

TCP Protocol

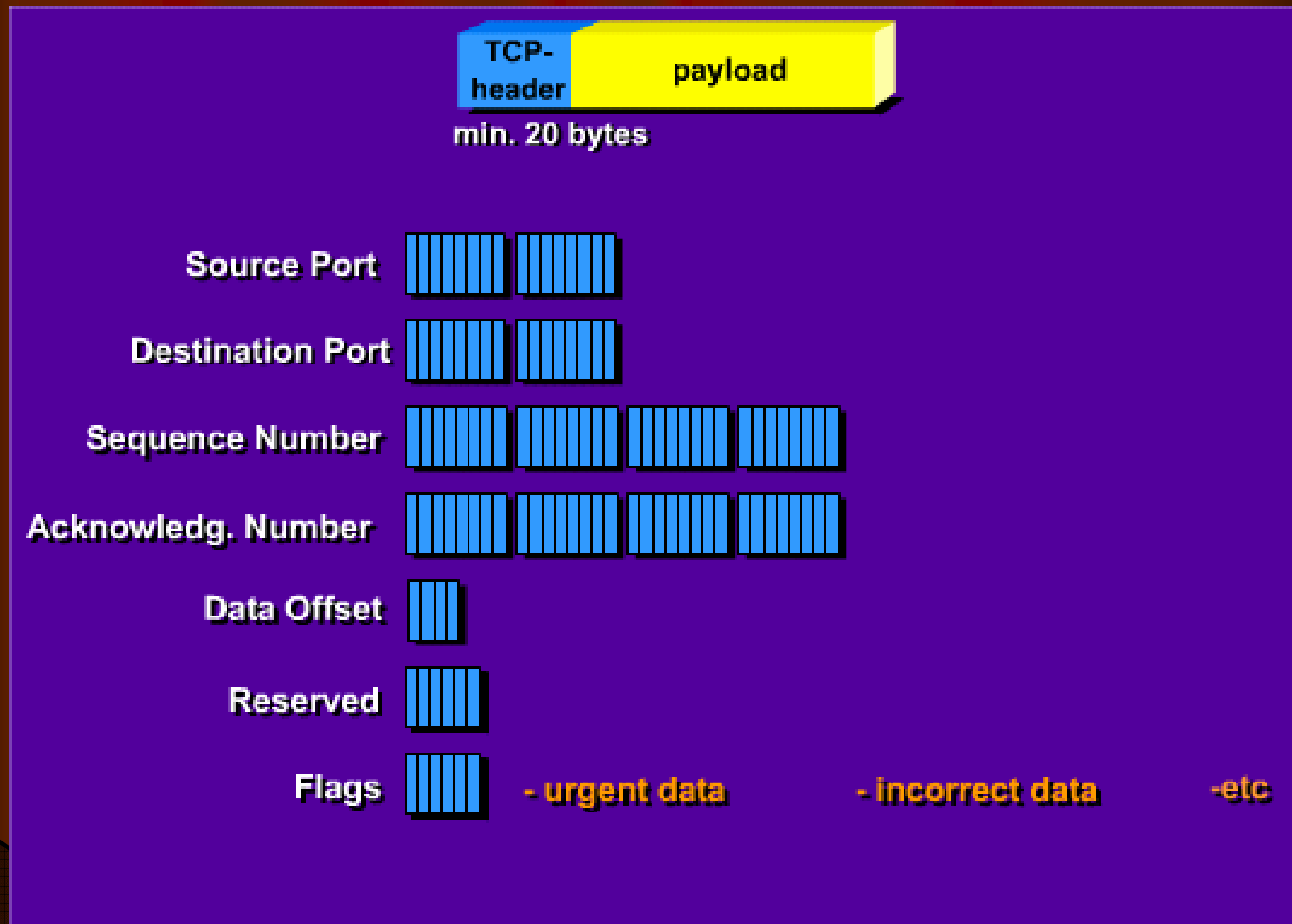
TCP Overview

- TCP merupakan protocol connection-oriented, yang artinya data hanya bisa ditransmisikan setelah ada proses negosiasi terlebih dahulu antara pengirim dan penerima
- Negosiasi diantaranya berupa : Berapa data yang bisa dikirim dalam satu waktu, nomor urut yang dipakai setiap pengiriman data dll.
- TCP biasanya merupakan komunikasi fully duplex, yang artinya Setiap host yang berkomunikasi mempunyai dua chanel logical untuk mengirim dan menerima message
- TCP Menyediakan transmisi data yang reliable, dengan cara.
 - Setiap paket data diberi sequence number, dan positive acknowledgement oleh receiver is expected, jika tidak harus retransmite data
 - Receiver akan membuang jika terjadi duplikasi data, dan resequences packets jika kedatangan tidak urut

