BAB I

PENDAHULUAN

1.1 Latar Belakang

Informasi, adalah suatu hal penting yang sudah menjadi kebutuhan utama umat manusia di dunia, di masa sekarang ini. Dalam kaitan itu, yaitu ketika informasi sudah menjadi kebutuhan utama saat ini, tentu saja dibutuhkan kemudahan-kemudahan untuk mendapatkan informasi tersebut. Salah satunya adalah melalui internet. Seiring dengan perkembangan teknologi pada dunia internet, fungsi dari internet pun berkembang dimana internet menyediakan akses untuk layanan telekomnunikasi dan sumber daya informasi untuk jutaan pemakainya yang tersebar di seluruh dunia

Internet adalah kumpulan dari jaringan-jaringan kecil dan besar yang saling terhubung secara real-time atau terus menerus di seluruh dunia. Internet merupakan salah satu contoh perkembangan teknologi informasi yang dapat memberi peluang kepada setiap orang. Mengapa demikian? hal itu disebabkan karena nilai ekonomi internet yang cukup besar dibandingakan dengan nilai ekonomi asuransi dan kendaraan yaitu sekitar 850 Milyar Dollar per tahunnya [AHM-08:19].

Diawali dari sebuah penelitian pertahanan Amerika Serikat berkembang menjadi sebuah sebuah mesin ekonomi global. Internet menyebabkan perubahan kondisi sosial dan ekonomi. Implikasinya terhadap ekonomi menyebabkan internet menjadi perhatian bagi khalayak akademis [ADD-07]. Dengan demikian, terbentuk kecenderungan penggunaan internet yang semakin hari semakin luas, dengan segala kemudahan dalam hal komunikasi dan transfer data.

Akan tetapi tak hanya keuntungan yang didapat dari berkembangnya penggunaan internet, tapi juga ada beberapa kelemahan, khususnya dalam faktor keamanan.

Dalam faktor keamanan ini biasanya perusahaan menempatkan administrator untuk menjaga. Tetapi fungsi administrator tentunya akan terbatas waktunya, saat jam kerja. Meskipun di jam kerja pun kadang kala karena terlalu banyaknya aliran data tentunya administrator tentunya akan kesulitan menganalisa apakah data yang diterima oleh server adalah data yang diharapkan atau data yang tidak diharapkan. Sedangkan suatu serangan ke sistem keamanan bisa terjadi kapan saja. Baik pada saat administrator sedang kerja atupun tengah malam dimana tidak ada yang menjaga server tersebut. Dengan demikian dibutuhkan sistem pertahanan didalam server itu sendiri yang bisa menganalisa langsung apakah setiap paket yang masuk tersebut adalah data yang diharapkan ataupun data yang tidak diharapkan.[PUJ-09]

Kalau paket tersebut merupakan data yang tidak diharapkan, diusahakan agar komputer bisa mengambil tindakan untuk mengantisipasi agar serangan yang terjadi tidak menimbulkan kerugian yang besar. Akan lebih baik kalau server bisa mengantisipasinya langsung, sehingga kerugian bisa mendekati nol atau tidak ada sama sekali.

Salah satu resiko keamanan dari sistem pertahanan internet adalah flooding data, yaitu Pengiriman data yang berlebihan baik dari besar paket maupun jumlah paket kedalam suatu jaringan dan umumnya merupakan data yang tidak berguna. Flooding data ini sangat mengancam, karena bisa menyebabkan kerusakan sistem

1.2 Rumusan Masalah

Berdasarkan pada permasalahan yang telah dijelaskan pada bagian latar belakang, maka rumusan masalah dikhususkan pada :

- Merancang sistem yang mampu untuk mengambil data dari ethernet card.
- Sistem mampu untuk mengklasifikasikan data-data yang ada
- Sistem juga dapat menyimpan data kedalam database
- Merancang sistem dalam jaringan agar dapat mendeteksi adanya flooding data
- Kemampuan sistem untuk melakukan blocking pada data yang terbukti Flood.

1.3 Ruang Lingkup Pembahasan

Ruang lingkup permasalahan dibatasi oleh:

- Sistem yang dibangun hanya digunakan untuk memantau TCP/IP, UDP, ICMP
- Sistem yang dibangun pada *Operating System* windows
- Sistem memiliki kemampuan mengirimkan perintah blok pada server

1.4 Tujuan Penulisan

Tujuan dari penelitian ini adalah untuk mencegah terjadinya flooding data secara otomatis dan untuk melakukan blok IP apabila terjadi data flood pada sebuah server secara otomatis

1.5 Sistematika Penulisan

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, ruang lingkup permasalahan, tujuan penulisan serta sistematika penulisan.

BAB II Tinjauan Pustaka

Membahas teori dasar dan penunjang mengenai jaringan komputer, topologi jaringan komputer, konsep TCP/IP, UDP, ICMP, dan dasar keamanan jaringan komputer.

BAB III Metode Penelitian

Berisi tentang metode yang digunakan dalam pembahasan skripsi. Metode yang digunakan antara lain studi literatur, perancangan dan implementasi sistem, pengujian dan analisis sistem, serta pengambilan kesimpulan dan saran.

BAB IV Analisis dan Perancangan

Berisi tentang pembahasan mengenai perancangan sistem dan perancangan antarmuka dengan konfigurasi yang sesuai dari Sistem Pencegahan Flooding Data dan Blocking IP

BAB V Implementasi

Berisi tentang pembahasan mengenai implementasi software/hardware yang digunakan, instalasi jaringan komputer, dan konfigurasi atau setting tiap-tiap perangkat yang digunakan dalam Sistem Pencegahan Flooding Data dan Blocking IP

BAB VI Pengujian

Berisi tentang pembahasan mengenai pengujian implementasi Sistem Pencegahan Flooding Data dan Blocking IP

BAB VII Penutup

Berisi tentang kesimpulan dan saran-saran yang diperlukan untuk mengembangkan aplikasi selanjutnya

BAB II
TINJAUAN PUSTAKA

Bab ini menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi ini yang berjudul ”Sistem Pencegahan Flooding Data dan Blocking IP Secara Otomatis Pada Jaringan Komputer”. Dasar teori yang diperlukan berdasarkan kajian pustaka untuk penyusunan skripsi ini adalah meliputi teori dasar dan penunjang mengenai jaringan komputer, topologi jaringan komputer, konsep TCP/IP, UDP, ICMP, dan dasar keamanan jaringan komputer., serta blocking IP sebagai upaya keamanan jaringan.

2.1 Jaringan Komputer

Kebutuhan akan adanya suatu jaringan informasi meningkat dengan pesat. Hal itu seiring dengan kebutuhan kita akan informasi yang bertambah besar. Bagi sebagian masyarakat, informasi telah menjadi barang kebutuhan primer, dan hal tersebut berkaitan erat dengan perkembangan dunia jaringan komputer.

Sebelum era penggunaan jaringan komputer, penggunaan komputer sangat terbatas untuk mesin-mesin stand-alone yang terpisah dan independen antara satu dengan yang lainnya. Tetapi setelah memasuki era penggunaan jaringan, kumpulan komputer-komputer stand-alone tersebut dihubungkan satu dengan yang lainnya dan menjadi suatu jaringan sehingga seluruh informasi dari masing-masing komputer dapat dikorelasikan.

Pengertian jaringan komputer sendiri, dalam standar sistem jaringan, bahwa yang dimaksud engan jaringan adalah beberapa komputer yang saling terhubung dan saling bertukar informasi [AND-06:2]. Di bagian lain, disebutkan bahwa Jaringan Komputer dapat diartikan sebagai suatu himpunan interkoneksi sejumlah komputer

otonom. Dua buah komputer dikatakan membentuk suatu network bila keduanya dapat saling bertukar informasi. Pembatasan istilah otonom disini adalah untuk membedakan dengan sistem master atau slave. Bila sebuah komputer dapat membuat komputer lainnya aktif atau tidak aktif dan mengontrolnya, maka komputer komputer tersebut tidak otonom. Sebuah sistem dengan unit pengendali (*control unit)* dan sejumlah komputer lain yang merupakan slave bukanlah suatu jaringan, demikian pula komputer besar dengan remote printer dan terminalpun bukanlah suatu jaringan [AHM-08:5].

Jaringan (*network*) adalah hubungan dari sistem komunikasi data yang melibatkan sebuah atau lebih sistem komputer yang dihubungkan dengan jalur transmisi dan alat komunikasi membentuk satu sistem. Dengan network, komputer yang satu dapat menggunakan data di komputer yang lain, dapat mencetak laporan di printer komputer yang lain, dapat memberi berita ke komputer yang lain walaupun berlainan area. Jaringan (*network*) merupakan cara yang sangat berguna untuk mengintergrasikan sistem informasi dan menyalurkan arus informasi dari satu area ke area yang lainnya. Jaringan (*network*) dan DDP (*Distributed Data Processing*) masih merupakan hal yang sulit dibedakan untuk beberapa orang. Jaringan dan DDP memang sangat berhubungan erat, tetapi berbeda konsep. Secara umum, jaringan mempunyai beberapa manfaat yang lebih dibandingkan dengan komputer yang berdiri sendiri dan dunia usaha telah pula mengakui bahwa akses ke teknologi informasi modern selalu memiliki keunggulan kompetitif dibandingkan pesaing yang terbatas dalam bidang teknologi.

Jaringan (*network*) merupakan konsep dari jaringan kerja sistem komunikasi data. Jaringan (*network*) dapat melibatkan hanya sebuah sistem komputer saja dengan beberapa terminal di lokasi yang berbeda atau melibatkan beberapa sistem komputer di lokasi yang berbeda. DDP harus melibatkan dua atau lebih sistem komputer yang independen tetapi dapat berhubungan satu dengan yang lainnya. Jadi DDP harus terdiri dari komunikasi dua atau lebih sistem komputer yang masing-masing dapat bekerja secara independen, sedang jaringan (*network*) dapat terdiri dari sebuah sistem komputer saja dengan beberapa terminal [AHM-08:6]

2.1.1 Sejarah Jaringan Komputer

Konsep jaringan komputer lahir pada tahun 1940-an di Amerika dari sebuah proyek pengembangan komputer MODEL I di laboratorium Bell dan group riset Harvard University yang dipimpin profesor H. Aiken. Pada mulanya proyek tersebut hanyalah ingin memanfaatkan sebuah perangkat komputer yang harus dipakai bersama. Untuk mengerjakan beberapa proses tanpa banyak membuang waktu kosong dibuatlah proses beruntun (*Batch Processing*), sehingga beberapa program bisa dijalankan dalam sebuah komputer dengan kaidah antrian.

Ditahun 1950-an ketika jenis komputer mulai membesar sampai terciptanya super komputer, maka sebuah komputer mesti melayani beberapa terminal, untuk itu ditemukan konsep distribusi proses berdasarkan waktu yang dikenal dengan nama TSS (*Time Sharing System*), maka untuk pertama kali bentuk jaringan (*network*) komputer diaplikasikan. Pada sistem TSS beberapa terminal terhubung secara seri ke sebuah host komputer. Dalam proses TSS mulai nampak perpaduan teknologi komputer dan teknologi telekomunikasi yang pada awalnya berkembang sendiri-sendiri.

Memasuki tahun 1970-an, setelah beban pekerjaan bertambah banyak dan harga perangkat komputer besar mulai terasa sangat mahal, maka mulailah digunakan konsep proses distribusi (*Distributed Processing*). Dalam proses ini beberapa host komputer mengerjakan sebuah pekerjaan besar secara paralel untuk melayani beberapa terminal yang tersambung secara seri disetiap host komputer. Dalam proses distribusi sudah mutlak diperlukan perpaduan yang mendalam antara teknologi komputer dan telekomunikasi, karena selain proses yang harus didistribusikan, semua host komputer wajib melayani terminal-terminalnya dalam satu perintah dari komputer pusat.

Selanjutnya ketika harga-harga komputer kecil sudah mulai menurun dan konsep proses distribusi sudah matang, maka penggunaan komputer dan jaringannya sudah mulai beragam dari mulai menangani proses bersama maupun komunikasi antar komputer (Peer to Peer) saja tanpa melalui komputer pusat. Untuk itu mulailah berkembang teknologi jaringan lokal yang dikenal dengan sebutan Local Area Network (LAN). Demikian pula ketika Internet mulai diperkenalkan, maka sebagian besar LAN

yang berdiri sendiri mulai berhubungan dan terbentuklah jaringan raksasa Wide Area Network (WAN).

2.1.2 Tipe Jaringan Komputer

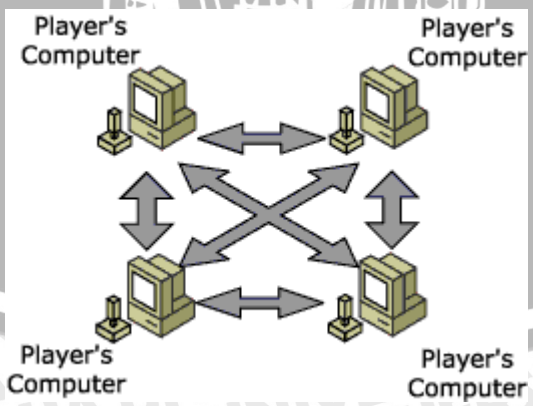
2.1.2.1 Tipe Jaringan Komputer Menurut Fungsinya

Menurut fungsi komputer pada sebuah jaringan, maka tipe jaringan komputer dapat dibedakan menjadi dua tipe, yaitu:

- Jaringan *peer to peer* atau *point to point*
- Jaringan *client-server*

2.1.2.1.1 Jaringan Peer to Peer

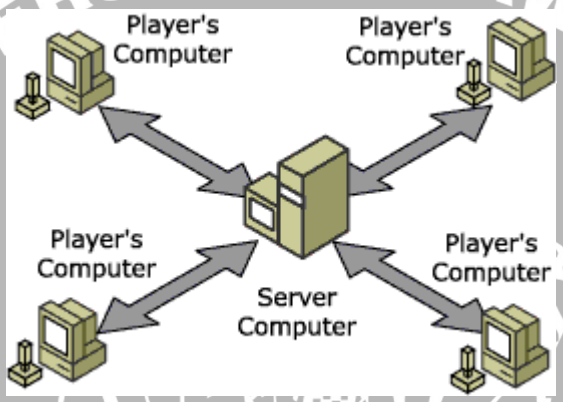
Pada jaringan *peer to peer* setiap komputer yang terhubung pada jaringan dapat berkomunikasi dengan komputer-komputer lain secara langsung tanpa melalui komputer perantara. Pada jaringan tipe ini sumber daya komputer terbagi pada seluruh komputer yang terhubung dalam jaringan tersebut, baik sumber daya yang berupa perangkat keras maupun perangkat lunak dan datanya. Gambar tipe jaringan *peer to peer* dapat dilihat dalam Gambar.



Gambar 2.1 Jaringan *peer to peer*

2.1.2.1.2 Jaringan Client-Server

Pada jaringan *client-server* terdapat satu atau lebih komputer yang berfungsi sebagai *server* sedangkan komputer-komputer yang lain berfungsi sebagai *client* atau disebut juga dengan *workstation*. Sesuai namanya maka komputer *server* berfungsi dan bertugas melayani seluruh komputer yang terdapat dalam jaringan tersebut, sedangkan komputer client menerima pelayanan dari komputer *server*. Gambar tipe jaringan *client-server* dapat dilihat dalam Gambar .



2 **Gambar 2.2** Jaringan *Client – Server*

2.1.2.2 Tipe Jaringan Komputer Menurut Cakupan Geografis

Sedangkan berdasarkan cakupan geografis jaringan komputer dibedakan menjadi 4 yaitu :

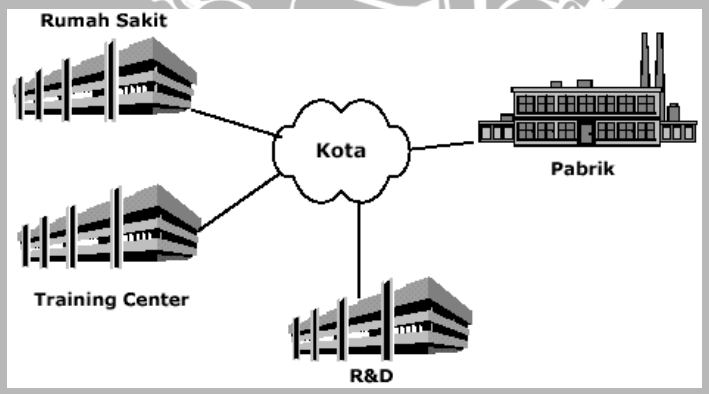
- LAN (Local Area Network)
- MAN (Metropolitan Area Network)
- WAN (Wide Area Network)
- GAN (Global Area Network)

2.1.2.2.1 LAN (Local Area Network)

LAN digunakan untuk menghubungkan komputer yang berada di dalam suatu area yang kecil, misalnya di dalam suatu gedung perkantoran atau kampus. Jarak antar komputer yang dihubungkannya bisa mencapai 5 sampai 10 km. Suatu LAN biasanya bekerja pada kecepatan mulai 10 Mbps sampai 100 Mbps. LAN menjadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya secara bersama-sama. Contoh dari sumber daya yang dapat digunakan itu misalnya suatu mainframe, file server, printer, dan sebagainya.

2.1.2.2.2 MAN (Metropolitan Area Network)

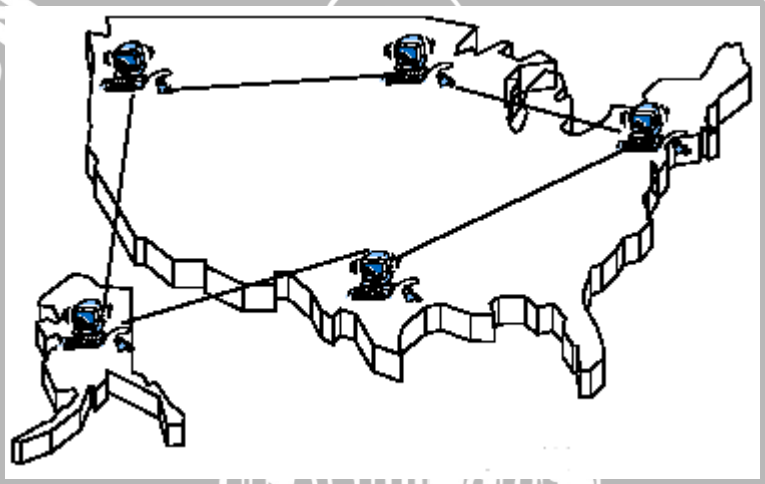
MAN merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN bisa mencapai 10 km sampai beberapa ratus km. Suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps. Pada Gambar anda dapat melihat suatu ilustrasi tentang MAN.



Gambar 2.3 Jaringan MAN

2.1.2.2.3 WAN (Wide Area Network)

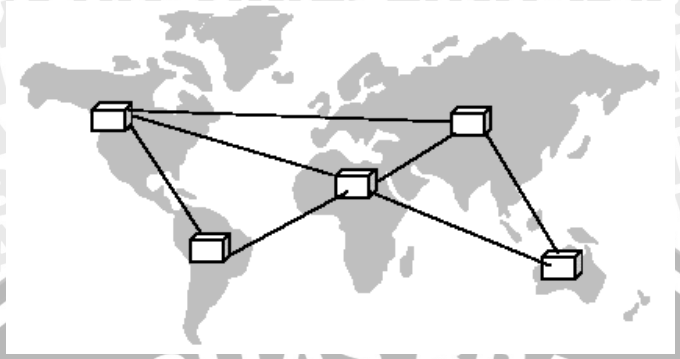
WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas, seperti hubungan dari satu kota ke kota lain di dalam suatu negara. Cakupan WAN bisa meliputi 100 km sampai 1.000 km, dan kecepatan antar kota bisa bervariasi antara 1,5 Mbps sampai 2,4 Gbps. Dalam WAN, biaya untuk peralatan transmisi sangat tinggi, dan biasanya jaringan WAN dimiliki dan dioperasikan sebagai suatu jaringan publik. Para pelaku bisnis dapat menyewa sistem transmisi tersebut untuk menghubungkan kantor-kantor cabang yang dimilikinya. Gambar mengilustrasikan suatu WAN.



4 **Gambar 2.4** Jaringan WAN

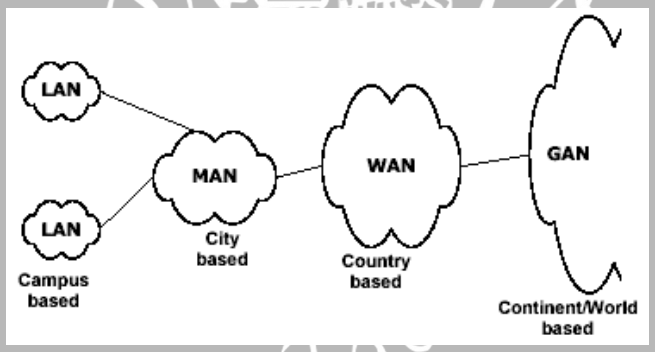
2.1.2.2.4 GAN (Global Area Network)

GAN merupakan suatu jaringan yang menghubungkan negara-negara di seluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps sampai dengan 100 Gbps dan cakupannya mencakapi ribuan kilometer. Contoh yang sangat baik dari GAN ini adalah Internet. Gambar memperlihatkan contoh suatu GAN.



5 **Gambar 2.5** Jaringan GAN

LAN, MAN, WAN dan GAN dapat berinteraksi satu sama lain. Gambar memperlihatkan interaksi antara jaringan-jaringan tersebut.



Gambar 2.6 interaksi antara LAN, MAN, WAN, GAN

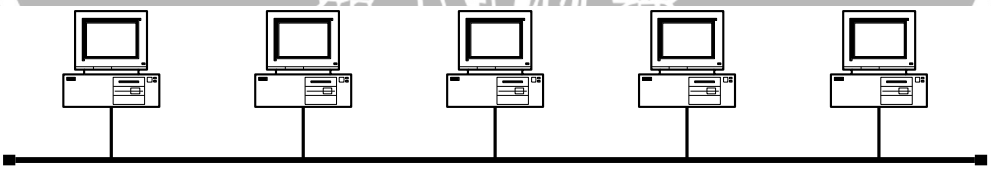
Interface yang digunakan antara jaringan-jaringan tersebut sudah ditentukan di dalam suatu standard interface internasional maupun regional. Standard-standard ini memungkinkan peralatan-peralatan yang berasal dari vendor yang berbeda dapat dihubungkan satu sama lain

2.1.3 Topologi Jaringan Komputer

Topologi jaringan komputer terbagi menjadi dua yaitu topologi fisik dan topologi logic. Topologi jaringan fisik berkaitan dengan bentuk fisik dari jaringan sedangkan untuk topologi logic berkaitan dengan teknologi atau logika hubungan yang digunakan dalam jaringan. Topologi fisik dibagi menjadi tiga yaitu topologi bus, topologi ring, dan topologi star.

