

Interface Dan Layanan

Fungsi setiap layer adalah untuk melayani keperluan layer yang berada di atasnya. **Entity** adalah elemen-elemen aktif dalam sebuah layer. Dapat berupa software entity (contoh : proses) atau hardware entity (contoh : intelligent I/O chip). **Peer entity** adalah entity yang berada pada layer yang sama namun pada mesin yang berbeda.

Entity pada layer n mengimplementasikan sebuah layanan yang dipakai oleh layer $n+1$. Dalam hal ini layer n disebut service provider dan layer $n+1$ disebut service user. Layer n dapat menggunakan layanan layer $n-1$ agar dapat melaksanakan tugas-tugasnya.

Layanan-layanan tersebut dapat dijumpai dalam **SAP (Service Access Point)**. SAP layer n adalah sebuah tempat dimana layer $n+1$ dapat mengakses layanan yang diberikannya. Setiap SAP mempunyai alamat yang unik untuk mengidentifikasi dirinya. SAP dalam sistem telepon adalah socket tempat telepon modular dihubungkan, sedangkan alamat SAP adalah nomor telpon dari socket yang bersangkutan. Sama halnya dengan sistem surat pos, alamat SAP adalah nama jalan dan nomor kotak pos. Agar dua layer dapat saling bertukar informasi, harus dibuat persetujuan tentang interface yang akan dipakai. Pada interface tertentu, entity layer $n+1$ melintasi sebuah IDU (*Interface Data Unit*) ke entity layer n melalui SAP. IDU terdiri dari sebuah SDU (*Service Data Unit*) dan beberapa informasi kontrol. SDU merupakan informasi yang dikirimkan pada jaringan ke peer entity hingga sampai di layer $n+1$. Informasi kontrol digunakan untuk membantu layer yang dibawah melaksanakan tugasnya (misalnya, jumlah byte pada SDU). Akan tetapi informasi kontrol bukan merupakan bagian dari data itu sendiri.

Pelayanan

Kualitas layanan disebut **andal** jika layanan yang bersangkutan tidak pernah kehilangan data. Biasanya dilengkapi dengan pemberitahuan (*acknowledgment*) si penerima bahwa pesan yang dikirim telah sampai (*delivery*).

1. Layanan *connection-oriented* dan *connectionless*

Layer memiliki 2 macam layanan bagi layer di atasnya yaitu :

1. Layanan *connection-oriented* / pelayanan berorientasi

Layanan dengan pembentukan sambungan terlebih dahulu kemudian membubarkannya setelah transmisi data selesai. Dibuat berdasarkan sistem telepon. Pengguna layanan membuat hubungan, menggunakan hubungan dan menghentikan hubungan (contoh : angkat nomor telepon, putar nomor telepon, berbicara, letakkan telepon)

2. Layanan *connectionless* / pelayanan tidak berorientasi

Layanan tanpa pembentukan sambungan dan tiap data yang dikirim mempunyai alamat tujuannya. Pengguna layanan : membuat koneksi, menggunakan koneksi dan menghentikan koneksi. Aspek penting sebuah koneksi adalah seperti halnya sebuah tabung, pengirim mendorong obyek (bit-bit) pada sebuah ujung, dan si penerima mengambil bit-bit itu di ujung lainnya sesuai dengan urutannya. Dibuat berdasarkan sistem surat pos. Setiap pesan (surat) memiliki alamat yang dituju, mempunyai rute masing-masing yang berdiri sendiri. Umumnya dua pesan yang mempunyai alamat tujuan yang sama, maka surat yang dikirim lebih dulu akan sampai di tujuan lebih dulu juga. Akan tetapi mungkin saja pesan yang dikirim lebih dahulu mengalami keterlambatan, sehingga pesan yang dikirim berikutnya yang akan sampai duluan. Hal ini tidak mungkin terjadi pada layanan *connection oriented*.

- Layanan *unreliable connectionless* (tidak menggunakan *acknowledgment*) disebut juga layanan *datagram*, seperti layanan telegram yang tidak memberikan pemberitahuan ke si pengirim.

- Layanan *acknowledgment datagram*, seperti pengiriman surat tercatat dan permintaan tanda terima. Pada saat tanda terima sudah kembali, pengirim akan merasa yakin bahwa suratnya telah dikirimkan ke alamat penerima dan tidak hilang di jalan.

