaringan komputer, bisa juga disebut dengan topologi fisik jaringan yaitu menjelaskan hubungan perkabelan dan lokasi node (simpul) atau *workstation*. Sedangkan arsitektur logik berkaitan dengan logika hubungan masing-masing komputer dalam jaringan atau menjelaskan aliran data dari satu *user* ke *user* lainnya dalam jaringan.

Topologi adalah bentuk hubungan dari suatu jaringan (*map of network*). Topologi fisik dari sebuah jaringan adalah merujuk pada konfigurasi kabel, komputer, dan perangkat lainnya. Terdapat tiga topologi fisik dasar yaitu: topologi *bus*, topologi *star*, dan topologi *ring.*

2.1.1.1 Topologi Bus

Pada topologi bus seluruh komputer dan *node* (simpul) dalam sebuah jaringan terhubung pada sebuah bus atau jalur komunikasi data (kabel). Piranti-piranti atau

simpul-simpul tersebut berkomunikasi dengan cara mengirim dan mengambil data di sepanjang bus tersebut. Topologi ini merupakan topologi jaringan paling sederhana dan biasanya jaringan ini menggunakan media yang berupa kabel koaksial. Karena seluruh proses komunikasi data menggunakan satu bus (jalur) saja maka topologi jaringan ini memiliki kelemahan pada tingkat komunikasi data yang cukup padat.



Gambar Error! No text of specified style in document.**.1 Topologi Fisik Bus**

Keuntungan dan kerugian topologi bus adalah sebagai berikut :

Keuntungan :

a. Mudah dikembangkan.
b. Jarak LAN tidak terbatas.
c. Keterandalan jaringan tinggi.
d. Kecepatan pengiriman tinggi.
e. Jumlah terminal dapat ditambah ataupun dikurangi tanpa menggangu operasi yang telah berjalan.
f. Tidak diperlukan pengendalian pusat.
g. Kondusif untuk konfigurasi jaringan pada gedung bertingkat

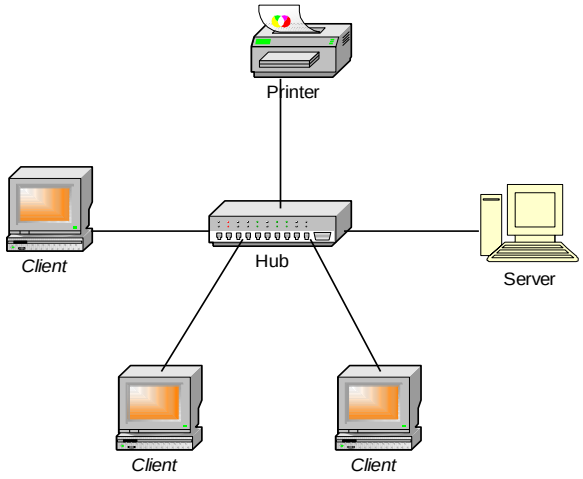
Kerugian :

a. Jika lalu lintas data terlalu tinggi dapat terjadi kemacetan (*Congestion*).
b. Diperlukan repeater untuk menguatkan sinyal pada pemasangan jarak jauh.
c. Operasional jaringan LAN bergantung pada setiap terminal.
d. Jika terjadi gangguan atau kerusakan pada salah satu lokasi (titik) dalam jaringan maka akan mempengaruhi jaringan secara keseluruhan, bahkan ada kemungkinan jaringan akan terhenti sama sekali

2.1.1.2 Topologi *Star* (Bintang)

Dalam topologi *star*, masing-masing komputer dalam jaringan dihubungkan ke pusat atau sentral dengan menggunakan jalur (bus) yang berbeda. Komunikasi pada jaringan diatur di sentral jaringan. Dengan digunakannya jalur yang berbeda untuk masing-masing komputer maka jika terjadi gangguan atau masalah pada salah satu titik dalam jaringan tidak akan mempengaruhi bagian jaringan yang lain. Hal ini juga memungkinkan pengaturan instalasi jaringan yang lebih fleksibel.

Pada gambar 2.2 digambarkan bahwa piranti-piranti atau simpul-simpul dalam jaringan tersebut terhubung ke pusat dengan menggunakan jalur masing-masing.



Gambar Error! No text of specified style in document..**2 Topologi Fisik *Star* (Bintang)**

Keuntungan dan kerugian topologi *star* adalah sebagai berikut :

Keuntungan :

a. Kehandalan topologi ini lebih besar dari topologi yang lain.
b. Mudah dikembangkan.
c. Keamanan data tinggi.
d. Kemudahan akses ke jaringan LAN lain.

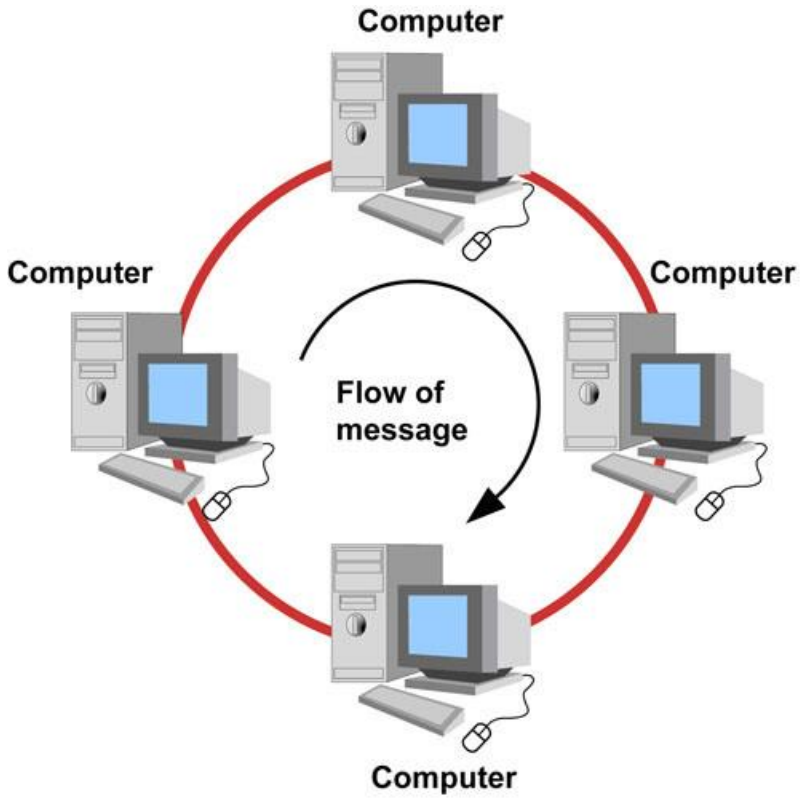
Kerugian :

a. Lalu lintas data yang padat menyebabkan jaringan lambat.

b. Jaringan tergantung pada terminal pusat.

2.1.1.3 Topologi *Ring* (Cincin)

Sesuai namanya *ring* atau cincin, maka pada topologi ini seluruh komputer dalam jaringan terhubung pada sebuah jalur data yang menghubungkan komputer satu dengan lainnya secara sambung-menyambung sedemikian rupa sehingga menyerupai sebuah cincin. Dalam sistem jaringan ini data dikirim secara berkeliling sepanjang jaringan. Setiap komputer yang akan mengirim data ke komputer lain dalam jaringan akan menempatkan data tersebut ke dalam cincin ini. Selanjutnya komputer yang dituju akan mengambil data tersebut dari cincin.



Gambar Error! No text of specified style in document.**..3 Topologi Fising *Ring* (cincin) [**6**]**

Keuntungan dan kerugian topologi *ring* adalah sebagai berikut :

Keuntungan :

a. Laju data (*transfer rate*) tinggi.
b. Dapat melayani lalu lintas data yang padat.
c. Tidak diperlukan *host*, relatif lebih murah.
d. Dapat melayani berbagai media pengiriman.
e. Komunikasi antar terminal mudah.
f. Waktu yang diperlukan untuk mengakses data optimal.

Kerugian :

a. Penambahan atau pengurangan terminal sangat sulit.
b. Kerusakan pada media pengiriman dapat menghentikan kerja seluruh jaringan.
c. Harus ada kemampuan untuk mendeteksi kesalahan dan metode pengisolasian kesalahan.
d. Kerusakan pada salah satu terminal dapat mengakibatkan kelumpuhan jaringan.
e. Tidak kondusif untuk pengiriman suara, gambar dan data.

2.1.2. Faktor-Faktor Yang Mempengaruhi Pemilihan Topologi Jaringan

Untuk membangun sebuah topologi jaringan, perlu diperhatikan beberapa faktor yang akan mempengaruhi proses yang terjadi dalam topologi jaringan yang akan dibangun, faktor-faktor tersebut yaitu:

1. Biaya.

   Biaya merupakan faktor mendasar untuk memutuskan dengan teknologi seperti apa topologi jaringan akan dibangun.
2. Kecepatan

   Kecepatan pada jaringan sangat erat kaitannya dengan efektifitas dan efisiensi pengalokasian sumberdaya *bandwidth* yang tersedia terhadap kesesuaian kecepatan yang dibutuhkan dalam sistem.
3. Lingkungan

Pembangunan infrastruktur jaringan sangat dipengaruhi lokasi dimana sistem tersebut dibangun sehingga perlu diperhatikan juga dalam pemilihan perangkat-perangkat jaringan yang digunakan

4. Ukuran

   Ukuran dilihat dari sejauh mana inftastruktur jaringan akan dibangun apakan berskala *enterprise* dengan memiliki banyak *server* ataukah tidak.

5. Konektivitas.

   Keandalan dari suatu infrastruktur jaringan juga perlu untuk diperhatikan guna mempermudah akses terhadap jaringan yang ada.

2.1.3. Konsep Topologi Jaringan

Dalam membangun sebuah topologi jaringan tidak selalu perangkat-perangkat yang terhubung mewakili cara akses dari setiap perangkat jaringan yang terhubung itu. Cara akses dan keterhubungan antar komponen jaringan dapat dipandang sebagai dua jenis topologi yang berbeda yang dilihat secara fisik dan *logic*.

Topologi fisik jaringan adalah cara yang digunakan untuk menghubungkan *workstation-workstation* di dalam LAN [**7**] atau dapat disebut sebagai suatu desain posisi dan letak *node* yang dilihat secara nyata untuk menentukan keterhubungan antar komponen jaringan melalui kabel atau media lainnya.

Sementara topologi logic jaringan adalah topologi yang dilihat dari metode accessnya [**7**] atau dapat disebut sebagai desain jaringan komputer untuk mentransmisikan data antar komputer yang dilihat dari cara kerjanya.

Sehingga dapat disimpulkan bahwa topologi jaringan jika dilihat dari keterhubungan antar komponen jaringan dan cara transmisi data yang berlangsung antar komputer dapat dibedakan sebagai topologi fisik jaringan dan topologi *logic* jaringan.

2.3.*Open Systems Interconnection* (OSI)

Model OSI digunakan untuk menjelaskan cara kerja jaringan komputer secara logika. Walaupun OSI merupakan sebuah model yang diakui di dunia saat

ini, namun tidak ada paksaan bagi pengembang *hardware* maupun *software* dan *user* untuk menggunakannya [**4**].

Model OSI terdiri atas 7 *layer* atau lapisan, berikut adalah tabel ketujuh *layer* tersebut berserta fungsinya:

Table Error! No text of specified style in document.**.1 OSI Layer [**4**]**

Layer	**Nama *Layer***	**Fungsi**
7	*Application*	Menyediakan *service* bagi berbagai aplikasi *network*
6	*Presentation*	Mengatur konversi dan translasi berbagai format data sepert kompresi data dan enkripsi data
5	*Session*	Mengatur sesi (*session*) yang meliputi *establishing* (memulai sesi), *maintaining* (mempertahankan sesi), dan *terminating* (mangakhiri sesi) antar entitas yang dimiliki oleh *presentation layer*
4	*Transport*	Menyediakan *end-to-end communication protocol*. Layer ini bertangggung jawab terhadap 'keselamatan data' dan 'segmentasi data', seperti: mengatur *flow control* (kendali aliran data), *error detection* (deteksi eror) *and correction* (koreksi), data *sequencing* (urutan data) dan *size of the packet* (ukuran paket)
3	*Network*	Mengatur rute yang dilalui oleh data. *Layer* ini menyediakan *logical addressing* (pengalamatan logika) dan *path determination* (penentuan rute tujuan)
2	*Data Link*	Menentukan pengalamatan fisik (*hardware address*), *error notification* (pendeteksi eror), *frame flow control* (kendali aliran *frame*) dan topologi *network*. Ada dua sublayer pada *data Link*, yaitu: *Logical Link Control* (LLC) dan *Media Access Control* (MAC) LLC mengatur komunikasi, seperti *error notification* dan *flow control*.

		Sedangkan MAC mangatur pengalamatan fisik yang digunakan dalam proses komunikasi antar-*adapter*.
1	*Physical*	*Layer* ini menentukan masalah kelistrikan/ gelombang/ medan dan berbagai prosedur/fungsi yang berkaitan dengan link fisik seperti besar tegangan/arus listrik, panjang maksimal media transmisi, pergantian fasa, jenis kabel dan konektor

2.4.Perangkat Jaringan

Perangkat jaringan merupakan alat atau piranti yang digunakan untuk membangun suatu sistem jaringan. Masing-masing perangkat jaringan memiliki fungsi dan tujuan tersendiri didalam suatu sistem jaringan. Pemilihan perangkat-perangkat jaringan yang diperlukan dapat disesuaikan dengan kebutuhan sistem jaringan yang akan dibangun.

2.1.4. Router

Router sering digunakan untuk menghubungkan beberapa *network*. Baik *network* yang sama maupun yang berbeda dari segi teknologinya. Seperti menghubungkan *network* yang menggunakan topologi *Bus*, *Star*, dan *Ring*. *Router* juga digunakan untuk membagi *network* besar menjadi beberapa buah *subnetwork* (*network-network* kecil). Setiap *subnetwork* seolah-olah "terisolir" dari *network* lain. Hal ini dapat membagi-bagi *traffic* yang akan berdampak positif pada performa *network* [**4**].

Sebuah *router* memiliki kemampuan *routing*. Artinya *router* secara cerdas dapat mengetahui kemana rute perjalanan informasi (yang disebut *packet*) akan dilewatkan. Apakah ditujukan untuk *host* lain dalam satu *network* ataukah berbeda network. Jika paket-paket ditujukan untuk *host* pada *network* yang sama maka *router* akan menghalangi paket-paket keluar, sehingga paket-paket tersebut tidak "membanjiri" *network* yang lain [**4**].