2.1.3.1 Topologi Bus

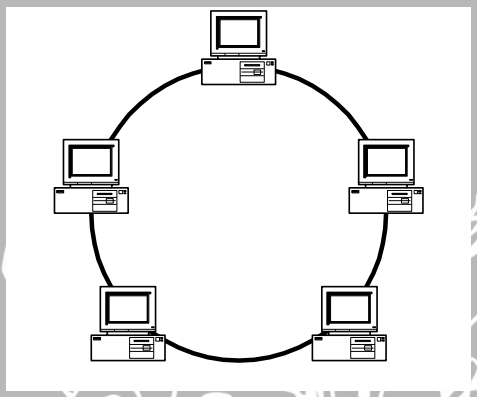
Pada topologi bus seluruh komputer dalam sebuah jaringan terhubung pada sebuah bus. Topologi ini menggunakan kabel coaxial RG-58 sebagai media komunikasi antar komputer. Gambar menunjukkan bentuk jaringan topologi bus. Topologi ini mempunyai kelemahan pada tingkat komunikasi data yang cukup padat, sehingga kemungkinan terjadinya tabrakan komunikasi antara beberapa komputer menjadi sangat besar. Hal ini akan menyebabkan turunnya kecepatan lalulintas data. Kelemahan yang lain adalah jika salah satu node dalam jaringan mengalami kerusakan maka seluruh jaringan akan terhenti sama sekali.



Gambar 2.7 jaringan dengan topologi bus

2.1.3.2 Topologi Ring

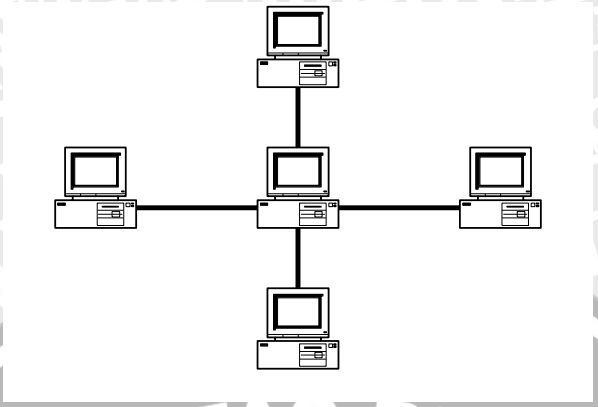
Topologi ring (Cincin), yaitu server dihubungkan ke beberapa workstation yang membentuk lingkaran (cincin), sehingga awal dan ujung kabel penghubung akan berada di server. Pada sistem ini, sebuah token1 dikirim dari satu simpul (node) ke simpul lainnya. Kelemahan topologi ini hampir sama dengan topologi bus yaitu jika terjadi kerusakan pada pada salah satu node dalam jaringan akan menghentikan komunikasi dalam jaringan tersebut. Untuk kecepatan topologi ring lebih cepat daripada menggunakan topologi bus. Gambar. memperlihatkan bentuk dari topologi ring.



8 **Gambar 2.8** jaringan dengan topologi ring

2.1.3.3 Topologi Star

Topologi star berbeda dengan dua topologi sebelumnya, topologi ini menggunakan jalur komunikasi untuk masing-masing komputer dalam jaringan yang dihubungkan ke pusat. Komunikasi pada jaringan ini diatur oleh pusat jaringan. Dengan digunakan jalur komunikasi yang berbeda ketika terjadi kerusakan pada salah satu node dalam jaringan tidak akan mempengaruhi kinerja jaringan secara keseluruhan. Topologi jaringan ini mempunyai kecepatan komunikasi lebih baik daripada topologi jaringan yang lain (bus dan ring), tetapi kinerja dari jaringan sangat dipengaruhi oleh pusat jaringan. Gambar dari topologi star dapat dilihat dalam Gambar .



9 **Gambar 2.9** jaringan dengan topologi star

Sedangkan topologi *logic* ada beberapa macam, bahkan terus dikembangkan bentuk-bentuk baru jaringan baru. Beberapa topologi *logic* yang telah ada antara lain adalah topologi ArcNet, Token Ring, Ethernet, dan lainnya. Ethernet yang saat ini banyak digunakan dalam jaringan komputer telah berkembang menjadi *Fast Ethernet* yang mempunyai kecapatan 100Mbps lebih cepat 10 kali lipat dari Ethernet yang mempunyai kecepatan 10 Mbps.

2.2 OSI (Open System Interconnection)

2.2.1 Model Referensi OSI

Untuk menyelenggarakan komunikasi berbagai macam vendor komputer diperlukan sebuah aturan baku yang standar dan disetejui berbagai fihak. Seperti halnya dua orang yang berlainan bangsa, maka untuk berkomunikasi memerlukan penerjemah/interpreter atau satu bahasa yang dimengerti kedua belah fihak. Dalam dunia komputer dan telekomunikasi interpreter identik dengan protokol. Untuk itu maka badan dunia yang menangani masalah standarisasi ISO (*International Standardization Organization*) membuat aturan baku yang dikenal dengan nama model referensi OSI (*Open System Interconnection*). Dengan demikian diharapkan semua vendor perangkat telekomunikasi haruslah berpedoman dengan model referensi ini dalam mengembangkan protokolnya. Model referensi OSI terdiri dari 7 lapisan, mulai dari

lapisan fisik sampai dengan aplikasi. Model referensi ini tidak hanya berguna untuk produk-produk LAN saja, tetapi dalam membangung jaringan Internet sekalipun sangat diperlukan. Hubungan antara model referensi OSI dengan protokol Internet bisa dilihat dalam tabel berikut:.

No	Lapisan
7	Aplikasi
6	Persentasi
5	Sessi
4	Transport
3	Network
2	Data link
1	Fisik

Table 2.1 model OSI

TCP/IP	Protokol TCP/IP	
	Nama Protokol	Kegunaan
Aplikasi	DHCP (Dynamic Host Configuration Protocol)	Protokol untuk distribusi IP pada jaringan dengan jumlah IP yang terbatas
	DNS (Domain Name Server)	Data base nama domain mesin dan nomer IP
	FTP (File Transfer Protocol)	Protokol untuk transfer file
	HTTP (HyperText Transfer Protocol)	Protokol untuk transfer file HTML dan Web
	MIME (Multipurpose Internet Mail Extention)	Protokol untuk mengirim file binary dalam bentuk teks
	NNTP (Networ News Transfer Protocol)	Protokol untuk menerima dan mengirim newsgroup
	POP (Post Office Protocol)	Protokol untuk mengambil mail dari server
	SMB (Server Message Block)	Protokol untuk transfer berbagai server file DOS dan Windows
	SMTP (Simple Mail Transfer Protocol)	Protokol untuk pertukaran mail
	SNMP (Simple Network	Protokol untuk manejemen jaringan

	Management Protocol)	
	Telnet	Protokol untuk akses dari jarak jauh
	TFTP (Trivial FTP)	Protokol untuk transfer file
	NETBIOS (Network Basic Input Output System)	BIOS jaringan standar
	RPC (Remote Procedure Call)	Prosedur pemanggilan jarak jauh
	SOCKET	Input Output untuk network jenis BSD-UNIX
Transport	TCP (Transmission Control Protocol)	Protokol pertukaran data beroriantasi (connection oriented)
	UDP (User Datagram Protocol)	Protokol pertukaran data non-oriantasi (connectionless)
Internet	IP (Internet Protocol)	Protokol untuk menetapkan routing
	RIP (Routing Information Protocol)	Protokol untuk memilih routing
	ARP (Address Resolution Protocol)	Protokol untuk mendapatkan informasi hardware dari nomer IP
	RARP (Reverse ARP)	Protokol untuk mendapatkan informasi nomer IP dari hardware
Network Interface	PPP (Point to Point Protocol)	Protokol untuk point ke point
	SLIP (Serial Line Internet Protocol)	Protokol dengan menggunakan sambungan serial
	Ethernet, FDDI, ISDN, ATM	

Tabel 2.2 Lapisan TCP/IP

Prinsip-prinsip yang digunakan bagi ketujuh layer tersebut adalah :

1. Sebuah layer harus dibuat bila diperlukan tingkat abstraksi yang berbeda.
2. Setiap layer harus memiliki fungsi-fungsi tertentu.

3. Fungsi setiap layer harus dipilih dengan teliti sesuai dengan ketentuan standar protocol internasional.
4. Batas-batas layer diusahakan agar meminimalkan aliran informasi yang melewati interface.
5. Jumlah layer harus cukup banyak, sehingga fungsi-fungsi yang berbeda tidak perlu disatukan dalam satu layer diluar keperluannya. Akan tetapi jumlah layer juga harus diusahakan sesedikit mungkin sehingga arsitektur jaringan tidak menjadi sulit dipakai.

2.2.2 Karakteristik Lapisan OSI

Ke tujuh lapisan dari model referensi OSI dapat dibagi ke dalam dua kategori, yaitu lapisan atas dan lapisan bawah.

Lapisan atas dari model OSI berurusan dengan persoalan aplikasi dan pada umumnya diimplementasi hanya pada software. Lapisan tertinggi (lapisan applikasi) adalah lapisan penutup sebelum ke pengguna (user), keduanya, pengguna dan lapisan aplikasi saling berinteraksi proses dengan software aplikasi yang berisi sebuah komponen komunikasi. Istilah lapisan atas kadang-kadang digunakan untuk menunjuk ke beberapa lapisan atas dari lapisan lapisan yang lain di model OSI.

Lapisan bawah dari model OSI mengendalikan persoalan transport data. Lapisan fisik dan lapisan data link diimplementasikan ke dalam hardware dan software. Lapisan-lapisan bawah yang lain pada umumnya hanya diimplementasikan dalam software. Lapisan terbawah, yaitu lapisan fisik adalah lapisan penutup bagi media jaringan fisik (misalnya jaringan kabel), dan sebagai penanggung jawab bagi penempatan informasi pada media jaringan. Tabel berikut ini menampilkan pemisahan kedua lapisan tersebut pada lapisan-lapisan model OSI.

Application	Application	Lapisan Atas
Presentation		
Session		
Transport	Data Transport	Lapisan Bawah
Network		
Data Link		
Physical		

Tabel 2.3 Pemisahan Lapisan atas dan Lapisan bawah pada model OSI

2.2.3 Protokol

Model OSI menyediakan secara konseptual kerangka kerja untuk komunikasi antar komputer, tetapi model ini bukan merupakan metoda komunikasi. Sebenarnya komunikasi dapat terjadi karena menggunakan protokol komunikasi. Di dalam konteks jaringan data, sebuah protokol adalah suatu aturan formal dan kesepakatan yang menentukan bagaimana komputer bertukar informasi melewati sebuah media jaringan. Sebuah protokol mengimplementasikan salah satu atau lebih dari lapisan-lapisan OSI. Sebuah variasi yang lebar dari adanya protokol komunikasi, tetapi semua memelihara pada salah satu aliran group: protokol LAN, protokol WAN, protokol jaringan, dan protokol routing. Protokol LAN beroperasi pada lapisan fisik dan data link dari model OSI dan mendefinisikan komunikasi di atas macam-macam media LAN. Protokol WAN beroperasi pada ketiga lapisan terbawah dari model OSI dan mendefinisikan komunikasi di atas macam-macam WAN. Protokol routing adalah protokol lapisan jaringan yang bertanggung jawab untuk menentukan jalan dan pengaturan lalu lintas. Akhirnya protokol jaringan adalah berbagai protokol dari lapisan teratas yang ada dalam sederetan protokol. [MOE–07:19]

2.2.4 Lapisan-lapisan Model OSI

2.2.4.1 Physical Layer

Physical Layer berfungsi dalam pengiriman raw bit ke channel komunikasi. Masalah desain yang harus diperhatikan disini adalah memastikan bahwa bila satu sisi mengirim data 1 bit, data tersebut harus diterima oleh sisi lainnya sebagai 1 bit pula, dan bukan 0 bit. Pertanyaan yang timbul dalam hal ini adalah : berapa volt yang perlu digunakan untuk menyatakan nilai 1? dan berapa volt pula yang diperlukan untuk angka 0?. Diperlukan berapa mikrosekon suatu bit akan habis? Apakah transmisi dapat diproses secara simultan pada kedua arahnya? Berapa jumlah pin yang dimiliki jaringan dan apa kegunaan masing-masing pin? Secara umum masalah-masalah desain yang ditemukan di sini berhubungan secara mekanik, elektrik dan interface prosedural, dan media fisik yang berada di bawah physical layer.[MOE-07:20]

2.2.4.2 Data Link Layer

Tugas utama data link layer adalah sebagai fasilitas transmisi raw data dan mentransformasi data tersebut ke saluran yang bebas dari kesalahan transmisi. Sebelum diteruskan kenetwork layer, data link layer melaksanakan tugas ini dengan memungkinkan pengirim memecag-mecah data input menjadi sejumlah data frame (biasanya berjumlah ratusan atau ribuan byte). Kemudian data link layer mentransmisikan frame tersebut secara berurutan, dan memproses acknowledgement frame yang dikirim kembali oleh penerima. Karena physical layer menerima dan mengirim aliran bit tanpa mengindahkan arti atau arsitektur frame, maka tergantung pada data link layer-lah untuk membuat dan mengenali batas-batas frame itu. Hal ini bisa dilakukan dengan cara membubuhkan bit khusus ke awal dan akhir frame. Bila secara insidental pola-pola bit ini bisa ditemui pada data, maka diperlukan perhatian khusus untuk menyakinkan bahwa pola tersebut tidak secara salah dianggap sebagai batas-batas frame.[MOE-07:20]

Terjadinya noise pada saluran dapat merusak frame. Dalam hal ini, perangkat lunak data link layer pada mesin sumber dapat mengirim kembali frame yang rusak

tersebut. Akan tetapi transmisi frame sama secara berulang-ulang bisa menimbulkan duplikasi frame. Frame duplikat perlu dikirim apabila acknowledgement frame dari penerima yang dikembalikan ke pengirim telah hilang. Tergantung pada layer inilah untuk mengatasi masalah-masalah yang disebabkan rusaknya, hilangnya dan duplikasi frame. Data link layer menyediakan beberapa kelas layanan bagi network layer. Kelas layanan ini dapat dibedakan dalam hal kualitas dan harganya.

Masalah-masalah lainnya yang timbul pada data link layer (dan juga sebagian besar layer-layer di atasnya) adalah mengusahakan kelancaran proses pengiriman data dari pengirim yang cepat ke penerima yang lambat. Mekanisme pengaturan lalu-lintas data harus memungkinkan pengirim mengetahui jumlah ruang buffer yang dimiliki penerima pada suatu saat tertentu. Seringkali pengaturan aliran dan penanganan error ini dilakukan secara terintegrasi.

Saluran yang dapat mengirim data pada kedua arahnya juga bisa menimbulkan masalah. Sehingga dengan demikian perlu dijadikan bahan pertimbangan bagi software data link layer. Masalah yang dapat timbul di sini adalah bahwa frame-frame acknoeledgement yang mengalir dari A ke B bersaing saling mendahului dengan aliran dari B ke A.

Jaringan broadcast memiliki masalah tambahan pada data link layer. Masalah tersebut adalah dalam hal mengontrol akses ke saluran yang dipakai bersama. Untuk mengatasinya dapat digunakan sublayer khusus data link layer, yang disebut medium access sublayer.

2.2.4.3 Network Layer

Network layer berfungsi untuk pengendalian operasi subnet. Masalah desain yang penting adalah bagaimana caranya menentukan route pengiriman paket dari sumber ke tujuannya. Route dapat didasarkan pada table statik yang “dihubungkan ke” network. Route juga dapat ditentukan pada saat awal percakapan misalnya session terminal. Terakhir, route dapat juga sangat dinamik, dapat berbeda bagi setiap paketnya. Oleh karena itu, route pengiriman sebuah paket tergantung beban jaringan saat itu.

Bila pada saat yang sama dalam sebuah subnet terdapat terlalu banyak paket, maka ada kemungkinan paket-paket tersebut tiba pada saat yang bersamaan. Hal ini dapat menyebabkan terjadinya bottleneck. Pengendalian kemacetan seperti itu juga merupakan tugas network layer.[MOE-07:22]

Karena operator subnet mengharap bayaran yang baik atas tugas pekerjaannya. seringkali terdapat beberapa fungsi accounting yang dibuat pada network layer. Untuk membuat informasi tagihan, setidaknya software mesti menghitung jumlah paket atau karakter atau bit yang dikirimkan oleh setiap pelanggannya. Accounting menjadi lebih rumit, bilamana sebuah paket melintasi batas negara yang memiliki tarip yang berbeda.

Perpindahan paket dari satu jaringan ke jaringan lainnya juga dapat menimbulkan masalah yang tidak sedikit. Cara pengalamatan yang digunakan oleh sebuah jaringan dapat berbeda dengan cara yang dipakai oleh jaringan lainnya. Suatu jaringan mungkin tidak dapat menerima paket sama sekali karena ukuran paket yang terlalu besar. Protokolnyapun bisa berbeda pula, demikian juga dengan yang lainnya. Network layer telah mendapat tugas untuk mengatasi semua masalah seperti ini, sehingga memungkinkan jaringan-jaringan yang berbeda untuk saling terinterkoneksi.

2.2.4.4 Transport Layer

Fungsi dasar transport layer adalah menerima data dari session layer, memecah data menjadi bagian-bagian yang lebih kecil bila perlu, meneruskan data ke network layer, dan menjamin bahwa semua potongan data tersebut bisa tiba di sisi lainnya dengan benar. Selain itu, semua hal tersebut harus dilaksanakan secara efisien, dan bertujuan dapat melindungi layer-layer bagian atas dari perubahan teknologi hardware yang tidak dapat dihindari.

Dalam keadaan normal, transport layer membuat koneksi jaringan yang berbeda bagi setiap koneksi transport yang diperlukan oleh session layer. Bila koneksi transport memerlukan throughput yang tinggi, maka transport layer dapat membuat koneksi jaringan yang banyak. Transport layer membagi-bagi pengiriman data ke sejumlah jaringan untuk meningkatkan throughput. Di lain pihak, bila pembuatan atau

pemeliharaan koneksi jaringan cukup mahal, transport layer dapat menggabungkan beberapa koneksi transport ke koneksi jaringan yang sama. Hal tersebut dilakukan untuk membuat penggabungan ini tidak terlihat oleh session layer.

Transport layer juga menentukan jenis layanan untuk session layer, dan pada gilirannya jenis layanan bagi para pengguna jaringan. Jenis transport layer yang paling populer adalah saluran error-free point to point yang meneruskan pesan atau byte sesuai dengan urutan pengirimannya. Akan tetapi, terdapat pula jenis layanan transport lainnya. Layanan tersebut adalah transport pesan terisolasi yang tidak menjamin urutan pengiriman, dan membroadcast pesan-pesan ke sejumlah tujuan. Jenis layanan ditentukan pada saat koneksi dimulai.

Transport layer merupakan layer end to end sebenarnya, dari sumber ke tujuan. Dengan kata lain, sebuah program pada mesin sumber membawa percakapan dengan program yang sama dengan pada mesin yang dituju. Pada layer-layer bawah, protokol terdapat di antara kedua mesin dan mesin-mesin lain yang berada didekatnya. Protokol tidak terdapat pada mesin sumber terluar atau mesin tujuan terluar, yang mungkin dipisahkan oleh sejumlah router. Perbedaan antara layer 1 sampai 3 yang terjalin, dan layer 4 sampai 7 yang end to end.[MOE-07:23]

Sebagai tambahan bagi penggabungan beberapa aliran pesan ke satu channel, transport layer harus hati-hati dalam menetapkan dan memutuskan koneksi pada jaringan. Proses ini memerlukan mekanisma penamaan, sehingga suatu proses pada sebuah mesin mempunyai cara untuk menerangkan dengan siapa mesin itu ingin bercakap-cakap. Juga harus ada mekanisme untuk mengatur arus informasi, sehingga arus informasi dari host yang cepat tidak membanjiri host yang lambat. Mekanisme seperti itu disebut pengendalian aliran dan memainkan peranan penting pada transport layer (juga pada layer-layer lainnya). Pengendalian aliran antara host dengan host berbeda dengan pengendalian aliran router dengan router. Kita akan mengetahui nanti bahwa prinsip-prinsip yang sama digunakan untuk kedua jenis pengendalian tersebut.

2.2.4.5 Session Layer

Session layer mengijinkan para pengguna untuk menetapkan session dengan pengguna lainnya. Sebuah session selain memungkinkan transport data biasa, seperti yang dilakukan oleh transport layer, juga menyediakan layanan yang istimewa untuk aplikasi-aplikasi tertentu. Sebuah session digunakan untuk memungkinkan seseorang pengguna log ke remote timesharing system atau untuk memindahkan file dari satu mesin kemesin lainnya.

Sebuah layanan session layer adalah untuk melaksanakan pengendalian dialog. Session dapat memungkinkan lalu lintas bergerak dalam bentuk dua arah pada suatu saat, atau hanya satu arah saja. Jika pada satu saat lalu lintas hanya satu arah saja (analog dengan rel kereta api tunggal), session layer membantu untuk menentukan giliran yang berhak menggunakan saluran pada suatu saat.

Layanan session di atas disebut manajemen token. Untuk sebagian protokol, adalah penting untuk memastikan bahwa kedua pihak yang bersangkutan tidak melakukan operasi pada saat yang sama. Untuk mengatur aktivitas ini, session layer menyediakan token-token yang dapat digilirkan. Hanya pihak yang memegang token yang diijinkan melakukan operasi kritis.

Layanan session lainnya adalah sinkronisasi. Ambil contoh yang dapat terjadi ketika mencoba transfer file yang berdurasi 2 jam dari mesin yang satu ke mesin lainnya dengan kemungkinan mempunyai selang waktu 1 jam antara dua crash yang dapat terjadi. Setelah masing-masing transfer dibatalkan, seluruh transfer mungkin perlu diulangi lagi dari awal, dan mungkin saja mengalami kegagalan lain. Untuk mengurangi kemungkinan terjadinya masalah ini, session layer dapat menyisipkan tanda tertentu ke aliran data. Karena itu bila terjadi crash, hanya data yang berada sesudah tanda tersebut yang akan ditransfer ulang.

2.2.4.6 Pressentation Layer

Pressentation layer melakukan fungsi-fungsi tertentu yang diminta untuk menjamin penemuan sebuah penyelesaian umum bagi masalah tertentu. Pressentation

Layer tidak mengijinkan pengguna untuk menyelesaikan sendiri suatu masalah. Tidak seperti layer-layer di bawahnya yang hanya melakukan pemindahan bit dari satu tempat ke tempat lainnya, presentation layer memperhatikan syntax dan semantik informasi yang dikirimkan.