- Layanan *request-reply*, dimana sebuah datagram berisi permohonan, balasannya merupakan jawaban. Umumnya digunakan untuk mengimplementasikan model client-server, dimana client mengajukan pertanyaan dan server akan menjawabnya

2. Layanan primitive

Suatu layanan secara resmi ditentukan oleh seperangkat primitive (operasi) yang tersedia bagi pengguna / entity lainnya untuk mengakses layanan tersebut. Primitive meminta layanan untuk membuat aksi / laporan tentang aksi yang dilakukan oleh sebuah peer entity, yang menggolongkannya menjadi 4 kelas yaitu :

1. *Connect.request* yang meminta dimulainya suatu hubungan atau menyebabkan dikirimkannya sebuah paket / suatu entity meminta layanan untuk mengerjakan sesuatu. Parameternya dapat menspesifikasikan kemana mesin tersambung, jenis layanan yang diinginkan dan ukuran pesan maksimum yang akan digunakan dalam hubungan yang bersangkutan.
2. *Connect.indication* yang memberitahu bahwa sebuah entity yang ada di suatu tempat perlu diinformasikan mengenai suatu kejadian / ingin membuat hubungan dengannya. Parameternya berisi identitas pemanggil, jenis layanan yang diinginkan dan ukuran pesan maksimum yang diajukan.
3. *Connect.response* yang memberitahukan bahwa ia menerima atau menolak pengajuan permintaan hubungan tersebut. Suatu entity ingin memberikan tanggapan terhadap suatu kejadian. Parameternya berisi penerimaan dan penolakan, jika ditolak akan meminta usulan ulang.
4. *Connect.confirm* yang mengkonfirmasi / menanggapi apa yang terjadi terhadap permintaan sebelumnya telah kembali. Parameternya berisi usulan ulang yang biasanya berisi permintaan perubahan ukuran pesan maksimum.

Negoisasi seperti ini merupakan bagian dari protokol.

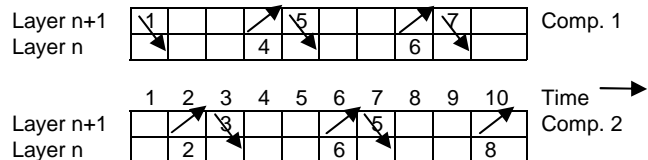
- *Layanan confirmed* memiliki sebuah request, indication, response dan confirm (contohnya connect).
- *Layanan unconfirmed* hanya memiliki request dan indication (contohnya disconnect). Tergantung apakah pengiriman memerlukan acknowledgment atau tidak.

Contoh pada sistem telpon :

1. connect.request : si A memutar tombol nomor telpon si B
2. connect.indication : telpon si B berdering
3. connect.response : si B mengangkat telpon
4. connect.confirm : si A mendengar bunyi nada panggil berhenti

5. data.request : si A mengundang si B belajar bersama
6. data.indication : si B mendengar undangan si A
7. data.request : si B berkata ingin memenuhi undangan si A
8. data.indication : si A mendengar perkataan menerima undangan tersebut
9. disconnect.request : si A memutuskan hubungan telepon, meletakkan gagang telepon
10. disconnect.indication : si B mendengar anda telah memutuskan hubungan, maka iapun menyimpan gagang teleponnya juga.

Contoh dalam jaringannya, dimana melibatkan dua buah layer yang ada pada sebuah komputer. Sebuah request / response menyebabkan sebuah indication / confirm di pihak lain beberapa saat kemudian. Pengguna layanan (si A dan si B) berada dalam layer n+1 dan penyedia layanan (sistem telepon) berada di layer n



Hubungan layanan dan protokol

Layanan merupakan sekumpulan primitive (operasi) yang disediakan sebuah layer bagi layer yang di atasnya, menentukan operasi yang disiapkan untuk dilakukan bagi pengguna-penggunanya, tetapi layanan ini tidak memberitahukan cara operasi tersebut diimplementasikan. Layanan dihubungkan oleh sebuah interface yang ada diantara 2 layer.

