

# TCP DAN UDP

Budhi Irawan, S.Si, M.T

# LAPISAN TRANSPOR

---

- adalah **Lapisan keempat** dari **Model Referensi OSI** yang bertanggung jawab untuk menyediakan *layanan-layanan* yang dapat diandalkan kepada *protokol-protokol* yang terletak di atasnya.

# LAYANAN PADA LAPISAN TRANSPOR

---

1. Mengatur *aliran (flow control)* untuk menjamin bahwa *perangkat* yang mentransmisikan data tidak mengirimkan *lebih banyak data* daripada yang *dapat ditangani* oleh *perangkat* yang menerimanya.

# LAYANAN PADA LAPISAN TRANSPOR (LANJ)

---

2. Mengurutkan *paket data* (*packet sequencing*), yang dilakukan untuk *mengubah data yang hendak dikirimkan* menjadi *segmen-segmen data* (proses ini disebut dengan proses *segmentasi/ segmentation*), dan tentunya memiliki *fitur untuk menyusunnya kembali (Reassembly)*.

# LAYANAN PADA LAPISAN TRANSPOR (LANJ)

---

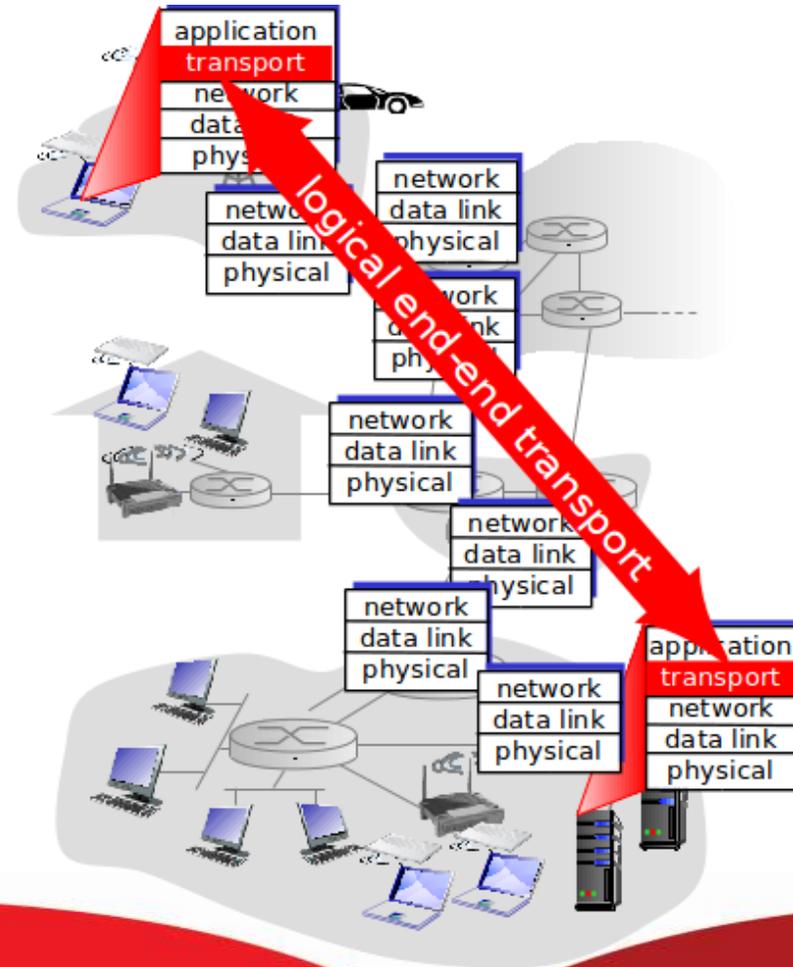
3. Penanganan *kesalahan* dan *fitur acknowledgment* untuk *menjamin bahwa data telah dikirimkan dengan benar* dan akan dilakukan *pengiriman ulang* ketika memang *data tidak sampai ke tujuan*.
4. Protokol Transpor adalah protokol *end2end* (hanya berada di *end sistem, tidak berada pada jaringan*)

# LAYANAN PADA LAPISAN TRANSPOR (LANJ)

---

5. *Multiplexing* yang digunakan untuk menggabungkan data dari *beberapa sumber* untuk dikirim melalui satu jalur data saja.
6. Pembentukan sirkuit virtual, yang dilakukan dengan cara membuat *sesi koneksi* antara dua node yang hendak berkomunikasi.