Satu contoh layanan pressentation adalah encoding data. Kebanyakan pengguna tidak memindahkan string bit biner yang random. Para pengguna saling bertukar data sperti nama orang, tanggal, jumlah uang, dan tagihan. Item-item tersebut dinyatakan dalam bentuk string karakter, bilangan interger, bilangan floating point, struktur data yang dibentuk dari beberapa item yang lebih sederhana. Terdapat perbedaan antara satu komputer dengan komputer lainnya dalam memberi kode untuk menyatakan string karakter (misalnya, ASCII dan Unicode), integer (misalnya komplemen satu dan komplemen dua), dan sebagainya. Untuk memungkinkan dua buah komputer yang memiliki presentation yang berbeda untuk dapat berkomunikasi, struktur data yang akan dipertukarkan dapat dinyatakan dengan cara abstrak, sesuai dengan encoding standard yang akan digunakan "pada saluran". Presentation layer mengatur data-struktur abstrak ini dan mengkonversi dari representation yang digunakan pada sebuah komputer menjadi representation standard jaringan, dan sebaliknya.

2.2.4.7 Application Layer

Application layer terdiri dari bermacam-macam protokol. Misalnya terdapat ratusan jenis terminal yang tidak kompatibel di seluruh dunia. Ambil keadaan dimana editor layar penuh yang diharapkan bekerja pada jaringan dengan bermacam-macam terminal, yang masing-masing memiliki layout layar yang berlainan, mempunyai cara urutan penekanan tombol yang berbeda untuk penyisipan dan penghapusan teks, memindahkan sensor dan sebagainya.

Suatu cara untuk mengatasi masalah seperti di atas, adalah dengan menentukan terminal virtual jaringan abstrak, serhingga editor dan program-program lainnya dapat ditulis agar saling bersesuaian. Untuk menangani setiap jenis terminal, satu bagian software harus ditulis untuk memetakan fungsi terminal virtual jaringan ke terminal

sebenarnya. Misalnya, saat editor menggerakkan cursor terminal virtual ke sudut layar kiri, software tersebut harus mengeluarkan urutan perintah yang sesuai untuk mencapai cursor tersebut. Seluruh software terminal virtual berada pada application layer.

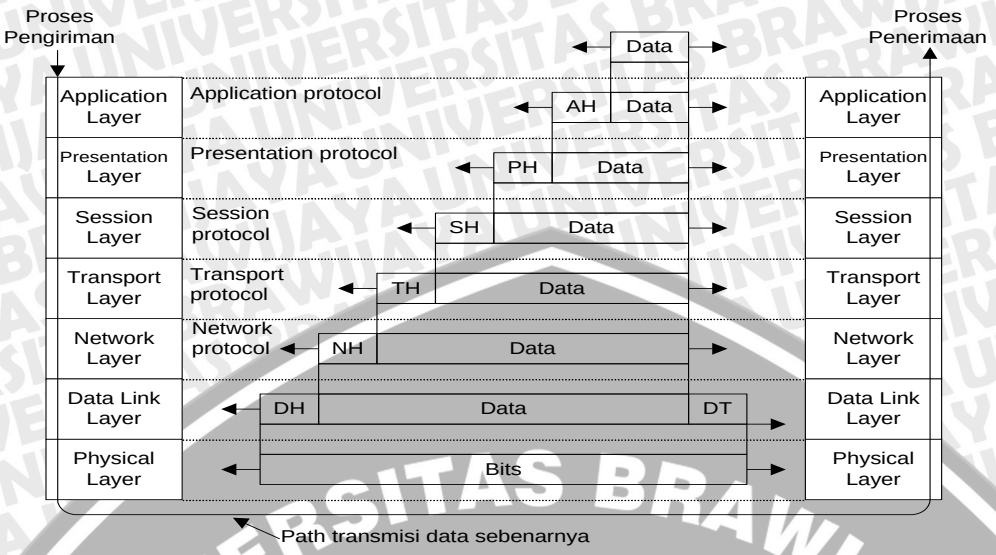
Fungsi application layer lainnya adalah pemindahan file. Sistem file yang satu dengan yang lainnya memiliki konvensi penamaan yang berbeda, cara menyatakan baris-baris teks yang berbeda, dan sebagainya. Perpindahan file dari sebuah sistem ke sistem lainnya yang berbeda memerlukan penanganan untuk mengatasi adanya ketidak-kompatibelan ini. Tugas tersebut juga merupakan pekerjaan appication layer, seperti pada surat elektronik, remote job entry, directory lookup, dan berbagai fasilitas bertujuan umum dan fasilitas bertujuan khusus lainnya.

2.2.5 Transmisi Data Pada Model OSI

Pada gambar di bawah ini dijelaskan sebuah contoh tentang bagaimana data dapat ditransmisikan dengan menggunakan model OSI. Proses pengiriman memiliki data yang akan dikirimkan ke proses penerima. Proses pengirim menyerahkan data ke application layer, yang kemudian menambahkan aplication header, AH (yang mungkin juga kosong), ke ujung depannya dan menyerahkan hasilnya ke presentation layer.

Pressentation layer dapat membentuk data ini dalam berbagai cara dan mungkin saja menambahkan sebuah header di ujung depannya, yang diberikan oleh session layer. Penting untuk diingat bahwa presentation layer tidak menyadari tentang bagian data yang mana yang diberi tanda AH oleh application layer yang merupakan data pengguna yang sebenarnya.

Proses pemberian header ini berulang terus sampai data tersebut mencapai physical layer, dimana data akan ditransmisikan ke mesin lainnya. Pada mesin tersebut, semua header tadi dicopoti satu per satu sampai mencapai proses penerimaan.



Gambar 2.10 Contoh tentang bagaimana model OSI digunakan

Yang menjadi kunci di sini adalah bahwa walaupun transmisi data aktual berbentuk vertikal seperti pada gambar, setiap layer diprogram seolah-olah sebagai transmisi yang bersangkutan berlangsung secara horizontal. Misalnya, saat transport layer pengiriman mendapatkan pesan dari session layer, maka transport layer akan membubuhkan header transport layer dan mengirimkannya ke transport layer penerima.

2.3 Internet Protocol (IP)

Lapisan internet bertanggung jawab untuk pengiriman data melalui antar jaringan. Protokol lapisan internet yang utama adalah Internet Protocol (IP). IP merupakan protocol internet yang mempunyai fungsi sebagai berikut :

1. Pengalamatan
2. Fragmentasi datagram pada antar jaringan
3. Pengiriman datagram antar jaringan

Diantara fungsi tersebut yang paling berkepentingan dengan administrator jaringan adalah fungsi pengalamatan. IP mempunyai pola pengalamatan yang unik yang membutuhkan waktu untuk membiasakannya.

2.3.1 Format Alamat IP

Alamat-alamat IP panjangnya 32 bit dan dibagi menjadi dua field:

1. Suatu field netid menunjukkan jaringan kemana host dihubungkan
2. Suatu field hostid memberikan suatu pengenal unik setiap host pada suatu jaringan.

Pada terminologi TCP/IP, suatu jaringan terdiri dari sekelompok host yang dapat berkomunikasi secara langsung tanpa router. Semua host TCP/IP yang menempati jaringan yang sama harus diberi netid yang sama. Host yang mempunyai netid harus bekomunikasi melalui router.

Suatu TCP/IP internetwork adalah sebuah jaringan dari beberapa jaringan, dan dapat menggabungkan banyak jaringan yang dihubungkan oleh router. Setiap jaringan pada internetwork harus diberi netid yang unik.

2.3.2 Macam-macam protokol IP

2.3.2.1 Internet Control Message Protocol (ICMP)

ICMP merupakan IP yang tidak didesain sebagai protokol yang handal. ICMP hanya bertugas untuk mengirimkan pesan-pesan khusus yang tidak memerlukan keamanan yang tinggi dan pengirimannya dalam bentuk data pada datagram IP. Pesan yang di bawa oleh ICMP antara lain :

1. Destination Unreachable. Pesan ini menyediakan informasi ketika host, jaringan port atau protokol tertentu tidak dapat dijangkau.
2. Time Exceeded. Pesan ini memberitahu pengirim bahwa datagram tidak dapat dikirim karena Time To Live sudah habis
3. Parameter Problem. Pesan ini menunjukkan adanya masalah pada parameter dan letak oktet dimana kesalahan terjadi.
4. Source Quench. Pesan ini dapat dikirimkan oleh router atau host tujuan yang terpaksa harus membuang datagram karena batasan yang ada pada ruang buffer.

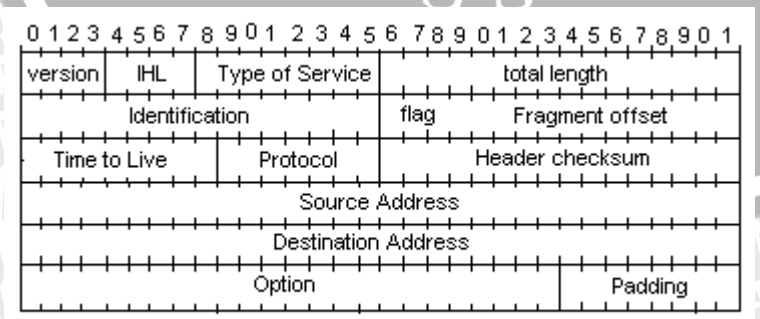
5. Redirect. Pesan ini dikirimkan ke host saat router menerima datagram yang dapat dirouting lebih cepat menggunakan gateway lain. Pesan ini memberi saran kepada host asal mengenai router yang lebih tepat untuk menerima datagram ini.
6. Echo Request dan Echo Replay Message. Pesan ini saling mempertukarkan data timestamp (penanda waktu pengiriman pesan) antara host
7. Information request dan information replay. Pesan ini digunakan agar host dapat mengetahui keadaan jaringan terhubung.

ICMP Router Discovery Message (RFC 1256) adalah perluasan ICMP yang menambahkan kemampuan bagi host untuk menentukan rute ke gateway. Pemberitahuan router (router Advertisement) dikirim secara multicast setiap periode waktu tertentu memberitahukan alamat IP untuk interface ke jaringan. Host memperoleh informasi rute dengan mendengarkan, pemberitahuan ini. Saat sebuah host dihidupkan , host dapat memberikan Router Solication untuk meminta pemberitahuan dengan segera . Teknik ini memberikan informasi mengenai router yang tersedia tetapi tidak dapat memberikan informasi jalur yang terbaik.

2.3.2.1.1 Format header ICMP

Operasi pada protokol IP dapat dibuat bermacam macam tergantung dengan penggunaan berbagai parameter yang terdapat pada header IP. Kebanyakan dari parameter ini dapat ditangani oleh Windows NT TCP/IP.

Format header ICMP :



Gambar 2.11 Format Header ICMP

Header IP berisi field-field berikut ini:

1. Version (4 bit). Menunjukkan format dari internet header. Versi saat ini sebagaimana dijelaskan pada RFC 791 adalah versi 4.
2. Internet header length (IHL: 4 bit). Menjelaskan panjang dari header menggunakan 32-bit word. Ukuran minimum header yang diijinkan adalah 5 word.
3. Type of service / jenis servis (8 bit). Data pada field ini menunjukkan kualitas layanan yang diinginkan.

   Pengaruh dari nilai field diatas bergantung pada teknologi jaringan yang dipakai dan nilai-nilai ini harus di konfigurasikan berdasarkan hal-hal tersebut. Tidak seluruh pilihan pada field ini kompatibel, saat sebuah layanan khusus diinginkan pilihan harus dibuat antara beberapa kemungkinan yang tersedia yaitu waktu tunda yang singkat (low delay), keterandalan yang tinggi (high realibility), atau kecepatan yang tinggi (high throughput). Untuk kerja yang lebih baik di satu bagian sering mengurangi kemampuan dari bagian lain. Hanya ada sedikit kasus yang memerlukan digunakannya tiga pilihan ini sekaligus.
4. Total Length/panjang keseluruhan (16 bit). Panjang keseluruhan ICMP dalam oktet, termasuk header dan data IP. Field ini memungkinkan datagram berisi sampai 66535 oktet. Standar yang ada menganjurkan tiap host bersiap siap untuk menerima datagram dengan panjang paling tidak 576 oktet.
5. Identification (16 bit). Field identifikasi digunakan untuk membantu proses penggabungan kembali pecahan-pecahan dari sebuah datagram.
6. Flag (3 bit). Field ini berisi tiga control flag.
7. Bit 0. Dicadangkan , harus 0
8. Bit 1 (DF). 0 = bisa dipecah menjadi fragmen; 1 = tidak boleh dipecah
9. Bit 2 (MF). 0 = fragmen terakhir; 1 = masih ada fragmen lagi
10. Bila sebuah datagram dipecah, MF bit untuk tiap fragmen kecuali yang terakhir bernilai 1

11. Fragment Offset / posisi fragmen (13 bit). Untuk datagram yang dipecah, menunjukkan posisi fragmen ini dalam datagram.
12. Time To Live / waktu hidup (8 bit). Menunjukkan waktu maksimum bagi sebuah datagram untuk berada dalam suatu jaringan. Bila field ini memberi nilai 0, datagram akan dibuang. Field ini di modifikasi selama tahap pemrosesan header IP dan umum nya dihitung dalam detik. Namun tiap modul IP yang menangani datagram harus mengurangi Time To Live ini dengan 1. mekanisme ini memastikan bahwa datagram yang tak terkirim suatu saat akan dibuang.
13. Protokol (8 bit). Protokol lapisan atas yang berhubungan dengan bagian data dari datagram.
14. Header checksum (16 bit) Sebuah nilai checksum untuk header saja. Nilai ini harus dihitung ulang tiap kali header dimodifikasi.
15. Source Address (32 bit). Alamat IP dari host yang mengirimkan datagram.
16. Destination Address (32 bit). Alamat IP dari host yang merupakan tujuan akhir datagram.
17. Option (0 sampai 11 32-bit word). Dapat berisi 0 atau lebih pilihan.

2.3.2.2 User Datagram Protocol (UDP)

User Datagram Protocol (UDP; RFC 768) memberikan alternatif transport untuk proses yang tidak membutuhkan pengiriman yang handal. UDP adalah protokol datagram yang tidak menjamin pengiriman data atau perlindungan duplikasi. Sebagai protokol datagram, UDP tidak mengurusi penerimaan aliran data dan pembuatan segmen yang sesuai untuk IP. Akibatnya, UDP adalah protokol sederhana yang berjalan dengan overhead jauh lebih kecil dari TCP.

Daftar berikut menjelaskan beberapa hal mengapa dipakai UDP sebagai protokol transport:

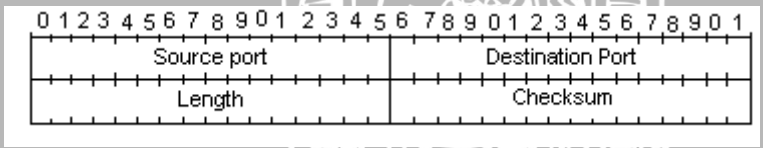
1. Pesan yang tidak membutuhkan jawaban. Overhead jaringan dapat dikurangi dengan menggunakan UDP. Pesan peringatan Simple Network Management

Protocol (SNMP) termasuk dalam kategori ini. Pada jaringan yang besar ada cukup banyak peringatan SNMP yang dikirimkan setiap kali peralatan SNMP mengalami update status. Namun jarang sekali kehilangan pesan ini berakibat fatal. Dengan menjalankan SNMP terhadap UDP, terjadi pengurangan overhead yang cukup berarti.

2. Pesan antara host bersifat sporadis. SNMP contoh yang baik dalam hal ini. Pesan SNMP dikirim pada interval yang tidak teratur. Overhead yang dibutuhkan untuk membuka dan menutup hubungan TCP untuk tiap pesan akan menghambat pengiriman dan manjatuhkan perfoma sistem.
3. Keterandalan dilakukan dalam level proses. Network File Sistem (NFS) adalah contoh dari proses yang memiliki fungsi keterandalannya sendiri dan berjalan terhadap UDP untuk meningkatkan perfoma jaringan

2.3.2.2.1 Format header UDP

Format header UDP adalah sebagai berikut

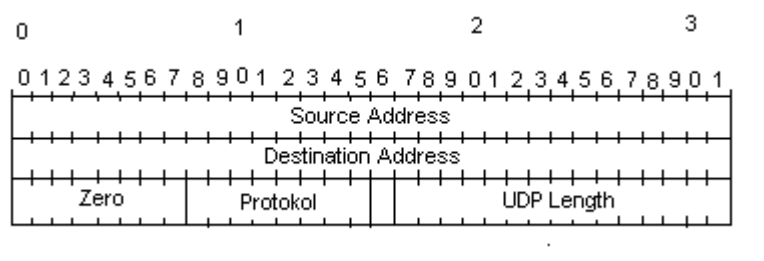


Gambar 2.12 Format Header UDP

1. Port asal / Source Port (16 bit). Field ini bersifat optional dan menunjukkan port asal saat menerima datagram dimungkinkan mengirimkan balasan . bila tidak nilai port asal adalah 0
2. Port tujuan / Destination Port (16 bit). Port tujuan pada host penerima.
3. Panjang (16 bit). Panjang datagram adalah satuan oktet, termasuk header dan data . Nilai minimum yang dimungkinkan oleh header adalah 8. dengan

demikian, datagram UDP memiliki panjang maksimum 65535 oktet, dimana 65527 oktet dapat dipergunakan untuk data.

4. Checksum (16 bit). Nilai checksum meliputi data pada pseudoheader, header UDP dan data.
5. Seperti TCP, UDP juga menghasilkan pseudoheader yang diteruskan oleh datagram UDP ke IP. Pseudoheader UDP berjaga-jaga terhadap datagram yang salah rute. Berikut ini menunjukkan pseudoheader pada UDP:



Gambar 2.13 Pseudoheader UDP

2.3.2.3 Transmission Control Protokol (TCP)

Transmission Control Protokol adalah rotocol yang handal. Protokol ini berusaha keras secara seksama untuk mengirimkan data ke tujuan, memeriksa kesalahan, mengirimkan data ulang bila diperlukan dan mengirimkan error kelapisan atas hanya bila TCP tidak berhasil mengadakan komunikasi. TCP di desain untuk memenuhi kebutuhan DoD akan pengiriman data akurat pada masa dimana jaringan wide-area masih belum begitu handal, dan tetap sesuai digunakan untuk aplikasi yang membutuhkan pengiriman data yang handal. Tetapi perlu dicatat bahwa keandalan TCP dapat tercapai dengan mengorbankan bandwith yang besar.TCP (RFC 793) memberikan komunikasi yang handal antar proses

yang berjalan pada host yang saling terhubung. Komunikasi host-to-host ini fungsinya independen terhadap struktur jaringan yang dipakai. TCP tidak mengurusi proses routing data melalui internetwork; infrastruktur jaringan adalah tanggung jawab IP.

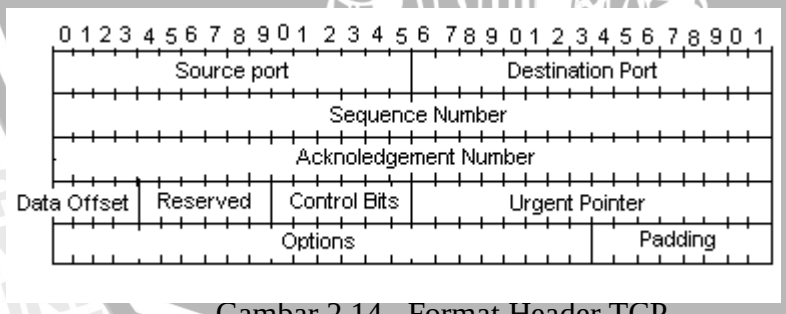
Pada lapisan host-to-host , TCP pada host yang satu berkomunikasi langsung dengan TCP pada host yang lain, tidak peduli apakah kedua host ini berada pada satu jaringan atau jaringan mereka terpisah satu dengan yang lainnya. Pada kenyataannya TCP tidak terdapat pada router kecuali fungsi router tersebut dilakukan pada host yang menjalankan proses lapisan atas (misal :windows NT dapat melakukan routing pada komputer yang digunakan untuk workstation).Pada kenyataanya, TCP diabaikan oleh jaringan. Banyak jenis teknologi jaringan lokal maupun wide-area, termasuk circuit switching dan paket switching. TCP mengenali host menggunakan IP address dan tidak memperdulikan alamat fisik.

Karakteristik dan fungsi TCP adalah sebagai berikut:

1. Penanganan aliran data dengan proses dan aplikasi lapisan atas.
2. Tersedianya komunikasi yang handal
3. Penanganan hubungan yang baik
4. Tersedianya jenjang dan keamanan

2.3.2.3.1 Format header TCP

TCP memiliki format header untuk tiap segmen yang dikirimkan ke IP, header TCP mengikuti header Ipdatagram.Dapat dilihat jelas sebagai berikut



Gambar 2.14 Format Header TCP

Segmen TCP disusun dalam 16-bit word. Bila sebuah segmen berisi jumlah oktet yang ganjil, akan ditambahkan oktet akhir yang berisi nol.

Field dalam header TCP adalah sebagai berikut:

1. Port Asal / Source Port (16 bit). Menunjukkan port pada modul TCP pengirim.
2. Port Tujuan / Destination Port (16 bit). Menunjukkan port pada modul TCP penerima .
3. Nomor urut / Sequence Number (32 bit). Menunjukkan posisi urutan dari oktet data pertama dari segmen. Saat sebuah segmen membuka hubungan ke jaringan yang lain (bit SYN di-set), nomor urut adalah nomor urut awal (Initial Sequence Number/ISN) dan oktet pertama pada field data adalah pada urutan ISN + 1.
4. Nomor Acknoledgment / acknoledgement Number (32 bit). Berisi nomor urutan berikutnya yang diharapkan oleh pengirim segmen, TCP menandakan bahwa field ini aktif dengan men-set bit ACK, yang selalu di set setelah sebuah hubungan terjadi.
5. Posisi Data / Data Offset (4 bit). Berisi jumlah 32 bit word pada header TCP. Option akan ditambahi oktet bernilai-0 bila dibutuhkan untuk melengkapi 32-bit word ini.
6. Reserved (6 bit). Selalu berisi nol. Dicadangkan untuk penggunaan mendatang.
7. Control Bit (6 bit). Enam kontol bit sebagai berikut:
   a. URG. Saat di set, field urgent pointer memiliki makna saat di clear (0), field tersebut diabaikan
   b. ACK, saat di set, field acknoledgement number memiliki makna
   c. PSH. Memulai fungsi push
   d. RST. Memaksa hubungan di reset.
   e. SYN. Melakukan sinkronisasi nomor urutan untuk hubungan. Bit ini di set sebuah segmen membuka hubungan baru.
   f. FIN. Tidak ada lagi data lagi. Hubungan di tutup.
   g. Window (16 bit). Berisi jumlah oktet, dimulai dari oktet yang disebutkan pada field acknoledgementy number, yang dapat diterima saat ini oleh pengirim segmen.

h. Checksum (16 bit). Checksum untuk error control yang meliputi header dan field data. Tidak termasuk penambahan data 0 sesudah option agar nomor oktet menjadi genap. Checksum juga meliputi 96-bit pseudoheader

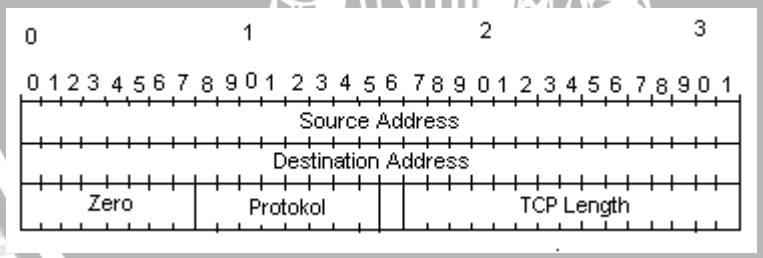
i. Urgent Pointer (16 bit) . menunjukkan nomor urutan oktet menyusul data yang mendesak. Urgent Pointer adalah bilangan positif berisi posisi dari nomor urutan pada segmen.

j. Option (variable). Option tersedia untuk berbagai fungsi termasuk akhir daftar option, no-operation, ukuran segmen maksimum, dan ukuran option data segmen maksimum.

k. Padding (variable). Oktet bernilai 0 yang ditambahkan pada header untuk memastikan bahwa header berukuran kelipatan 32-bit word.

l. Terdapat 12 oktet pseudoheader pada TCP yang berisi alamat asal dan tujuan, protokol dan ukuran segmen. Informasi ini diteruskan oleh segmen ke IP untuk melindungi TCP dari segmen yang salah rute. Nilai dari field panjang segmen (segmen legth) termasuk header dan data TCP, tetapi tidak termasuk panjang dari pseudoheader.