Protokol merupakan sejumlah aturan yang mengatur format dan arti sebuah frame, paket / pesan yang dipertukarkan diantara 2 peer entity dalam sebuah layer agar dapat mengimplementasikan definisi-definisi layanannya.

Contoh Model-model Referensi

1. Model Referensi OSI
2. Model Referensi TCP/IP

Model referensi OSI

OSI (*Open System Interconnection*) dibuat oleh ISO (*International Standards Organization*) sebagai standarisasi protokol internasional yang digunakan pada berbagai layer yang ditujukan pada penyambungan sistem terbuka (*open system*) yang dapat berkomunikasi dengan sistem yang lain.

Membagi protokol untuk komunikasi data atas 7 layer / lapisan. Dimana tiap layer independent tetapi fungsi masing-masing layer tergantung dari keberhasilan operasi layer sebelumnya.

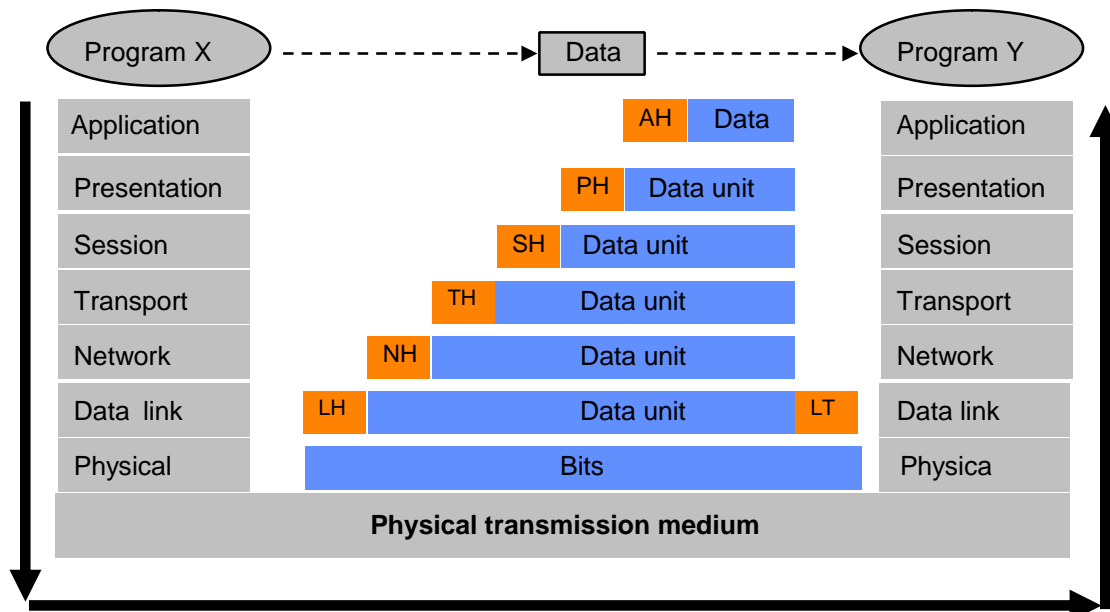
- Layer tertentu di Tx hanya perlu berhubungan dengan layer yang sama di Rx, menggunakan protokol.
- Layer tertentu berhubungan dengan layer di atasnya dan dibawahnya dimana layer tersebut berada
- Proses pengiriman memiliki data yang akan dikirim ke proses penerima. Tx menyerahkan data ke application layer, kemudian menambahkan application layer, AH (yang mungkin kosong) ke ujung depannya dan menyerahkan hasilnya ke presentation layer.
- Presentation layer dapat mengubah data dalam berbagai cara dan mungkin saja menambahkan sebuah header di ujung depan, memberikan hasilnya ke session layer. Presentation layer tidak menyadari tentang bagian data yang mana diberi tanda AH oleh application layer yang merupakan data pengguna sebenarnya.
- Proses pemberian header ini berulang terus sampai data tersebut mencapai physical layer. Pada layer ini data akan ditransmisikan ke mesin lainnya.