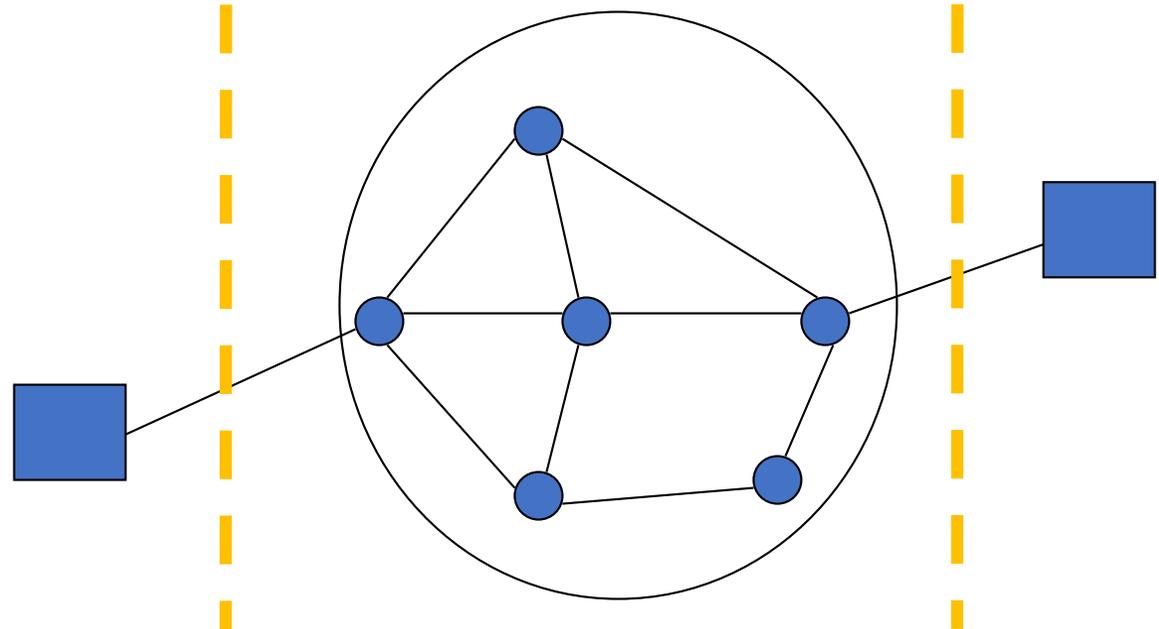
# LAYANAN PADA LAPISAN TRANSPOR (LANJ)



# TOPOLOGI LAPISAN TRANSPOR

---

Node ●  
Link —  
Terminal ■  
Jaringan ○



# PROTOKOL LAPISAN TRANSPOR

---

Protokol yang bekerja pada lapisan transpor adalah **Transmission Control Protocol (TCP)** dan **User Datagram Protocol (UDP)** yang tersedia dari kumpulan protokol TCP/IP

# PROTOKOL TCP

---

- Transmission Control Protocol (TCP) adalah suatu protokol yang berada di lapisan transpor yang berorientasi sambungan (*connection-oriented*) dan dapat *diandalkan (reliable)*.
- TCP dispesifikasikan dalam Dokumen RFC 793

# FUNGSI PROTOKOL TCP

---

1. Menyediakan komunikasi logika antar proses aplikasi yang berjalan pada host yang berbeda dan berjalan pada end systems
2. Bertugas untuk mengirimkan File (file transfer)
3. Untuk keperluan Remote Login
4. Dipergunakan untuk mendukung Computer Mail
5. Dapat mengelola Network File System (NFS)
6. Bisa dilakukan Remote Execution
7. Untuk manajemen Name Servers

# SKEMA PAKET TCP

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Source port																
Destination port																
Sequence number																
Acknowledge number																
Header length				Reserved				URG	ACK	PSH	RST	<u>SYN</u>	FIN			
Windows																
Checksum																
Urgent pointer																
Options																
								Padding								
User data = besarnya tidak ditentukan																