Berikut ini format dari pseudoheader dari TCP :



Gambar 2.15 Pseudoheader TCP

2.4 TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol) adalah salah satu jenis protokol yg memungkinkan kumpulan komputer untuk berkomunikasi dan bertukar data didalam suatu network (jaringan). Dalam pengertian lain, TCP/IP adalah sekelompok protokol yang mengatur komunikasi data komputer di internet. Karena menggunakan bahasa/protokol yang sama, perbedaan jenis komputer dan sistem operasi tidak menjadi masalah dalam komunikasi data. [FON-07:4]

TCP/IP menjadi penting karena TCP/IP merupakan protokol yg telah diterapkan pada hampir semua perangkat keras dan sistem operasi. Tidak ada rangkaian protokol lain yg tersedia pada semua sistem berikut ini :

1. Novel Netware.
2. Mainframe IBM.
3. Sistem digital VMS.
4. Server Microsoft Windows NT
5. Workstation UNIX, LinuX, FreeBSD
6. Personal komputer DOS.

Konsep TCP/IP berawal dari kebutuhan DoD (Departement of Defense) AS akan suatu komunikasi di antara berbagai variasi komputer yg telah ada. Komputer-komputer DoD ini seringkali harus berhubungan antara satu organisasi peneliti dg organisasi peneliti lainnya, dan harus tetap berhubungan sehingga pertahanan negara tetap berjalan selama terjadi bencana, seperti ledakan nuklir. Oleh karenanya pada tahun 1969 dimulailah penelitian terhadap serangkaian protokol TCP/IP. Di antara tujuan-tujuan penelitian ini adalah sebagai berikut :

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

Tahun 1968 DoD ARPAnet (Advanced Reseach Project Agency) memulai penelitian yg kemudian menjadi cikal bakal packet switching . Packet switching inilah yg kemungkinkan komunikasi antara lapisan network (dibahas nanti) dimana data dijalankan dan disalurkan melalui jaringan dalam bentuk unit-unit kecil yg disebut paket. Tiap-tiap packet ini membawa informasi alamatnya masing-masing yg ditangani dengan khusus oleh jaringan tersebut dan tidak tergantung dengan paket-paket lain.

Jaringan yg dikembangkan ini, yg menggunakan ARPAnet sebagai tulang punggungnya, menjadi terkenal sebagai internet. Protokol-protokol TCP/IP dikembangkan lebih lanjut pada awal 1980 dan menjadi protokol-protokol standar untuk ARPAnet pada tahun 1983.

Protokol-protokol ini mengalami peningkatan popularitas di komunitas pemakai ketika TCP/IP digabungkan menjadi versi 4.2 dari BSD (Berkeley Standard Distribution) UNIX. Versi ini digunakan secara luas pada institusi penelitian dan pendidikan dan digunakan sebagai dasar dari beberapa penerapan UNIX komersial, termasuk SunOS dari Sun dan Ultrix dari Digital. Karena BSD UNIX mendirikan hubungan antara TCP/IP dan sistem operasi UNIX, banyak implementasi UNIX sekarang menggabungkan TCP/IP.

Ciri-ciri protokol TCP/IP sendiri dapat dijelaskan sebagai berikut:

1. dikembangkan menggunakan standar protokol yang terbuka.
2. dikembangkan dengan tidak tergantung pada sistem operasi atau perangkat keras tertentu
3. dikembangkan dengan konsensus dan tidak tergantung pada vendor tertentu
4. independen terhadap perangkat keras jaringan
5. pengalamatan TCP/IP bersifat unik dalam skala global
6. memiliki fasilitas routing sehingga dapat diterapkan pada inter-network
7. memiliki banyak jenis layanan.

2.4.1 Layanan TCP/IP

2.4.1.1 Pengiriman File (FileTransfer)

File Transfer Protokol (FTP) memungkinkan pengguna komputer yg satu untuk dapat mengirim ataupun menerima file ke komputer jaringan. Karena masalah keamanan data, maka FTP seringkali memerlukan nama pengguna (user name) dan password, meskipun banyak juga FTP yg dapat diakses melalui anonymous, alias tidak berpassword.

2.4.1.2 Remote Login

Network terminal Protokol (telnet) memungkinkan pengguna komputer dapat melakukan log in ke dalam suatu komputer didalam suatu jaringan. Jadi hal ini berarti bahwa pengguna menggunakan komputernya sebagai perpanjangan tangan dari komputer jaringan tersebut.

2.4.1.3 Computer Mail

Digunakan untuk menerapkan sistem elektronik mail.

2.4.1.4 Network File System (NFS)

Pelayanan akses file-file jarak jauh yg memungkinkan klien-klien untuk mengakses file-file pada komputer jaringan jarak jauh walaupun file tersebut disimpan secara lokal.

2.4.1.5 Remote Execution

Memungkinkan pengguna komputer untuk menjalankan suatu program didalam komputer yg berbeda. Biasanya berguna jika pengguna menggunakan komputer yg terbatas, sedangkan ia memerlukan sumber yg banyak dalam suatu system komputer. Ada beberapa jenis remote execution, ada yg berupa perintah-perintah dasar saja, yaitu yg dapat dijalankan dalam system komputer yg sama dan ada pula yg menggunakan "prosedure remote call system", yg memungkinkan program untuk memanggil

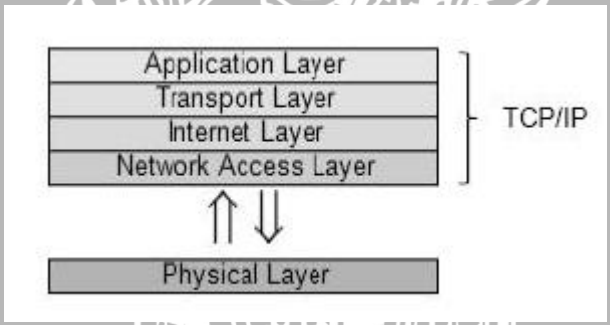
subroutine yg akan dijalankan di system komputer yg berbeda. (sebagai contoh dalam Berkeley UNIX ada perintah “rsh” dan “rexec”)

2.4.1.6 Name Servers

Nama database alamat yg digunakan pada internet.

2.4.2 Arsitektur TCP/IP

Dikarenakan TCP/IP adalah serangkaian protokol di mana setiap protokol melakukan sebagian dari keseluruhan tugas komunikasi jaringan, maka tentulah implementasinya tak lepas dari arsitektur jaringan itu sendiri. Arsitektur rangkaian protokol TCP/IP mendifinisikan berbagai cara agar TCP/IP dapat saling menyesuaikan. Berikut dijelaskan dalam diagram gambar arsitektur TCP/IP.



Gambar 2.16 Arsitektur TCP/IP [FON-07:5]

2.4.2.1 Network Access Layer

Protokol pada layer ini menyediakan media bagi sistem untuk mengirimkan data ke peralatan lain yang terhubung secara langsung. Network Access Layer merupakan gabungan antara Network, Data Link dan Physical Layer dalam standar OSI. Fungsi dalam layer ini adalah:

1. mengubah IP datagram ke frame yang ditransmisikan oleh network.

2. memetakan IP Address ke *physical address* yang digunakan dalam jaringan. IP Address ini harus diubah ke alamat apapun yang diperlukan oleh *physical layer* untuk mentransmisikan datagram.

2.4.2.2 Internet Layer

Dalam layer ini terdapat empat buah protokol yaitu:

1. IP (Internet Protocol)
   - Protokol IP merupakan inti dari protokol TCP/IP. Seluruh data yang berasal dari protokol pada layer di atas IP harus dilewatkan, diolah oleh protokol IP, dan dipancarkan sebagai paket IP, agar sampai ke tujuan.
   - Dalam melakukan pengiriman data, IP memiliki sifat yang dikenal sebagai unreliable, connectionless, datagram delivery service.
2. ICMP (Internet Control Message Protocol)

   Menyediakan fungsi kontrol dalam pengiriman pesan, seperti memberitahukan kepada pengirim pesan bahwa pesan yang dikirimkan tidak sampai ke tujuan.
3. ARP (Address Resolution Protocol)

   Menentukan alamat data link layer untuk IP address yang telah dikenal.
4. RARP (Reverse Address Resolution Protocol)

   Menentukan *Network Address* pada saat alamat data link diketahui.

2.4.2.3 Transport Layer

Pada transport layer terdapat dua buah protocol :

1. *TCP*, bersifat *connection, oriented, reliable*, dan *byte stream service*.
   - Connection Oriented berarti sebelum melakukan pertukaran data, dua aplikasi pengguna TCP harus melakukan hubungan (handshake) terlebih dahulu.

- Reliable berarti TCP menerapkan proses deteksi kesalahan paket dan retransmisi.
- Byte Stream Service berarti paket dikirimkan dan sampai ke tujuan secara berurutan.

2. *UDP*, bersifat *connectionless* dan *unreliable*.
   - Walaupun bertanggung jawab untuk mentransmisikan pesan/data, tidak ada software yang mengecek pengantara setiap segmen yang dikirimkan oleh layer ini.
   - Keuntungan penggunaan UDP adalah kecepatannya, karena pada UDP tidak ada *acknowledgement* sehingga trafik yang lewat jaringan rendah.

2.4.2.4 Application Layer

Menyediakan spesifikasi standar untuk panduan pembuatan aplikasi yang berjalan di atas protokol TCP/IP. Beberapa contoh standar di antaranya:

- TELNET, yaitu Network Terminal Protocol, yang menyediakan remote login dalam jaringan.
- FTP, File Transfer Protocol, digunakan untuk file transfer.
- SMTP, Simple Mail Transfer Protocol, digunakan untuk mengirimkan *electronic mail*.
- DNS, Domain Name Service, untuk memetakan IP Address ke dalam nama tertentu.
- HTTP, Hyper Text Transfer Protokol, protokol untuk web browsing.

2.5 Keamanan Jaringan Komputer

Jaringan komputer merupakan kumpulan dari beberapa komputer yang memliki koneksi satu dengan yang lain. Ketika semua komputer saling terhubung dalam suatu jaringan, keamanan merupakan hal yang harus diperhatikan. Fungsi keamanan adalah membuat jaringan komputer menjadi stabil, terstruktur, kuat serta mampu mengatasi berbagai gangguan. Logikanya, bila dalam suatu jaringan komputer ternyata memiliki sisi keamanan yang lemah tentu hal ini berdampak merusak kestabilan jaringan komputer tersebut. Banyaknya gangguan yang masuk akibat lemahnya keamanan yang dibuat bisa merusak kinerja transfer data pada jaringan komputer. Oleh karena itu, gangguan menjadi parameter untuk mengukur tingkat keamanan.[YOD - 07]

2.5.1 Flood Data

Traffic data yang ada dalam suatu jaringan akan mengalami turun naik selama pemakaiannya. Pada jam-jam sibuk *traffic* suatu data akan sangat padat, sehingga traffic data tersebut akan terganggu. Baik data yang akan dikirim maupun data yang akan datang akan mengalami antrian data yang mengakibatkan kelambatan dalam pengiriman dan penerimaan data.

Tetapi adakalanya data-data yang berada dalam *traffic* merupakan data yang tidak perlu. Data-data tersebut memang sengaja di dikirim oleh sesorang untuk merusak jaringan data yang ada. Perngiriman data tersebut dapat mengakibatkan lambatnya jalur *traffic* yang ada dalam jaringan, dan juga bisa mengakibatkan kerugian lain yang cukup berarti, misalnya kerusakan alat ataupun kerusakan program karena adanya *intruder* yang masuk ke dalam jaringan. Pengiriman data yang berlebihan baik dari besar paket maupun jumlah paket kedalam suatu jaringan dan umumnya merupakan data yang tidak berguna biasa disebut Flood.

Macam-macam Flood attack :

1. Ping of death

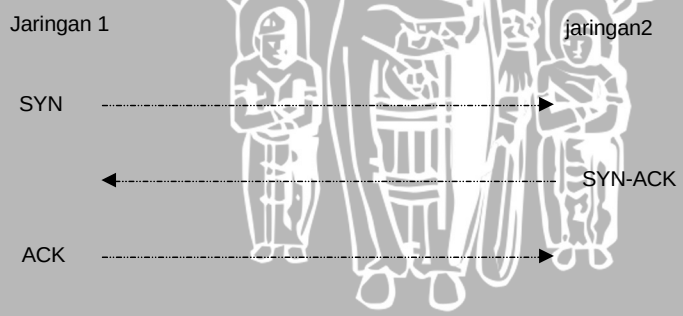
Pengiriman paket *echo request* ICMP ke dalam suatu jaringan secara berlebihan. Pengiriman paket ini dapat mengakibatkan sistem *crash, hang* ataupun *reboot*

2. Smurf Attack

Hampir sama dengan *Ping of death* tetapi untuk *smurf attack* paket ICMP tidak dikirim secara langsung ke korban, melainkan melalui perantara. Pada awalnya dikirim sebuah paket ICMP *echo request* ke sebuah host lain, paket ini bertujuan agar host tersebut mengirimkan paket ICMP PING secara terus menerus ke korban terakhirnya.

3. Syn Flooding

Dalam proses pengiriman data yang melalui TCP, proses data yang terjadi adalah sebagai berikut :



Gambar 2.17 Proses data TCP

Hubungan TCP dimulai dengan mengirimkan paket SYN-TCP ke host yang dituju, pengiriman paket SYN adalah merupakan pembuka untuk membuka jalur koneksi antara dua *host* melalui protokol TCP. Apabila hubungan tersebut disetujui host tujuan akan mengirimkan paket SYN-ACK sebagai tanda bahwa jalur sudah

terbentuk. Dan bagian terakhir adalah pengiriman paket ACK dari *host* awal ke host tujuan sebagai konfirmasi.

Sedangkan flood SYN terjadi bila suatu host hanya mengirimkan paket SYN TCP saja secara kontinyu tanpa mengirimkan paket ACK sebagai konfirmasinya. Hal ini akan menyebabkan host tujuan akan terus menunggu paket tersebut dengan menyimpannya kedalam *backlog*. Meskipun besar paket kecil, tetapi apabila pengiriman SYN tersebut terus menerus akan memperbesar *backlog*. Hal yang terjadi apabila backlog sudah besar akan mengakibatkan host tujuan akan otomatis menolak semua paket SYN yang datang, sehingga host tersebut tidak bisa dikoneksi oleh host-host yang lain.

4. UDP flood

Pengiriman data UDP secara berlebihan kedalam suatu jaringan, pengiriman UDP flood ini akan membentuk suatu jalur hubungan dengan suatu servis UDP dari *host* tujuan. Flood UDP ini akan mengirimkan karakter-karakter yang akan mengetes jaringan korban. Sehingga terjadi aliran data yang tidak perlu dalam jaringan korban tersebut.

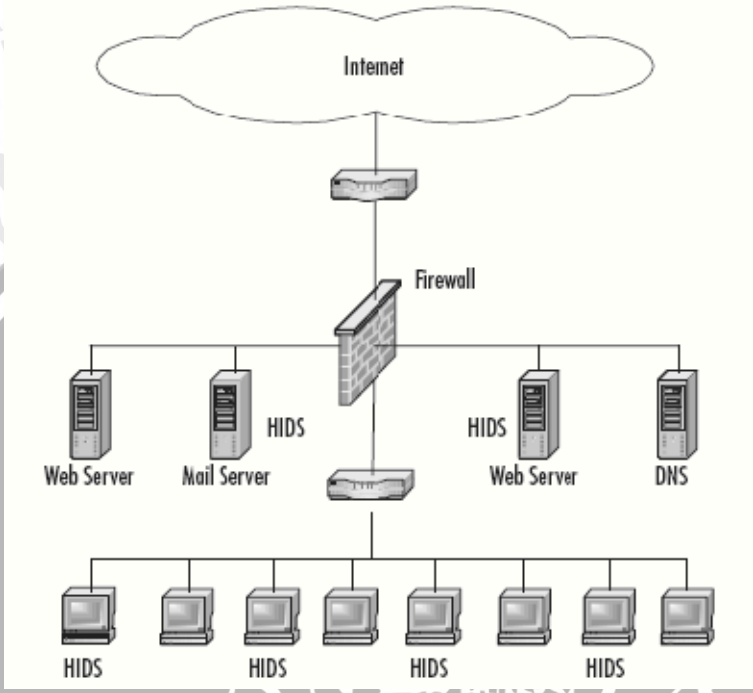
2.6 Intrusion Detection System (IDS)

Suatu Intrusion Detection System (IDS) dapat didefinisikan sebagai alat, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer [ARI-07:27]. Dilihat dari kemampuan mendeteksi penyusupan pada jaringan, IDS dibagi menjadi dua.

2.6.1 Host Intrusion Detection System (HIDS)

Host-based IDS bekerja pada host yang akan dilindungi. Host-based IDS memperoleh informasi dari data yang dihasilkan oleh sistem pada sebuah komputer yang diamati. Data Host-based IDS biasanya berupa log yang dihasilkan dengan memonitor sistem file, event, dan keamanan pada Windows NT dan syslog pada

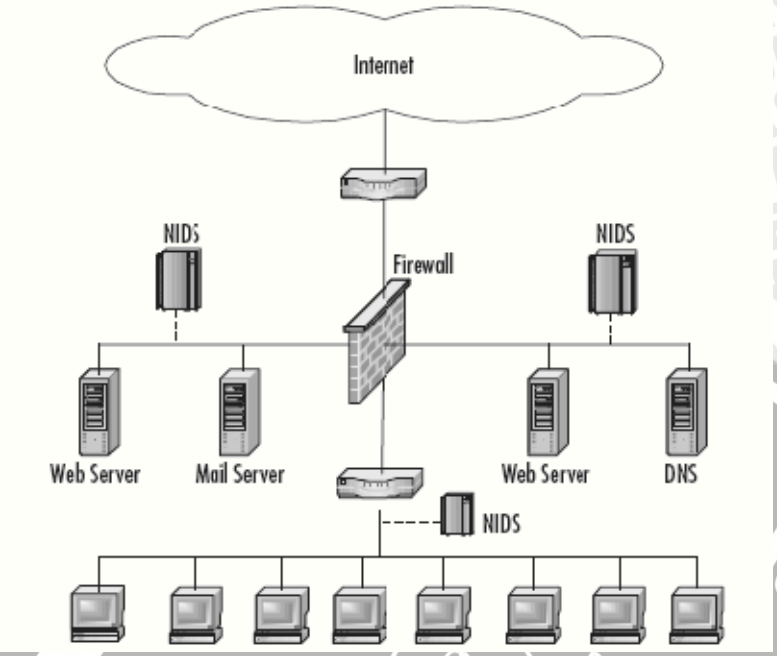
lingkungan sistem
operasi UNIX [ARI-07:40]. Berikut gambar yang mengilustrasikan Host-based IDS.



Gambar 2.18 Host Based IDS

2.6.2 Network Intrusion Detection System (NIDS)

Network-based IDS akan mengumpulkan paket-paket data yang terdapat pada jaringan dan kemudian menganalisanya serta menentukan apakah paket-paket itu berupa suatu paket yang normal atau suatu serangan atau berupa aktivitas yang mencurigakan [ARI-07:36]. Salah satu contoh aplikasi pada network-based adalah Snort [ARI-07:46]. Berikut gambar yang mengilustrasikan Network Based IDS



Gambar 2.19 Network Based IDS

Sedangkan dilihat dari cara kerja dalam menganalisa apakah paket data dianggap sebagai penyusup atau bukan, IDS dibagi dua, yaitu

2.6.3 Knowledge Based IDS

Knowledge-based IDS dapat mengenali adanya penyusupan dengan cara menyadap paket data kemudian membandingkannya dengan database rule IDS (berisi signature-signature paket serangan). Jika paket data mempunyai pola yang sama dengan (setidaknya) salah satu pola di database rule IDS, maka paket tersebut dianggap sebagai serangan, dan demikian juga sebaliknya, jika paket data tersebut sama sekali tidak mempunyai pola yang sama dengan pola di database rules IDS, maka paket data tersebut dianggap bukan serangan.

2.6.4 Behavior-based IDS

Behavior based (anomaly-based) dapat mendeteksi adanya penyusupan dengan mengamati adanya kejanggalan-kejanggalan pada sistem atau adanya penyimpangan-penyimpangan dari kondisi normal, sebagai contoh ada penggunaan memori yang melonjak secara terus menerus atau ada koneksi parallel dari 1 buah IP dalam jumlah banyak dan dalam waktu yang bersamaan. Kondisi-kondisi di atas dianggap kejanggalan yang kemudian oleh IDS jenis anomaly based dianggap sebagai serangan.