Flow Control

- Selain itu, TCP mensupport Flow Control untuk menghindari terlalu banyak data yang dikirim pada satu waktu dan overload pada jaringan router
- Flow Control artinya harus ada kesepakatan berada besar data yang dikirim dalam satu waktu antara pengirim dan penerima.
- Flow Control mengindikasikan ukuran buffer penerima yang free yang bisa diisi dalam waktu tertentu

Struktur TCP



TCP Header

- TCP header panjangnya bervariasi. Panjang minimal 20 bytes. Terdiri dari 7 field : Source Port, Destination Port, Sequence Number, Ack. Number, Data Offset, Reserver dan Flag.
- 2 byte masing –masing untuk Source Port and the Destination Port. Sama seperti UDP.
- 4 byte sequence Number yang berisi nomor urut transmisi data dalam satu segment
 - Ini digunakan checking ketika semua byte telah diterima
- Acknowledgement Number terdiri dari 4 byte.
 - Berisi Sequence number berikutnya dari penerima
- Data Offset mengindikasikan awal data. Ini berhubungan dengan ukuran TCP header.
- Diikuti 6 bit reserve untuk penggunaan kedepan, diset 0.
- Flags menentukan tipe informasi pada segment.

Flag

Flags



Urgent		
Acknowledgement		!sending acknowledgement!
Push		!push function!
Reset		!connection reset!
Synchronization		!connection set-up!
Final		!end of the byte stream!

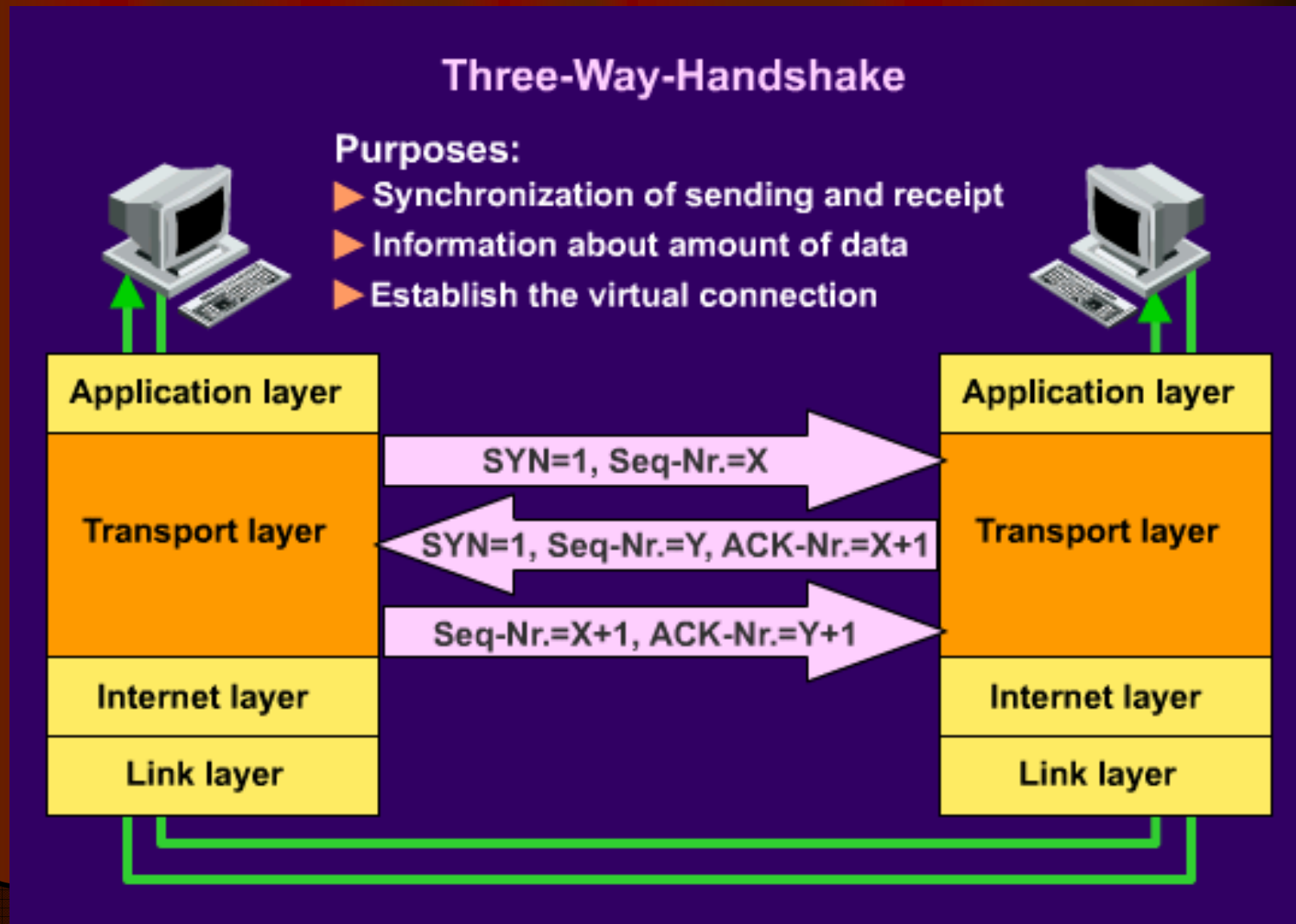
Tahapan Komunikasi pada TCP

- TCP adalah Protocol connection-oriented.
- Sebelum data ditransmisikan, koneksi yang dibuat bisa diset atau dirubah sesuai keadaan.
- Tiga tahap komunikasi dahulu connection set-up
 - data transfer
 - connection release
- Data yang ditransmisikan bisa dipakai untuk tiga keadaan tadi