# DESKRIPSI SKEMA TCP

---

- Source port dan destination port ( $2^{16}$ ) : misal : **10.14.64.54:80** berbeda port dengan **10.14.64.54:102**
- Sequence Number ( $2^{32}$ ) : Nomor paket yang dikirim
- Acknowledge number ( $2^{32}$ ): Nomor acknowledge untuk paket telah diterima
- URG,ACK,PSH,RST,SEQ,FIN : bit-bit status pada paket.
  - URG = Urgent pointer
  - ACK = Acknowledgment
  - PSH = Deliver data on receipt of this segment
  - RST = Reset the sequence/acknowledgment numbers
  - SEQ = Sequence number
  - FIN = End of byte stream from sender

# SIFAT DAN KARAKTERISTIK TCP

---

1. **Reliable** berarti data ditransfer ke tujuannya dalam suatu urutan seperti ketika dikirimkan.
2. Berorientasi sambungan (**connection-oriented**) yang mana sebelum data dapat ditransmisikan antara **dua host**, dua proses yang berjalan pada lapisan aplikasi harus melakukan **negosiasi** untuk membuat **sesi koneksi** terlebih dahulu. Koneksi TCP ditutup dengan menggunakan proses **terminasi koneksi TCP (TCP connection termination)**.

# SIFAT DAN KARAKTERISTIK TCP

---

3. *Full-duplex* dimana untuk setiap *host TCP*, koneksi yang terjadi antara dua host terdiri atas *dua buah jalur*, yakni *jalur keluar* dan *jalur masuk*.
4. Memiliki layanan **Flow control** guna *mencegah data terlalu banyak dikirimkan pada satu waktu*, yang akhirnya *membuat padat* dalam suatu jaringan internetwork IP dan TCP mengimplementasikan *layanan flow control* yang dimiliki oleh pihak pengirim yang *secara terus menerus memantau* dan *membatasi jumlah data yang dikirimkan* pada satu waktu.

# SIFAT DAN KARAKTERISTIK TCP

---

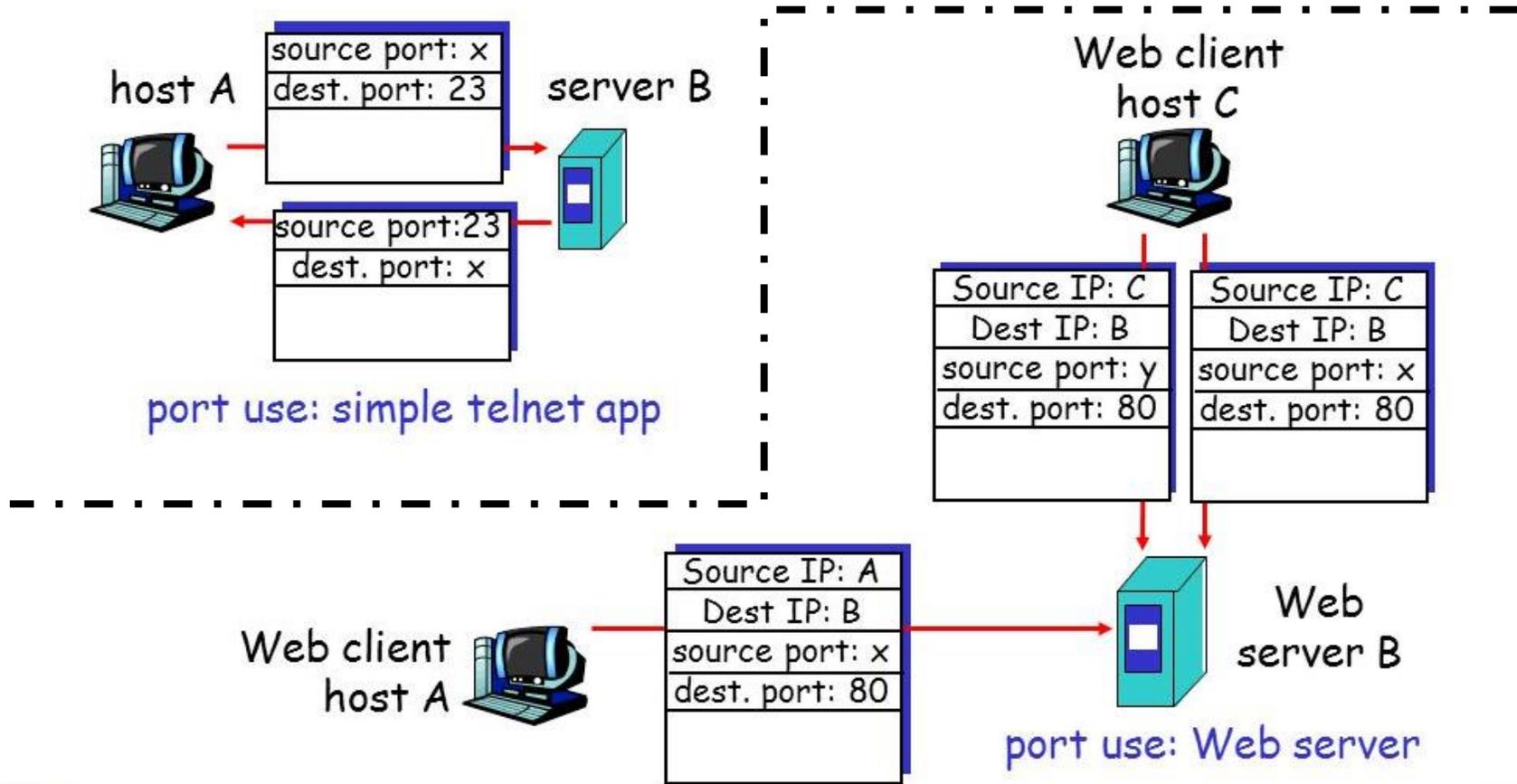
5. Melakukan *segmentasi* terhadap data yang datang dari lapisan aplikasi.
6. Mengirimkan paket secara “*one-to-one*” karena memang TCP harus membuat sebuah *sirkuit logis/ sirkuit virtual* antara *dua buah protokol lapisan aplikasi* agar saling dapat berkomunikasi.