2.7 Snort

Snort merupakan suatu perangkat lunak untuk mendeteksi penyusup dan mampu menganalisis paket yang melintasi jaringan secara real time traffic dan logging ke dalam database serta mampu mendeteksi berbagai serangan yang berasal dari luar jaringan. Snort dapat digunakan pada platform sistem operasi Linux, BSD, Solaris, MacOS dan Windows [ARI-07:145]. Snort pertama kali dibuat oleh Marty Roesch dan dikembangkan oleh Sourcefire (www.sourcefire.com) [SET-06]

Snort bisa dioperasikan dengan tiga mode:

1.Paket sniffer: untuk melihat paket yang lewat di jaringan. Beberapa contoh perintah-nya terdapat di bawah ini:

./ snort -v

./ snort -vd

./ snort -vde

./ snort -v -d -e

Dengan menambahkan beberapa option –v, -d, -e, akan menghasilan beberapa keluaran yang berbeda, yaitu:

-v, untuk melihat header paket TCP/IP yang lewat

-d, untuk melihat isi paket

-e, untuk melihat header link layer paket seperti Ethernet header

2. paket Logger, untuk mencatat semua paket yang lewat di jaringan unuk dianalisis di kemudian hari. Beberapa perintah yang mungkin dapat digunakan untuk mencatat paket yang ada adalah:

./snort –dev –l ./log –b

./snort –dev –l ./log –h 192.168.0.0/24

./snort –dev –l ./log –b

perintah yang paling penting untuk me-log paket yang lewat adalah

-l ./log

yang menentukan bahwa paket yang lewat akan di log / di catat ke file ./log. Beberapa perintah tambahan dapat digunakan seperti –h 192.168.0.0/24 yang menunjukan bahwa yang di catat hanya packet dari host mana saja, dan –b yang memberitahukan agar file yang di log dalam format binary, bukan ASCII. Untuk membaca file log dapat dilakukan dengan menjalankan Snort dengan ditambahkan perintah –r nama file log-nya, seperti:

./snort –dv –r packet.log

./snort –dvr packet.log icmp

3. Network Intrusion Detection System (NIDS): pada mode ini Snort akan berfungsi untuk mendeteksi serangan yang dilakukan melalui jaringan komputer. Untuk menggunakan mode IDS ini diperlukan setup dari berbagai rules atau aturan yang akan membedakan sebuah paket normal dengan paket yang membawa serangan. Beberapa contoh perintah untuk mengaktifkan snort untuk melakukan pendeteksian penyusup, seperti

./snort –dev –l ./log –h 192.168.0.0/24 –c snort.conf

./snort –d –h 192.168.0.0/24 –l ./log –c snort.conf

2.8 Metoda Pengambilan data

Agar bisa mengidentifikasikan suatu data apakah data tersebut termasuk data yang diperlukan oleh user dari server tersebut ataukah data yang termasuk data yang tidak dibutuhkan dengan kata lain data serangan. Maka terlebih dahulu kita harus bisa mendapatkan keterangan-keterangan dari semua data yang masuk. Kemana tujuan dari data itu, darimana datangnya data itu ataupun berapa jumlah byte yang dibawa oleh data. Keterangan-keterangan yang kita butuhkan tersebut dapat kita peroleh langsung dari data itu.

Yang jadi permasalahan adalah "Bagaimana kita mendapatkan data yang terdapat pada paket tersebut dari jaringan yang ada ?". Semua komunikasi data yang terjadi di jaringan akan melewati router sebagai perantaranya. Baik data dari jaringan lokal yang akan keluar ke internet, maupun data yang datang dari luar jaringan lokal yang menuju ke lokal. Bisa di katakan router merupakan satu-satunya tempat yang memungkinkan kita untuk mendapatkan semua data paket yang ada. Pengambilan data paket yang akan berfungsi sebagai masukan untuk mengidentifikasikan paket bisa juga dilakukan di router tersebut.

Tetapi fungsi router sebagai pengatur bisa terganggu kalau harus dilakukan pengecekan yang rutin terhadap sejumlah database yang besar. Untuk itu di buat suatu komputer yang di tujukan untuk mengatur lalulintas keluar masuknya data di tempat lain.

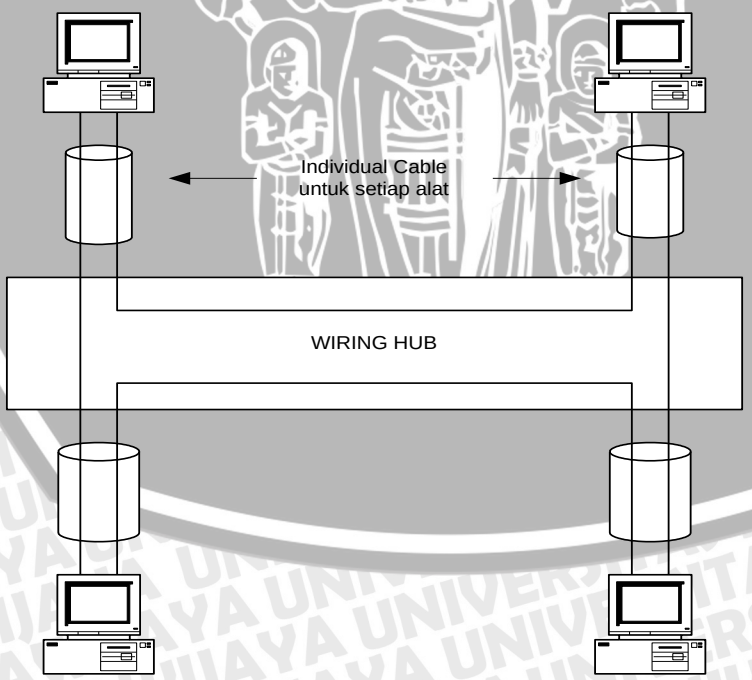
Untuk pengambilan data tersebut dilakukan oleh sniffer, Sniffer disini adalah suatu program yang dapat mengambil setiap paket yang masuk dan keluar didalam suatu jaringan. Sehingga setiap ada data yang melewati jaringan tersebut bisa terdeteksi dan bisa dilihat isi dari data tersebut, dengan cara mengidentifikasikannya sesuai dengan aturan yang ada disetiap protokol pembawanya (aturan tersebut sudah dijelaskan di sub bab terdahulu).

Tetapi proses sniffing ini mempunyai kelemahan bila ditempatkan di luar router. Kelemahannya ini bergantung dari penghubung yang ada didalam jaringan tersebut. Penghubung dalam suatu jaringan bisa dalam bentuk HUB atau SWITCH HUB, yang

mempunyai karakteristik masing-masing dalam menyalurkan data nya Berikut dijelaskan karakteristik masing-masing penghubung.

a. Karakteristik HUB

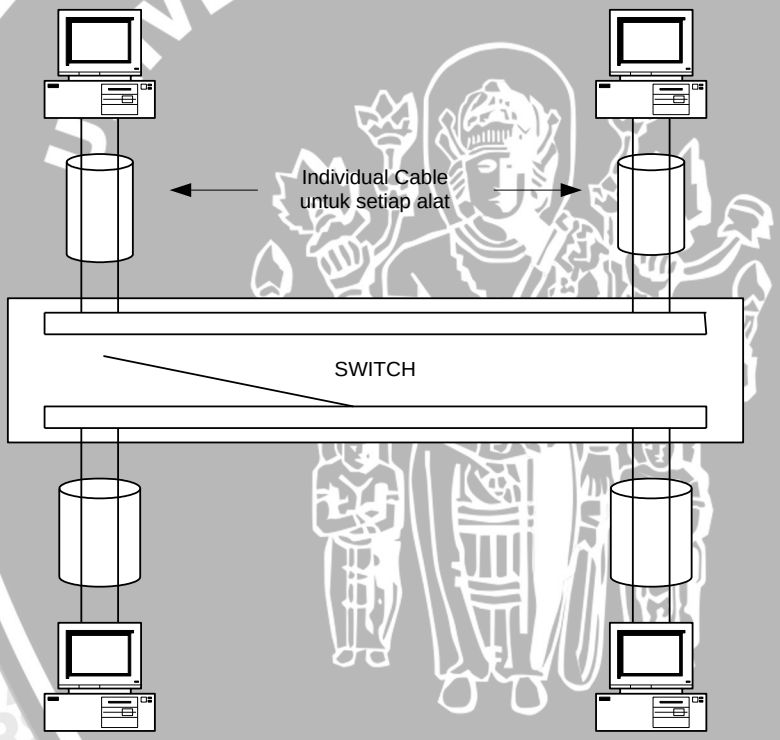
Hub adalah suatu penghubung dalam jaringan sehingga suatu jaringan bisa memiliki user lebih dari satu. Prinsip dasar dari dari HUB adalah token ring, yaitu suatu konfigurasi hubungan internet yang pada mulanya digunakan untuk mengelakkan kekurangan dari CSMA, dengan menggunakan token ring maka secara periodik setiap titik dalam jaringan dapat tersambungkan. Meskipun demikian karena jalur yang digunakan adalah satu maka kesempatan yang di dapat juga probabilitas, dengan artian user dapat mengirim atau menerima data jika memang pada saat pengiriman jalur kosong. Tetapi jika tidak maka harus menunggu untuk mengirim sampai jalur kosong. Sistem penyaluran data nya adalah broadcast, jadi setiap komputer sebenarnya bisa mengetahui aliran data tersebut.



Gambar 2.20 Jaringan pada Hub

b. Karakteristik SWITCH

Seperti halnya Hub, switch juga merupakan penghubung dalam suatu jaringan. Tetapi dasar yang dipakai adalah switching, sehingga memungkinkan alur dari lalu lintas data lebih banyak option-nya. Pada waktu sibuk user tidak terlalu lama untuk menungu giliran mengirimkan data karena jalan yang dipakai banyak (tidak cuma satu). Metode pengerimannya adalah point-to-point, dianggap jika ada yang datang langsung terhubung satu sama lain secara kontinyu.



Gambar 2.21 Jaringan pada Switch

Dengan demikian dapat dijelaskan apabila program sniffer diletakkan diluar router maka penghubung yang digunakan haruslah Hub. Agar paket yang datang dan pergi dalam jaringan dapat diambil semuanya, tidak hanya paket yang menuju ke komputer dimana program itu berada.

2.7 Metode Pemblokiran IP

Pemblokiran IP tersebut disesuaikan dengan operating system yang ada di router, apakah Linux, Windows 2000, ataukah FreeBSD

a. WINDOWS 2000

Di dalam windows 2000 server telah dilengkapi dengan cara untuk mengatur IP baik itu mengeblok IP maupun melewatkan suatu IP. Program tersebut adalah IPSECPOL. Utility ini hampir sama kegunaannya pada iptables dan ipchains dalam program LINUX. Hanya saja untuk utility ini hanya bekerja pada windows 2000 server.

Pengaturan IPSECPOL pada tampilan windows dapat dijumpai pada "IP Security Policies on Local Machine" yang berada pada "Computer Configuration Security Settings" di MMC (Microsoft Management Console).

Yang pada defaultnya terdapat 3 ketentuan yang telah ditetapkan :

a. **Client (Respond Only)**

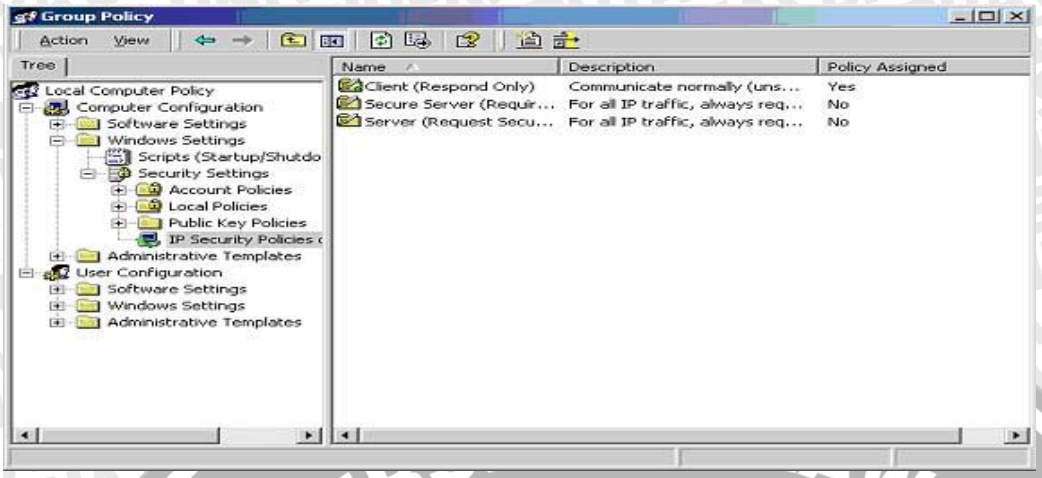
Digunakan oleh client untuk memberikan respon kepada windows 2000 server pada saat ada permintaan menggunakan servis yang ada didalamnya.

b. **Secure Server (Require Security)**

Ketentuan ini digunakan pada windows 2000 server dan windows 2000 *host* yang menghasilkan *network-based services* untuk meyakin kan bawa tidak ada *non-authentication* dan *non-encryption traffic* yang di abaikan.

c. **Server (Request Security)**

Ketentuan ini hampir sama dengan ketentuan yang ada pada *Secure Server*, yang menjadi perbedaan adalah pada ketentuan ini terdapat ketentuan untuk mengadakan hubungan enkripsi pada tingkat lebih tinggi di user.



Gambar 2.22 IPSECPOL pada tampilan windows

b. Linux

Untuk sistem pemblokiran dengan menggunakan operating system ini dengan menggunakan aplikasi yang sudah tersedia yaitu dengan menggunakan IPTABLES atau IPCHAINS tergantung versi yang digunakan. Pada aplikasi ini tersedia berbagai fungsi tentang routing baik forwarding, accepting ataupun blocking.

c. FreeBSD

FreeBSD juga mempunyai aplikasi untuk pengaturan routing yang fungsinya mirip dengan IPTABLES pada linux ataupun IPSECPOL pada windows, hanya saja pada sistem FreeBSD untuk pengaturannya menggunakan perintah IPFW.

BAB III
METODE PENELITIAN

Tahap ini akan menjelaskan langkah-langkah yang akan dilakukan untuk merancang sistem yang akan dibuat hingga dapat berfungsi sebagaimana yang diharapkan. Langkah-langkah tersebut adalah sebagai berikut:

3.1 Studi Literatur

Studi literature yang dilakukan bertujuan untuk mengkaji hal-hal yang berhubungan dengan teori-teori yang mendukung dalam perencanaan dan perancangan sistem, yaitu:

1. Kajian pustaka mengenai protocol TCP/IP, yaitu suatu protocol yang digunakan oleh suatu komputer dalam suatu jaringan computer agar dapat saling berkomunikasi dengan komputer lain.
2. Kajian pustaka mengenai topologi jaringan computer, yaitu suatu cara untuk menyusun komputer-komputer secara fisik sehingga membentuk sebuah jaringan komputer.
3. Kajian pustaka mengenai Snort, yaitu suatu program untuk mendeteksi adanya kejanggalan di suatu jaringan yang terjadi sesuai dengan tipe data yang ada berdasarkan *rules*.
4. Kajian pustaka mengenai program *Firewall* bawaan dari sistem operasi Windows yang digunakan dalam perancangan sistem, yang berfungsi untuk melakukan blocking IP suatu paket data.

3.2 Perancangan dan Implementasi Sistem

Pada tahap ini,sistem untuk pencegahan flooding data pada jaringan, menggunakan program Snort dan Firewall bawaan dari Sistem Operasi Windows, dan akan dirancang sebagaimana berikut:

1. Perancangan topologi jaringan
2. Pemasangan sistem operasi pada komputer di jaringan komputer.
3. Pemasangan (install) Snort pada komputer.
4. Pengkonfigurasian Snort dan Firewall bawaan sistem operasi pada computer sehingga dapat bekerja sesuai dengan yang diharapkan.

3.3 Pengujian dan Analisis Sistem

Pada tahap pengujian dan analisis sistem ini, perancangan Snort dan Firewall yang telah dibangun untuk pencegahan flooding data dan blocking IP otomatis, akan diuji dengan beberapa cara. Salah satunya adalah dengan cara port scanning dan fingerprinting dengan menggunakan Nmap. Hasil dari pengujian akan dianalisis, apakah sistem yang dibangun telah memenuhi harapan sesuai dengan fungsinya atau belum.

3.4 Pengambilan Kesimpulan dan Saran

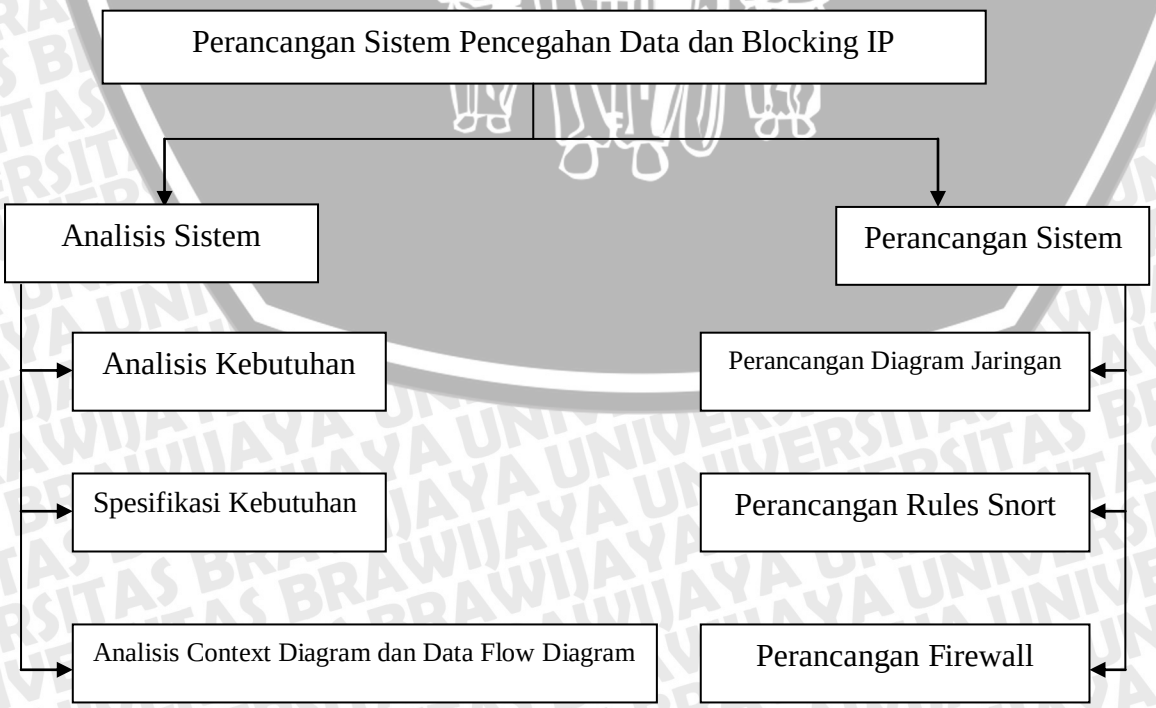
Tahap ini adalah tahap pengambilan kesimpulan dari sistem yang telah dibuat. Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan dan pengujian sistem telah selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktek. Kesimpulan ini merupakan informasi akhir dari perancangan sistem yang berisi mengenai berhasil atau tidaknya sistem tersebut dijalankan.

Tahap terakhir dari penulisan adalah saran yang dmaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi serta menyempurnakan penulisan.

BAB IV
ANALISIS DAN PERANCANGAN

Pada bab ini dibahas tentang bagaimana proses yang digunakan untuk merancang dan mengimplementasikan sistem pengamanan server yang akan dibuat. Pembahasan yang akan dilakukan meliputi spesifikasi dan mekanisme kerja program yang diinginkan. Setelah itu berlanjut ke pembahasan mengenai alat bantu yang akan digunakan. Setelah masing-masing alat bantu itu di jabarkan, pembahasan menyentuh hal-hal yang lebih spesifik dari sistem, yaitu bagaimana data diambil dan bagaimana pengolahan data-data tersebut. Kemudian akan dijabarkan secara lebih jelas mengenai desain sistem secara keseluruhan. Proses implementasinya dijelaskan melalui algoritma dari program-program yang akan di buat.

Berikut ini dijelaskan dalam diagram pohon perancangan yang akan dilakukan, dengan software utama menggunakan Snort yang berbasis open source.



Gambar 4.1 Diagram Pohon Perancangan

sumber: [perancangan]

Perancangan sistem pencegahan pada jaringan akan dibagi dalam dua tahap, yaitu analisis sistem dan perancangan sistem. Analisis sistem yang dilakukan meliputi beberapa tahap, yaitu analisis kebutuhan, spesifikasi kebutuhan, dan analisis context diagram dan data flow diagram.

Sedangkan perancangan sistem yang dilakukan dibagi menjadi beberapa tahap, yaitu perancangan diagram jaringan komputer, perancangan rules Snort dan perancangan firewall.

4.1 Analisis Sistem

Analisis sistem adalah proses yang menggunakan prinsip-prinsip sistem untuk mengidentifikasi, merekontruksi, mengoptimalkan, dan mengontrol sebuah sistem. Proses analisis sistem pencegahan flooding data ini meliputi analisis kebutuhan, spesifikasi kebutuhan, dan pemodelan analisis sistem. Pemodelan analisis sistem dilakukan dengan membuat Context Diagram dan Data Flow Diagram (DFD).

4.1.1 Analisis Kebutuhan

Analisis kebutuhan sistem pencegahan penyusupan pada jaringan menggunakan pendekatan analisis kebutuhan perangkat lunak yaitu proses yang digunakan untuk mendapatkan, menganalisis, dan memvalidasi kebutuhan-kebutuhan sistem. Proses analisis kebutuhan sistem pencegahan flooding data pada jaringan meliputi definisi konseptual dan penentuan kebutuhan fungsional.

4.1.1.1 Definisi Konseptual

Definisi konseptual merupakan ruang lingkup proses security atau batasan dari sistem yang harus dilindungi. Definisi konseptual memiliki peranan yang penting untuk membantu mendefinisikan tangggung jawab dan ruang lingkup -individu yang terlibat dalam proses security. Definisi konseptual pencegahan penyusupan pada jaringan adalah melindungi network internal dari serangan-serangan jaringan yang berasal dari network eksternal dan internet.

4.1.1.2 Penentuan kebutuhan Fungsional

Definisi kebutuhan fungsional merupakan analisis detail tentang apa yang diperlukan untuk mengurangi resiko yang diidentifikasikan dengan membandingkan status sistem jaringan komputer yang aktual dengan status sistem jaringan komputer yang dijadikan sebagai objek. Definisi kebutuhan dibagi menjadi beberapa aktifitas yaitu review internal, review eksternal, dan analisis resiko. Produk akhir dari penentuan kebutuhan adalah sebuah definisi kebutuhan.

Definisi Kebutuhan Fungsional

- Komputer router pada network internal memiliki Snort dan Iptables firewall sebagai lapisan pelindung tambahan
- Snort pada komputer router dapat memblokir TCP SYN flood, UDP Flood.
- Iptables firewall pada komputer router dapat memblokir TCP SYN flood, UDP Flood.
- Paket IP yang diloloskan oleh komputer router adalah paket IP yang dibutuhkan oleh client. Paket data yang tidak dibutuhkan akan didrop oleh firewall.
- Software pendeteksi penyusup pada jaringan yang digunakan adalah Snort.
- Software firewall yang digunakan adalah Iptables pada sistem operasi Linux

- Spesifikasi hardware router yang digunakan merupakan spesifikasi hardware komputer router yang telah memenuhi minimum hardware requirement untuk software firewall.

4.1.2 Spesifikasi Kebutuhan

Spesifikasi kebutuhan dibutuhkan untuk menjelaskan kebutuhan sistem pencegahan Flooding Data pada jaringan yang telah didefinisikan pada Sub Bab 4.1 secara lebih detail dan tepat yang akan menjadi dasar bagi perancangan dan implementasi.