Three Way Handshake

- Koneksi TCP diawali oleh prosedur yang biasa disebut dengan Three-Way-Handshake.
- Tujuannya untuk melakukan sinkronisasi antara pengirim dan penerima. Hal yang diinformasikan selama Three Way Handshake adalah Jumlah data yang bisa ditransmisikan dalam satu waktu, Sequence number yang dipakai.
- Untuk setup koneksi, host melakukan session inisialisasi dengan menset flag sinkronisasi ke 1.
- Segment juga berisi sequence number yang mengindikasikan awal byte yang ingin dikirim berikutnya. Juga berisi acknowledgement yang terdiri dari sequence number berikutnya untuk menerima data.
- Setelah Three Way Handshake dilakukan baru dianggap session established, dan koneksi dua arah siap dilaksanakan

Three Way Handshake



3 Way Handshake

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

```

Packet 1: source: 130.57.20.10  dest.:130.57.20.1
TCP: ----- TCP header -----
      TCP: Source port              = 1026
      TCP: Destination port         = 524
      TCP: Initial sequence number = 12952
      TCP: Next expected Seq number= 12953
      TCP:      .... ..1. = SYN
      TCP: Window                   = 8192
      TCP: Checksum                  = 1303 (correct)
      TCP: Maximum segment size     = 1460 (TCP Option)

```

```

Packet 2: source: 130.57.20.1  dest: 130.57.20.10
TCP: ----- TCP header -----
      TCP: Source port              = 524
      TCP: Destination port         = 1026
      TCP: Initial sequence number = 2744080
      TCP: Next expected Seq number= 2744081
      TCP: Acknowledgment number   = 12953
      TCP:      .... ..1. = SYN
      TCP: Window                   = 32768
      TCP: Checksum                  = D3B7 (correct)
      TCP: Maximum segment size     = 1460 (TCP Option)

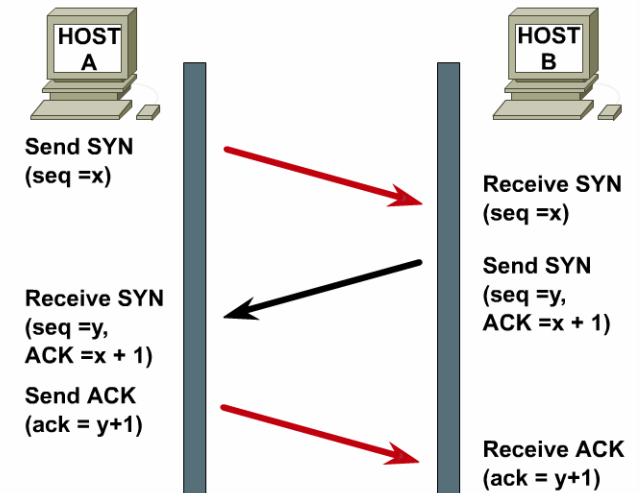
```

```

Packet 3: source: 130.57.20.10  dest: 130.57.20.1
TCP: ----- TCP header -----
      TCP: Source port              = 1026
      TCP: Destination port         = 524
      TCP: Sequence number          = 12953
      TCP: Next expected Seq number= 12953
      TCP: Acknowledgment number   = 2744081
      TCP:      ...1 .... = Acknowledgment
      TCP: Window                   = 8760
      TCP: Checksum                  = 493D (correct)
      TCP: No TCP options