# IMPLEMENTASI TCP

---

- TCP umumnya digunakan ketika protokol lapisan aplikasi membutuhkan *layanan transfer data* yang bersifat *andal*, yang mana layanan tersebut tidak dimiliki oleh protokol lain pada lapisan transpor.
- Contoh dari protokol yang menggunakan TCP pada lapisan aplikasi adalah **HTTP** dan **FTP**.

# IMPLEMENTASI TCP



# MANAJEMEN KONEKSI TCP

---

## Pada saat Setup Koneksi

1. Client mengirimkan *kontrol TCP SYN* ke server, dengan memberikan *sequence number inisial*.
2. Server menerima *TCP SYN*, dan membalasnya dengan *kontrol SYNACK*.
  - *ACK* yang menyatakan telah menerima *SYN*.
  - Mengalokasikan buffer.
  - Menghasilkan sequence number untuk ke client.

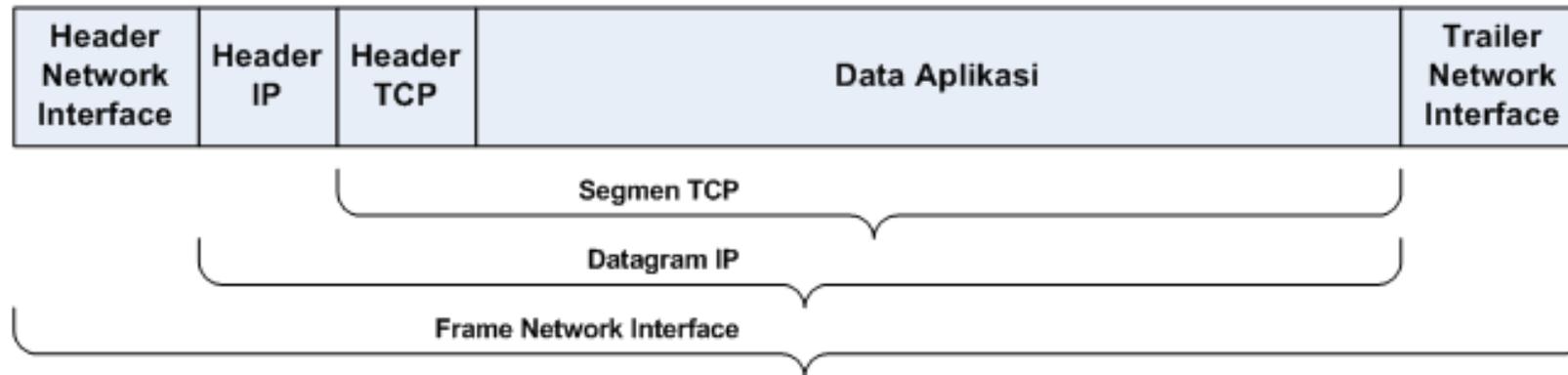
# MANAJEMEN KONEKSI TCP

---

## Pada saat Menutup Koneksi

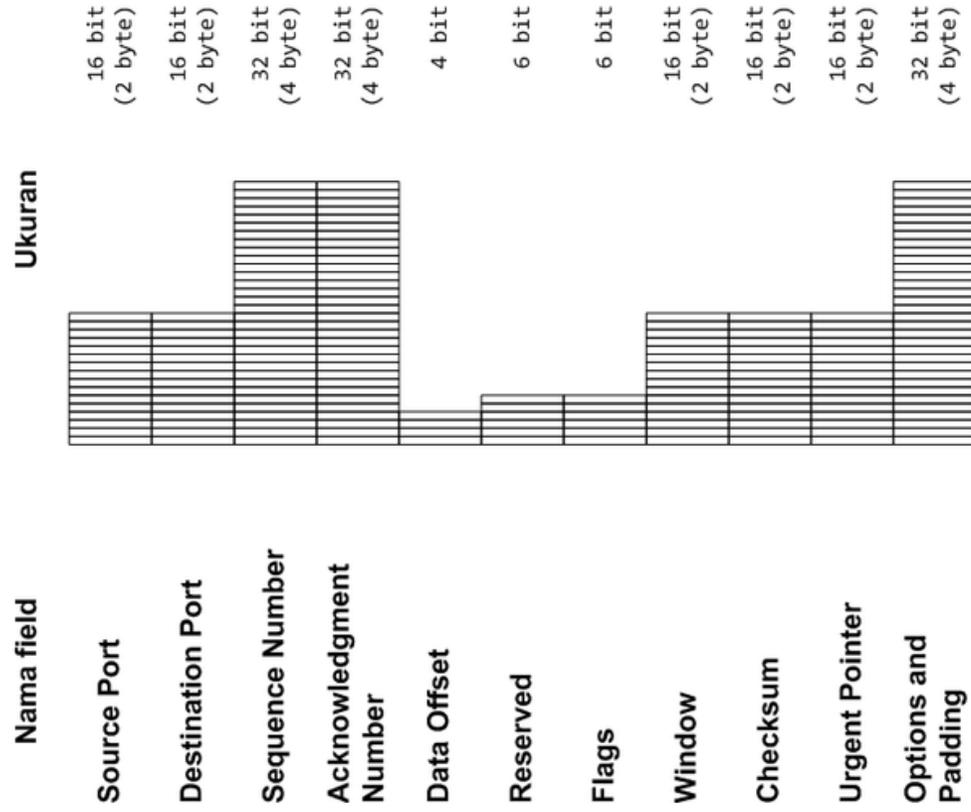
1. Client mengirim kontrol *TCP FIN* ke server
2. Server menerima *FIN*, dan membalas dengan *ACK* kemudian menutup koneksi dan mengirimkan *FIN* ke client.
3. Client menerima *FIN* dan membalas *ACK*
  - Masuk pada masa menunggu balasan *ACK* terhadap dari server
4. Server menerima *ACK* dan koneksi tertutup.

# SEGMENT TCP



- *Segmen-segmen TCP* akan dikirimkan sebagai *datagram-datagram IP*
- Sebuah *Segmen TCP* terdiri atas sebuah *Header TCP* dan segmen data aplikasi (*payload*), yang *dienkapsulasi* dengan menggunakan *header IP* dari Internet Protocol.