Definisi:

1. Komputer Server

Spesifikasi:

1.1 Komputer router ditempatkan pada jaringan internal

1.2 Sistem operasi yang digunakan dalam computer server adalah Linux distribusi Fedora Core versi 9

Definisi:

2. Sistem pencegahan flooding data pada jaringan terdiri dari sebuah software pendeteksi flooding data, yaitu Snort, kemudian BlockIt sebagai penghubung Snort dengan IPS, dan firewall dalam sistem operasi, yaitu IPtables pada Linux.

Spesifikasi:

2.1 Snort ditempatkan pada komputer host/client

2.2 BlockIt ditempatkan pada komputer host/client

2.3 IPtables terinstall pada komputer host/client

Definisi:

3. Software IDS (Intrusion Detection System) yang digunakan adalah Snort

Spesifikasi:

3.1 Snort yang digunakan adalah Snort versi 2.7.0

3.2 Rules Snort yang digunakan adalah snortrules-snapshot-2.6

Definisi:

4. BlockIt adalah program yang menghubungkan Snort dengan IPtables

Spesifikasi:

4.1 BlockIt yang digunakan adalah versi 1.4.2

Definisi:

5. Software firewall yang digunakan adalah Iptables pada sistem operasi Linux

Spesifikasi:

5.1 Program firewall yang digunakan adalah Iptables versi 1.4

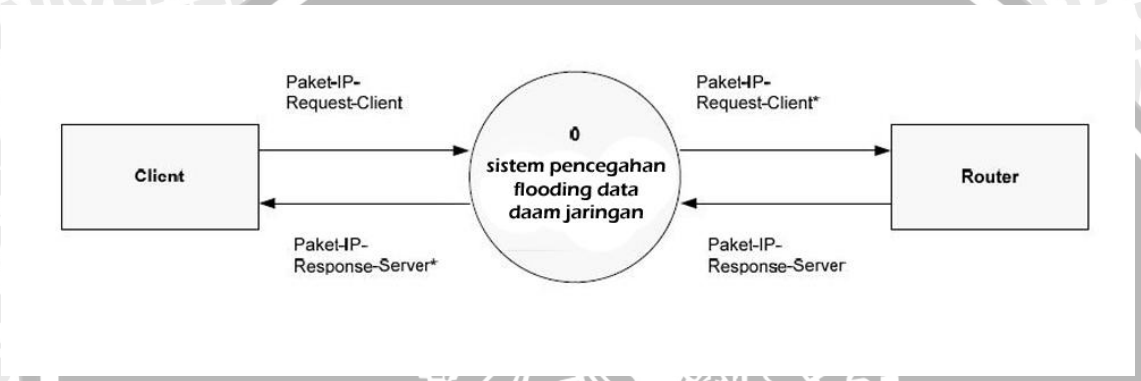
4.1.3 Analisis Context Diagram dan Data Flow Diagram (DFD)

Context diagram atau diagram konteks merupakan diagram yang menampilkan masukan proses, proses dan keluaran proses dari sistem secara umum. Diagram Konteks adalah DFD yang pertama kali dibuat dikenal dengan DFD level 0.

sistem pencegahan Flooding Data pada jaringan memiliki beberapa tipe aliran data pada sistem yaitu berupa paket IP request dan paket IP response.

4.1.3.1 DFD Level 0 Client ke Router

DFD level 0 client ke server merupakan diagram yang menampilkan masukan proses, proses, dan keluaran proses secara umum antara client yang berada di network eksternal, Sistem Pencegahan Flooding Data pada Jaringan, dan router yang berada di network.



Gambar 4.2 DFD Level 0

Sumber: [Perancangan]

Berdasarkan gambar, Sistem Pencegahan Flooding Data mempunyai masukan:

- Paket-IP-Request-Client

Paket-IP-Request-Client merupakan aliran data paket IP berisi request aplikasi tertentu dari client ke server

- Paket-IP-Response-server

Paket-IP-Response- server merupakan aliran data paket IP berisi response aplikasi tertentu dari server

Berdasarkan gambar, Sistem Pencegahan Flooding Data mempunyai keluaran:

- Paket-IP-Request-Client

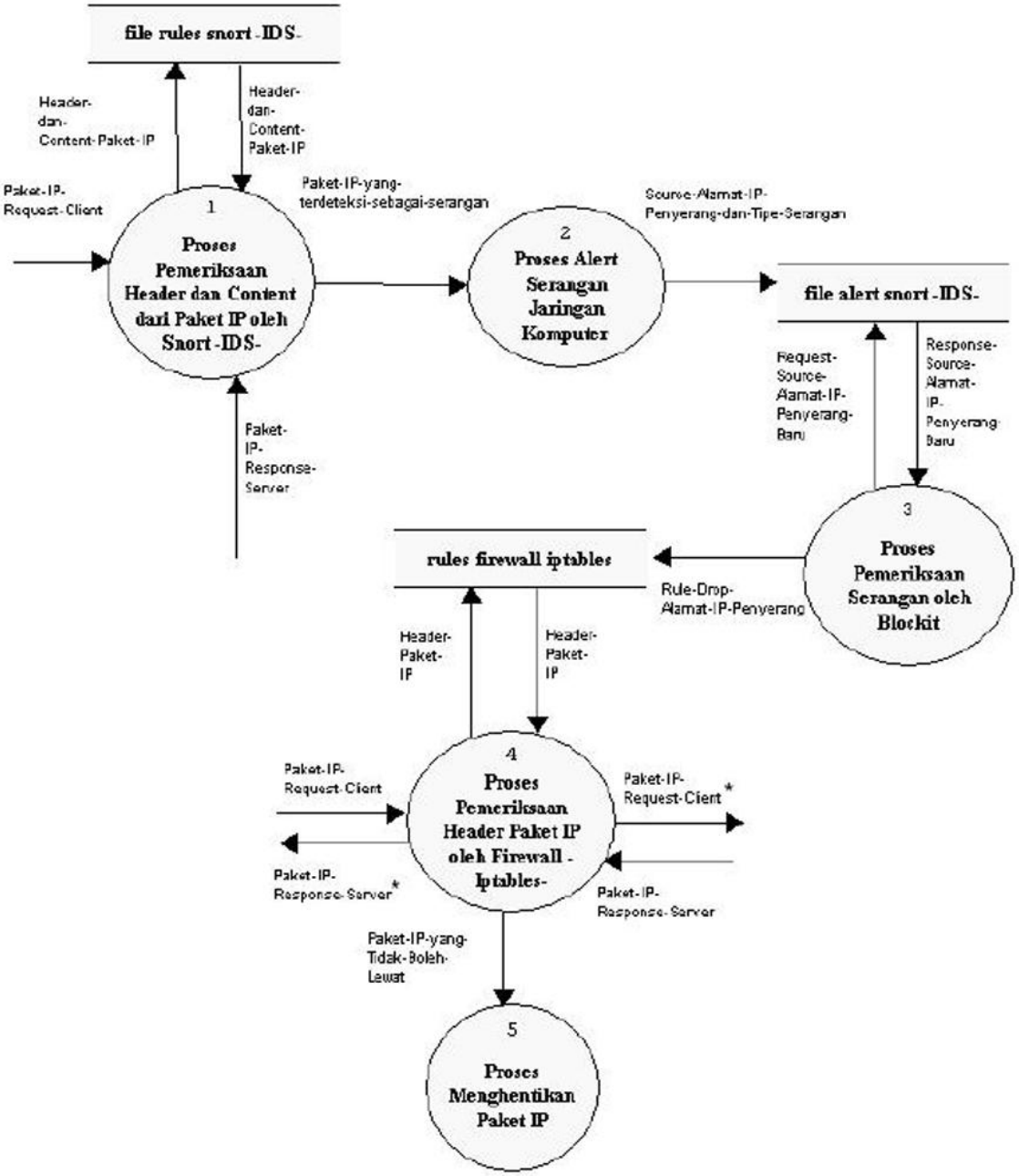
Paket-IP-Request-Client merupakan aliran data paket IP berisi request aplikasi tertentu dari client

- Paket-IP-Response- server

Paket-IP-Response- server merupakan aliran data paket IP berisi response aplikasi tertentu dari server ke client

4.1.3.2 DFD Level 1 Client ke server

DFD level 1 merupakan penjabaran dari DFD level 0, seperti yang ditunjukkan pada gambar berikut



Gambar 4.3 DFD level 1 Client ke Server

Sumber: [perancangan]

Penjabaran proses-proses tersebut antara lain:

1. Proses Penyaringan Header dan Content dari Paket IP oleh Snort

Proses ini terletak pada proses program Snort. Pada proses ini header paket IP yang masuk ke komputer router akan diperiksa apakah diperbolehkan masuk atau tidak berdasarkan file rules Snort.

2. Proses Alert Serangan Jaringan Komputer

Proses ini menjawab request Paket IP yang terdeteksi sebagai serangan. Jawaban ini berupa response berupa sumber alamat IP penyerang dan tipe serangan.

3. Proses Pemeriksaan Serangan Oleh Blockit

Proses ini terletak pada proses program Blockit. Pada proses ini sumber alamat IP penyerang yang terdeteksi oleh Snort akan diperiksa dan Blockit membuat rules DROP alamat IP penyerang tersebut pada Iptables.

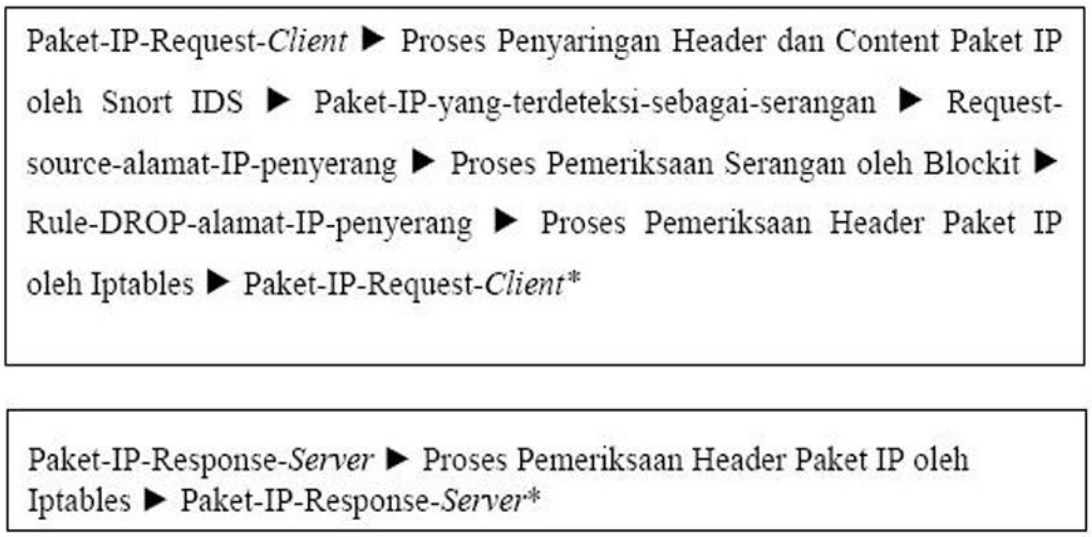
4. Proses Pemeriksaan Header Paket IP oleh Iptables firewall

Proses ini terletak pada Iptables. Pada proses ini header paket IP yang berasal dari alamat IP penyerang akan diperiksa apakah diperbolehkan keluar atau tidak berdasarkan rules tabel filter rantai OUTPUT

5. Proses Menghentikan IP

Proses ini akan melakukan penghentian paket IP yang tidak boleh masuk ke komputer client sesuai dengan rules Iptables.

DFD level 1 Client ke Router memiliki 2 jenis aliran data paket IP yaitu paket IP request client dan paket IP response server. Urutan paket IP ini :



Gambar 4.3 Urutan Paket IP

Sumber: [perancangan]

Keterangan:

* menandakan paket IP ini tidak diblok/dihentikan berdasarkan rules.

4.2 Perancangan Sistem

4.2.1 Spesifikasi Sistem

Sebelum melakukan proses pembuatan sistem, terlebih dahulu ditentukan spesifikasi sistem. Spesifikasi sistem akan menjadi titik tolak sekaligus menjadi acuan untuk pembuatan sistem dan juga menentukan kapabilitas dan kemampuan apa saja yang harus bisa dipenuhi sistem yang dimaksud.

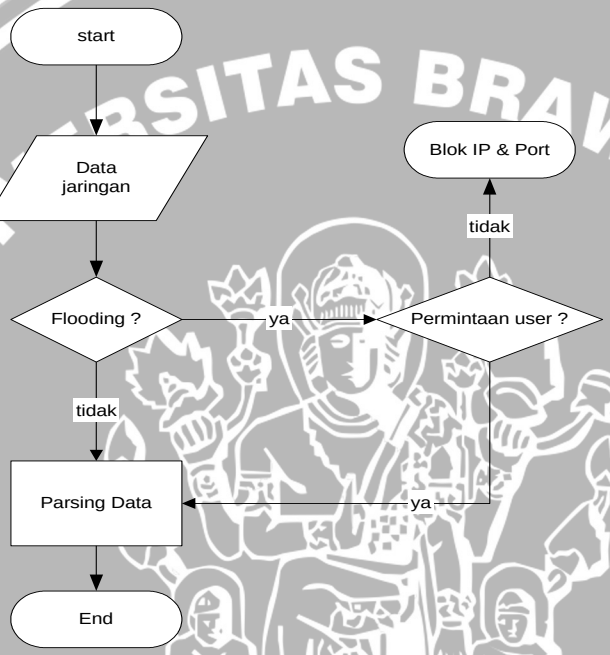
Sistem yang dibangun memiliki spesifikasi sebagai berikut:

1. Sistem beroperasi pada platform Linux
2. Resource yang digunakan harus seminimal mungkin
3. Sistem harus bersifat *multiuser* dan *multitasking*. Dikembangkan dengan alat bantu yang mudah dan/atau gratis (open source).

4.2.2 Perancangan Diagram Sistem

4.2.2.1 Desain Sistem Secara Umum

Secara umum sistem yang akan dibangun adalah sebagai berikut :



Gambar 4.4 Desain umum program blokir otomatis pada flood

Sumber: [perancangan]

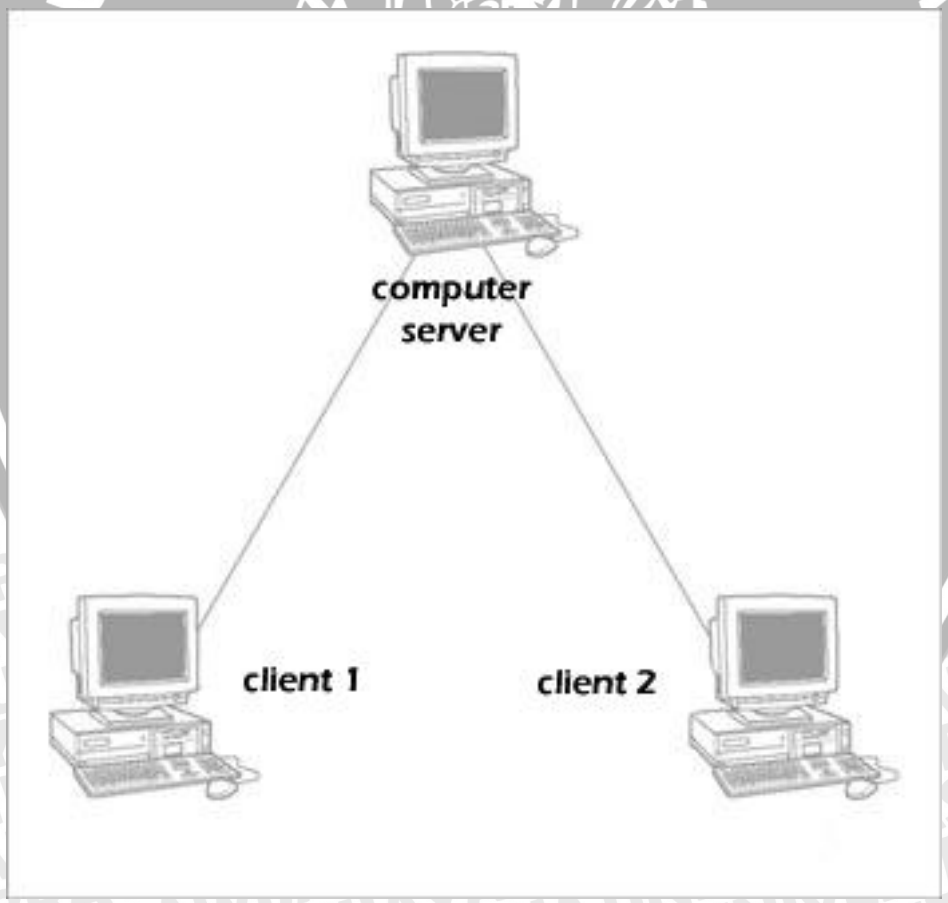
Keterangan gambar:

Input dari program adalah data jaringan yang masuk kemudian akan di proses apakah data yang ada tersebut melakukan flooding atau tidak. Jika data yang datang adalah flooding maka computer akan mencari apakah data merupakan permintaan user atau tidak. Jika terbukti tidak maka secara otomatis akan memblok ip dan port darimana data itu berasal dan kalau ya berarti data akan ditujukan kepada tujuanya.

4.2.3 Perancangan Diagram Jaringan

Diagram jaringan yang akan dibangun pada sistem ini adalah diagram jaringan sederhana, yang menggunakan basis topologi star. Topologi star menghubungkan setiap komputer langsung ke pusat sehingga tidak perlu berhubungan dengan komputer lain. Keistimewaan topologi ini adalah cepat berhubungan dengan pusat (server), sehingga kontrol data dapat dilakukan dengan cepat karena data terpusat di server. Tetapi, jika server mengalami kerusakan atau gangguan, maka semua komputer yang terhubung tidak dapat beroperasi.[FLY - 09]

Berikut diagram jaringan komputer yang dibangun dalam sistem, seperti pada gambar.

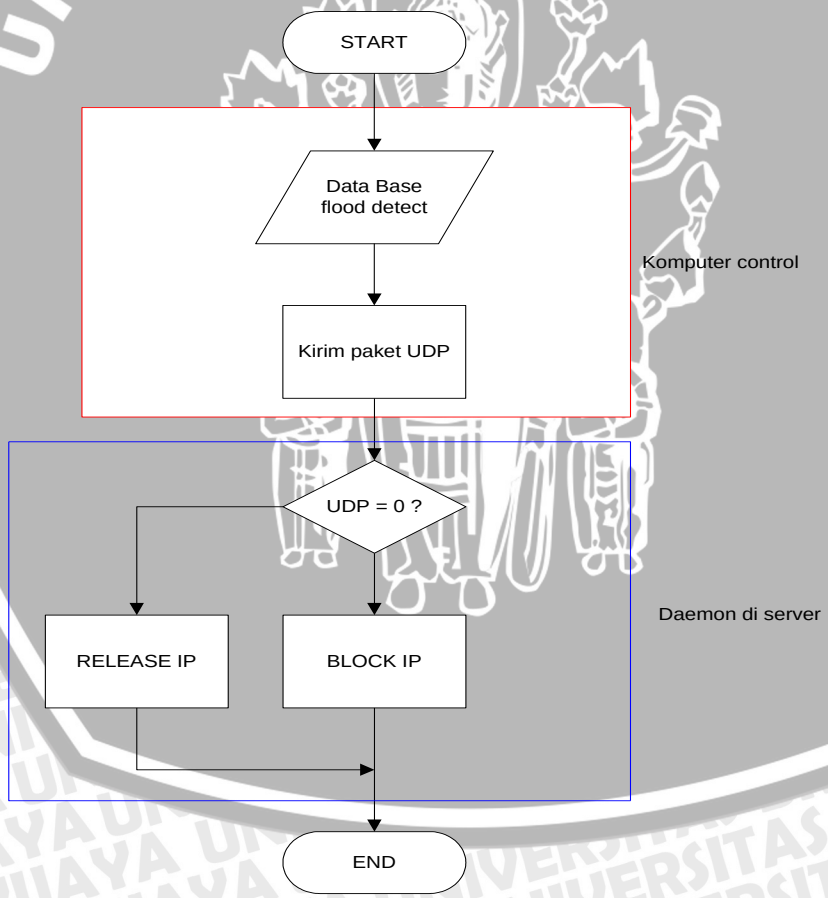


Gambar 4.5 desain perancangan jaringan sistem

Sumber: [perancangan]

4.2.4 Desain Pemblokiran IP

Setelah data terbukti melakukan flooding pada jaringan maka sistem akan mengirimpkan paket UDP ke server untuk mengirimkan perintah blooking kepada IP yang besangkutan. Sebelumnya program daemon sudah diletakkan didalam server terlebih dahulu dan dijalankannya,untuk program daemon akan ditanyakan pakah paket UDP sama dengan nol,jika sama maka data akan ditujukan ketujuanya,sebaliknya jika tidak maka data akan diblok.

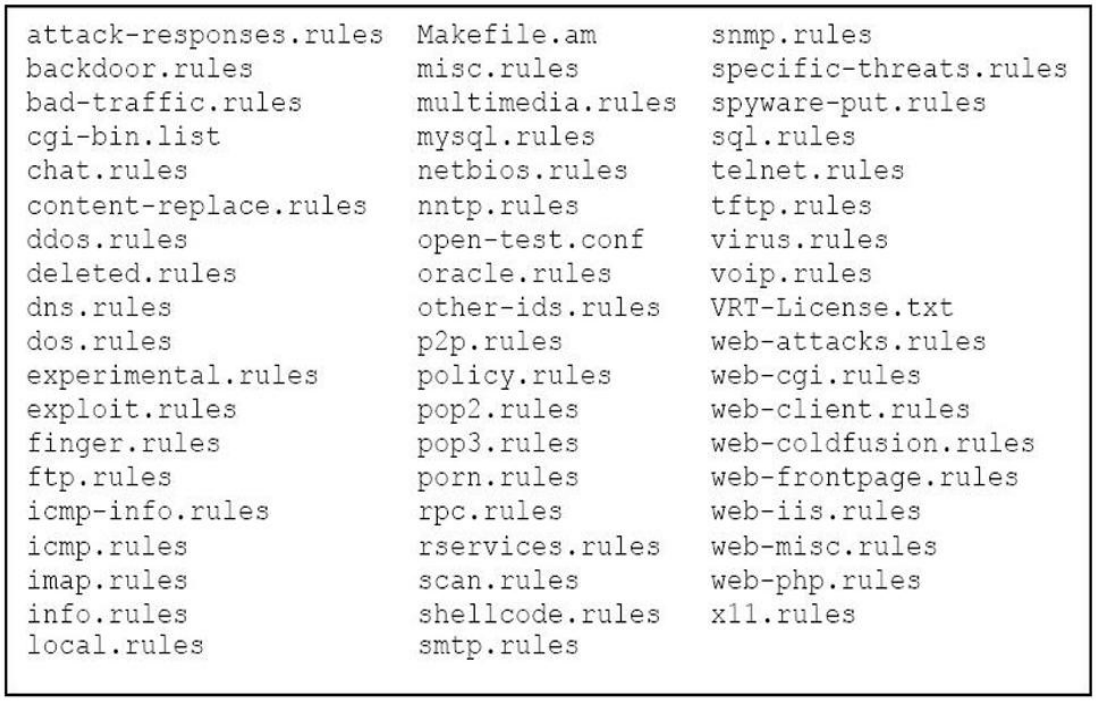


Gambar 4.6 Desain blokir IP

Sumber: [perancangan]

4.2.5 PerancanganRules Snort

Rules Snort merupakan database yang berisi pola-pola serangan berupa signature jenis-jenis serangan. Rules Snort IDS ini, harus diupdate secara rutin agar ketika ada suatu teknik serangan yang baru, serangan tersebut dapat terdeteksi [HEL - 09]. Rules Snort dapat di download di http://www.snort.org. Berikut beberapa rules tersebut:



```
attack-responses.rules   Makefile.am        snmp.rules
backdoor.rules           misc.rules         specific-threats.rules
bad-traffic.rules        multimedia.rules   spyware-put.rules
cgi-bin.list             mysql.rules        sql.rules
chat.rules               netbios.rules      telnet.rules
content-replace.rules    nntp.rules         tftp.rules
ddos.rules               open-test.conf     virus.rules
deleted.rules            oracle.rules       voip.rules
dns.rules                other-ids.rules    VRT-License.txt
dos.rules                p2p.rules          web-attacks.rules
experimental.rules       policy.rules       web-cgi.rules
exploit.rules            pop2.rules         web-client.rules
finger.rules             pop3.rules         web-coldfusion.rules
ftp.rules                porn.rules         web-frontpage.rules
icmp-info.rules          rpc.rules          web-iis.rules
icmp.rules               rservices.rules    web-misc.rules
imap.rules               scan.rules         web-php.rules
info.rules               shellcode.rules    x11.rules
local.rules              smtp.rules
```

Gambar 4.7 Rules Snort

Sumber:[ORG]

4.2.6 Perancangan Firewall

4.2.6.1 BlockIt

Program BlockIt digunakan untuk menghubungkan antara Snort dengan Iptables sehingga jika ada paket data yang dicurigai oleh Snort, maka BlockIt akan membaca log dari Snort yang berada di file "/var/log/snort/alert" dan akan memberitahu Iptables untuk memblok paket data tersebut.