```

TCP Three-Way Handshake/ Open Connection



- Maksimum ukuran segment dan ukuran windows yang dinegosiasikan juga dikirim

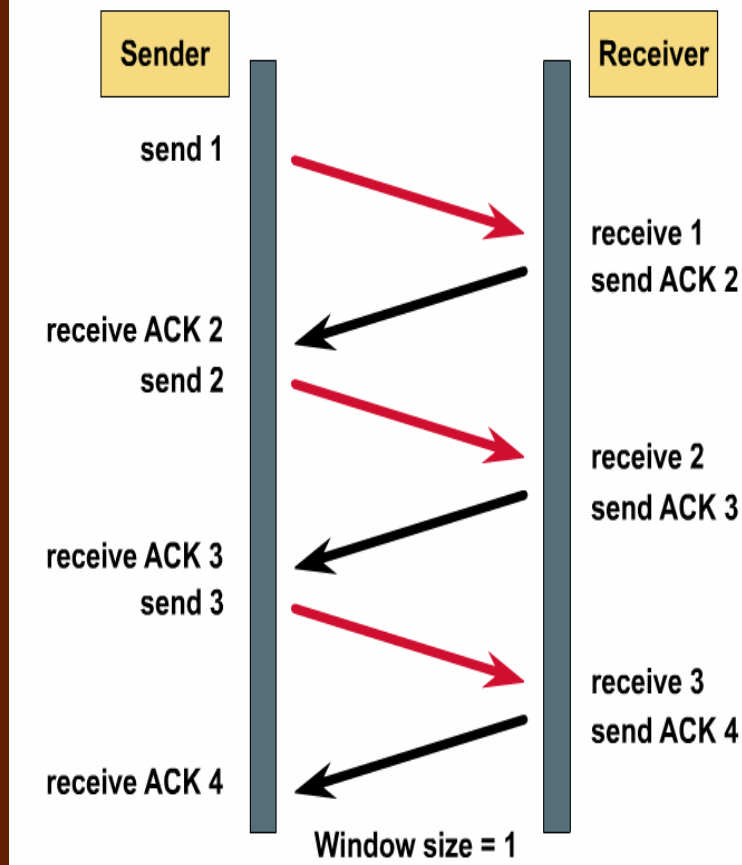
*

PAR

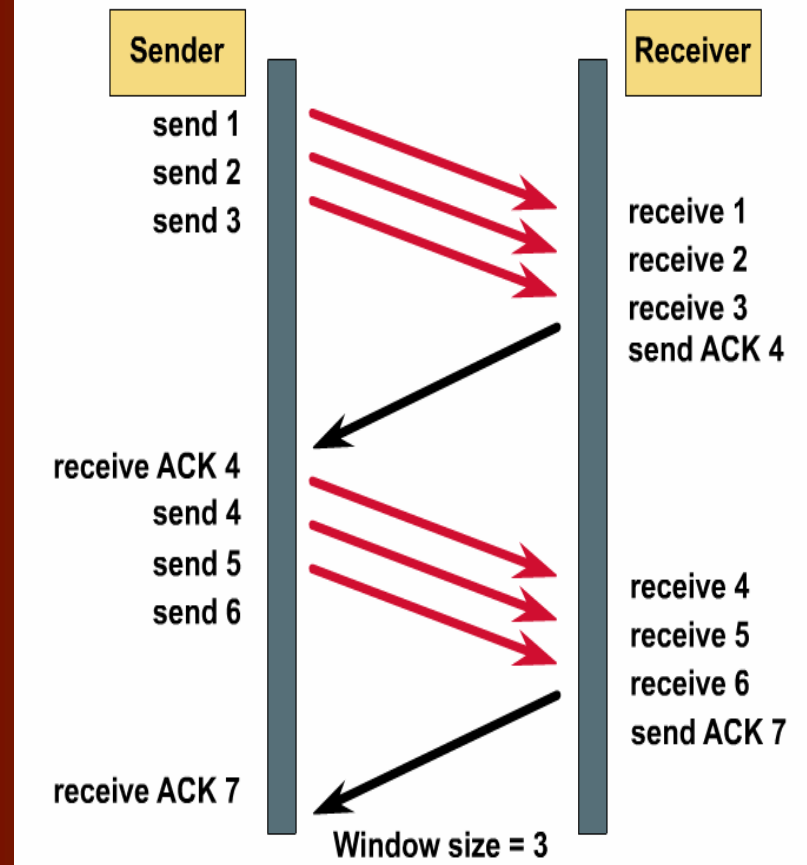
- Setelah koneksi established, ini harus dimantain sampai salah satu partner komunikasi ingin mengakhiri komunikasi. System Transfer Data didasarkan pada mekanisme PAR - Positive Acknowledgement with Retransmission. Yang artinya bahwa untuk kebenaran data yang diterima maka penerima data harus mengirimkan acknowledgement ke pengirim.
- Untuk efisiensi, acknowledgements hanya berisi paket selanjutnya yang harus dikirim, tidak untuk setiap individu paket