2.1.5. Mikrotik

Mikrotik adalah sebuah perusahaan yang bergerak di bidang produksi perangkat keras (hardware) dan perangkat lunak (Software) yang berhubungan dengan sistem jaringan komputer yang berkantor pusat di Latvia, bersebelahan dengan Rusia. Mikrotik didirikan pada tahun 1995 untuk mengembangkan router dan sistem ISP (Internet Service Provider) nirkabel.

Mikrotik dibuat oleh **MikroTikls** sebuah perusahaan di kota Riga, Latvia. Latvia adalah sebuah negara yang merupakan "pecahan" dari negara Uni Soviet dulunya atau Rusia sekarang ini. Mikrotik awalnya ditujukan untuk perusahaan jasa layanan Internet (PJI) atau *Internet Service Provider* (ISP) yang melayani pelanggannya menggunakan teknologi nirkabel atau *wireless*. Saat ini MikroTikls memberikan layanan kepada banyak ISP nirkabel untuk layanan akses Internet dibanyak negara di dunia dan juga sangat populer di Indonesia. MikroTik sekarang menyediakan hardware dan software untuk konektivitas internet di sebagian besar negara di seluruh dunia. Produk hardware unggulan Mikrotik berupa Router, Switch, Antena, dan perangkat pendukung lainnya. Sedangkan produk Software unggulan Mikrotik adalah MikroTik RouterOS.

2.1.1.4 RouterOS

adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan komputer manjadi router network yang handal, mencakup berbagai fitur yang dibuat untuk ip network dan jaringan wireless, cocok digunakan oleh ISP dan provider hotspot.

2.1.1.5 Routerboard

Router embedded produk dari mikrotik. Routerboard seperti sebuah pc mini yang terintegrasi karena dalam satu board tertanam prosesor, ram, rom, dan memori flash. Routerboard menggunakan os RouterOS yang berfungsi sebagai router jaringan, bandwidth management, proxy server, dhcp, dns server dan bisa juga berfungsi sebagai hotspot server.

Ada beberapa seri routerboard yang juga bisa berfungsi sebagai wifi. sebagai wifi access point, bridge, wds ataupun sebagai wifi client. seperti seri RB411, RB433, RB600. dan sebagian besar ISP wireless menggunakan

routerboard untuk menjalankan fungsi wirelessnya baik sebagai ap ataupun client. Dengan routerboard Anda bisa menjalankan fungsi sebuah router tanpa tergantung pada PC lagi. karena semua fungsi pada router sudah ada dalam routerboard. Jika dibandingkan dengan pc yang diinstal routerOS, routerboard ukurannya lebih kecil, lebih kompak dan hemat listrik karena hanya menggunakan adaptor. untuk digunakan di jaringan wifi bisa dipasang diatas tower dan menggunakan PoE sebagai sumber arusnya.

Mikrotik pada standar perangkat keras berbasiskan Personal Computer (PC) dikenal dengan kestabilan, kualitas kontrol dan fleksibilitas untuk berbagai jenis paket data dan penanganan proses rute atau lebih dikenal dengan istilah **routing**. Mikrotik yang dibuat sebagai router berbasiskan PC banyak bermanfaat untuk sebuah ISP yang ingin menjalankan beberapa aplikasi mulai dari hal yang paling ringan hingga tingkat lanjut. Contoh aplikasi yang dapat diterapkan dengan adanya Mikrotik selain *routing* adalah aplikasi kapasitas akses (*bandwidth*) manajemen,*firewall*, *wireless access point (WiFi)*, *backhaul link*, sistem *hotspot*, *Virtual Private Netword* (VPN)*server* dan masih banyak lainnya.

2.1.6. Bridge

Bridge atau kadangkala disebut *transparent bridge* merupakan perangkat *network* yang digunakan untuk menghubungkan dua buah LAN atau membagi sebuah LAN menjadi dua buah segmen. Tujuannya adalah untuk mengurangi *traffic* sedemikian rupa sehingga dapat meningkatkan performa *network* [**4**].

Bridge dan *router* secara sepintas terlihat sama dan dapat menghubungan dua buah LAN. Namun sesungguhnya cara kerja dan fungsi utama kedua perangkat tersebut berbeda [**4**].

Tabel Error! No text of specified style in document.**.1 Perbandingan *Router* dan *Bridge* [4]**

Router	**Bridge**
Mendukung berbagai *network protocol address*, seperti IP, IPX,	Tidak mendukung *network protocol address*. Hanya mengenali MAC

Apple Talk.	*address*.
Dapat menghubungkan beberapa subnet yang menggunakan teknologi yang berbeda-beda.	Menghubungkan dua buah segmen. Semua segmen dipandang sebagai sebuah subnet.
Mampu memblok *traffic* antar subnet.	Tidak dapat memblok *traffic* dari subnet lain.
Cocok digunakan pada sembarang protokol *network*.	Cocok digunakan pada protokol *non-routable* seperti NetBIOS dan DECnet.
Instalasi dan konfigurasi memerlukan keahlian khusus.	Instalasi relatif mudah, pasang dan nyalakan.
Cocok digunakan pada Internet dan Intranet.	Cocok digunakan pada LAN atau Intranet.

2.1.7. Switch

Dalam sebuah jaringan, *switch* berfungsi sebagai sentral atau konsentrator pada sebuah *network*. Cara kerja *switch* mirip dengan *bridge*, sehingga *switch* disebut sebagai *multiport bridge*. Ada dua arsitektur dasar yang digunakan pada switch, yaitu *cut-through* atau *fast forward* dan *store and forward*.

Cut-through merupakan jenis *switch* 'tercepat' diantara jenis lainnya. Switch jenis ini hanya mengecek alamat tujuan (yang ada pada *header frame*). Selanjutnya *frame* akan diteruskan ke *host* tujuan. Kondisi ini dapat mengurangi 'waktu tunggu' atau *latency*. Karena hanya mengecek alamat tujuan, kelemahan switch jenis ini yaitu tidak dapat mengecek *frame-frame* yang *error*. *Frame* yang *error* akan tetap diteruskan ke *host* tujuan [**4**].

Store and forward switch merupakan jenis *switch* yang akan menyimpan semua *frame* untuk sementara waktu sebelum diteruskan ke *host* tujuan. Seluruh *frame* akan dicek melalui mekanisme CRC (*Cyclic Redundancy Check*). Jika ditemukan *error* maka *frame* akan 'dibuang' dan tidak diteruskan ke *host* tujuan. *Switch* jenis ini paling 'terpercaya' diantara jenis lainnya. Kelemahan *switch* jenis

ini adalah meningkatnya *latency* akibat adanya proses pengecekan seluruh *frame* yang melalui *switch* [**4**].

2.1.8. Hub

Hub mirip dengan *switch* yaitu sebagai konsentrator. Namun, *hub* tidak ‘secerdas’ *switch*. Jika informasi dikirim ke *host* target melalui hub maka informasi akan mengalir ke semua *host*. Kondisi semacam ini dapat menyebabkan beban *traffic* yang tinggi. Oleh sebab itu, sebuah *hub* biasanya hanya digunakan pada *network* berskala kecil [**4**].

2.1.9. Repeater

Repeater termasuk satu dari perangkat keras jaringan komputer yang dipasang di titik-titik tertentu dalam jaringan untuk memperbaharui sinyal yang ditransmisikan agar mencapai kembali kekuatan dan bentuknya semula guna memperpanjang jarak tempuh. *Repeater* berfungsi untuk menguatkan sinyal.

2.1.10. Network Interface Card

Kartu Jaringan (NIC) merupakan salah satu dari perangkat keras yang menyediakan media untuk menghubungkan antar komputer. Kebanyakan kartu jaringan adalah kartu internal, yaitu kartu jaringan yang di pasang pada slot ekspansi di dalam komputer. Kartu jaringan yang banyak terpakai saat ini adalah: kartu jaringan *Ethernet*, *LocalTalk* konektor, dan kartu jaringan *Token Ring*.

2.5.Topologi Jaringan

Topologi jaringan merupakan suatu aturan atau cara untuk menghubungkan komputer yang satu dengan komputer yang lainnya sehingga membentuk suatu jaringan. Topologi jaringan dapat didefinisikan juga sebagai gambaran secara fisik dari pola hubungan antar komponen jaringan. [**8**]

Ada beberapa jenis topologi jaringan yang dikenal secara luas yang memiliki kelebihan dan kekurangan tersendiri. Topologi jaringan juga sangat berpengaruh terhadap performasi maupun kompleksitas jaringan yang dibangun,

oleh karena itu dalam pemilihan topologi jaringan harus disesuaikan dengan kebutuhan terhadap kesesuaian infrastruktur jaringan yang akan dibangun.

2.1.11. Faktor-Faktor Yang Mempengaruhi Pemilihan Topologi Jaringan

Untuk membangun sebuah topologi jaringan, perlu diperhatikan beberapa faktor yang akan mempengaruhi proses yang terjadi dalam topologi jaringan yang akan dibangun, faktor-faktor tersebut yaitu:

1. Biaya.

   Biaya merupakan faktor mendasar untuk memutuskan dengan teknologi seperti apa topologi jaringan akan dibangun.
2. Kecepatan

   Kecepatan pada jaringan sangat erat kaitannya dengan efektifitas dan efisiensi pengalokasian sumberdaya *bandwidth* yang tersedia terhadap kesesuaian kecepatan yang dibutuhkan dalam sistem.
3. Lingkungan

   Pembangunan infrastruktur jaringan sangat dipengaruhi lokasi dimana sistem tersebut dibangun sehingga perlu diperhatikan juga dalam pemilihan perangkat-perangkat jaringan yang digunakan

4. Ukuran

   Ukuran dilihat dari sejauh mana inftastruktur jaringan akan dibangun apakan berskala *enterprise* dengan memiliki banyak *server* ataukah tidak.
5. Konektivitas.

   Keandalan dari suatu infrastruktur jaringan juga perlu untuk diperhatikan guna mempermudah akses terhadap jaringan yang ada.

2.6. *Transmision Control Protocol / Internet Protocol* (TCP/IP)

TCP/IP adalah sekumpulan protokol yang terdapat di dalam jaringan komputer (*network*) yang digunakan untuk berkomunikasi atau bertukar data antar komputer. TCP/IP merupakan protokol standar pada jaringan internet yang

menghubungkan banyak komputer yang berbeda jenis mesin maupun sistem operasi agar dapat berinteraksi satu sama lain [**9**].

Konsep TCP/IP berawal dari kebutuhan DoD (*Departement of Defense*) USA akan suatu komunikasi di antara berbagai variasi komputer yang telah ada. Komputer-komputer DoD ini seringkali harus menghubungkan antara satu organisasi peneliti dengan organisasi peneliti lainnya. Komputer tersebut harus tetap berhubungan karena terkait dengan pertahanan negara dan sumber informasi harus tetap berjalan meskipun terjadi bencana alam besar, seperti ledakan nuklir. Oleh karenanya pada tahun 1969 dimulailah penelitian terhadap serangkaian protokol TCP/IP [**9**].

Adapun tujuan penelitian tersebut adalah sebagai berikut:

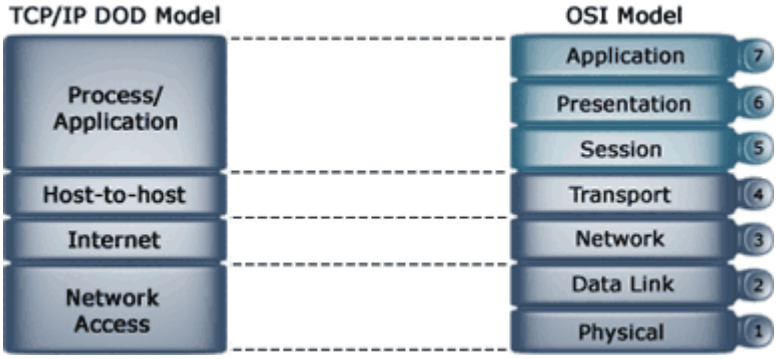
1. Terciptanya protokol-protokol umum, (DoD memerlukan suatu protokol yang dapat dipergunakan untuk semua jenis jaringan).
2. Meningkatkan efisiensi komunikasi data.
3. Dapat dipadukan dengan teknologi WAN (*Wide Area Network*) yang telah ada.
4. Mudah dikonfigurasi.

Tahun 1968 DoD ARPAnet (*Advanced Research Project Agency*) memulai penelitian yang kemudian menjadi cikal bakal *packet switching*. *Packet switching* inilah yang memungkinkan komunikasi antara lapisan *network*, dimana data dijalankan dan disalurkan melalui jaringan dalam bentuk unit-unit kecil yang disebut *packet*. Tiap-tiap paket ini membawa informasi alamat masing-masing yang ditangani dengan khusus oleh jaringan tersebut dan tidak tergantung dengan paket-paket lain. Jaringan yang dikembangkan ini adalah awal mula jaringan yang memiliki cakupan yang luas dan menjadi terkenal sebagi internet [**9**].

2.1.12. Arsitektur TPC/IP

Arsitektur rangkaian protokol TCP/IP didefinisikan dengan berbagai cara agar fungsi protokol-protokol TCP/IP tersebut dapat saling menyesuaikan. Protokol TCP/IP merupakan protokol standar yang terdapat pada Referensi Model

DoD atau TCP/IP maupun Refernesi Model OSI. Jika model OSI memiliki 7 lapisan *layer* maka pada model TCP/IP memiliki 4 lapisan *layer* yang diperlihatkan sebagai berikut:



Gambar Error! No text of specified style in document.**.4 TCP/IP Model dan OSI Model**

Hampir sama dengan OSI model, fungsi dari setiap layer pada TCP/IP model adalah sebagai berikut:

1. *Layer Application*

   Layer Application merupakan *layer* teratar dalam model TCP/IP *layer* ini bertanggung jawab untuk menyediakan akses kepada apliaksi terhadap layanan jaringan TCP/IP.

2. *Layer Host-to-Host* atau *layer Transport*

   Bertanggung jawab untuk komunikasi antar aplikasi dengan melakukan sesi koneksi yang bersifat *connection-oriented* (TCP) dan *connectionless* (UDP)

3. *Layer Internet*

   Layer internet ini berfungsi untuk melakukan pemetaan (*routing*) dan enkapsulasi *packet*. Salah satu protokol yang terlibat didalamnya adalah *Internet Protocol* (IP)

4. *Layer Network Access*

   Berfungsi untuk meletekkan *frame-frame* jaringan di atas media jaringan yang digunakan.