4.2.6.2 IPtables

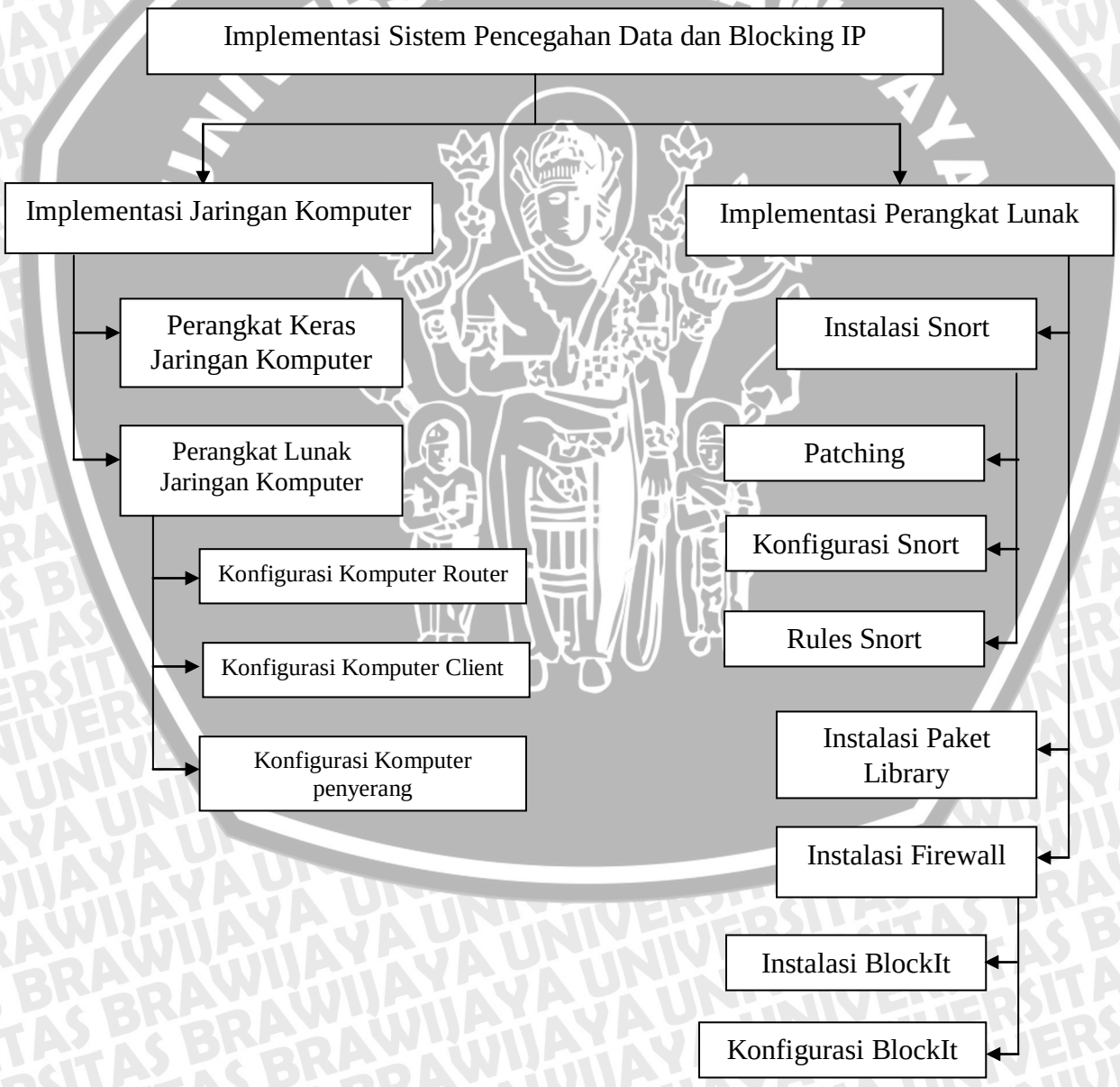
Dalam sistem yang akan dibangun ini, yang digunakan sebagai firewall adalah Iptables dalam sistem operasi Linux. Aturan IPtables mengacu pada spesifikasi kebutuhan seperti yang telah disebutkan sebelumnya. Setelah program BlockIt terinstall, maka secara otomatis akan ada penambahan *chain* BlockIt pada Iptables. Default aturannya diberlakukan jika paket data yang melewati Snort sesuai dengan salah satu rules Snort. Default aturan dari Iptables memiliki *policy* DROP, sehingga jika paket IP yang melewati Snort cocok dengan salah satu rules maka paket IP tersebut akan di DROP oleh Iptables.

DROP sendiri adalah perintah yang berfungsi untuk membuang paket data. Paket data yang dibuang tersebut akan hilang tanpa meninggalkan tanda-tanda atau peringatan apapun. Sistem pengirim paket data hanya akan mendapat pesan request time out yang dapat diterjemahkan menjadi banyak hal.

BAB V

IMPLEMENTASI

Bab ini menjelaskan tentang implementasi Sistem Pencegahan Penyusupan pada Jaringan dengan menggunakan Snort dan Iptables firewall. Implementasi yang dilakukan dapat digambarkan dengan diagram pohon seperti dalam Gambar 5.1 berikut:



Gambar 5.1 Implementasi Sistem Pencegahan Flooding Data

Implementasi ini memiliki dua tahap utama yang dilakukan, yaitu implementasi jaringan komputer dan implementasi perangkat lunak.

5.1 Implementasi Jaringan Komputer

Implementasi ini terkait dengan network internal. Pada tahapan ini, rancangan mengenai jaringan komputer yang telah dibuat pada bab IV akan diwujudkan. Jaringan komputer terdiri dari dua bagian, yaitu perangkat keras dan perangkat lunak.

5.1.1 Perangkat Keras Jaringan Komputer

Pada bagian perangkat keras terdiri dari tiga buah komputer. Satu buah komputer digunakan sebagai router, 1 buah komputer digunakan sebagai client di network internal dan 1 buah komputer digunakan sebagai komputer penyerang di network eksternal.

- Komputer yang akan berfungsi sebagai router memiliki spesifikasi sebagai berikut:

  Intel Celeron M processor 440 1,86 GHz

  80 GB HDD

  512 MB DDR2

  LAN CARD Realtek RTL-8139C

- Komputer yang akan berfungsi sebagai client memiliki spesifikasi sebagai berikut:

  Intel Pentium 4 processor 440 1,86 GHz

  80 GB HDD

  256 MB DDR

  LAN CARD Realtek RTL-8139C

- Komputer yang akan berfungsi sebagai komputer penyerang memiliki spesifikasi sebagai berikut:

Intel Celeron M 1,73 GHz

80 GB HDD

512 MB DDR

LAN CARD VIA Rhine II Fast Ethernet Adapter

Perangkat keras lainnya adalah sebagai berikut:

- Hub : TP-LINK dengan Port 16 buah
- Kabel UTP : kabel BELDEN CDT NETWORKING
- Konektor : RJ-45 AMP buatan Tyco Corporation

Perangkat keras tersebut kemudian disusun sehingga membentuk sebuah jaringan dengan Topologi Star seperti yang diperlihatkan dalam bab IV.

5.1.2 Perangkat Lunak Jaringan Komputer

Pada tahap ini, perangkat keras yang telah disusun, dikonfigurasi perangkat lunaknya. Perangkat lunak yang dikonfigurasi adalah komputer jaringan pada komputer router dan komputer client.

5.1.2.1 Konfigurasi Komputer Router

Konfigurasi pada komputer router dilakukan setelah Sistem Operasi Fedora Core 9 terinstall. Semua konfigurasi yang berupa perintah-perintah linux, dilakukan pada Terminal-Command Line.

Konfigurasi yang perlu dilakukan agar komputer router dapat berfungsi dengan baik adalah sebagai berikut:

a. Mengaktifkan fungsi untuk meneruskan paket data

   Konfigurasi ini dilakukan dengan melakukan perubahan variabel dari 0 menjadi 1 pada parameter `net.ipv4.ip_forward` yang terdapat dalam file `sysctl.conf`

```
[root@ips-jar ~] vi
/etc/sysctl.conf
```

```
# Controls IP packet forwarding
Net.ipv4.ip_forward=1
```

b. Memasang default gateway

Konfigurasi ini dilakukan dengan menambahkan baris gateway=172.17.8.1 dalam file network

```
[root@ips-jar ~] vi
/etc/sysconfig/network
```

```
NETWORKING=yes
HOSTNAME=ips-jar.brawijaya.ac.id
GATEWAY=172.17.8.1
GATEWAYDEV=eth0
```

c. Memasang DNS

Konfigurasi ini dilakukan dengan menambahkan baris `nameserver 202.162.208.99` dan `nameserver 202.162.208.100` dalam file `resolv.conf`

```
[root@ips-jar ~] vi
/etc/resolv.conf
```

```
nameserver 202.162.208.99
nameserver 202.162.208.100
```

d. Konfigurasi eth0 sebagai uplink

Konfigurasi ini dilakukan dengan melakukan perubahan pada file `ifcfg-eth0` seperti di bawah ini :

```
[root@ips-jar ~] vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
DEVICE=eth0
BOOTPROTO=static
BROADCAST=172.17.8.127
HWADDR= 00:90:08:A3:F8:94
IPADDR=172.17.8.12
NETMASK=255.255.255.128
NETWORK=172.17.8.0
ONBOOT=yes
DNS1=202.162.208.99
DNS2=202.162.208.100
```

e. Konfigurasi eth1 sebagai downlink

Konfigurasi ini dilakukan dengan melakukan perubahan pada file `ifcfg-eth1` seperti di bawah ini :

```
[root@ips-jar ~] vi /etc/sysconfig/network-scripts/ifcfg-eth1
```

```
DEVICE=eth1
BOOTPROTO=static
BROADCAST=168.18.18.127
HWADDR=00:14:2A:80:E0:FB
IPADDR=168.18.18.1
NETMASK=255.255.255.128
NETWORK=168.18.18.0
ONBOOT=yes
DNS1=202.162.208.99
DNS2=202.162.208.100
```

f. Konfigurasi service network

Konfigurasi ini dilakukan agar setiap komputer router melakukan booting langsung menjalankan service network untuk eth0 dan eth1. Konfigurasi ini dilakukan pada file `rc.local` dengan menambahkan perintah `SERVICE NETWORK RESTART`

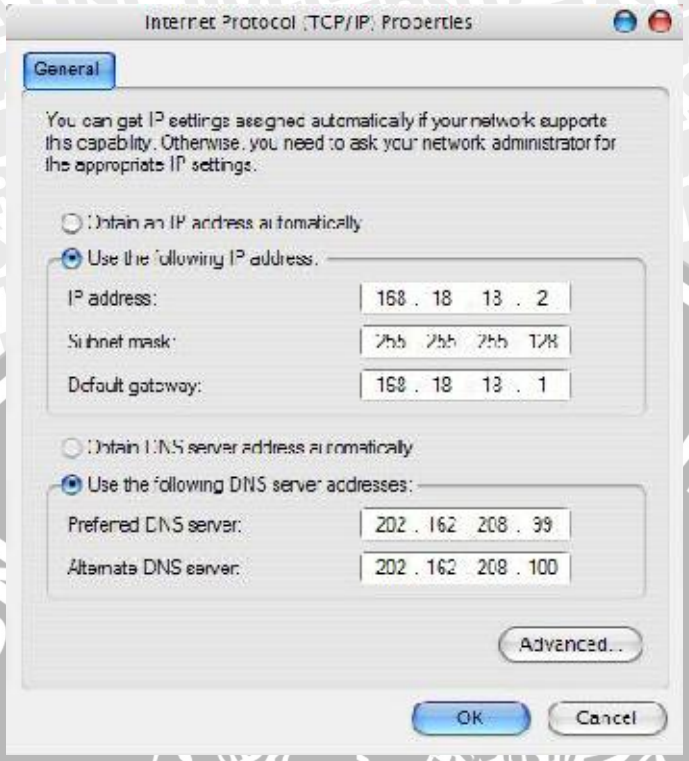
```
[root@ips-jar ~] vi /etc/rc.lokal
```

```
#!/bin/sh
/etc/init.d/network restart
```

5.1.2.2 Konfigurasi TCP/IP pada Komputer Client

Konfigurasi TCP/IP pada komputer client dilakukan agar komputer tersebut dapat berkomunikasi dengan komputer gateway, network eksternal maupun Internet.

Konfigurasi alamat IP komputer client dilakukan dengan konfigurasi alamat IP secara manual. Konfigurasi alamat IP secara manual pada sistem operasi Microsoft Windows XP Professional Edition Service Pack 2 terletak pada: Start > Control Panel > Local Area Connection > Properties > Internet Protocol TCP/IP > Properties .

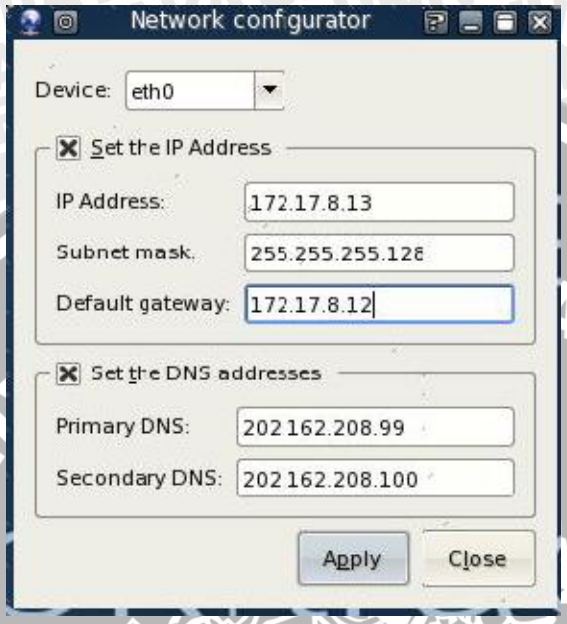


Gambar 5.2 konfigurasi IP Address pada Client

5.1.2.3 Konfigurasi TCP/IP pada Komputer Penyerang

Konfigurasi TCP/IP pada komputer penyerang dilakukan agar komputer dapat berkomunikasi dengan komputer gateway, network internal maupun internet.

Konfigurasi alamat IP komputer penyerang dilakukan dengan konfigurasi alamat IP secara manual. Konfigurasi alamat IP secara manual pada sistem operasi BackTrack 2.0 terletak pada: Start > Internet > Set IP Address.



Gambar 5.3 konfigurasi IP Address pada Komputer Penyerang

5.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak terdiri dari tiga bagian utama yaitu instalasi Snort, instalasi BlockIt dan konfigurasi Iptables.

5.2.1 Instalasi Snort

Snort yang digunakan adalah Snort versi 2.7.0.1 yang didapatkan dari url http://download.fedora.redhat.com/pub/fedora/linux/releases/9/Everything/i386/os/Packages/snort-2.7.0.1-6.fc9.i386.rpm.

Adapun instalasi Snort dilakukan dengan cara sebagai berikut:

- Melakukan perintah install berikut:

```
[root@ips-jar ~]# rpm -ivh
snort-2.7.0.1-6.fc9.i386.rpm.
```

5.2.1.1 Patching

Program yum dikonfigurasikan menggunakan repository (tempat data program disimpan dan dirawat) yang diarahkan ke url

ftp://ftp.brawijaya.ac.id/linux/fedora/linux/releases/9/Fedora/i386/os/

dengan melakukan konfigurasi pada file fedora.repo seperti dibawah ini:

```
[root@ips-jar ~]# vi
/etc/yum.repos.d/fedora.repo
```

```
[fedora]
name=Fedora $releasever - $basearch
baseurl=ftp://ftp.brawijaya.ac.id/linux/fed
ora/linux
/releases/9/Fedora/i386/os/
#mirrorlist=http://mirrors.fedoraproject.or
g/mirrorl
ist?repo=fedora-$releasever&arch=$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-
fedora
file:///etc/pki/rpm-gpg/RPM-GPG-KEY
```

5.2.1.2 Konfigurasi Snort

Konfigurasi dilakukan pada file snort.conf yang terletak di direktori /etc/snort/.

```
[root@ips-jar ~]# vi
/etc/snort/snort.conf
```

5.2.1.3 Rules Snort

Rule Snort merupakan database yang berisi pola-pola serangan berupa signature jenis-jenis serangan. Rule Snort dapat didownload dari url

http://www.snort.org/pub-bin/downloads.cgi/Download/vrt_os/snortrules-snapshot-2.6.tar.gz.

```
[root@ips-jar ~]wget
http://www.snort.org/pub-
bin/downloads.cgi/Download/vrt_os/snort
rules-snapshot-2.6.tar.gz
```

Kemudian rules tersebut di un-tar pada direktori `/etc/snort/` :

```
[root@ips-jar ~]# tar zxvf
snortrules-snapshot-2.6.tar.gz -C
/etc/snort/
```

Berikut merupakan beberapa hasil rules Snort pada implementasi:

```
[root@ips-jar rules]# ls
attack-responses.rules          smtp.rules
Makefile.am                     snmp.rules
backdoor.rules                  misc.rules
specific-threats.rules          bad-traffic.rules
multimedia.rules                spyware-put.rules
cgi-bin.list                    mysql.rules
sql.rules                       chat.rules
netbios.rules                   telnet.rules
content-replace.rules           nntp.rules
tftp.rules                      ddos.rules
open-test.conf                  virus.rules
deleted.rules                   oracle.rules
voip.rules                      dns.rules
other-ids.rules                 VRT-License.txt
dos.rules                       p2p.rules
web-attacks.rules               experimental.rules
policy.rules                    web-cgi.rules
exploit.rules                   pop2.rules
web-client.rules                finger.rules
pop3.rules                      web-coldfusion.rules
ftp.rules                       porn.rules
web-frontpage.rules             icmp-info.rules
rpc.rules                       web-iis.rules
icmp.rules                      rservices.rules
web-misc.rules                  imap.rules
scan.rules                      web-php.rules
info.rules                      shellcode.rules
x11.rules                       local.rules
```

5.2.2 Instalasi Paket Library

Dibutuhkan library agar Sistem Pencegahan Flooding data ini dapat berjalan, beberapa diantaranya adalah:

- Iptables-devel

  Iptables-devel yang digunakan adalah iptables-devel-1.3.7-2.i386.rpm, didownload:

```
root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/r
eleases/9/Everything/i386/os/Packages/iptables-
devel-1.4.0-4.fc9.i386.rpm
```

  Sedangkan cara instalasinya adalah:

```
[root@ips-jar ~]# rpm -ivh
iptables-devel-1.4.0-
4.fc9.i386.rpm
```

- Libnet

  Libnet yang digunakan adalah libnet10-1.0.2a-12.fc7.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/libnet10-1.0.2a-
14.fc9.i386.rpm
```

  Sedangkan cara instalasinya adalah:

```
[root@ips-jar ~]# rpm -ivh
libnet10-1.0.2a-14.fc9.i386.rpm
```

- Libnet-devel

  Libnet-devel yang digunakan adalah libnet-devel-1.1.2.1-10.fc7.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/libnet-devel-
1.1.2.1-12.fc9.i386.rpm
```

  Cara instalasinya adalah:

```
[root@ips-jar ~]# rpm -ivh libnet-
devel-1.1.2.1-12.fc9.i386.rpm
```

- Libpcap

  Libpcap yang digunakan adalah libpcap-0.9.5-1.fc7.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/libnet-devel-
1.1.2.1-12.fc9.i386.rpm
```

  Sedangkan cara instalasinya:

```
[root@ips-jar ~]# rpm -ivh
libpcap-0.9.8-2.fc9.i386.rpm
```

- Libpcap-devel

  Libpcap-devel yang digunakan adalah libpcap-devel-0.9.5-1.fc7.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/libpcap-devel-
0.9.8-2.fc9.i386.rpm
```

Cara instalasinya:

```
[root@ips-jar ~]# rpm -ivh
libpcap-devel-0.9.8-2.fc9.i386.rpm
```

- Pcre

Pcre yang digunakan adalah pcre-7.0-2.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/pcre-7.3-
3.fc9.i386.rpm
```

Dan cara instalasinya:

```
[root@ips-jar ~]# rpm -ivh pcre-
7.3-3.fc9.i386.rpm
```

- Pcre-devel

Pcre-devel yang digunakan adalah pcre-devel-7.0-2.i386.rpm, didownload:

```
[root@ips-jar ~]# wget
http://download.fedora.redhat.com/pub/fedora/linux/re
leases/9/Everything/i386/os/Packages/pcre-devel-7.3-
3.fc9.i386.rpm
```

Dan cara instalasinya adalah:

```
[root@ips-jar ~]# rpm -ivh pcre-
devel-7.3-3.fc9.i386.rpm
```

5.2.3 Instalasi Firewall

5.2.3.1 Instalasi BlockIt

Program BlockIt digunakan untuk menghubungkan antara Snort dengan Iptables sehingga jika ada paket data yang dicurigai oleh Snort, maka BlockIt akan membaca log dari Snort yang berada di file "/var/log/snort/alert" dan akan memberitahu Iptables untuk memblok paket data tersebut.

BlockIt yang digunakan adalah versi 1.4.2 yang didapatkan dari url

http://www.teknofx.com/proggie/blockit-1.4.2.tar.gz

Adapun cara instalasi program ini sebagai berikut:

```
[root@ips-jar ~]# tar zxvf blockit-1.4.2.tar.gz -C
/tmp/
[root@ips-jar blockit-1.4.2]# sh install.sh
Please Enter Install Directory
[/usr/local/blockit]:
/etc/blockit
ln: `/etc/blockit/conf' and `/etc/blockit/conf'
are the same file
Do you want to configure MySQL support? [y/n]: y
Enter Username : ips
Enter Password : this_ips
If using PF add a line saying 'anchor blockit' in
your /etc/pf.conf!
```

5.2.3.2 konfigurasi BlockIt

Konfigurasi BlockIt dilakukan pada file bernama blockit.conf di direktori /etc/blockit/

```
[root@ips-jar ~]# vi
/etc/blockit/blockit.conf
```

Berikut beberapa konfigurasi yang harus dilakukan:

- default yang digunakan adalah Iptables, maka konfigurasi dilakukan pada firewall options adalah "FirewallType=0".