- Pada mesin tersebut, semua header tadi dilepas satu persatu sampai mencapai proses penerima.
- Transmisi data aktual berbentuk vertikal, setiap layer diprogram seolah-olah transmisi berlangsung secara horisontal. Seperti seorang diplomat berbahasa Indonesia sedang menyampaikan pesan ke diplomat berbahasa rusia, ia dianggap seolah-olah ia berbicara langsung dengan diplomat rusia tersebut, padahal sebenarnya ia hanya berbicara ke panerjemahnya jika dipandang sebagai rincian teknis.

Pengendalian komunikasi dalam bentuk layer menambah overhead karena tiap layer berkomunikasi dengan lawannya melalui header, tetapi layer lebih mudah administrasi dan standarisasinya.

Tujuan OSI :

1. Koordinasi berbagai kegiatan
2. Penyimpanan data
3. Manajemen sumber dan proses
4. Keandalan dan keamanan sistem pendukung perangkat lunak
5. membuat kerangka agar sistem / jaringan yang mengikutinya dapat saling berkomunikasi / bertukar informasi, sehingga tidak tergantung merk dan model peralatan.
6. 3 layer pertama adalah interface antara terminal dan jaringan yang dipakai bersama, 4 layer terakhir adalah hubungan antara software.
7. Antar layer berlainan terdapat interface, layer yang sama terdapat protokol.



Missa Lamsani

7 lapisan OSI tersebut adalah :

1. Lapisan fisik

berhubungan dengan fisik saluran yang digunakan untuk transmisi, berupa peralatan mekanis dan elektrik / listrik, prosedur interface dan medium transmisi untuk memulai, memelihara dan membubarkan hubungan fisik untuk penyaluran bit. Misalkan mengirimkan bit 1 harus diterima bit 1 pula bukan bit 0. Lapisan harus memungkinkan digunakannya beberapa macam media fisik untuk interkoneksi dengan beberapa macam kontrol yang berbeda. Memberikan standarisasi hubungan antar pin dari connector DTE dan DCE.

2. Lapisan hubungan data

Tugas utamanya adalah sebagai fasilitas transmisi data dan mentransformasi data tersebut ke saluran yang bebas dari kesalahan transmisi. Merupakan teknik khusus yang digunakan pada prosedur untuk memungkinkan pengiriman data melalui media yang relatif memiliki error data rate (misal pada saluran telepon) yang tinggi dengan cara memecah data yang dikirimkan menjadi blok-blok, dimana tiap blok ditambahkan informasi untuk mengenali error, sehingga lapisan tersebut dapat mengenali blok yang diterima dengan baik / meminta pengiriman kembali bagi blok-blok yang mengalami error. Mengatur hubungan antara Tx dan Rx sehingga dapat berhubungan dan saling mengenal. Sambungan data tersebut dapat berbentuk :

- DTE melalui circuit switched network
- DTE melalui circuit switch jaringan pribadi
- Simpul ke terminal melalui jaringan paket
- Simpul ke simpul melalui jaringan paket

3. Lapisan jaringan

Tugas utamanya sebagai pengendalian operasi subnet. Mengatur hubungan antara Tx dan Rx yang menentukan jalan / rute yang harus ditempuh oleh data / informasi yang dikirimkan. Menjamin adanya connection path agar informasi yang dikirim dapat sampai pada alamat yang dituju dan bekerja sebagai traffic controller yang mengatur prioritas pengiriman informasi. Routing dan switched yang perlu untuk membentuk jaringan fisik dilakukan disini. Mengendalikan kemacetan yang terjadi.

4. Lapisan transport

Tugas utamanya adalah menerima data dari session layer. Mengatur lalu lintas / flow control secara keseluruhan melalui beberapa node sampai ke tujuan dan mengatur agar blok-blok yang dikirimkan dapat diterima dengan ukuran yang benar dan mencari cara yang paling baik dalam memanfaatkan karakteristik jaringan

yang dapat ditransmisikan paling efektif. Fungsinya untuk memecah berita dalam paket dan merakitnya kembali pada tujuan, melalui kendali aliran paket dalam hubungan transport.

5. Lapisan session

Bertugas untuk mengatur, mengorganisir dan mensinkronkan dialog dalam pertukaran data, sinkronisasi antara Tx dan Rx, pemeriksaan password, menentukan hubungan full dan half duplex, serta sebagai manajemen token. Fungsinya untuk memeriksa urutan berita jika layer dibawahnya tidak memberikan jaminan dalam hal penyampaian dan urutan, menghubungkan berita dan tanggapan atasnya dengan cara kendali (contohnya nomor urut), mengendalikan cara percakapan dilaksanakan yaitu aturan untuk dialog.