2.1.13. TCP dan IP

TCP dan IP merupakan protokol yang bekerja pada suatu *layer* dan menjadi penghubung antara satu komputer dengan komputer yang lain di dalam *network*, meskipun kedua komputer tersebut memiliki OS yang berbeda. Dalam melakukan komunikasi didalam jaringan, TCP dan IP memiliki tugas masing-masing yaitu sebagai berikut:

1. TCP, *Transmission Control Protocol* merupakan *connection-oriented*, yang berarti bahwa kedua komputer yang ikut serta dalam pertukaran data harus melakukan hubungan terlebih dulu sebelum pertukaran data berlangsung. Selain itu TCP juga bertagnggungjawab untuk meyakinkan bahwa data yang dikirimkan sampai ke tujuan, memeriksa kesalahan dan mengirimkan *error* ke lapisan atas jika TCP tidak berhasil melakukan hubungan. Jika ukuran data terlalu besar untuk satu datagram, TCP akan membaginya ke dalam beberapa datagram.
2. IP, *Internet Protocol* bertanggung jawab setelah hubungan berlangsung. Tugasnya adalah untuk me-rute-kan paket data di dalam *network*. IP hanya bertugas sebagai pengirim dari TCP dan mencari jalur yang terbaik dalam penyampaian datagram. IP tidak bertanggung jawab jika data tersebut tidak sampai dengan utuh, karena IP tidak memiliki informasi mengenai isi data yang dikirimkan, namun IP akan mengirimkan pesan kesalahan (*error message*) melalui ICMP (*Internet Control Message Protocol*) jika hal ini terjadi dan kemudian kembali ke sumber data berasal.

   Karena IP hanya mengirimkan data tanpa mengetahui urutan data mana yang akan disusun berikutnya, maka hal ini menyebabkan IP mudah untuk dimodifikasi di daerah sumber dan tujuan datagram. Hal inilah yang menyebabkan adanya paket data yang hilang sebelum sampai ke tujuan .

2.7.*Internet Protocol version* 4 (IPv4)

IP *address* merupakan pengenal atau identifier yang digunakan untuk memberi alamat pada tiap-tiap komputer dalam jaringan. Format IPv4 address adalah 32 bit yang tiap 8 bit-nya dipisahkan oleh tanda titik, format bakunya

adalah seperti demikian xxxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx, dengan x merupakan bilangan biner 0 atau 1. Atau dengan bentuk empat bilangan dengan titik sebagai pemisahnya seperti berikut xxx.xxx.xxx.xxx, di mana xxx merupakan nilai desimal dari 8 bit biner.

2.1.14. *Network ID* dan *Host ID*

Dalam 1 alamat IPv4 dibagi menjadi 2 bagain yaitu *network ID* dan *host ID* yang ditandai oleh *subnet mask*. Untuk menandai suatu *network ID subnet mask* di set ke angka 1 dan untuk menandai suatu *host ID subnet mask* di set ke angka 0, misalkan alamat 192.168.10.100 dengan subnet masuk 255.255.255.0 menandakan bahwa 192.168.10.0 merupakan alamat *network* dan .100 merupakan alamat *host*.

Alamat *host ID* dalam 1 *network ID* harus unik, karena *host ID* harus merupakan satu-satunya alamat yang dimiliki oleh suatu *host* dalam satu jaringan, jika terdapat 2 *host ID* dalam 1 *network ID* maka akan terjadi konflik dalam jaringan tersebut.

Alamat IPv4 dibagi menjadi 3 jenis yaitu :

1. Alamat *Unycast*

   Alamat ini digunakan dalam komunikasi *point to point.*
2. Alamat *Broadcast*

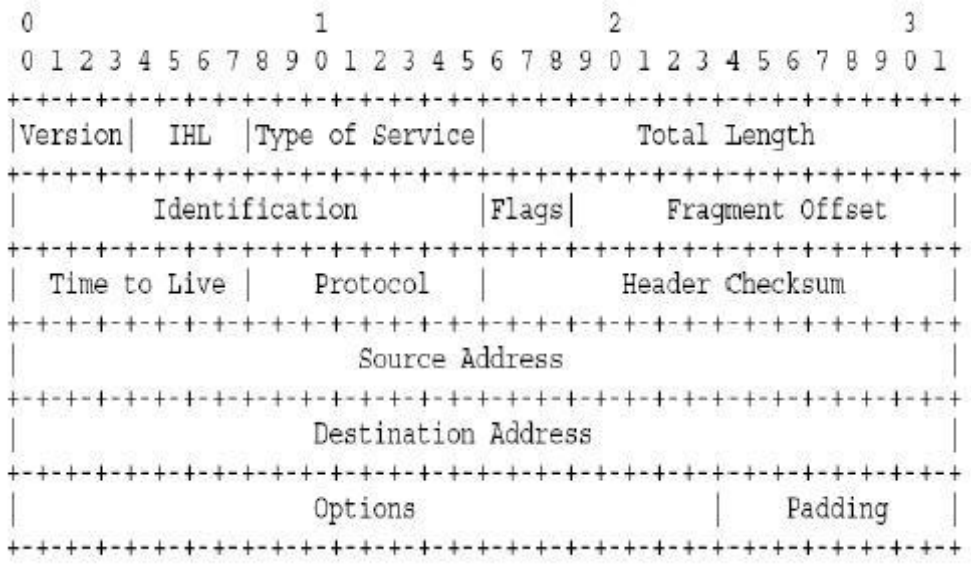
   Alamat ini digunakan dalam komunikasi *one to everyone.* Biasanya dipakai dalam satu segmen jaringan yang sama.
3. Alamat *Multicast*

   Alamat ini digunakan dalam komunikasi *one to many.*

2.1.15. Struktur *Header Internet Protocol version* 4

IPv4 dibentuk oleh struktur *header* yang memiliki 14 *field* dan panjang 32 bit. Setiap *field* memiliki fungsi dan tujuan masing-masing. Pada *header* IPv4 masih ditemukan kekurangan-kekurangan yang dapat dijadikan celah bagi penyusup atau orang-orang yang tidak bertanggungjawab untuk melakukan aksi pencurian data pada *packet* IPv4.

Berikut merupakan struktur *header* dari IPv4:



Gambar Error! No text of specified style in document.**.5 Format Header IPv4**

Header IPv4 terdiri atas beberapa *field*, dengan penjelasan sebagai berikut:

1. *Version*

   Digunakan sebagai tanda *version* dari *header* IP yang digunakan, yaitu IPv4.

2. *Internet Header Length*

   Digunakan untuk mengindikasikan ukuran *header* IP.

3. *Type of Service*

   Field ini digunakan untuk menentukan kualitas transmisi dari sebuah datagram IP.

4. *Total Length*

   Merupakan panjang total dari datagram IP, yang mencakup *header* IP dan muatannya.

5. *Identification*

   Digunakan untuk mengidentifikasi sebuah *packet* IP tertentu yang akan di fragmentasi.

6. *Flags*

   Berisi dua buah *flag* yang berisi apakah sebuah datagram IP mengalami fragmentasi atau tidak.

7. *Fragment Offset*

   Digunakan untuk mengidentifikasikan *offset* dimana fragmen yang bersangkutan dimulai, dihitung dari permulaan muatan IP yang belum pecah.

8. *Time to Live*

   Digunakan untuk mengidentifikasikan berapa banyak saluran jaringan dimana sebuah datagram IP dapat berjalan-jalan sebelum sebuah *router* mengabaikan datagram tersebut.

9. *Protocol*

   Digunakan untuk mengidentifikasikan jenis protokol lapisan yang lebih tinggi yang dikandung oleh muatan IP.

10. *Header Checksum*

    Field ini berguna hanya untuk melakukan pengecekan integritas terhadap *header* IP.

11. *Source IP Address*

    Field ini mengandung alamat IP dari sumber *Host* yang mengirimkan datagram IP tersebut.

12. *Destination IP Address*

    Field ini mengandung alamat IP tujuan kemana datagram IP tersebut akan disampaikan.

13. *Option*

    Fitur pendukung IP yang dapat dipilih, biasanya fitur kemanan.

14. *Padding*

    Bit-bit 0 tambahan yang ditambahkan ke dalam *field* ini untuk memastikan *header* IPv4 tetap berukuran 32 bit.

2.8.*Internet Protocol version* 6 (IPv6)

Internet Protocol version 6 atau IPv6 merupakan *internet protocol* yang dirancang untuk menggantikan IPv4. Jumlah 4.3 miliyar alamat yang dimiliki oleh IPv4 dirasa akan mengalami kekurangan untuk memenuhi jumlah kebutuhan

alamat IP. IPv6 mulai dikembangkan pada 14 Juli 1999 [**10**]. Selain untuk memenuhi jumlah alamat yang diperlukan, IPv6 juga didesain untuk memperbaiki kekurangan-kekurangan yang ada pada IPv4.

Kekurangan pada sisi jumlah alamat IP disebabkan oleh kebutuhan manusia yang kian hari kian meningkat serta alih fungsi yang terjadi pada jaringan yang sudah menjadi kebutuhan disebagian aspek kegiatan manusia.

2.1.16. Fitur IPv6

IPv6 memiliki beberapa fitur yang mampu mengantisipasi perkembangan aplikasi masa depan dan mengatasi kekurangan yang dimiliki IPv4. Fitur-fitur tersebut adalah [**9**]:

1. Jumlah IP address yang sangat banyak

   Sesuai dengan salah satu tujuannya, IPv6 dibuat untuk memenuhi jumlah kebutuhan alamat IP yang sangat banyak. Alamat IPv6 memiliki panjang 128 bit atau setara dengan 3.4 x 10 ^38.

2. *Autoconfiguration*

   IPv6 dirancang agar penggunanya tidak dipusingkan dengan konfigurasi IP *address*. Komputer pengguna yang terhubung dengan jaringan IPv6 akan mendapatkan IP *address* langsung dari *router* sehingga akan sangat berguna untuk peralatan *mobile* internet karena pengguna tidak direpotkan dengan konfigurasi sewaktu berpindah tempat dan jaringan.

3. *Security*

   Kebutuhan *security* yang hanya menjadi optional pada IPv4 telah menjadi suatu kebutuhan pada IPv6. IPv6 telah dilengkapi dengan protokol IPSec, sehingga semua aplikasi telah memiliki sekuriti yang optimal bagi berbagai aplikasi yang membutuhkan keamanan.

4. *Quality of Services*

   IPv6 memiliki protokol QoS yang terintegrasi dengan baik sehingga semua aplikasi yang berjalan di atas IPv6 memiliki jaminan QoS.

2.1.17. Jenis Alamat IPv6

Pada alamat IPv6, jenis alamat *broadcast* pada IPv4 ditiadakan, namun pada IPv6 muncul sebuah pengalamatan baru dengan nama *multicast address*. Sehingga jenis alamat yang terdapat pada IPv6 sama dengan IPv4 yaitu sebanyak 3 jenis dengan penjelasan sebagai berikut [**11**]:

1. *Unycast address*

   Unicast address merupakan jenis IP *address* yang digunakan untuk identifikasi sebuah *interface* saja. Paket yang dikirimkan ke *unicast address* hanya akan diterima oleh sebuah *interface* yang menggunakan alamat tersebut.

2. *Anycast address*

   Anycast address merupakaan jenis IP *address* yang digunakan untuk identifikasi sekumpulan *interface*. Paket yang dikirim ke *anycast address* akan diterima oleh *interface* terdekat (salah satu *interface*) dari sekumpulan *interface* yang menggunakan alamat tersebut. Penentuan *interface* terdekat adalah berdasarkan pengukuran jarak dari protokol *routing*. *Anycast address* tidak lain adalah *unicast address* yang diberikan pada sekumpulan *interface* dengan persyaratan khusus, yaitu:

   a. *Anycast address* hanya digunakan pada *router* IPv6 saja. Tidak boleh digunakan untuk alamat *host*.
   b. *Anycast address* tidak boleh digunakan sebagai alamat asal dari paket IPv6

3. *Multicast address*

   Multicast address digunakan untuk identifikasi sekumpulan *interface*. Paket yang dikirim ke *multicast address* akan diterima oleh semua *interface* yang menggunakan alamat tersebut. Secara umum, *multicast address* pada IPv6 berfungsi sama dengan multicast address pada IPv4.

2.1.18. Format Penulisan IPv6

Secara umum format penulisan IPv6 adalah x:x:x:x:x:x:x:x yang dipisahkan oleh *double-colon* (titik dua ‘:’) dan x yang merepresentasikan panjang

16 bit atau 2 octec kemudian diikuti dengan tanda slash (/) untuk menandakan prefix jaringan. Untuk menuliskan alamat IPv6, dapat dilakukan dengan cara sebagai berikut yaitu:

Alamat IPv6 : 2001:0716:0250:0000:0000:0000:0000:4/64

Dapat ditulis juga dengan bentuk:

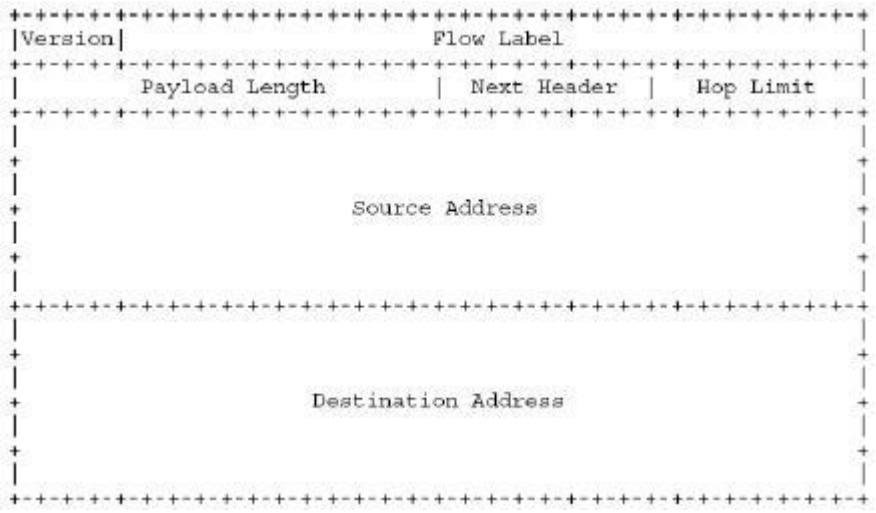
2001:716:250:0:0:0:0:4/64 atau 2001:716:250::4/64

Tanda /64 menunjukan bahwa 64 bit pertama merupakan alamat jaringan yang digunakan. Penulisan 2 *double-colon* menunjukan bahwa bit yang ada didalamnya adalah bit 0 dan penggunaan 2 *double-colon* tersebut tidak dapat diberikan sebanyak 2 kali pada 1 alamat IPv6 karena dapat menyebabkan alamat IPv6 menjadi tidak valid.