```
# - Firewall Type -
# Type of Firewall Program to use
# 0 = IPTABLES --- DEFAULT
# 1 = IPCHAINS
# 2 = IPFWADM
# 3 = CHECKPOINT
# 4 = IPFW
# 5 = IPFILTER
# 6 = PF

FirewallType = 0
```

- Melakukan path direktori Iptables sebagai Firewall default

```
# - Path To Firewall Binary -
FirewallPath = /sbin/iptables
```

- Menyesuaikan file dari log Snort yang mengacu pada direktori /var/log/snort/alert.

```
# - Snort's alert file. -
AlertFile = /var/log/snort/alert
```

Dan untuk menjalankan program BlockIt ini, harus masuk ke dalam direktori/etc/blockit/.

```
[root@ips-jar blockit]# ./blockit start
Autodetected Gateway Address: 172.17.8.1
Ignoring local ip: 168.18.18.1
Autodetected Host IP Address: 172.17.8.13
BlockIt v1.4.2

Loaded 0 addresses from
/etc/blockit/blockit.ignore
Loaded 0 addresses from
/etc/blockit/blockit.sigid
Loaded 0 addresses from
/etc/blockit/blockit.hosts
Applying Intruders File
Loaded 0 addresses from
/etc/blockit/blockit.intruders
Becoming a daemon..
```

5.2.4 Konfigurasi NAT

Konfigurasi ini digunakan agar network internal dapat melakukan komunikasi dengan network eksternal atau internet. NAT yang digunakan adalah jenis NAT yang mentranslasikan banyak IP address internal menjadi satu IP address eksternal. Konfigurasi ini diletakkan pada file iptables dengan isi konfigurasi:

```
[root@ips-jar ~]# vi
/etc/synconfig/iptables
```

```
# Generated by iptables-save v1.4.0 on 21:33:34 2008
*nat
:PREROUTING ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A POSTROUTING -s 168.18.18.0/25 -o eth0 -j SNAT --to-
source 172.17.8.13
COMMIT
```

BAB VI

PENGUJIAN

Bab ini menjelaskan tentang pengujian Sistem Pencegahan Flooding Data dan Blocking IP secara otomatis pada Jaringan Komputer. Pengujian yang dilakukan adalah pengujian DoS (Denial of Service). Pada tahap pengujian, terdapat dua kondisi yang dibandingkan, yaitu pada saat Snort tidak aktif dan pada saat Snort aktif. Dua kondisi yang berbeda ini digunakan untuk membandingkan keamanan jaringan.

Pengujian ini bertujuan mengetahui apakah paket IP DoS dapat dideteksi oleh Snort lalu dihentikan oleh Iptables. Sistem ini dikatakan aman jika dapat menghentikan paket IP tersebut.

6.1 Spesifikasi Komputer Pengujian

Komputer yang digunakan untuk pengujian ini terdiri dari tiga buah yang terdiri dari 1 buah komputer penyerang, 1 buah komputer router dan salah satu komputer yang berada di network internal. Spesifikasi komputer yang ada sebagai berikut:

- Komputer Penyerang

Operating System (OS): BackTrack 2.0 Final

Processor : Intel Celeron M - 1.73 GHz

Memory : 1.214 MB RAM

Harddisk : 80 GB

Komputer penyerang berada pada network internal:

IP Address : 168.18.18.5

Default Gateway : 168.18.18.1

- Komputer Router

Operating System (OS): Linux Fedora Core 9

Processor : Intel Pentium 4 - 2.66 GHz

Memory : 256 MB RAM

Harddisk : 20 GB

IP Address : eth0 : 172.17.8.12

eth1 : 168.18.18.1

Default Gateway : 172.17.8.1

- Komputer network internal

Operating System (OS): Windows XP Professional SP 2

Processor : Intel Pentium 4 - 2.66 GHz

Memory : 256 MB RAM

Harddisk : 80 GB

IP Address : 168.18.18.2

Default Gateway : 168.18.18.1

6.2 Pengujian DoS (Denial of Service)

Tujuan utama dari pengujian ini adalah mengetahui apakah paket IP DoS dapat dihentikan atau tidak oleh Iptables firewall. Pengujian ini dilakukan dengan mengirimkan paket yang tidak dibutuhkan dalam jumlah besar dengan interval waktu yang sangat pendek. Pengujian ini dilakukan dengan cara teknik ping flood attack.

Pengujian ping flood dilakukan dengan dua kondisi yaitu Snort aktif dan Snort tidak aktif. Pengujian ini dilakukan dari komputer penyerang yang terletak pada network internal.

6.2.1 Pengujian Ping Flood dari network internal

1. Snort Tidak Aktif

A. Tujuan

Pengujian ini dilakukan untuk mengetahui pada saat Snort tidak aktif, apakah paket IP ping flood ke komputer router tidak dihentikan oleh Iptables.

B. Prosedur

Prosedur pengujian yang dilakukan di komputer router adalah mematikan Snort dengan menjalankan perintah berikut:

```
[root@ips-jar ~]# service
snortd stop
```

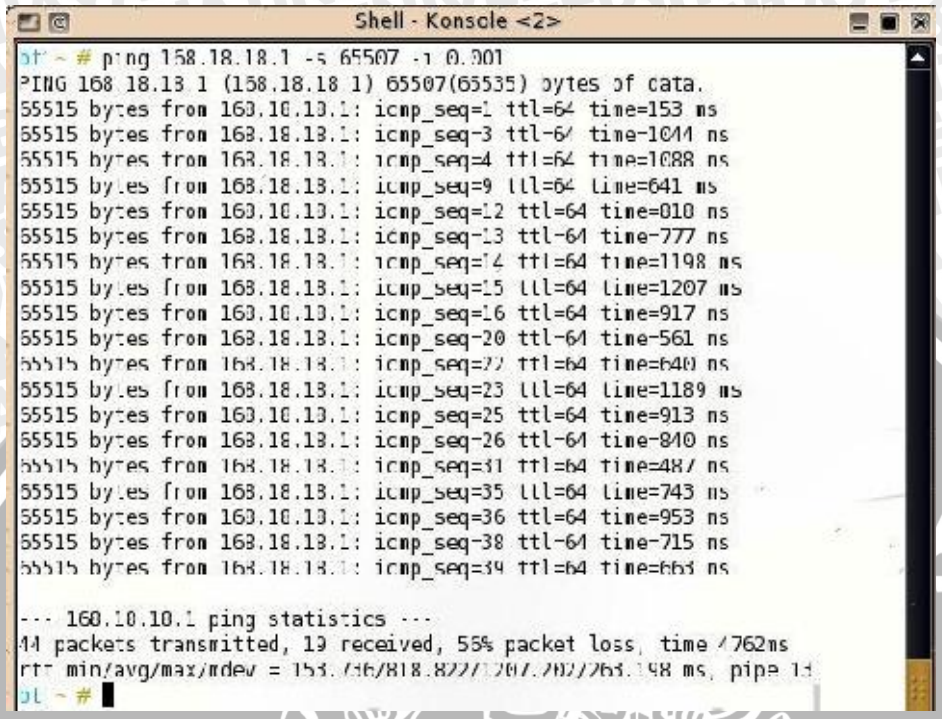
Prosedur pengujian yang dilakukan di komputer penyerang adalah menjalankan program ping flood dengan perintah sebagai berikut:

```
[root@ips-jar ~]# ping
168.18.18.1 -s 65507 -i
0.001
```

C. Hasil Pengujian yang Diharapkan

Paket IP ping flood ke komputer router tidak dihentikan oleh Iptables.

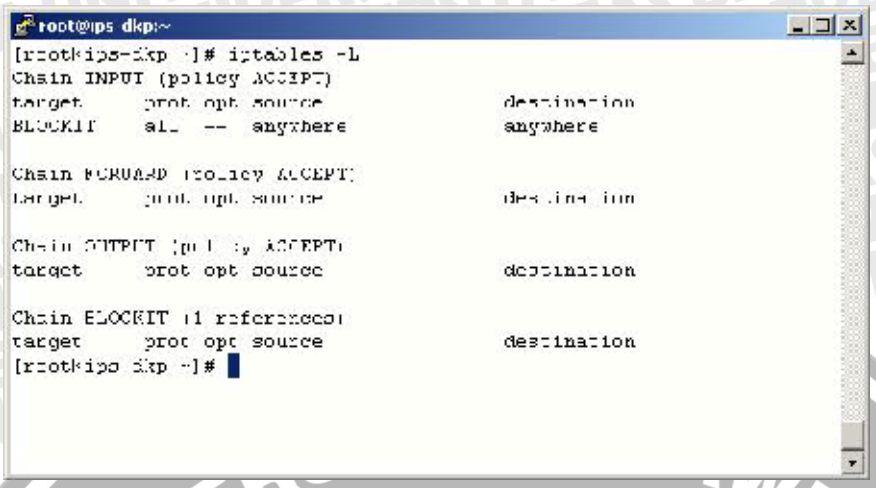
D. Hasil Pengujian



Gambar 6.1 Ping Flood dari komputer penyerang network internal saat Snort tidak aktif

E. Analisis Hasil Pengujian

Hasil pengujian pada saat Snort dalam keadaan tidak aktif menunjukkan bahwa komputer penyerang melakukan ping flood secara terus menerus tanpa dihentikan oleh Iptables. Dapat dilihat pada Iptables pada Gambar



Gambar 6.2 Iptables meloloskan alamat IP komputer penyerang (network internal) saat melakukan ping flood

F. Kesimpulan

Hasil pengujian ping flood pada saat Snort tidak aktif menunjukkan bahwa ping flood yang menyerang komputer router tidak dihentikan oleh Iptables firewall.

2. Snort Aktif

A. Tujuan

Pengujian ini dilakukan untuk mengetahui pada saat Snort aktif, apakah paket IP ping flood ke komputer router dihentikan oleh Iptables atau tidak.

B. Prosedur

Prosedur pengujian yang dilakukan di komputer router adalah mengaktifkan Snort dengan menjalankan perintah berikut:

```
[root@ips-jar ~]# service
snortd start
```

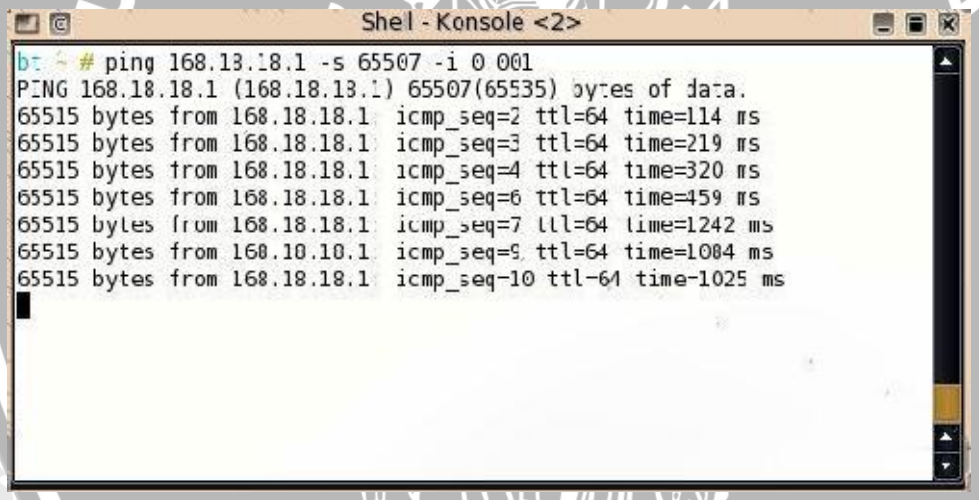
Prosedur pengujian yang dilakukan di komputer penyerang adalah menjalankan program ping flood dengan perintah sebagai berikut:

```
[root@ips-jar ~]# ping
168.18.18.1 -s 65507 -i
0.001
```

C. Hasil yang Diharapkan

Pengujian ini dilakukan untuk mengetahui pada saat Snort aktif, apakah paket IP ping flood ke komputer router dihentikan oleh Iptables atau tidak.

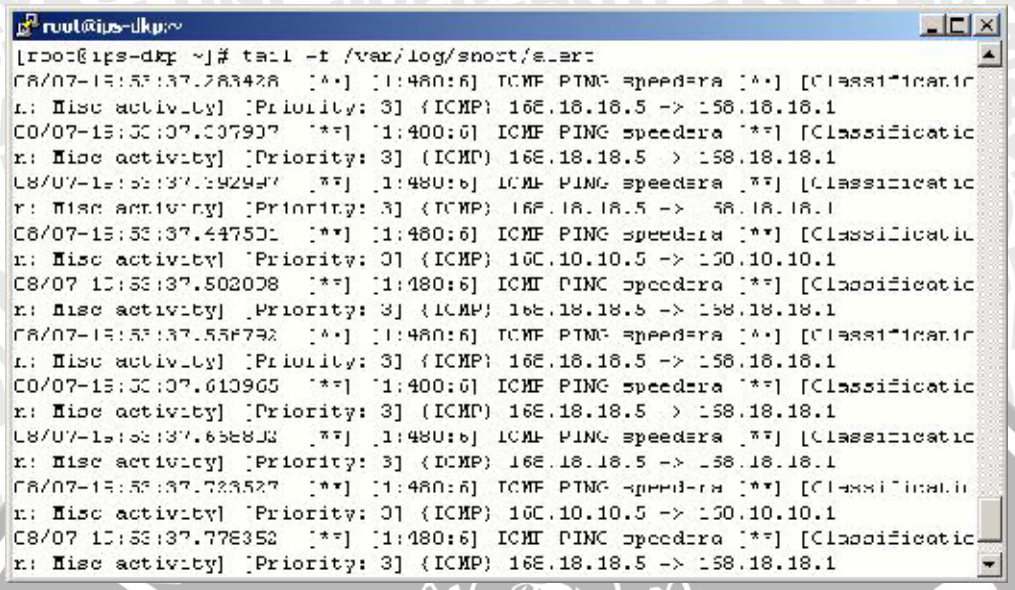
D. Hasil Pengujian



Gambar 6.3 Flood dari komputer penyerang network internal saat Snort aktif

E. Analisis Hasil Pengujian

- Hasil pengujian menunjukkan bahwa pada saat komputer penyerang melakukan ping flood langsung dihentikan oleh Iptables, sehingga komputer penyerang tidak dapat melakukan ping flood terus menerus.
- Pada saat melakukan ping flood ke komputer router, Snort mendeteksi paket data ping yang dikirim oleh komputer penyerang, dapat dilihat pada Gambar



Gambar 6.4 Snort mendeteksi ping flood dari komputer penyerang network internal

- Paket data ping flood tersebut terdeteksi oleh Snort karena mempunyai pola yang sama pada rule Snort, yakni icmp.rules

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any
(msg:"ICMP PING speedera"; itype:8;
content:"89|3A 3B|<=>?"; depth:100;
classtype:misc-activity; sid:480; rev:6;)
```

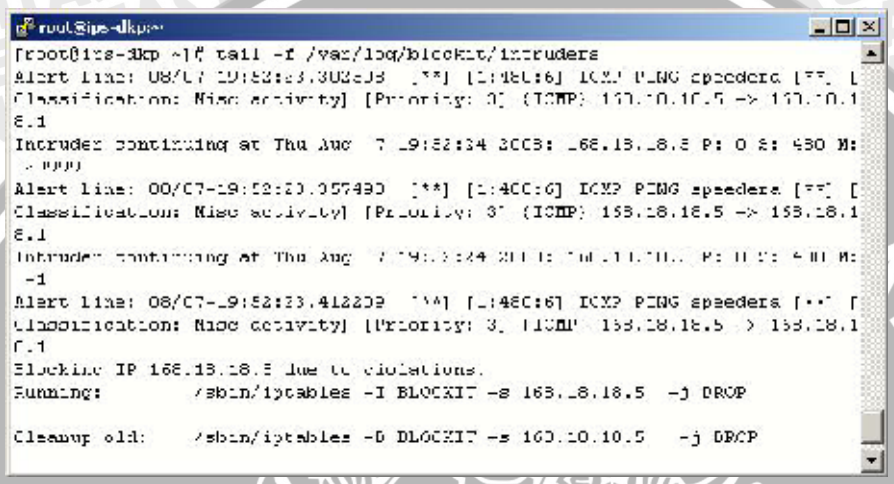
Keterangan:

`alert tcp $EXTERNAL_NET any -> $HOME_NET any` : akan memberikan alert jika ada paket data dari network eksternal yang memasuki ke network internal.

`msg:"ICMP PING speedera"` : memberikan pesan "ICMP PING speedera"

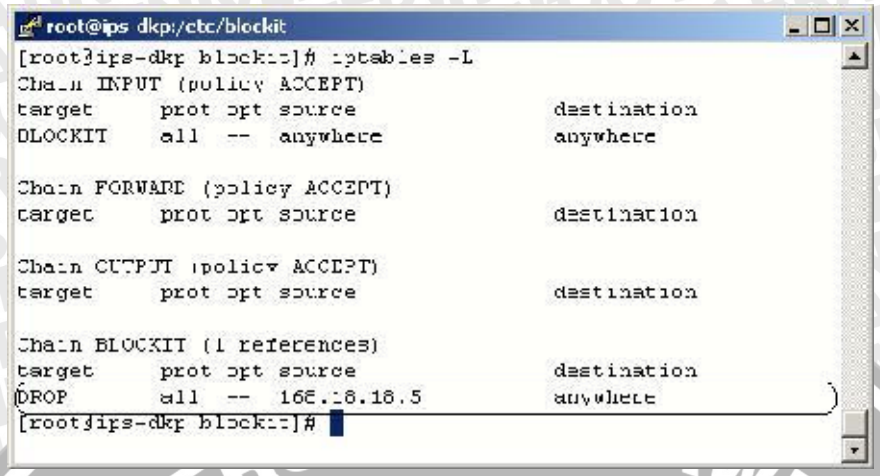
`classtype:misc-activity` : penyusupan yang bertipe aktivitas mencurigakan.

- BlockIt mencatat alamat IP komputer penyerang yang melakukan ping flood dari log Snort lalu disimpan pada file blockit.intruder, kemudian memerintahkan Iptables untuk melakukan DROP. Dapat dilihat pada Gambar



Gambar 6.5 Blockit mencatat IP komputer penyerang network internal yang melakukan ping flood

- Alamat IP komputer penyerang langsung di DROP oleh Iptables, dapat dilihat pada gambar



Gambar 6.6 Iptables memblok alamat IP komputer penyerang network internal yang melakukan ping flood

F. Kesimpulan

Hasil pengujian ping flood pada saat Snort aktif menunjukkan bahwa ping flood yang menyerang komputer router dihentikan oleh Iptables firewall.

BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis sistem pencegahan flooding data, dapat disimpulkan bahwa :

1. Sistem dapat mendeteksi flooding data

   Data yang keluar masuk akan dideteksi,sehingga semua data bisa dilihat apakah data itu merupakan flooding atau bukan, sehingga data bisa mengklasifikasikan bahwa data tersebut benar-benar melakukan flooding atau tidak.

2. Sistem dapat bekerja meskipun di berikan flood yang besar

   Karena pembatasan paket datang yang masuk merupakan variabel yang bisa diubah besar kecilnya maka berapapun besar flood yang masuk dapat di deteksi dan diatasi, selain itu pengolahan data bukan semua data yang ada melainkan data-data yang sudah sangat terseleksi.

3. Sistem dapat bekerja meskipun tidak ada admin

   Karena sifat dari sistem yang otomatis keberadaan seorang admin untuk mengatur server apabila flood terjadi tidak diperlukan lagi. Sistem mampu mengatasi sendiri dengan melakukan pengambilan keputusan data masuk apakan flood atau tidak. Dan juga sekaligus melakukan tindakan akhir apabila flood benar-benar terjadi yaitu dengan melakukan blooking data.

4. Keamanan data lebih terjamin Dengan adanya penanggulangan yang dini atas flooding data maka keamanan dari jaringan akan lebih terjamin baik dari segi kemanan alat maupun dari segi keamanan data

5. Sistem yang diimplementasikan menggunakan program Snort dan IPTables. Program tersebut diletakkan pada komputer router.
6. Snort yang telah dikonfigurasi dapat mendeteksi paket data dari alamat IP komputer penyerang yang melakukan host reconnaissance dan serangan flooding data yang ditunjukkan pada log Snort di direktori var/log/snort/alert.
7. Firewall yang menggunakan program Iptables dapat menghentikan host reconnaissance dan serangan flooding data yang ditunjukkan dengan status ping time out pada sisi komputer penyerang.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem ini adalah:

1. Flooding data yang terjadi hanya bisa dicegah sampai titik server saja, dan hanya bisa mencegah data masuk kedalam jaringan yang bisa menyebabkan kerusakan yang lebih parah. Tetapi proses pengiriman data oleh pelaku flooding masih tetap berlangsung tanpa bisa dihentikan. Sebagai akibat pengiriman data yang terus menerus itu tentunya traffic yang ada masih mengalami ganguan, yaitu berupa penuhnya jaringan yang ada. Proses pengiriman data dan penerimaan data akan mengalami kelambatan. Sehingga masih diperlukan suatu sistem untuk menyempurnakan sistem ini dengan menambahkan suatu komunikasi dari server ke server. Dalam hal ini hubungan server lokal ke server yang lebih tinggi. Tujuan komunikasi ini adalah untuk mengadakan pemblokiran IP pada server yang lebih tinggi sehingga gangguan yang ada lebih bisa dikurangi lagi. Traffic di jaringan lokal akan kembali normal karena data yang sebelumnya datang sudah di blokir di tingkat lebih atas.
2. Untuk mempermudah pengelolaan rule perlu user interface (front end) yang friendly seperti Webmin yang ditambahkan plugin Snort rule
3. Untuk mempermudah analisa terhadap catatan-catatan Snort (security event) perlu ditambahkan modul tambahan seperti ACID (Analysis Console for Intrusion Databases).