# HEADER TCP



# PORT TCP

---

- *Port TCP* mampu mengindikasikan sebuah lokasi tertentu untuk menyampaikan segmen-segmen TCP yang dikirimkan yang diidentifikasi dengan *TCP Port Number*.
- Nomor-nomor di bawah angka **1024** merupakan port yang umum digunakan dan ditetapkan oleh *IANA*

# TCP FLAG

---

<b>URG</b>
<b>ACK</b>
<b>PSH</b>
<b>RST</b>
<b>SYN</b>
<b>FIN</b>

- **URG** : Mengindikasikan bahwa beberapa bagian dari segmen TCP mengandung data yang sangat penting
- **ACK** : Mengindikasikan field Acknowledgment mengandung oktet selanjutnya yang diharapkan dalam koneksi
- **PSH** : Mengindikasikan bahwa isi dari TCP Receive buffer harus diserahkan kepada protokol lapisan aplikasi
- **RST** : Mengindikasikan bahwa koneksi yang dibuat akan digagalkan
- **SYN** : Mengindikasikan bahwa segmen TCP yang bersangkutan mengandung Initial Sequence Number (ISN).
- **FIN** : Menandakan bahwa pengirim segmen TCP telah selesai dalam mengirimkan data dalam sebuah koneksi TCP.

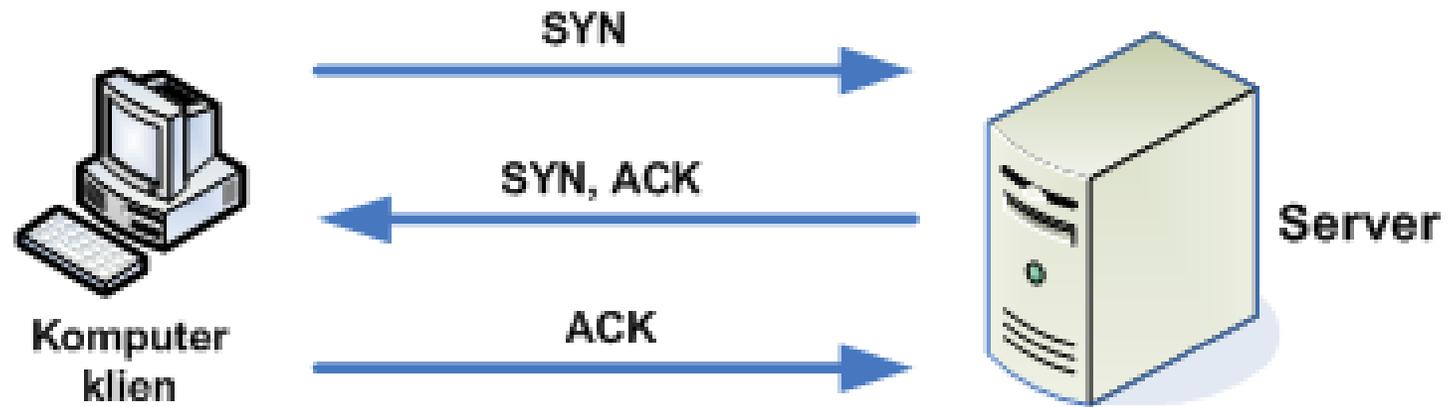
# TCP THREE-WAY HANDSHAKE

---

- Proses pembuatan koneksi TCP disebut juga dengan "Three-way Handshake".
- Tujuan metode ini adalah agar dapat melakukan *sinkronisasi* terhadap *nomor urut* dan *nomor acknowledgement* yang dikirimkan oleh *kedua pihak* dan *saling bertukar ukuran TCP Window*.

# TCP THREE-WAY HANDSHAKE

---



# PROSES TCP THREE-WAY HANDSHAKE

---

1. *Host pertama* (yang ingin membuat koneksi) akan mengirimkan sebuah *segmen TCP* dengan **flag SYN** diaktifkan kepada *host kedua* (yang hendak diajak untuk berkomunikasi).
2. *Host kedua* akan meresponsnya dengan mengirimkan segmen dengan *acknowledgment* dan juga **SYN** kepada *host pertama*.
3. *Host pertama* selanjutnya akan mulai saling **bertukar data** dengan *host kedua*.

# USER DATAGRAM PROTOCOL

---

- *UDP (User Datagram Protocol)* adalah *protokol lapisan transpor* yang mendukung *komunikasi yang tidak handal* dan *tanpa adanya koneksi terlebih dahulu*.
- UDP adalah protokol pada *layer transpor* yang bersifat *connectionless*.