6. Lapisan presentasi

Bertugas untuk menjamin penemuan sebuah penyelesaian umum bagi masalah tertentu, sebagai presentasi dan manipulasi data sehingga berhubungan dengan syntax (presentasi data) dan tidak bersangkutan dengan semantik (arti dari data) yaitu hanya dimengerti oleh layer berikutnya serta sebagai data encoding. Peralatan / sistem yang menggunakan data format yang berlainan dapat saling berkomunikasi merupakan layer penerjemah yang mengatur komposisi data, konversi data dalam bentuk yang dapat diterapkan untuk mengirimkan data dan menerima data.

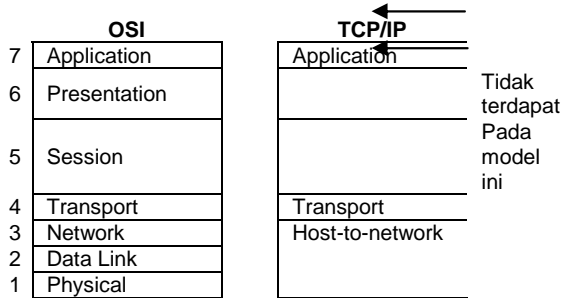
7. Lapisan application

Proses informasi agar dapat dimengerti oleh suatu proses aplikasi yang dapat berupa proses manual / komputer. Mengatur segala sesuatu yang berhubungan dengan pertukaran data / informasi antar pemakai, penerjemah lunak aplikasi / peralatan suatu sistem komputer.

Model referensi TCP/IP

TCP/IP adalah arsitektur dari suatu jaringan ARPANET. **ARPANET** adalah jaringan riset yang disponsori DoD (Departemen Pertahanan AS – US Department of Defence) yang menghubungkan ratusan universitas dan instalasi pemerintah dengan menggunakan kabel telepon sewaan. Kemudian saat jaringan satelit dan radio ditambahkan ke sistem, ternyata protokolnya mendapat kesulitan dalam menghubungkan komputer yang telah tergabung, karena itu diperlukan model referensi yang baru. Dengan demikian kemampuan untuk menghubungkan jaringan komputer secara bersama-sama tanpa melihat adanya perbedaan merupakan tujuan utamanya, ditambah dengan mengusahakan agar

jaringan yang telah ada mampu mempertahankan diri dari hilangnya perangkat keras subnet (mesin Tx dan Rx), dengan percakapan yang ada tidak terputus.



1. **Lapisan internet** (internet = dalam jaringan)
 Menggunakan jaringan packet-switching yang didasarkan pada connectionless internet-working layer, yang merupakan simpul yang mengikat keseluruhan bentuk arsitektur secara bersamaan. Lapisan internet menentukan format paket dan protokolnya disebut IP (*Internet Protocol*). Bertugas untuk memungkinkan host mengirim paket-paket IP ke jaringan / ke tempat tujuan dan memungkinkan paket itu berjalan sendiri ke tempat tujuannya.

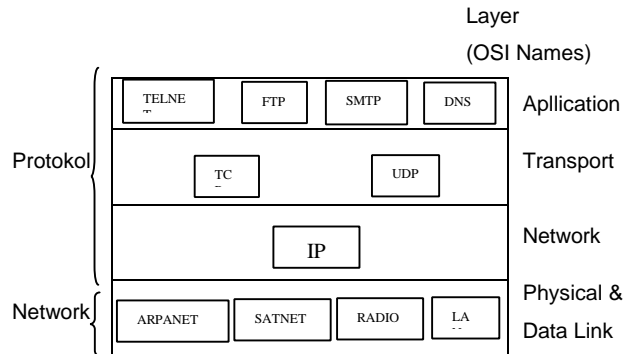
2. **Lapisan transport**
 Dirancang untuk memungkinkan peer entity ada host sumber dan tujuan melakukan percakapan. Terdapat 2 buah protokol end-to-end yaitu :