2.1.19. Struktur *Header Internet Protocol version* 6

IPv6 memiliki beberapa fitur yang mampu mengantisipasi perkembangan aplikasi masa depan dan mengatasi kekurangan yang dimiliki pendahulunya, yaitu IPv4. IPv6 dirancang sebagai perbaikan dari IPv4. *Header* pada IPv6 terdiri dari dua jenis, yang pertama, yaitu *field* yang dibutuhkan oleh setiap paket disebut *header* dasar, sedangkan yang kedua yaitu *field* yang tidak selalu diperlukan pada packet disebut *header* ekstensi, dan *header* ini didifinisikan terpisah dari *header* dasar. *Header* dasar selalu ada pada setiap paket, sedangkan *header* tambahan hanya jika diperlukan diselipkan antara *header* dasar dengan data. *Header* tambahan, saat ini didefinisikan selain bagi penggunaan ketika paket dipecah, juga didefinisikan bagi fungsi sekuriti dan lain-lain. *Header* tambahan ini, diletakkan setelah *header* dasar, jika dibutuhkan beberapa *header* maka *header* ini akan disambungkan berantai dimulai dari *header* dasar dan berakhir pada data. *Router* hanya perlu memproses *header* yang terkecil yang diperlukan saja, sehingga waktu pemrosesan menjadi lebih cepat. Hasil dari perbaikan ini, meskipun ukuran *header* dasar membesar dari 20 bytes menjadi 40 bytes namun jumlah *field* berkurang dari 12 menjadi 8 buah saja.

Berikut ini merupakan format *header* dari IPv6.



Gambar Error! No text of specified style in document.**.6 Format *Header* IPv6**

Field-field pada *header* IPv6 dapat dijelaskan secara singkat sebagai berikut:

1. *Version*

   Field 4 bit yang menunjukkan versi *Internet Protocol*, yaitu 6.

2. *Prior*

   Field 4 bit yang menunjukkan nilai prioritas. *Field* ini memungkinkan pengirim paket mengidentifikasi prioritas yang diinginkan untuk paket yang dikirimkan, relatif terhadap paket-paket lain dari pengirim yang sama.

3. *Flow Label*

   Field 24 bit yang digunakan oleh pengirim untuk memberi label pada paket-paket yang membutuhkan penanganan khusus dari *router* IPv6, seperti *quality of service* yang bukan default, misalnya *service-service* yang bersifat *real-time.*

4. *Payload Length*

   Field berisi 16 bit yang menunjukkan panjang *payload,* yaitu sisa paket yang mengikuti *header* IPv6, dalam oktet.

5. *Next Header*

   Field 8 bit yang berfungsi mengidentifikasi *header*

   Berikut ini merupakan yang mengikuti *header* IPv6 utama.

6. *Hop Limit*

   Field berisi 8 bit *unsigned integer*. Menunjukkan jumlah *link* maksimum yang akan dilewati paket sebelum dibuang. Paket akan dibuang bila *Hop Limit* berharga nol.

7. *Source Address*

   Field 128 bit, menunjukkan alamat pengirim paket.

8. *Destination Address*

   Field 128 bit, menunjukkan alamat penerima paket.

Di bawah ini tabel perbandingan antara IPv4 dan IPv6 :

Table Error! No text of specified style in document.**.2 Perbandingan IPv4 dan IPv6**

IPv4	**IPv6**
Panjang alamat 32 bit (4 bytes)	Panjang alamat 128 bit (16 bytes)
Konfigurasi secara manual atau DHCP IPv4.	Tidak harus dikonfigurasi secara manual, bisa menggunakan address *auto-configuration*.
Dukungan terhadap IPSec opsional	Dukungan terhadap IPSec dibutuhkan
Fragmentasi dilakukan oleh pengirim dan pada router, menurunkan kinerja router.	Fragmentasi dilakukan hanya oleh pengirim.
Tidak mensyaratkan ukuran paket pada link-layer dan harus bisa menyusun kembali paket berukuran 576 byte.	Paket link-layer harus mendukung ukuran paket 1280 byte dan harus bisa menyusun kembali paket berukuran 1500 byte
Untuk mengelola keanggotaan grup pada subnet lokal digunakan *Internet Group Management Protocol(*IGMP)	IGMP telah digantikan fungsinya oleh *Multicast Listener Discovery* (MLD)

Checksum termasuk pada *header*	Cheksum tidak masuk dalam *header*
Header mengandung *option*	Data opsional dimasukkan seluruhnya ke dalam *extensions header*.
Menggunakan ARP Request secara broadcast untuk menterjemahkan alamat IPv4 ke alamat link-layer	ARP Request telah digantikan oleh *Neighbor Solitcitation* secara multicast.

2.9.Dual-Stack

Dual-Stack merupakan strategi migrasi dengan menggunakan protokol IPv6 dan IPv4 dalam satu infrastruktur jaringan. Infrastruktur tersebut dapat berupa *host* komputer maupun perangkat jaringan lainnya seperti *router*. Sehingga *host* atau lingkungan jaringan yang menerapkan mekanisme *Dual-Stack* dapat menjalankan kedua protokol IPv6 dan IPv4 secara paralel sesuai dengan kebutuhan yang digunakan saat komunikasi berlangsung. [**12**]

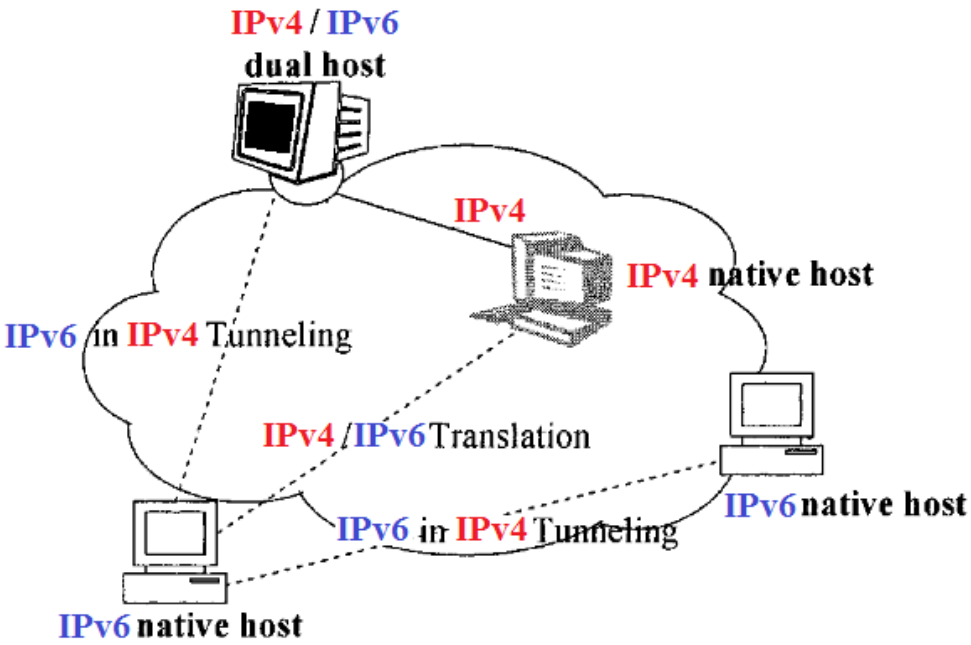
Sebagai contoh, misalkan *node Dual Stack* berkomunikasi dengan *IPv4-only node*, maka *node Dual Stack* tersebut akan menggunakan protokol IPv4 sebagai jalur komunikasinya dan sebaliknya misalkan *node Dual Stack* tersebut berkomunikasi dengan *IPv6-only node*, maka *node Dual Stack* tersebut akan menggunakan protokol IPv6 sebagai jalur komunikasinya. Namun jika kedua node tersebut sama-sama menerapkan mekanisme *dual stack* dan memiliki IPv6 maupun IPv4 maka komunikasi akan berlangsung berdasarkan protokol *preferred* yang dioperasikan. [**12**]

Walaupun kedua protokol digunakan secara bersama-sama, namun salah satu dari kedua protokol tersebut dapat di *disable*-kan karena suatu alasan tertentu. Terdapat 3 kemungkinan yang dapat dioperasikan terhadap *node Dual Stack*, yaitu:

1. *IPv4 stack enabled* dan *IPv6 stack disabled*.
2. *IPv6 stack enabled* dan *IPv4 stack disabled*.
3. *IPv4* dan *IPv6 stack enable* kedua-duanya. [**13**]

Dual Stack merupakan konsep transisi yang paling biasa digunakan, karena memungkinkan untuk kedua protokol berjalan masing-masing tanpa saling

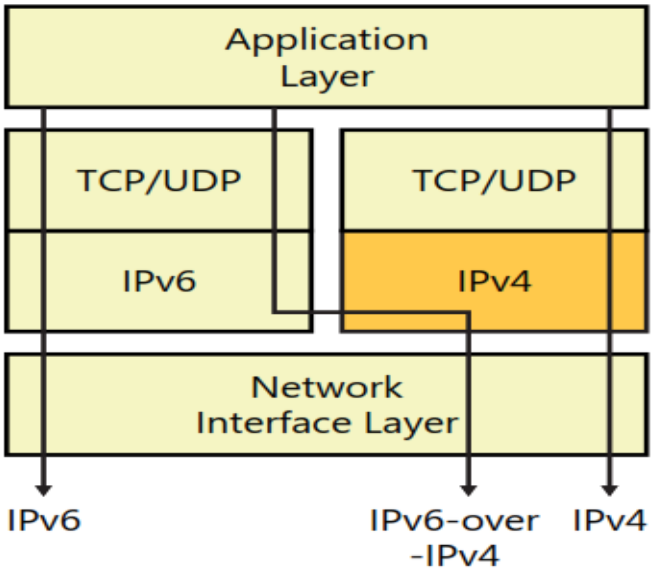
mempengaruhi. *Dual Stack* juga merupakan kemampuan dasar yang dimiliki oleh perangkat jaringan yang dapat beroperasi dengan IPv4 maupun IPv6.



Gambar Error! No text of specified style in document.**..7 Skenario Jaringan *Dual Stack* IPv4 dan IPv6 [**13**]**

2.1.20. Dual Stack Architecture

Arsitektur *Dual-Stack* terdiri dari IPv4 dan IPv6 *internet layers*, tetapi keduanya berada pada *protocol stacks* yang berbeda yang berisi implementasi terpisah dari *Transport layer protokol* seperti TCP dan UDP. [**14**]



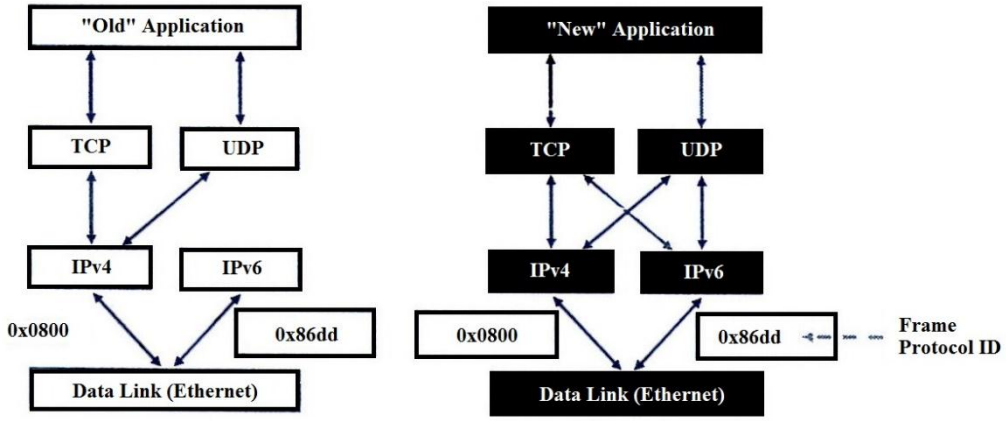
Gambar Error! No text of specified style in document.**.8 Arsitektur *Dual Stack* [**14**]**

2.1.21. Dual Stack Applications

Mekanisme *Dual-Stack* tidak hanya diimplemntasikan pada perangkat-perangkat jaringan. Di sisi aplikasi, penerapan mekanisme *Dual Stack* menjadi yang lebih diminati karena memiliki 2 protokol IPv4 dan IPv6 dalam satu *node* yang sama.

Pada pengembangan aplikasi-aplikasi terbaru, perubahan serta modifikasi dibutuhkan agar aplikasi dapat berjalan diatas protokol jaringan IPv6 sementara untuk aplikasi-aplikasi yang lama masih tetap dapat berjalan dan berkomunikasi diatas protokol jaringan IPv4.

Application Programming Interface (API) didefinisikan untuk dapat mendukung pengalamatan dan *Domain Name Services request* IPv4 dan IPv6. API terbaru ini mengganti yang disebut ***gethostnyname*** dan ***gethostbyaddr*** pada aplikasi. Perubahan API ini membuat aplikasi dapat berjalan diatas protokol IPv4 maupun IPv6 dan selama proses peralihan tersebut aplikasi tetap dapat berjalan diatas protokol jaringan IPv4 [**15**].



Gambar Error! No text of specified style in document.**.9 *Dual Stack Applications* [**15**]**

2.1.22. Proses Pemilihan Alamat *Host Dual Stack*

Pengaturan pemilihan alamat dan DNS *resolution* pada *host Dual-Stacking* dilakukan dengan proses sebagai berikut:

1. Step 1: DNS *request* terkirim

   Dual Stack apllication mengirim DNS *request* dengan menggunakan kedua jenis protokol ke *destination hostname*, contoh *hostname* tujuan: http://www.example.com.
2. Step 2: DNS *server* mengirim pesan balasan

   DNS *server* mengirim pesan balasan kesemua alamat *host* yang tersedia, dengan menggunakan IPv6 dan IPv4.
3. Step 3: Memilih alamat yang sesuai.