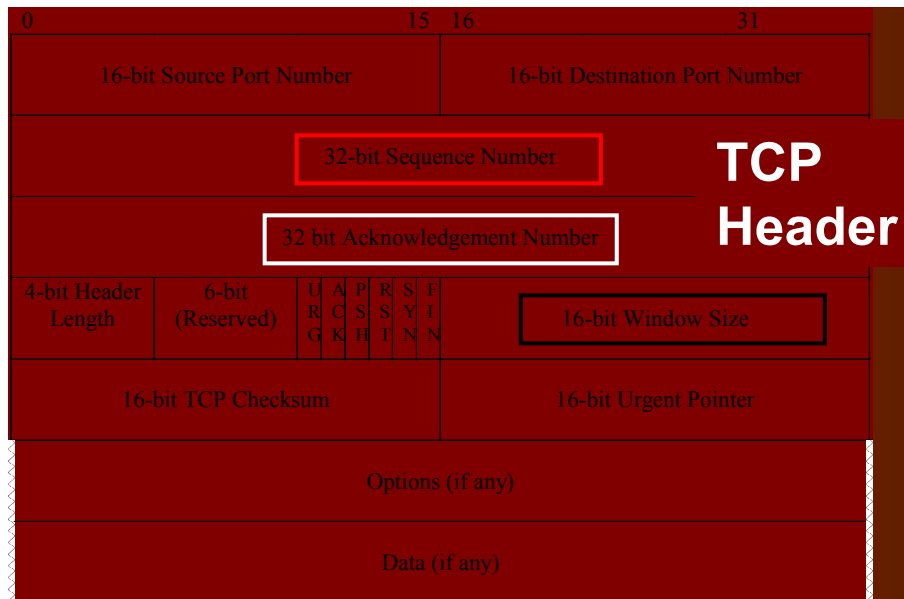
PAR

TCP Simple Acknowledgment

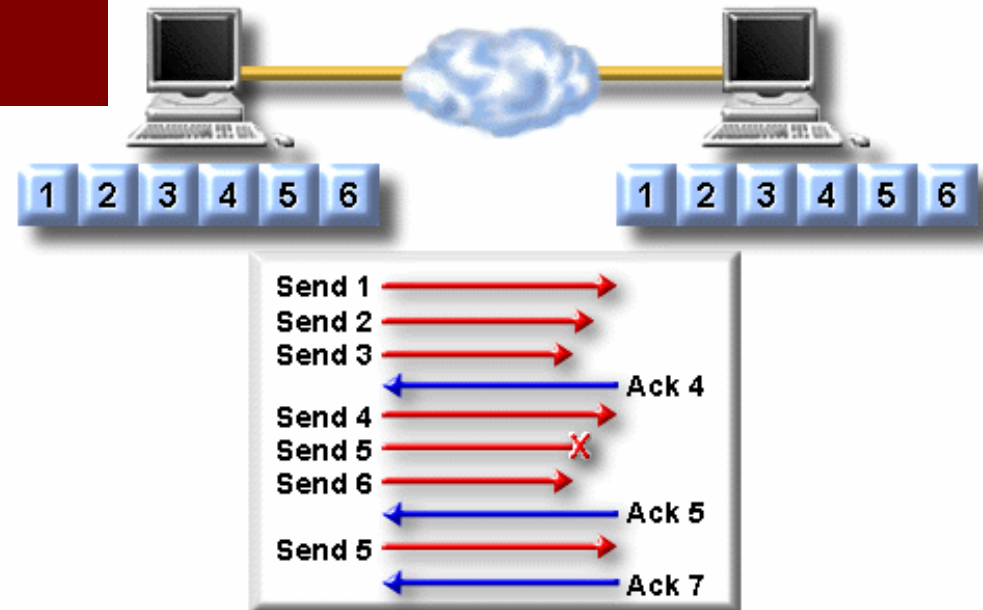


TCP Sliding Window





An Acknowledgment Technique

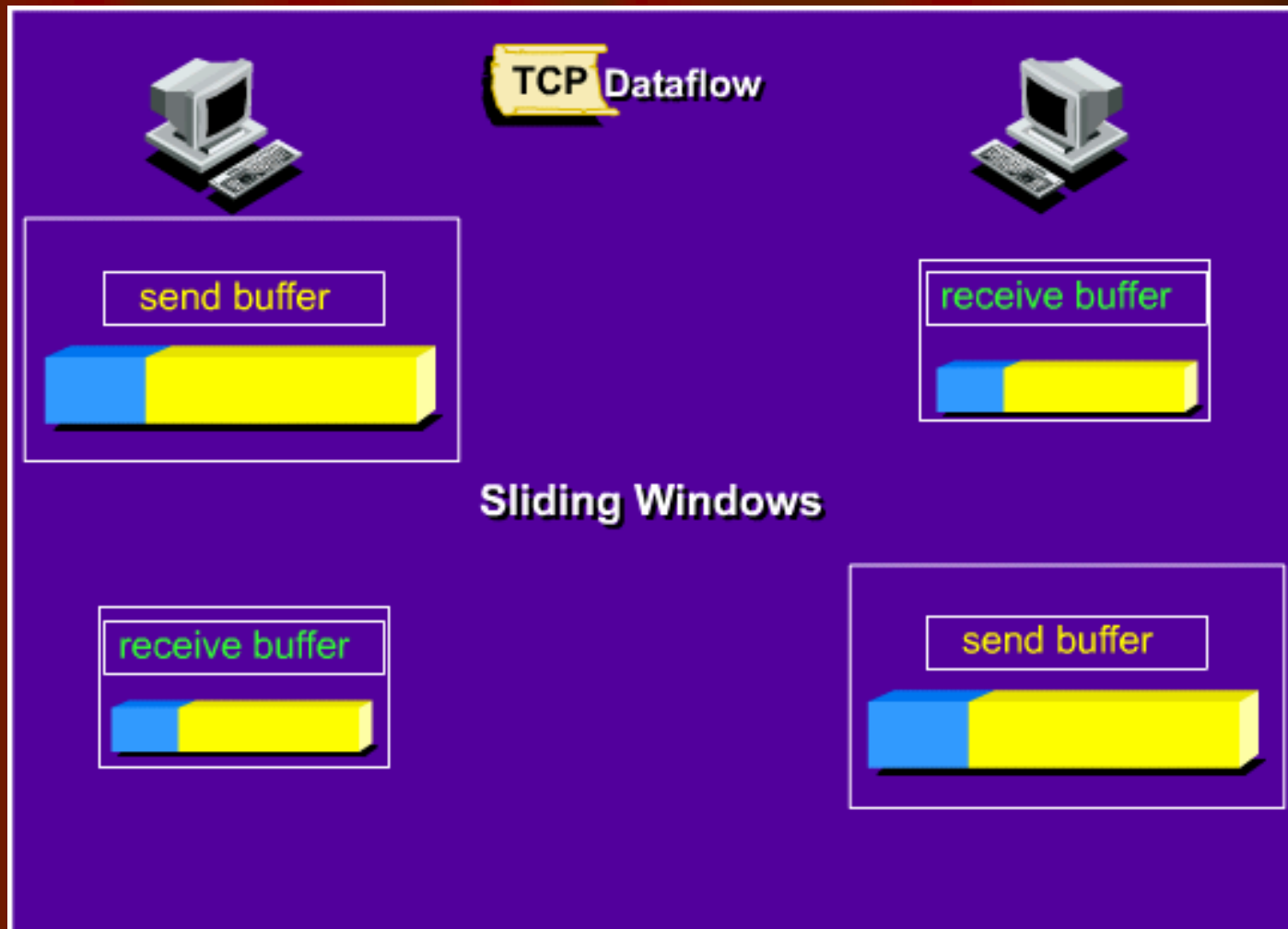


- Paket mungkin didrop sepanjang jalan, time out atau rusak
- Jika misal 4, 5, dan 6 dikirim, tapi 5 lost, receiver hanya akan memberi ack sampai 4, mengirim ack 5.
- Pengirim akan mengirim ulang paket 5 dan menunggu untuk mendengar dari penerima paket mana yang selanjutnya dikirim
- Receiver mengirim Ack 7, jadi pengirim tahu dapat memulai lagi mengirim paket ke-7 dan seterusnya

Sliding Window

- Untuk melakukan transmisi data, penerima menyiapkan buffer, untuk mekanisme ini TCP menggunakan mekanisme sliding