1. **TCP** (*Transmission Control Protokol*) merupakan protokol yang berorientasi pada hubungan yang andal yang mengijinkan sebuah aliran byte yang berasal dari suatu mesin untuk dikirimkan tanpa error ke sebuah mesin yang ada di internet. Dan memecah aliran byte data menjadi pesan diskret dan meneruskannya ke internet layer. Pada mesin tujuan, proses TCP penerima merakit kembali pesan yang diterimanya menjadi aliran output. TCP juga menangani pengendalian aliran untuk memastikan bahwa pengirim yang cepat tidak akan membanjiri pesan yang akan diterima panerima yang lambat.
2. **UDP** (*User Datagram Protokol*) merupakan protokol yang tidak andal dan tanpa sambungan bagi aplikasi yang tidak memerlukan pengurutan TCP / pengendalian aliran dan bagi aplikasi yang ingin melayani dirinya sendiri. Mengutamakan pengiriman yang cepat dibandingkan pengiriman yang akurat.

3. **Lapisan aplikasi**
 TCP/IP tidak mempunyai session dan presentation layer karena dianggap tidak diperlukan.

Lapisan aplikasi berisi macam-macam protokol tingkat tinggi, seperti :

1. **TELNET** (terminal virtual), yang mengijinkan pengguna pada sebuah mesin untuk log ke mesin yang ada ditempat yang jauh dan bekerja di terminal jarak jauh itu.
2. **FTP** (transfer file), memungkinkan pengiriman data dari mesin yang satu ke mesin yang lain secara efisien.
3. **SMTP** (surat elektronik), awalnya merupakan salah satu jenis transfer file, tetapi dibuat protokol khusus untuk itu.
4. **DNS** (*Domain Name Service*), untuk memetakan nama host ke alamat jaringannya.
5. **NNTP**, yaitu protokol yang digunakan untuk memindahkan artikel berita (*newsgroup*).
6. **HTTP**, protokol yang berguna untuk mengambil halaman (*page*) di world wide web.
7. Dll



Contoh-contoh jaringan

1. **Novell NetWare**
 Dirancang untuk digunakan oleh perusahaan yang ingin menurunkan ukuran dari mainframe menjadi jaringan PC (model client-server). Setiap pengguna memiliki desktop PC yang berfungsi sebagai client. Dan sejumlah PC yang lebih besar bertindak sebagai server yang bertugas menyediakan layanan file, database, dll ke sejumlah client. Model referensi Novell Netware memakai stack protokol yang dibuat berdasarkan pada sistem jaringan Xerox (Xerox Network System – XNS™).

Layer			
Application	SAP	File Server
Transport	NCP		SPX
Network	IPX		
Data Link	Ethernet	Token Ring	ARCnet
Physical	Ethernet	Token Ring	ARCnet

Physical layer dan data link dapat dipilih dari bermacam-macam standar industri, seperti ethernet, token ring, ARCnet. Network layer menjalankan protokol antarjaringan tanpa sambungan dan tidak dapat diandalkan yang disebut IPX (Internet Packet eXchange). IPX menyampaikan paket dari sumber ke tujuan secara transparan, meski sumber dan tujuan terdapat pada jaringan yang berlainan. IPX memiliki fungsi yang mirip dengan IP, hanya saja IPX menggunakan alamat 12-byte, bukan 4-byte. Transport layer yang connection oriented, yaitu NCP (Network Core Protocol) menyediakan layanan transport data pengguna, dll dan merupakan jantung Netware. Sedangkan SPX (Sequenced Packet

eXchange) hanya menyediakan layanan transport saja.

Pada NetWare tidak terdapat session layer dan presentation layer. Berbagai protokol aplikasi ditempatkan di application layer.

2. ARPANET

ARPA (Advanced Research Projects Agency) merupakan research jaringan dari jaringan telepon circuit-switched yang dianggap tidak aman, karena jika 1 jalur / switch hilang, maka percakapan yang menggunakan jaringan akan hilang.

3. USENET

USENET (*Use Network*) adalah jaringan yang menawarkan suatu pelayanan network news.

4. NSFNET / CSNET

NSF (*The US National Science Foundation*) membuat sebuah jaringan maya yang disebut **CSNET** (*Computer Science Network*).