   Aplikasi memilih alamat yang digunakan secara *default*, contoh menggunakan IPv6.
4. Step 4: Aplikasi terhubung dengan *host* tujuan

   Aplikasi sebagai alamat *source* terhubung dengan *destination host* dengan menggunakan alamat IPv6. [**15**]

2.1.23. Langkah-Langkah Mengimplementasikan *Dual Stack*

Berikut adalah langkah-langkah yang dibutuhkan saat akan mengimplementasikan mekanisme *Dual Stack*:

1. Me-*review* jaringan, aplikasi dan keamanan *network environment*.
2. Meng-*upgrade host*, *router* dan *infrastructure services* untuk mendukung IPv6 jika dibutuhkan.
3. Memungkinkan *support* IPv6.
4. Meng-*upgrade* semua servis yang memungkinkan dapat menyediakan layanan IPv6.
5. Memastikan bahwa implemntasi *Dual Stack* telah benar dan berhasil dioperasikan [**15**].

2.1.24. Kelebihan dan Kekurangan *Dual Stack*

Kelebihan penerapan mekanisme *Dual Stack*:

1. Relatif mudah diimplementasikan.
2. Dapat dilakukan dengan biaya yang kecil dalam jangka pendek.
3. Dapat mengembangkan infrastruktur IPv4 yang sudah ada.
4. Memberikan kesinambungan terhadap keberadaan IPv4 dan IPv6 secara bersama-sama.

Kekurangan penerapan mekanisme *Dual Stack*:

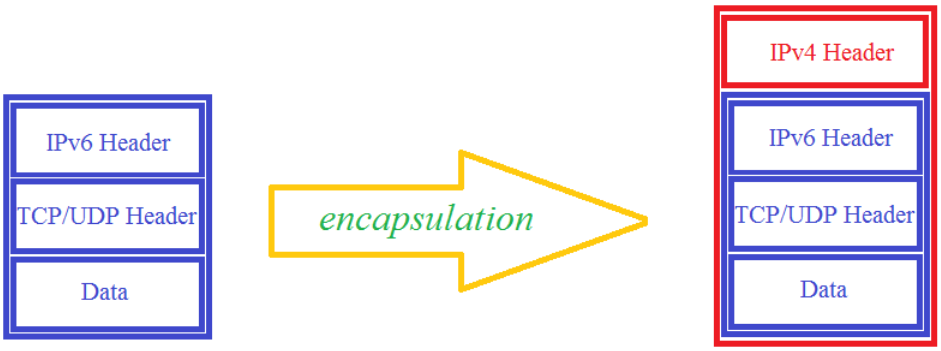
1. Dapat menyebabkan pengeluaran biaya yang mahal dalam jangka panjang.
2. Dapat meningkatkan kompleksitas jaringan. [**15**]

2.10. IPv6 over IPv4 Tunnel

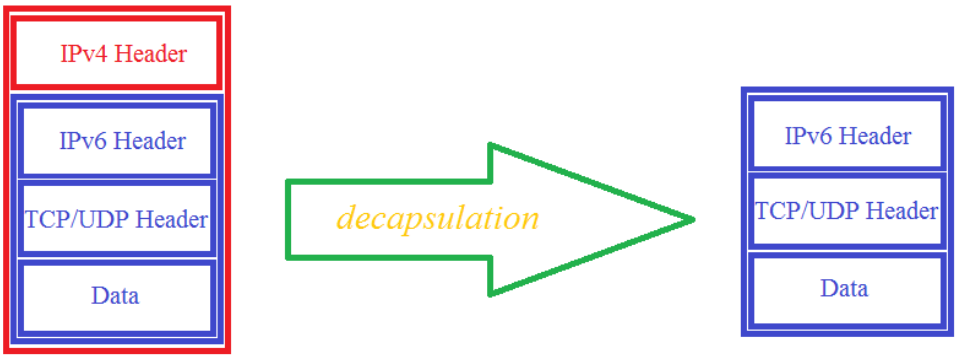
Mekanisme *tunneling* merupakan suatu konsep migrasi dimana *packet* dengan *header* IPv6 akan ‘dibungkus’ dengan *header* IPv4 untuk dilewatkan pada jaringan IPv4. Ada banyak jenis mekanisme *tunneling* yang dapat digunakan untuk melakukan migrasi IP, yaitu:

1. *IPv6 configure tunnel*
2. *Automatic tunnel with IPv4-compatible IPv6 address*

3. *6over4 tunnel*
4. *6to4 tunnel*
5. ISATAP
6. DSTM
7. *Tunnel Broker*
8. *Teredo*



Gambar Error! No text of specified style in document.**..10 Enkapsulasi *Packet* IPv6 menjadi *Packet* IPv4 [**13**]**



Gambar Error! No text of specified style in document.**..11 Dekapsulasi *Packet* IPv4 menjadi *Packet* IPv6 [**13**]**

Mekanisme *tunneling* secara umum telihat pada gambar 2.12 dan gambar 2.11 dan prosedur tunneling secara umum dilakukan sebagai berikut [**13**]:

1. *Tunnel entrance node* (enkapsulator) membangun *header* IPv4, kemudian membungkus *packet* IPv6 dengan *header* IPv4 kemudian melewakan *packet* tersebut ke dalam jaringan IPv4.
2. *Tunnel exit node* (dekapsulator) menerima *packet* kemudian membuka *header packet* IPv4 agar *packet* kembali menjadi *packet* IPv6

Dengan melakukan *tunneling*, proses *routing* pada jaringan akan tetap menggunakan proses *routing* pada packet IPv4 sehingga kelebihan IPv6 yang memiliki kecepatan *routing* yang lebih baik dibandingkan IPv4 tidak dapat dirasakan. Namun hal tersebut tetap membuat *packet* dengan *header* IPv6 sampai ke tujuan dengan alamat IPv6 walaupun melewati jaringan IPv4.

2.11. Keamanan Jaringan

Keamanan atau security adalah mekanisme dan teknik untuk melindungi sesuatu yang dapat berupa data atau informasi di dalam sistem. Pada dasarnya secutity adalah sistem yang digunakan untuk melindungi sistem dalam suatu jaringan keamanan agar tetap terjaga.

Pada dasarnya keamanan adalah proteksi perlindungan atas sumber-sumber fisik dan konseptual dari bahaya alam dan manusia. Kemanan terhadap sumber konseptual meliputi data dan informasi. Namun disamping itu banyak sekali upaya manusia untuk menjaga keamanan atas sumber-sumber fisik maupun konseptual dengan berbagai cara, sehingga untuk lebih memudahkan pengedalian keamanan maka dilakukan kontrol manajemen dimana seorang dapat mengatur keamanan dari berbagai macam serangan.

Keamanan atau Security haruslah memiliki beberapa bagian penting di dalamnya, yaitu :

1. Availability yaitu menjaga akses untuk masuk ke dalam informasi
2. Confidentianlity yaitu menjaga informasi secara rahasia dan hanya dapat dibuka oleh yang memiliki hak resmi untuk mengaksesnya.
3. Anonymity yaitu menyembunyikan identitas dari entitas yang terlibat dalam prosesnya

4. Privacy yaitu memiliki hak dan kewajiban yang mengatur akusisim rahasia pribadi, dan informasi rahasi yang lain.
5. Identification and Authentication yaitu cara mengetahui identitas user dalam jaringan komputer.

Tujuan keamanan dapat dicapai dengan suatu metode keamanan jaringan yang dapat melindungi sistem baik dari dalam maupun dari luar jaringan, namun bukan hanya melindungi tetapi harus dapat bertindak apabila terjadi serangan yang ada di dalam jaringan. Salah satu metode tersebut yaitu *Intrusion Dtection System (IDS)* dan *Intrusion Prevention System (IPS)*. Namun, selain metode tersebut dibutuhkan juga suatu pemahaman tentang menentukan kebijakan (*security Policy*) dalam keamanan jaringan. Jika ingin menentukan apa saja yang harus dilindungi maka harus mempunyai perencanaan keamanan yang matang dan baik berdasarkan pada prosedur dan kebijakan keamanan jaringan, karena apabila tidak direncanakan maka tidak akan sesuai dengan yang diharapkan dalam perlindungan jaringan.

2.1.25. Analisis keamanan jaringan

Insiden keamanan jaringan komputer adalah suatu aktifitas yang berkaitan dengan jaringan komputer dimana aktivitas tersebut memberikan implikasi terhadap keamanan.

Secara garis besar insiden yang sering terjadi dan memiliki potensi besar untuk melakukan serangan pada jaringan dapat diklasifikasikan sebagai berikut:

1. Probe/scan: usaha-usaha yang tidak lazim untuk memperoleh akses ke dalam sisten, atau untuk menemukan informasi tentang sistem tersebut. Kegiatan probe dalam jumlah besar dengan menggunakan tool secara otomatis biasa disebut *scan*. Bermacam- macam tool yang dipergunakan untuk keperluan ini seperti: network mapper,

port mapper, network scanner, port scanner, atau vulnerability scanner. Informasi yang diperoleh misalkan:

c. Topologi dari jaringan target
d. Tipe trafik yang melewati *firewall*
e. *Host* yang aktif
f. Sistem operasi pada host
g. Perangkat lunak yang berjalan pada server dan versinya.

2. *Account Compromise*: penggunaan account sebuah komputer secara ilegal oleh seseorang yang bukan pemilik account, dimana account tersebut tidak mempunyai privilege sebagai administrator sistem.
3. *Root Compromise*: mirip dengan account compromise tetapi tidak mempunyai hak akses sebagai adminstrator sistem. Tindakan ini biasanya dilakukan untuk menguasai sitem dengan tujuan tidak baik untuk menghancurkan sistem maupun memodifikasi sistem tanpa sepengetahuan admin.
4. *Paket Sniffer*: perangkat lunak/ keras yang digunakan untuk memperoleh informasi yang melewati jaringan komputer, biasanya dengan menggunakan NIC. Penggunaan perangkat ini digunakan oleh para intruder dengan memotivasi yang sama dengan tindakan probe atau scan.
5. Denial Of Service (DOS) Serangan DoS (bahasa Inggris: denial-of-service attacks') adalah jenis serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (*resource*) yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang tersebut.

Dalam sebuah serangan *Denial of Service*, si penyerang akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, yakni sebagai berikut:

1. Membanjiri lalu lintas jaringan dengan banyak data sehingga lalu lintas jaringan yang datang dari pengguna yang terdaftar menjadi tidak dapat masuk ke dalam sistem jaringan. Teknik ini disebut sebagai *traffic flooding*.
2. Membanjiri jaringan dengan banyak request terhadap sebuah layanan jaringan yang disedakan oleh sebuah host sehingga request yang datang dari pengguna terdaftar tidak dapat dilayani oleh layanan tersebut. Teknik ini disebut sebagai *request flooding*.
3. Mengganggu komunikasi antara sebuah host dan kliennya yang terdaftar dengan menggunakan banyak cara, termasuk dengan mengubah informasi konfigurasi sistem atau bahkan perusakan fisik terhadap komponen dan server.

6. Port Scanning : Port Scan adalah awal dari masalah besar yang akan datang melalui jaringan, untuk itu perlu mendeteksi adanya Port Scan. Port Scan merupakan awal serangan dan hasil Port Scan membawa beberapa informasi kritis yang sangat penting untuk pertahanan mesin & sumber daya yang kita miliki. Keberhasilan untuk menggagalkan Port Scan akan menyebabkan tidak berhasilnya memperoleh informasi strategis yang dibutuhkan sebelum serangan yang sebetulnya dilakukan.
7. IP-Spoofing: Menurut Felten et al *spoofing* dapat didefinisikan sebagai "Teknik yang digunakan untuk memperoleh akses yang tidak sah ke suatu komputer atau informasi, dimana penyerang berhubungan dengan pengguna dengan berpura-pura memalsukan bahwa mereka adalah *host* yang dapat dipercaya" teknik spoofing atau kadang dikenal dengan nama Source Address Spoofing ini melakukan Pemalsuan IP Address dari si attacker agar korban menganggap bahwa IP Address itu

bukan berasal dari luar jaringan. Analoginya adalah bayangkan jika anda ingin mengirimkan surat ancaman kepada seseorang,dan anda tidak ingin orang yang anda ancam mengetahui bahwa surat ancaman itu berasal dari anda. Jadi yang anda lakukan adalah mengirimkan surat tersebut dengan menggunakan nama dan alamat orang lain sehingga anda akan aman dan tidak terlacak.

8. ICMP flood: Melakukan eksploitasi sistem agar dapat membuat suatu target *client* menjadi *Crash* yang dilakukan oleh penyerang. Sehingga menjadi *crash* karena diakibatkan oleh pengiriman sejumlah paket yang besar kearah target *client. Exploting* sistem ini dilakukan dengan mengirimkan suatu perintah ping dengan tujuan *broadcast* atau *multicast* di mana si pengirim dibuat seolah-olah adalah target *client.* Semua pesan balasan dikembalikan ke target *client.* Hal inilah yang membuat target *client* menjadi *crash* dan menurunkan kinerja jaringan. Bahkan hal ini dapat mengakibatkan *denial of service.*
9. UDP flood: Pada dasarnya mengkaitkan dua sistem tanpa disadarinya. Dengan cara spoofing. *User Datagram Protocol* (UDP) flood attack akan menempel pada service UDP chargen di salah satu mesin, yang untuk keperluan "percobaan" akan mengirimkan sekelompok karakter yang diterima melalui *service chargen*. Karena paket UDP tersebut di spoofing antara ke dua mesin tersebut, maka yang terjadi adalah banjir tanpa henti kiriman karakter yang tidak berguna antara ke dua mesin tersebut, untuk menanggulangi UDP flood, kita dapat men-disable semua servis UDP di semua mesin di jaringan, atau yang lebih mudah memfilter pada firewall semua servis UDP yang masuk.

Meskipun *administrator* pada jaringan berusaha meminimalkan vulnerabilitas yang memungkinkan serangan dengan memberlakukan bebeapa mekanisme kepada sistem , ada banyak situasi yang tidak memungkinkan hal ini dilakukan, diantaranya:

a. Pada beberapa sistem, tidak bisa dilakukan patch atau update sistem operasi. Hal ini sedapatnya tidak sering dilakukan pada jaringan dikarenakan update sistem operasi akan menyebabkan semua klien tidak dapat menggunakan sumber daya jaringan.
b. Administrator tidak memiliki waktu atau sumber daya yang mencukupi untuk melacak dan menginstall semua patch yang diperlukan. Hal ini terjadi karena jaringan pada umumnya memiliki sejumlah host dengan hardware dan software yang berbeda.
c. User menggunakan layanan protokol uang merupakan sumber vulnerabilitas. Hal ini dikarenakan adminm membuka bebeapa port untuk memudahkan user mengaksus dan mentransfer informasi dari internet. Terbukanya beberapa port ini memberikan peluang yang besar bagi user untuk menggunakan layanan protokol yang merupakan sumber vulnerabilitas.
d. Baik user maupun administrator : bisa membuat kesalahan dalam melakukan konfigurasi dan penggunaan sistem
e. Terjadi disparitas policy yang memungkinkan user melakukan tindakan yang melibihi kewenanagnn. Hal ini dikarenakan kebijakan panggunaan account jaringan pada beberapa institusi pada umumnya tidak dilaksanakan dengan tegas sehingga banyak user yang masih sering melakukan pelanggaran.