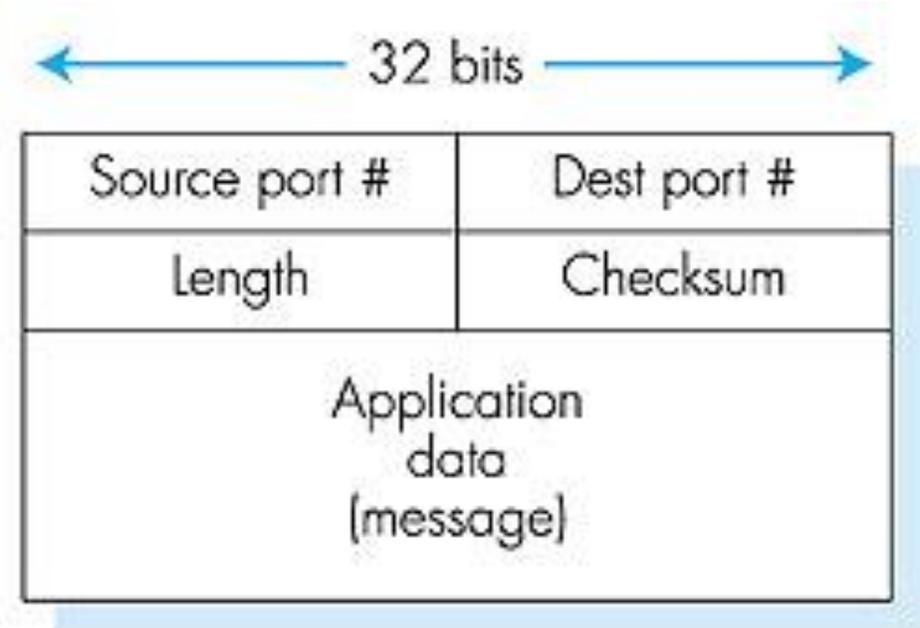
# USER DATAGRAM PROTOCOL

---

- UDP tidak mementingkan bagaimana *keadaan koneksi*, jadi jika terjadi pengiriman data maka *tidak akan dijamin sampai tidaknya data ke tujuan*.
- Pada UDP tidak ada *pemecahan data (segmentasi)*, oleh karena itu tidak dapat dilakukan pengiriman data dengan *ukuran yang besar*.

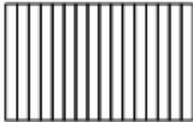
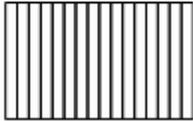
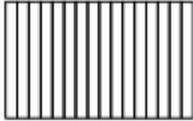
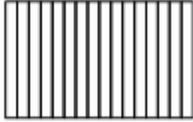
# SKEMA USER DATAGRAM PROTOCOL

- **Source port** = alamat host /port sumber
- **Destination port** = alamat server/ port tujuan
- **Length** = panjang total paket UDP
- **Checksum** = fungsi deteksi error sederhana



# HEADER UDP

---

Field		Panjang
Source Port		16 bit (2 byte)
Destination Port		16 bit (2 byte)
Length		16 bit (2 byte)
Checksum		16 bit (2 byte)

# SOURCE /DESTINATION PORT

---

- Port Sumber, dengan *panjang 16 bit*, jika host sumber sebagai *client*, kebanyakan *nomor port* ditentukan oleh *software UDP* yang berjalan di *host sumber*, namun jika *host sumber* sebagai *server*, Maka *nomor port* menggunakan *port yang umum digunakan*.
- Port Tujuan, *panjang 16 bit*, jika *host tujuan* adalah *server*, biasanya *nomor port* adalah yang *biasa digunakan*, jika *host tujuan* adalah *client*, *nomor port* adalah nomor yang *disalin* dari nomor port sementara yang diterima pada paket.

# LENGTH DAN CHECKSUM

---

- Panjang Data, sepanjang 16 bit menyatakan *panjang total user datagram* dan *header*.
- Sesungguhnya angka panjang user datagram di UDP *tidak terlalu penting* karena *user datagram terbungkus* dalam *IP datagram*, dan didalamnya sudah terdapat *definisi panjang totalnya*.
- Checksum, bagian ini digunakan untuk mengetahui adanya *error* pada user datagram.