windows. Setiap host mempunyai akses ke dua windows: satu mengirim data dan yang lain menerima data. Ukuran windows mengindikasikan jumlah buffer yang disiapkan untuk data

Sliding Window



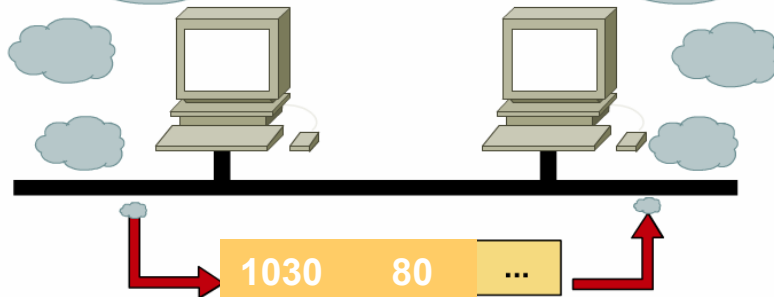
Contoh Penggunaan Transport Layer

TCP/UDP Port Numbers

Source Port	Destination Port	...
-------------	------------------	-----

http to
www.cisco.com

Dest. Port = 80 Send
packets to web
server application



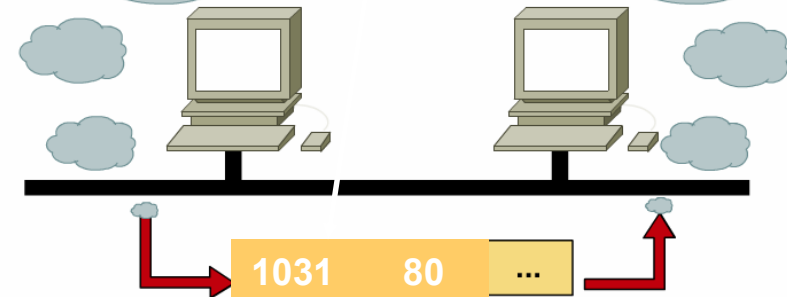
Netscape Navigator



Pada setiap session http antara client dan server yang sama, mempunyai destination port yang sama, tapi berbeda Source Port (unik), untuk mengidentifikasi setiap session sehingga pengembalian permintaan masuk ke sesi yang benar