Dengan adanya kondisi-kondisi di atas maka dapat disimpulkan bahwa jaringan membutuhkan sistem keamanan jaringan otomatis yang dapat memantau pelanggaran penggunaan account oleh user di dalam sistem maupun memantau penyusupan yang dilakukan oleh pihak di luar sistem.

2.12. Intrusion Detection System (IDS)

IDS (Intrusion Detection System) merupakan sistem untuk mendeteksi adanya “intrusion” yang dilakukan oleh “intruder” atau “pengganggu atau penyusup” di jaringan. IDS (Intrusion Detection System) sangat mirip seperti alarm, yaitu IDS (Instrusion Detection System) akan memperingati bila terjadinya atau adanya penyusupan pada jaringan. IDS (Intrusion Detection System) dapat didefinisikan sebagai kegiatan yang bersifat anomaly, incorrect, inappropriate yang terjadi di jaringan atau host. IDS (Intrusion Detection System) adalah sistem keamanan yang bekerja bersama Firewall untuk mengatasi Intrusion.[2].

IDS (Intrusion Detection System) juga memiliki cara kerja dalam menganalisa apakah paket data yang dianggap sebagai intrusion oleh intruser. Intrusion itu sendiri didefinisikan sebagai kegiatan yang bersifat anomaly, incorrect, inappropite yang terjadi di jaringan atau di host tersebut. Intrusion tersebut kemudian akan diubah menjadi “rules” ke dalam IDS (Intrusion Detection System). Sebagai contoh, intrusion atau gangguan seperti port scanning yang dilakukan oleh intruder. Oleh karena itu IDS (Intrusion Detection System) ditujukan untuk meminimalkan kerugian yang dapat ditimbulkan dari intrusion.

2.1.26. Mengenali Adanya Intruder

Cara kerja IDS (Intrusion Detection System) dibagi menjadi dua, yaitu :

1. Knowledge Based (Misuse Detection)

   Knowledge Based pada IDS (Intrusion Detection System) adalah cara kerja IDS(Intrusion Detection System) dengan mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkannya dengan database rule pada IDS (Intrusion Detection System) tersebut. Database rule tersebut dapat berisi signature – signature paket serangan. Jika pattern atau pola paket data tersebut terdapat kesamaan dengan rule pada database rule pada IDS (Intrusion Detection System), maka paket data tersebut dianggap sebagai seranganm dan demikian juga sebaliknya, jika paket data tersebut tidak memiliki

kesamaan dengan rule pada database rule pada IDS(Intrusion Detection System), maka paket data tersebut tidak akan dianggap serangan.

2. Behavior Based (Anomaly Based)

Behavior Base adalah cara kerja IDS (Intrusion Detection System) dengan mendeteksi adanya penyusupan dengan mengamati adanya kejanggalan – kejanggalan pada sistem, aatu adanya keanehan dan kejanggalan dari kondisi pada saat sistem normal, sebagai contoh : adanya penggunaan memory yang melonjak secara terus menerus atau terdapatnya koneksi secara paralel dari satu IP dalam jumlah banyak dan dalam waktu yang bersamaan. Kondisi tersebut dianggap kejanggalan yang selanjutnya oleh IDS (Intrusion Detection System) Anomaly Based ini dianggap sebagai serangan.

Biasanya rule dan signature hanya berisi pola serangan yang selalu di *Update* secara rutin, hal ini dikarenakan adanya serangan baru setiap hari. Namun untuk proses deteksi *anomaly* tidak menggunakan rule dan signature, hanya mengamati kondisi normal dari sistem jaringan, jika suatu waktu kondisi dari jaringan tidak normal, hal seperti ini dianggap sebagai suatu serangan. Keunggulan dari sistem deteksi ini dapat mengenali serangan baru yang polanya tidak ada pada rule dan signature dasil dari pembelajaran sistem deteksi itu sendiri.

Kekurangan dari sistem seteksi *anomaly* ini adalah banyaknya *alert false positive* yang dikirim ke *user*. Contoh jika suatu waktu server menerima banyak *request* dari true *client internal* dan kinerja sistem meningkat dengan cepat (memory, prosesor), maka sistem deteksi akan melaporkan sebagai serangan. IDS yang dibangun menggunakan dua sistem deteksi ini untuk mengatasi masalah serangan yang terjadi, jika suatu pola serangan tidak ada pada rule dan signature, maka sistem deteksi *anomaly* berfungsi untuk mencari pola serangan baru

2.1.27. Karakteristik dan sifat *Intrusion Detection System* (IDS)

1. Suitability

   Aplikasi IDS yang cenderung memfokuskan berdasarkan skema manajemen dan arsitektur jaringan yang dihadapkannya.

2. Flexibility

   Aplikasi IDS yang mampu beradaptasi dengan spesifikasi jaringan yang akan dideteksi oleh aplikasi tersebut.

3. Protection

   Aplikasi IDS yang secara ketat memproteksi gangguan yang sifatnya utama dan berbahaya.

4. Interoperability

   Aplikasi IDS yang secara umum mampu beroperasi secara baik dengan perangkat-perangkat keamanan jaringan serta manajemen jaringan lainnya.

5. Comprehensiveness

   Kelengkapan yang dimiliki oleh aplikasi IDS ini mampu melakukan sistem pendeteksian secara menyeluruh seperti pemblokiran semua yang berbentuk *Java Applet*, memonitor isi dari suatu email serta dapat memblokir *address url* secara spesifik.

6. Event Management

   Konsep IDS yang mampu melakukan proses manajemen suatu jaringan serta proses pelaporan pada saat dilakukan setiap pelacakan, bahkan aplikasi ini mampu melakukan *updating* pada sistem basis data pola suatu gangguan.

7. Active Response

   Pendeteksi gangguan ini mampu secara cepat untuk mengkonfigurasi saat munculnya suatu gangguan, biasanya aplikasi ini berintegrasi dengan aplikasi lainnya seperti aplikasi *Firewall* serta aplikasi IDS ini dapat mengkonfigurasi ulang spesifikasi *router* pada jaringannya.

8. Support

   Lebih bersifat mendukung pada suatu jenis produk apabila diintegrasikan dengan aplikasi lain.

2.1.28. Cara kerja IDS

Ada beberapa cara bagaimana IDS bekerja. Cara yang paling populer adalah dengan menggunakan pendeteksian berbasis signature (seperti halnya yang dilakukan oleh beberapa antivirus), yang melibatkan pencocokan lalu lintas jaringan dengan basis data yang berisi cara-cara serangan dan penyusupan yang sering dilakukan oleh penyerang. Sama seperti halnya antivirus, jenis ini membutuhkan pembaruan terhadap basis data signature IDS yang bersangkutan.

Metode selanjutnya adalah dengan mendeteksi adanya anomali, yang disebut sebagai Anomaly-based IDS. Jenis ini melibatkan pola lalu lintas yang mungkin merupakan sebuah serangan yang sedang dilakukan oleh penyerang. Umumnya, dilakukan dengan menggunakan teknik statistik untuk membandingkan lalu lintas yang sedang dipantau dengan lalu lintas normal yang biasa terjadi. Metode ini menawarkan kelebihan dibandingkan signature-based IDS, yakni ia dapat mendeteksi bentuk serangan yang baru dan belum terdapat di dalam basis data signature IDS. Kelemahannya, adalah jenis ini sering mengeluarkan pesan false positive. Sehingga tugas administrator menjadi lebih rumit, dengan harus memilah-milah mana yang merupakan serangan yang sebenarnya dari banyaknya laporan false positive yang muncul.

Teknik lainnya yang digunakan adalah dengan memantau berkas-berkas sistem operasi, yakni dengan cara melihat apakah ada percobaan untuk mengubah beberapa berkas sistem operasi, utamanya berkas log. Teknik ini seringnya diimplementasikan di dalam HIDS, selain tentunya melakukan pemindaian terhadap log sistem untuk memantau apakah terjadi kejadian yang tidak biasa. [**16**]

2.1.29. Kebutuhan Pengguna IDS

Pemahaman tentang tingginya resiko pada jaringan adalah kunci suksesnya penerapan IDS. Untuk menangani semua gangguan yang mengancam keamanan jaringan, maka jaringan memerlukan sistem

keamanan yang harus dipasang pada jaringan dan dapt dipantau setiap saat. Sistem keamanan ini berfungsi untuk memberikan jalur yang aman antara entitas yang saling bertukar informasi dan juga untuk menjamin adanya perlindungan data-data penting yang ada pada sistem. [**16**]

Dalam dunia keamanan jaringan telah banyak dipakai teknik atau sistem yang telah umum berfungsi untuk mengamankan jaringan yaitu firewall dan kriptografi. Firewall didesain untuk melewatkan, menghentikan atau menolak trafik, akan tetapi tidak akan pernah memberikan alert atau peringatan terhadap trafik yang mencurigakan. Kriptografi berfungsi untuk mengamankan data dari pihak yang mempunyai hak akses, akan tetapi juga tidak pernah memberikan peringatan terhadap akses yang mencurigakan. Sementara itu IDS didesain untuk memberitahu kita kapan aktivitas yang mencurigakan terjadi. Baik teknologi firewall, krptigrafi dan IDS ketiga-tiganya perlu diimplementasikan pada jaringan untuk dapt saling melengkapi dalam menjalankan fungsi keamanan jaringan. Oleh karena itu, sistem keamanan yang diharapkan selain dapat mencegah penyusupan juga haruslah dapat melakukan deteksi penyusupan dari luar sistem dan sekaligus dapat melakukan deteksi penyalahgunaan account dari dalam sistem.

IDS akan membantu administrator jaringan dengan cara meningkatkan kemampuan penemuan resiko yang membahayakan sistem secara cepat dan *real time*. Dengan demikian, fungsi IDS adalah memberi peringatan kepada administrator atas serangan yang terjadi sehinggs administrator dapat memperoleh keuntungan sebagai berikut:

1. Mencegah resiko timbulnya masalah.
2. Mendeteksi serangan dan pelanggaran keamanan lain yang tidak dicegah oleh perangkat keamanan lainnya. Biasanya penyusupan berlangsung dalam tahapan yang bisa diprediksi. Tahapan pertama adalah probing, atau eksploitasi pencarian titik masuk. Pada sistem tanpa IDS, penyusup memiliki kebebasan melakukannya dengan resiko ketahuan lebih kecil. IDS yang mendapati probing bila melakukan blok akses dan

memberitahukan tenaga keamanan yang selanjutnya mengambil tindakan lebih lanjut.

3. Mendeteksi usaha penyusupan yang berkaitan dengan serangan misal probing dan aktivitas DoS.
4. Untuk mendokumentasikan ancaman yang perrnah terjadi dan mungkin terjadi lagi pada jaringan. IDS akan mampu menggolongkan ancaman baik dari dalam maupun dari luar jaringan sehingga membantu membuat keputusan untuk alokasi sumber daya keamanan jaringan.
5. Untuk bertindak sebagai pengendali kualitas pada administrasi dan perancangan keamanan pada jaringan. Saat IDS dijalankan dalam waktu tertentu, pola dari pemakaian sistem dan masalah yang ditemui bisa nampak. Sehingga akan membantu pengelolaan keamanan dan memperbaiki kekurangan sebelum menyebabkan insiden.
6. Untuk memberikan informasi yang berguna mengenai penyusupan yang terjadi, peningkatan diagnosa, recovery, dan perbaikan dari faktor penyebab. Meskipun pada IDS yang bertipe passive response tidak melakukan blok serangan. Tetapi IDS ini masih bisa mengumpulkan informasi yang relevan mengenai serangan, sehingga membantu penanganan insiden dan recovery. Dengan demikian IDS akan membantu konfigurasi atau kebijakan organisasi.

Semua teknologi keamanan memiliki resiko. Firewall memiliki resiko untuk dapat dipenetrasi, sementara enkripsi memiliki resiko untuk dibobol atau didescypt, maka IDS pun memiliki resiko kesa;ahan dalam mendeteksi jaringan. Resiko kesalahan dalam mendeteksi jaringan ini dikenal dengan false positive. Apabila suatu IDS menghasilkan banyak sekali false positive maka penggunaan IDS pada suatu jaringan justru akan menyibukkan pekerjaan admin dalam merespon banyaknya alarm yang salah melakukan pendeteksian yang dihasilkan setiap harinya. [**16**]

Ada beberapa macam jenis dan produk IDS yang ada di pasaran. Beberapa IDS tersebut memiliki kemampuan yang berbeda-beda dalam medeteksi beragamnya jenis serangan yang ada. Ada IDS tertentu yang

mampu mendeteksi suatu jenis serangan dengan tepat akan tetapi lemah dalam mendeteksi serangan yang lain sehingga menghasilkan banyak false positive untuk serangan itu. Sementara beberapa IDS yang lain bisa jadi memilii kemampuan sebaliknya. Karena itu diperlukan suatu IDS yang mampu memberikan fungsionalitas yang sebaik—baiknya dalam mendeteksi penyusupan tanpa banyak menghasilkan false positive maupun false negative. Berikut adalah beberapa kriteria yang diinginkan untuk suatu IDS yang ideal:

1. Meminimalkan overhead sistem untuk tidak mengganggu operasi normal sehingga tidak mengganggu aktivitas jaringan
2. Mudah dikonfigurasi untuk disesuaikan dengan kebijakan keamanan sistem
3. Mudah diinstalasi
4. Mudah beradaptasi dengan perubahan sistem dan perilaku user, misalnya pemakain aplikasi atau resource baru.
5. Mampu memonitor sejumlah host dengan tetap memberikan hasil yang cepat dan tepat.
6. Dampak negatif yang minimal
7. Memungkinkan konfigurasi dinamis. Khususnya bila pemantauan dilakukan pada sejumlah besar host.
8. Berjalan secara kontinu dengan supervisi minimal dari manusia
9. Mampu mendeteksi serangan :
    h. Tidak salah menandai virus yang legitimate (false positive)
    i. Tidak gagal mendeteksi serangan sesungguhnya (false negative)
    j. Segera melakukan pelaporan penyusupan yang terjadi.
    k. Cukup general untuk berbagai tipe serangan.
        1) Mampu fault tolerant dalam arti:
            a) Bisa melakukan rcover dari sistem yang crash baik secara insidental atau karena aktivitas tertentu
            b) Setelah itu bisa melanjutkan state sebelumnya tanpa mempengaruhi operasinya

2) Mampu menolak usaha pengubahan:
    a) Akan memberikan kesulitan yang tinggi bila penyerang mencoba memodifikasi dan menyerang IDS itu sendiri.
    b) Mampu memonitor dirinya sendiri dan mendetaksi bila dirinya telah dirubah oleh penyerang.