# SIFAT – SIFAT UDP

---

1. **Connectionless** (tanpa koneksi)
2. **Unreliable** (tidak andal)

# CONNECTIONLESS DAN UNRELIABLE

---

- *Connectionless* adalah pesan-pesan UDP akan dikirimkan *tanpa harus dilakukan proses negosiasi koneksi antara dua host* yang hendak bertukar informasi.
- Pesan-pesan UDP akan dikirimkan sebagai *datagram tanpa adanya nomor urut atau pesan acknowledgment (ack)* sehingga sering disebut *Unreliable*.
- *Protokol lapisan aplikasi* yang berjalan di atas UDP harus melakukan *pemulihan terhadap pesan-pesan yang hilang selama transmisi*.

# PORT UDP

---

- UDP memiliki saluran untuk mengirimkan informasi antar host, yang disebut dengan *UDP Port*.
- Untuk menggunakan *protokol UDP*, sebuah aplikasi harus menyediakan *alamat IP* dan *nomor UDP Port* dari host yang dituju.

# FUNSI PORT UDP

---

- Sebuah UDP port berfungsi sebagai sebuah *multiplexed message queue*, yang berarti bahwa UDP port tersebut dapat menerima beberapa pesan secara sekaligus.
- Setiap port diidentifikasi dengan *nomor yang unik*, seperti halnya TCP, tetapi meskipun begitu, UDP Port berbeda dengan TCP Port meskipun memiliki nomor port yang sama.

# PORT UDP

Nomor Port UDP	Digunakan oleh
53	Domain Name System (DNS) Name Query
67	BOOTP client (Dynamic Host Configuration Protocol [DHCP])
68	BOOTP server (DHCP)
69	Trivial File Transfer Protocol (TFTP)
137	NetBIOS Name Service
138	NetBIOS Datagram Service
161	Simple Network Management Protocol (SNMP)
445	Server Message Block (SMB)
520	Routing Information Protocol (RIP)
1812/1813	Remote Authentication Dial-In User Service (RADIUS)

# TABEL WELL-KNOWN PORT TCP & UDP

---

Table- Well-known port assignments			
Protocol	Port	Protocol	Purpose
echo	7	TCP/UDP	Echo is a test protocol used to verify that two machines are able to connect by having one echo back the other's input.
discard	9	TCP/UDP	Discard is a less useful test protocol in which all data received by the server is ignored.
daytime	13	TCP/UDP	Provides an ASCII representation of the current time on the server.
FTP data	20	TCP	FTP uses two well-known ports. This port is used to transfer files.
FTP	21	TCP	This port is used to send FTP commands like <code>put</code> and <code>get</code> .
SSH	22	TCP	Used for encrypted, remote logins.
telnet	23	TCP	Used for interactive, remote command-line sessions.
smtp	25	TCP	The Simple Mail Transfer Protocol is used to send email between machines.

# TABEL WELL-KNOWN PORT TCP & UDP

---

Table- Well-known port assignments			
Protocol	Port	Protocol	Purpose
time	37	TCP/UDP	A time server returns the number of seconds that have elapsed on the server since midnight, January 1, 1900, as a four-byte, signed, big-endian integer.
whois	43	TCP	A simple directory service for Internet network administrators.
finger	79	TCP	A service that returns information about a user or users on the local system.
HTTP	80	TCP	The underlying protocol of the World Wide Web.
POP3	110	TCP	Post Office Protocol Version 3 is a protocol for the transfer of accumulated email from the host to sporadically connected clients.
NNTP	119	TCP	Usenet news transfer; more formally known as the "Network News Transfer Protocol".
IMAP	143	TCP	Internet Message Access Protocol is a protocol for accessing mailboxes stored on a server.
RMI Registry	1099	TCP	The registry service for Java remote objects.

# PERBEDAAN TCP & UDP

---

TCP	UDP
Reliable	Unreliable, cepat, dan Low Overhead
Connection-oriented	Connectionless
Acknowledgement	Tanpa Acknowledgement
Mengirim ulang data yang hilang	Tidak ada pengiriman ulang
Sequencing.	Tidak ada sequencing, data diberikan ke layer atas sesuai dengan datangnya data.
PDU disebut Segment	PDU disebut Datagram
Overhead 20 bytes	Overhead 8 bytes.
Web, email, file transfer	Video streaming, VoIP