http to
www.cisco.com

Dest. Port = 80 Send
packets to web
server application

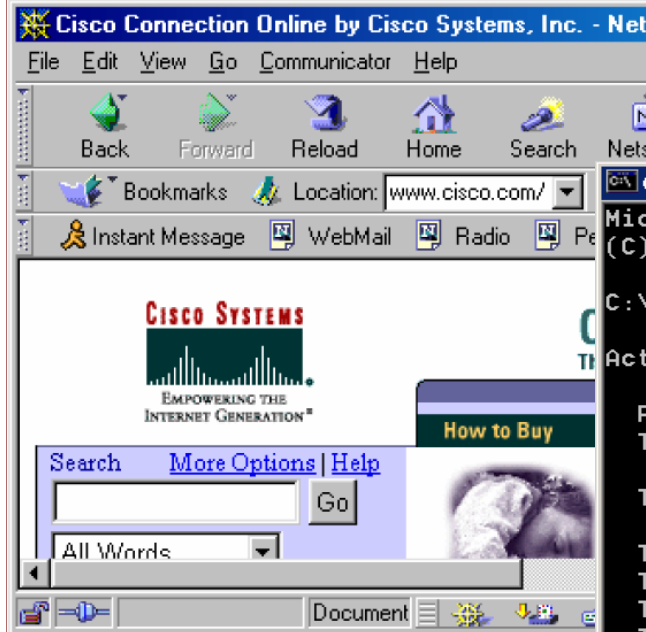


Netscape Navigator



Ini menunjukkan contoh dua browser windows dengan URL yang sama.
TCP/IP menggunakan source port numbers untuk pengembalian informasi

Netscape Navigator



TCP or
UDP

Source IP

Destination IP

Connection State

Source Port

Destination Port

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>netstat

Active Connections

Proto Local Address Foreign Address State
TCP RICK-GRAZIANI:1430 RICK-GRAZIANI.cabrillo-lab.net:1431 ESTABLISHED
TCP RICK-GRAZIANI:1431 RICK-GRAZIANI.cabrillo-lab.net:1430 ESTABLISHED
TCP RICK-GRAZIANI:1414 65.104.127.219:http ESTABLISHED
TCP RICK-GRAZIANI:1432 boris.cabrillo.cc.ca.us:imap ESTABLISHED
TCP RICK-GRAZIANI:1919 216.239.37.126:http CLOSE_WAIT
TCP RICK-GRAZIANI:1920 216.239.37.126:http CLOSE_WAIT
TCP RICK-GRAZIANI:1937 216.239.37.126:http CLOSE_WAIT
TCP RICK-GRAZIANI:1938 216.239.37.126:http CLOSE_WAIT
TCP RICK-GRAZIANI:1975 208.44.195.174:http TIME_WAIT
TCP RICK-GRAZIANI:1977 www.cisco.com:http ESTABLISHED
TCP RICK-GRAZIANI:1978 www.cisco.com:http ESTABLISHED
TCP RICK-GRAZIANI:1979 www.cisco.com:http ESTABLISHED
TCP RICK-GRAZIANI:1980 www.cisco.com:http ESTABLISHED
TCP RICK-GRAZIANI:1985 cco-sj-1.cisco.com:http ESTABLISHED
TCP RICK-GRAZIANI:1986 63.209.4.18:http TIME_WAIT