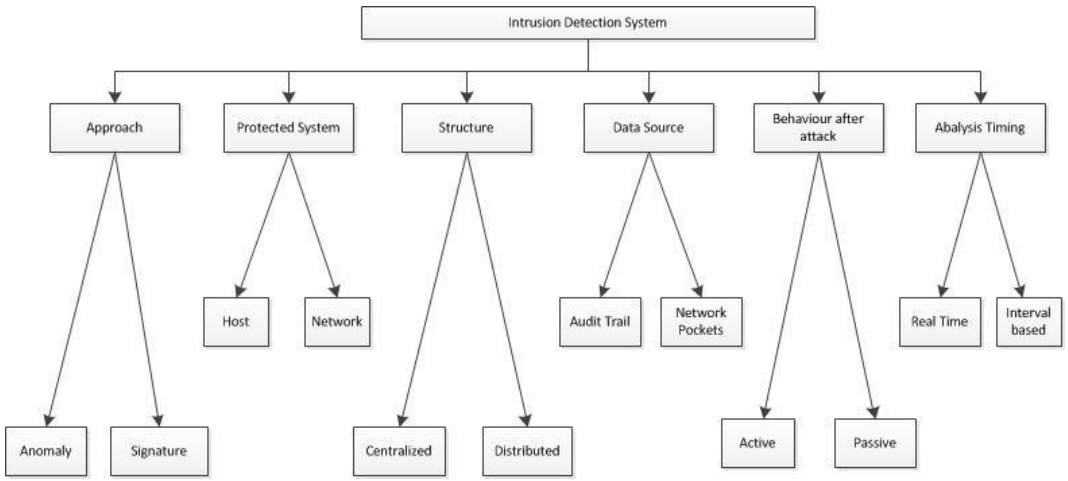
2.1.30. Metode Pendeteksian Serangan

Metode pendeteksian serangan-serangan pada jaringan komputer dapat menggunakan salah satu sistem pendeteksian yaitu *intrusion detection system* (IDS). *Intrusion detection system* adalah sebagai *tools* yang memberikan bantuan untuk melakukan identifikasi dan memberikan laporan terhadap aktifitas jaringan komputer. IDS mampu mencatat hasil pendeteksiannya ke dalam sebuah file atau database, sehingga administrator dapat melihatnya dikemudian hari. [**17**]

2.1.1.6 Analisis Intrusion Detection System

Intrusion Detection System (IDS) dapat didefenisikan sebagai *tool*, metode atau sumber daya yang memberikan bantuan utnuk melakukan identifikasi dan memberikan laporan terhadap aktivitas jaringan komputer. IDS merupakan bagian kecil dari sistem keamanan jaringan komputer.

Intrusion detection System memiliki beberapa kategori dan klarifikasi seperti yang terdapat pada Gambar 2.12



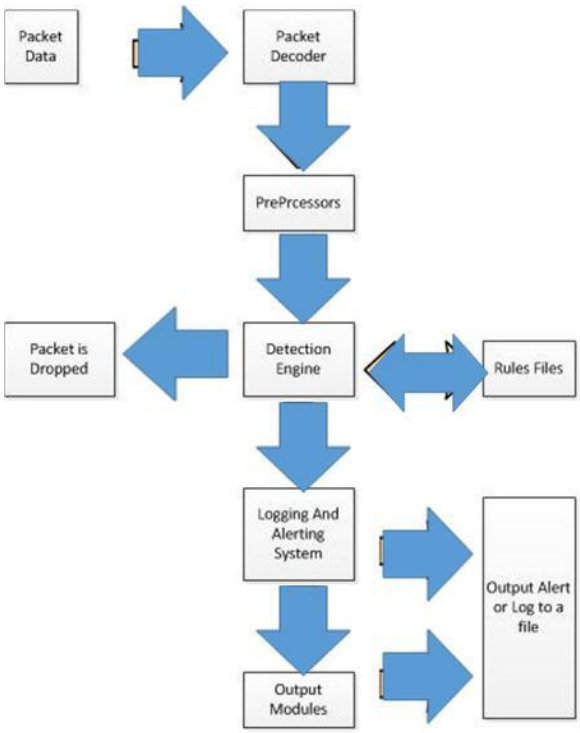
Gambar Error! No text of specified style in document.**.12 Klasifikasi Kategori *Intrusion detection System***

Dari kategori yang digambarkan pada gambar 2.12 dapat dijelaskan bahwa IDS memiliki beberapa klarifikasi cara atau metode dalam penerapannya untuk mendeteksi serangan, berikut ini akan dijelaskan beberapa keuntungan dan kerugian jenis-jenis klasifikasinya.

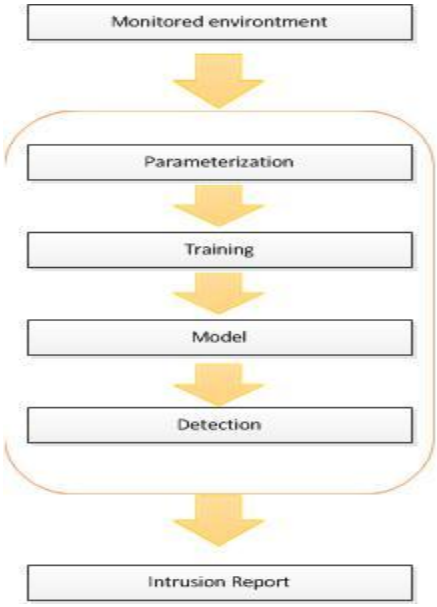
1.1.1.1.2 Anomaly Vs. Signature

Pada dasarnya terdapat dua pendekatan atau metode pada *intrusion detection sistem* yaitu *anomaly based* dan *signature based*. Masing-masing metode memiliki teknik yang berbeda dalam pendeteksian serangan pada suatu jaringan. Metode *signature based* bekerja seperti *antivirus,* yaitu dengan mendeteksi *patern-patern intrusion* yang ada pada *rules* atau *database* yang terdapat pada IDS tersebut. *Software* IDS akan mengumpulkan paket data yang selanjutnya akan dibandngkan dengna *rules* yang sudah didefenisikan dan selanjutnya akan memutuskan apakah paket data tersebut serangan atau bukan (lihat Gambar 3.6). kelemahan metode ini yaitu tidak dapat mendeteksi serangan yang baru apabila *rule* atau *database* tidak di *update.* Sedangkan metode *anpmaly based* adalah metode pendeteksian serangan pada IDS dengan cara memonitor pola-pola pada jaringan, jika terdapat pola paket data yang mencurigakan maka akan dianggap sebagai sebuah interusi, kelemahan metode ini yaitu banyak

menghasilkan peringatan interusi yang sebenarnya paket data itu bukan merupakan serangan (*file alrm*). Proses yang dilakukan oleh metode *anomaly* dapat dilihat pada gambar 2.13.



Gambar Error! No text of specified style in document.**.13 Proses Signatured Based Detection**



Gambar Error! No text of specified style in document.**.14 Proses *Anomaly Detection***

Dari kedua metode tersebut terdapat beberapa keunggulan dan kelemahan dari masing-masing metode, berikut akan dijelaskan perbandingan keunggulaan diantara kedua metode tersebut pada tabel 2.3.

Table Error! No text of specified style in document.**.3 Perbandingan *Anomaly* dan *Signature***

NO	Anomaly Based	Signature Based
1	Memungkinkan untuk mendeteksi jenis serangan yang baru	Mendeteksi serangan yang sudah dibuatkan Rule-nya
2	Banyak menghasilkan kesalahan pada saat mendeteksi serangan (f*alse alarm*).	Peringatan terjadinya interupsi akan muncul jika sesuai dengan rule yang ditetapkan.
3	Paket data tidak akan dipantau selama proses pembangunan pelatihan profile serangan	Tidak ada proses pelatihan dan pembangunan profile serangan

Dari hasil penelitian yang dilakukan oleh Noviana Sagita dalam penelitiannya yang berjudul "*Perbandingan Performansi antara Signature Based dan Anomaly Based dalam pendeteksian Intrusi*" pada tahun 2011, telah

dilakukan beberapa pengujian terhadap kedua metode ini dengan melakukan beberapa penyerangan dan hasilnya dapat dilihat pada tabel 2.4

Table Error! No text of specified style in document.**.4 Hasil Pengujian Serangan terhadap metode IDS**

Jenis Pengujian	Anomali (Ourmon)	Signature (snort)
Port Scanning	Tidak Terdeteksi	Terdeteksi
SYN Flooding attack	Terdeteksi	Terdeteksi
Downloading File	Terdeteksi	Tidak Terdeteksi

Kesimpulan dari pengujian itu yaitu metode *signature based* dapat mendeteksi dua pengujian dan tidak dapat mendeteksi pengujian *downloading file*, sedangkan metode *anomaly* dapat mendeteksi dua pengujian dan tidak dapat mendeteksi pengujian *port scanning*. Pengujian *downloading file* bukan merupakan jenis aktifitas serangan. Contoh aplikasi yang menggunakan metode anomaly yaitu AAFID, MIDAS, NADIR, UNICORN, dan lain-lain. Contoh aplikasi IDS yang menggunakan metode signature yaitu ARMD, ASIM, Bro, CSM, CyberCop, GRIDS, Snort, Stalker, Suricata, Tripware dan lain-lain.

1.1.1.1.3 Host Vs. Network

***a. Network Intrusion Detection System* (NIDS)**

Memantau Anomali di Jaringan dan mampu mendeteksi seluruh host yang berada satu jaringan dengan host implementasi IDS (Intrusion Detection System) tersebut. NIDS (Network Instrusion Detection System) pada umumnya bekerja dilayer 2 pada OSI layer, IDS (Intrusion Detection System) menggunakan "raw traffic" dari proses sniffing kemudian mencocokknannya dengan signature yang telah ada dalam policy. Jika terdapat kecocokan antara signature dengan raw traffinc hasil sniffing paket, IDS (Intrusion Detection System) memberikan allert atau peringgatan sebagai tanda adanya proses intrusi ke dalam sistem. NIDS (Network Instrusion Detection System) yang cukup banyak dipakai adalah snort karena signature yang customizable, sehingga setiap vulnerability baru ditemukan

dapat dengan mudah ditambahkan agar jika terjadi usaha punyusupan atau intrusion dari intruder akan segera terdeteksi.. Idealnya semua traffic yang berasal dari luar dan dalam jaringan di lakukan di scan, namun cara ini dapat menyebabkan bottleneck yang mengganggu kecepatan akses di seluruh jaringan .

a. *Host Intrusion Detection System* (HIDS)

Mamantau Anomali di Host dan hanya mampu mendeteksi pada host tempat implementasi IDS (Intrusion Detection System) tersebut. HIDS (Host Instrusion Detection System) biasanya berupa tools yang mendeteksi anomali di sebuah host seperti perubahan file password dengan penambahan user ber UID 0, perubahan loadable kernel, perubahan ini script, dan gangguan bersifat anomali lainnya. IDS jenis ini berjalan pada host yang berdiri sendiri atau perlengkapan dalam sebuah jaringan. Sebuah HIDS melakukan pengawasan terhadap paket-paket yang berasal dari dalam maupun dari luar hanya pada satu alat saja dan kemudian memberi peringatan kepada user atau administrator sistem jaringan akan adanya kegiatan-kegiatan yang mencurigakan yang terdeteksi oleh HIDS. [**17**]

Table Error! No text of specified style in document.**.5 Perbedaan NIDS dan HIDS**

No	NIDS	HIDS
1	Ruang lingkup yang luas (mengamati semua aktifitas jaringan)	Ruang lingkup yang terbatas (mengamati hanya pada host tertentu)
2	Lebih mudah melakukan setup	Setup yang kompleks
3	Lebih baik untuk mendeteksi serangan yang berasal dari luar jaringan	Lebih baik mendeteksi serangan yang berasal dari dalam jaringan
4	Menganalisa paket header	Paket header diabaikan
5	Response *real time*	Response tergantung konfigurasi timing
6	Sistem Operasi independen	Sistem Operasi harus spesifik

Selain perbedaan, ada beberapa perbandingan dari HIDS dan NIDS, perbandingan ini diambil dari analisa yang dilakukan oleh Ricky M. Magalehaes

yang ditulisnya dalam artikel yang berjudul "*Host-Based IDS vs Network-Based IDS*" pada website www.windowsecurity.com. Pada Tabel 2.6 dipaparkan beberapa analisis antara HIDS dan NIDS.

Table Error! No text of specified style in document.**.6 Analisis Perbandingan antara HIDS dan NIDS**

NO	Function	HIDS	NIDS
1	Protection on LAN	****	****
2	Protection off LAN	****	-
3	Ease of Administration	****	****
4	Versality	****	**
5	Price	***	*
6	Ease od Implementation	****	**
7	Little Training required	****	**
8	Total cost of ownership	***	**
9	Bandwidth requirements on (LAN)	0	2
10	Network overhead	1	2
11	Bandwidth requirements (internet)	**	**
12	Spanning port switching requirements	-	****
13	Update frequency to clients	****	-
14	Cross platforms compability	**	****
15	Local machine registry scans	****	-
16	Logging	***	***
17	Alarm Functions	***	***
18	PAN scan	****	-
19	Packet rejection	-	****
20	Specialist knowledge	***	****
21	Central management	**	***

22	Disable risk factor	*	****
23	Upgrade potential	***	***
24	Multiple LAN detection nodes	****	**

1.1.1.1.4 Centralised vs distributed

Intrusion detection system memiliki dua tipe infrastruktur yaitu *centralised intrusion detection system* (CIDS) dan *distributed intrusion detecion system* (DIDS). CIDS yaitu tipe IDS yang seluruh kendalinya baik itu monitoring, deteksi dan pelaporannya dikendalikan secara terpusat. Sedangkan DIDS adalah tipe IDS yang monitoring dan pendeteksian dilakukan oleh beberapa node lokal dengan pelaporan pada satu atau beberapa pusat.

Model CIDS dan DIDS memiliki beberapa kelemahan dan keunggulan masing-masing terlihat pada

Table Error! No text of specified style in document.**.7 Perbandingan CIDS dan DIDS**

No	Centralised	Distributed
1	Mudah dalam melakukan konfigurasi.	Relatif sulit dalam konfigurasi
2	Biaya yang dikeluarkan cukup murah karena hanya menggunakan satu IDS	Biaya yang dikeluarkan relatif mahal karena menggunakan beberapa IDS.
3	Tidak dapat memonitor paket data jika trafik sedang padat	Dapat memonitor paket data hingga 1000 Mbs tergantung dari jumlah NIDS nya.

Dari tabel 2.7 didapat kesimpulan bahwa CIDS lebih cocok digunakan pada jaringan komputer yang memiliki *host* sedikit. Contoh aplikasi yang menggunakan CIDS adalah ARMD, ASIM, BRO, CMDS, CSM, CyberCop, MIDAS, NADIR, NIDAS, Stalker, Tripwire, dan lain-lain. Sedangkan untuk jaringan yang memiliki *host* banyak disarankan menggunakan DIDS, contoh aplikasi IDS yang menggunakan DIDS yaitu AAFID, CSM, GRIDS, Snort, dan lain-lain.

1.1.1.1.5 *Audit Trail* vs *Network Packets*

Intrusion detection system memiliki dua tipe pengambilan sumber data yaitu *audit trail* dan *network packets.* Audit trail biasanya digunakan oleh jenis HIDS, IDS akan mengenali suatu serangan dari *file log* yang ada pada *server,* sedangkan untuk *network packets,* IDS akan mengenali serangan dari paket-paket data pada jaringan. Dari kedua jenis sumber data yang digunakan IDS ini memiliki beberapa kelemahan dan keunggulan, diantaranya dijelaskan pada tabel 2.8.

Table Error! No text of specified style in document..**8 Perbandingan *Audit Trail* dan *Network Packet***

No	Audit Trail	Network Packet
1	Monitoring data tidak real-time	Monitoring dan Real-time
2	Lebih lambat dalam proses pendeteksian	Cepat dalam mengenali suatu serangan
3	Diterapkan pada HIDS	Diterapkan pada NIDS

Contoh aplikasi IDS yang menggunakan *audit trail* yaitu ASIM, NADIR, Stalker, Tripwire. Contoh aplikasi IDS yang menggunakan *network packet* yaitu AAFID, ARMD, Bro, CMDS, CSM, CyberCop, MIDAS, Snort, dan lain-lain.

1.1.1.1.6 Active vs. Passive

Intrusion detection system memiliki dua aksi yang diterapkan jika mendeteksi suatu pola serangan yaitu active dan passive. IDS active adalah IDS yang akan melakukan aksi berupa pembolkiran terhadap host yang diduga melakukan penyerangan biasanya IDS jenis ini digabungkan dengan IPS (*intrusion prevention System*) berupa *firewall* atau *proxy server.* Sedangkan IDS passive yaitu IDS yang hanya melakkan pencatatan file dan pelaporan kepada administrator jika terdeteksi pola serangan.

1.1.1.1.7 Real time vs Interval-Based

Intrusion Detection System memiliki dua jenis analisis timing yaitu *real time* dan *interval-based.* IDS real time adalah IDS yang melakukan pendeteksian paket data secara *real time,* setiap data yang lewat pada jaringan akan dianalisa

paket datanya. IDS *interval-based* adalah IDS yang melakukan pendeteksian berdasarkan waktu yang ditetapkan, IDS jenis ini akan meng-*audit file log* pada *host* untuk menganalisa paket datanya.

2.1.31. Snort

Snort IDS merupakan IDS open source yang secara defacto menjadi standar IDS di industri. Snort dapat diimplementasikan dalam jaringan yang multiplatform, salah satu kelebihannya adalah mampu mengirimkan alert dari mesin Unix atupun Linux ke platform Microsoft Windows dengan melalui Server Message Block (SMB). Snort dapat bekerja dalam 3 modus: sniffer mode(penyadap), packet loggerdan network intrusion detection mode. Modus kerja yang akan digunakan dalam membangun sistem pencegahan penyusupan adalah modus kerja network intrusion detection. Snort memiliki kemampuan untuk mengumpulkan data log seperti alert dan yg lainnya ke dalam database [**17**].

Snort diciptakan untuk menjadi IDS berbasiskan open source yang berkualitas tinggi. Snort di desain untuk dapat digabung dengan tools yang sudah ada dan kemampuan ekspansi yang tinggi. [**16**].

2.1.1.7 Komponen Snort

Snort dibagi kedalam beberapa komponen. Komponen ini bekerjasama untuk mendeteksi serangan yang berbeda dan untuk menghasilkan keluaran pada format yang diinginkan pada sistem deteksi. IDS berbasiskan Snort umumnya terdiri dari komponen : [**17**]

1. **Dekoder Paket**

Dekoder paket mengambil paket dari beberapa jenis jaringan yang berbeda dari antarmuka jaringan dan mempersiapkan paket untuk di proses atau di kirim menuju detection engine. Antarmukanya dapat berupa Ethernet, SLIP, PPP dan yang lain.

2. Preprocessor

Preprocessor merupakan komponen atau plug-ins yang dapat digunakan pada snort untuk menyusun atau mengubah paket data sebelum detection engine

melakukan beberapa operasi untuk mencari tahu jika paket digunakan oleh penyusup. Preprocessor pada snort dapat mendekode-kan URL HTTP, men-defragmentasi paket, menggabungkan kembali aliran TCP dan yang lain. Fungsi ini merupakan bagian yang sangat penting pada sistem deteksi intrusi.

3. Detection Engine

Detection engine adalah bagian terpenting dari Snort. Tugasnya adalah untuk mendeteksi jika terjadi aktifitas penyusup pada paket. Detection engine mempekerjakan rules Snort untuk tujuan ini. Rules dibaca ke dalam struktur atau rantai data internal kemudian dicocokkan dengan paket yang ada. Jika paket sesuai dengan rules yang ada, tindakan akan diambil, jika tidak paket akan dibuang. Tindakan yang diambil dapat berupa logging paket atau mengaktifkan alert.

Detection engine merupakan bagian dari snort yang sangat bergantung pada waktu tanggap. Waktu tanggap merupakan waktu yang dibutuhkan untuk merespon paket, hal ini bergantung pada seberapa bagus server yang ada dan seberapa banyak rules yang telah didefinisikan. Berikut hal-hal yang mempengaruhi waktu tanggap pada detection engine:

1. Jumlah rules
2. Kekuatan mesin pada sistem
3. \Kecepatan bus internal server snort
4. Beban pada jaringan

Ketika mendesain sistem keamanan komputer berbasiskan NIDS (Network Intrusion Detection System), faktor ini harus dipertimbangkan. Catatan bahwa sistem deteksi dapat membelah paket dan menerapkan rules pada bagian paket yang berbeda. Bagian tersebut diantaranya :

1. IP header paket
2. Header layer transport. Header ini dapat berupa TCP, UDP atau header Layer transport lainnya. Hal ini juga dapat bekerja pada header ICMP
3. Header level layer aplikasi. Header layer aplikasi termasuk header DNS, FTP, SNMP dan SMTP, namun tidak dibatasi hanya header ini saja.

4. Packet payload. Hal ini berarti dapat diciptakan rules yang digunakan oleh detection engine untuk mencari string di dalam data yang ada pada paket.

Detection engine bekerja dengan cara yang berbeda tergantung versi dari Snort. Pada versi 1.x detection engine akan berhenti bekerja jika paket telah sesuai dengan rules. Detection engine mengambil tindakan yang berbeda dengan me-logpaket atau mengaktifkan alert, tergantung pada rules. Hal ini berarti jika paket sesuai dengan kriteria beberapa rules, hanya rules pertama yang akan diterapkan pada paket dan rules yang lain tidak akan dihiraukan. Penanganan jenis ini baik, kecuali pada satu permasalahan. Rules dengan prioritas rendah mengaktifkan alert dengan prioritas rendah, bahkan dengan rules berprioritas tinggi akan menghasilkan alert berprioritas tinggi akan ditempatkan kemudian pada rantai rules. Permasalahan ini kemudian diralat pada snort versi 2 dimana semua rulestelah dicocokkan dengan paket sebelum mengaktifkan alert. Setelah mencocokkan semua rules, rules berprioritas tertinggi memilih untuk mengaktifkan alert.

4. Sistem Log dan Alert

Berdasarkan apa yang ditemukan detection engine pada paket, paket dapat digunakan untuk me-log kegiatan atau mengaktifkan alert, bergantung pada apa yang ditemukan detection engine pada paket. Log di simpan pada format file teks sederhana, file berjenis tcpdump atau bentuk yang lain. File log disimpan di direktori /var/log/snort secara default. Perintah snort -l pada command line dapat digunakan untuk memodifikasi lokasi dari log dan alert yang dihasilkan.

5. Modul Output

Modul Output dapat melakukan beberapa operasi berbeda tergantung bagaimana cara penyimpanan keluaran yang dihasilkan sistem log dan alert dari Snort. Pada dasarnya modul ini mengatur jenis keluaran yang dihasilkan oleh sistem log dan alert.

2.1.1.8 MySQL

MySQL dapat digunakan untuk membuat dan mengola database beserta isinya.Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah dan menghapus data yang berada dalam database.MySQL merupakan sisitem manajemen database yang bersifat at relational. Artinya data-data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat. MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar.MySQL juga dapat menjalankan perintah-perintah Structured Query Language (SQL) untuk mengelola database-database yang ada di dalamnya.Hingga kini, MySQL sudah berkembang hingga versi 5. MySQL 5 sudah mendukung trigger untuk memudahkan pengelolaan tabel dalam database.

Secara umum, SQL terdiri dari dua bahasa, yaitu Data Definition Language (DDL) dan Data Manipulation Language (DML). Implementasi DDL dan DML berbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umum implementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan ANSI. Artikel ini akan menggunakan bentuk paling umum yang dapat digunakan pada kebanyakan SMBD.

Tujuan dari pemakaian SQl dalam menyelesaikan tugas-tugas yang berkaitan dengan database, diantaranya :

1. Memanggil data dari satu table atau lebih
2. Memanipulasi data
3. Mendapatkan ringkasan informasi
4. Membuat, memodifikasi atau menghapus table
5. Membuat atau menghapus index

2.1.1.9 Apache

Server HTTP Apache atau Server Web/WWW Apache adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk

melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini mengunakan HTTP [**16**].

Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah.Apache merupakan perangkat lunak sumber terbuka dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation.

Apache adalah komponen server web dari paket perangkat lunak LAMP (Linux, Apache, MySQL, PHP/Perl/bahasa pemrograman Python). Karena berbagai keunggulan dan kelebihan yang dimiliki web server apache, server web ini menjadi sebuah web server yang paling populer dikalangan pengguna dengan berbagai kelebihan sebagai berikut:

1. Open Source, Free software
2. Apache dapat berjalan di beberapa sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya).
3. Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigurasi, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah.
4. Fleksibel, mudah settingnya (fleksibilitas untuk di setting dengan PHP dan MySQL).
5. Kehandalannnya telah teruji.

2.1.1.10 PHP

PHP merupakan singkatan dari ” Hypertext Preprocessor”, PHP adalah sebuah bahasa scripting atau sering disebut bahasa pemrograman yang terpasang pada HTML. Sebagian besar sintaksnya mirip dengan bahasa pemrograman C, Java, asp dan Perl, ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti. Sejarah PHP, awalnya PHP merupakan kependekan dari Personal

Home Page (situs personal) dan PHP itu sendiri pertama kali di buat oleh Rasmus Lerdorf pada tahun 1995, dan pada saat PHP masih bernama FI (Form Interpreter), yang wujudnya berupa sekumpulan sript yng digunakan untuk mengolah data form dari web dan selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP.

Kegunaan dari PHP adalah untuk membuat tampilan web menjadi lebih dinamis, dengan php kita bisa menampilkan atau menjalankan beberapa file dalam 1 file dengan cara di include atau require, dan php itu sendiri sudah bisa beriteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda, yaitu seperti :

1. FilePro (Personic, Inc),
2. DBM,
3. Informix,
4. Ingres,
5. InterBase,
6. Microsoft Access,
7. MSSQL,
8. MySQL,
9. Oracle
10. PostgrSQL,
11. Sybase.

Dalam cara kerja PHP, yaitu pertama client web browser atau pengguna memakai komputer kemudian pengguna tersebut menjalankan file PHP itu di web browser atau yang biasa di sebut Browser saja dan kemudian File PHP itu di kirim ke web server, Web server mengirimkannya lagi ke Engine PHP atau mesin PHP dan di dalam mesin PHP itu diproses dan setelah diproses oleh mesin PHP maka akan berbentuk file HTML, dan file HTML ini akan di kirimkan ke web server dan web server akan memberikan ke pengguna Software-software Yang Digunakan Software -software yang digunakan dalam menjalakan PHP ini yaitu TexEditor, Web Browser, Web Server, Software PHP dan Database Untuk TextEditor kita bisa memakai notepad bawaannya Windows atau engga kita bisa

juga menggunakan notepad++, TSPad, Dreamweaver atau yang lainnya dan Untuk Web Browser kita bisa menggunaka IE (Internet Explorer) atau Mozilla Firefox atau yang lainnya.

2.1.1.11 BASE (Basic Alnalysis Security Engine)

BASE membaca log yang ditulis Snort pada database. Secara otomatis BASE akan menampilkan nya sebagai alert dan list log dengan antarmuka yang mudah dipaami dan dilihat oleh administrator. BASE juga didefenisikan sebagi berikut:

1. Program berbasis WEB yang memungkinkan implementasi antar Platform.
2. Merupakan perintis antarmuka GUI untuk snort dan paling banyak digunakan oleh pengguna IDS. BASE merupakan rekomendasi dari SNORT.org sendiri
3. Multi language, antarmuka memiliki beberapa bahasa selain bahasa Inggris
4. Layanan Peringatan yang *real time*
5. *Open Source*
6. Dapat diimplementasikan pada IDS manapun selain SNORT