C:\>
```

www.google.com

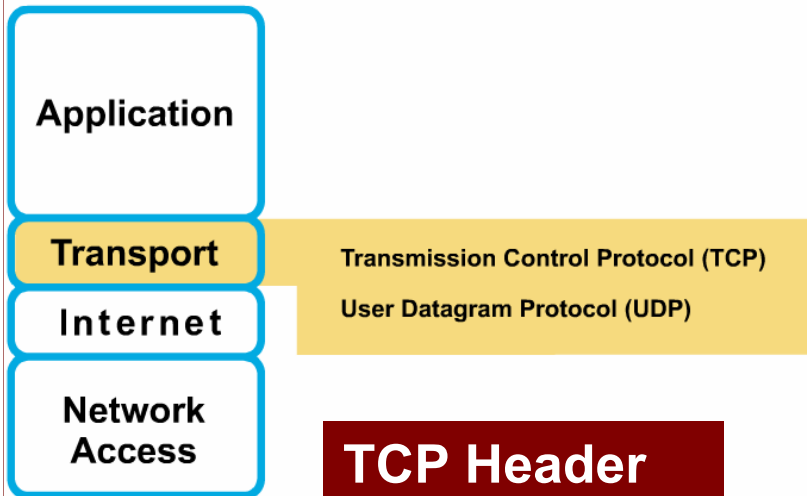
www.cisco.com

netstat command

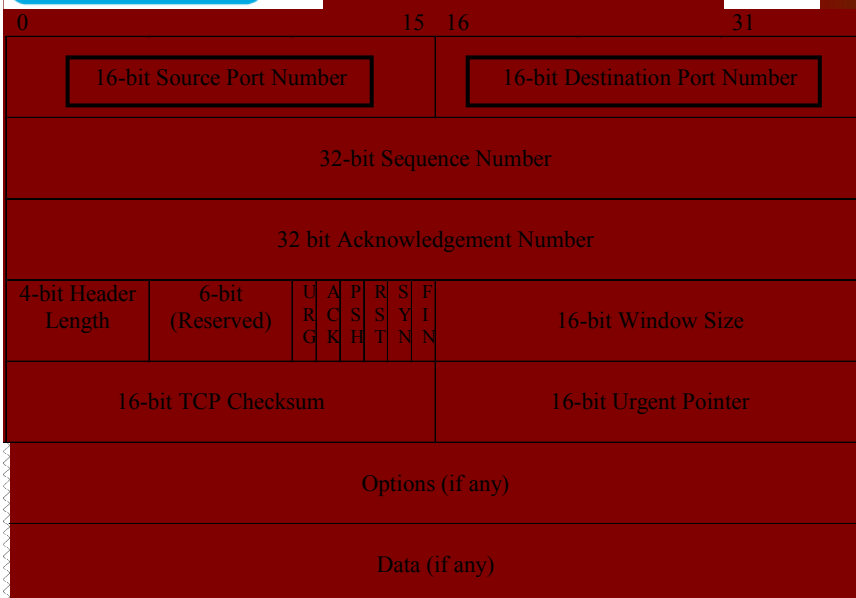
- **Aktualnya**, ketika kita membuka sebuah halaman html, maka session TCP dibangun.
- Jika membuka beberapa halaman web maka multiple koneksi TCP dibentuk, setiap koneksi di client akan membentuk port yang berbeda-beda, dengan port tujuan sama.

Hubungan antara Aplikasi, Port dan Transport Layer

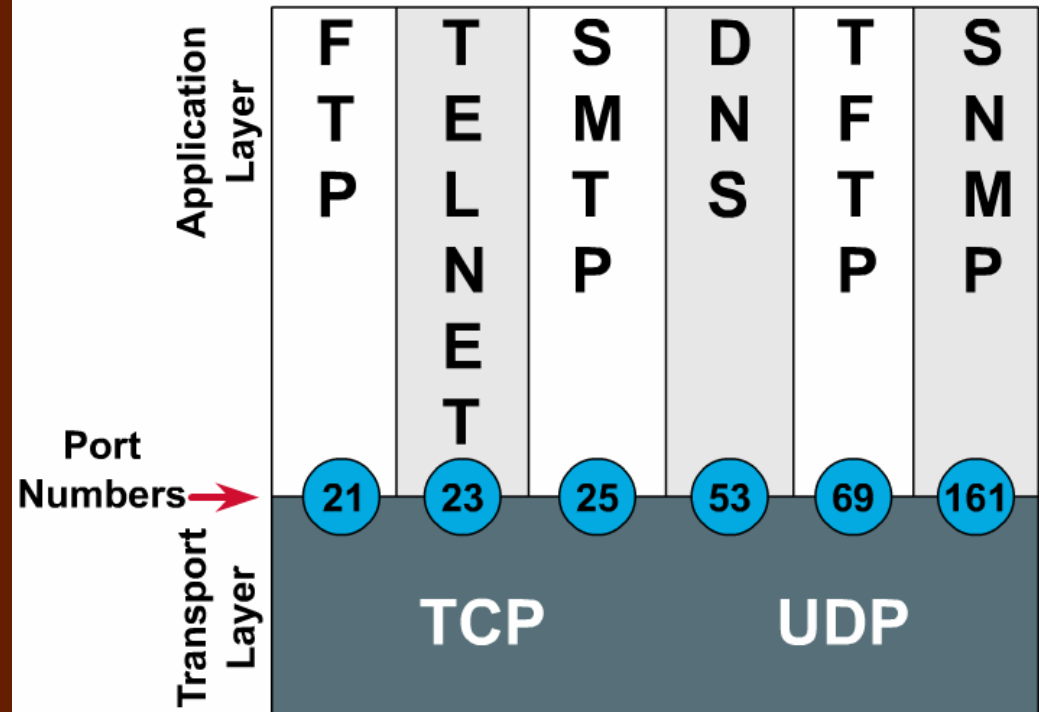
Transport Layer Overview



TCP Header



Port Numbers



- **Keduanya TCP dan UDP** menggunakan Port untuk meneruskan informasi ke layer di atasnya

The Protocol Field

