

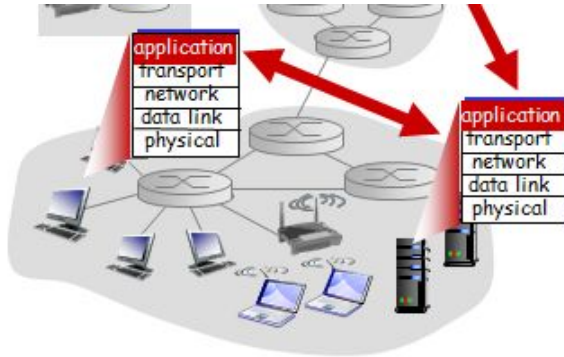
Week 3

Muhammad Michael Maulana - 1806191181 - Jarkomdat A

Principles of network applications & Web and HTTP

Cara membuat network apps:

Membuat program yang bekerja pada end systems, bisa berinteraksi dalam satu jaringan.
Network core device tidak menjalankan applications



Application architectures

- Client-server
Client: berkomunikasi dengan server, memiliki dynamic IP address, tidak berkomunikasi secara langsung dengan client lain.
Server: always-on host, permanent IP address
- Peer-to-peer (P2P)
Self scalability

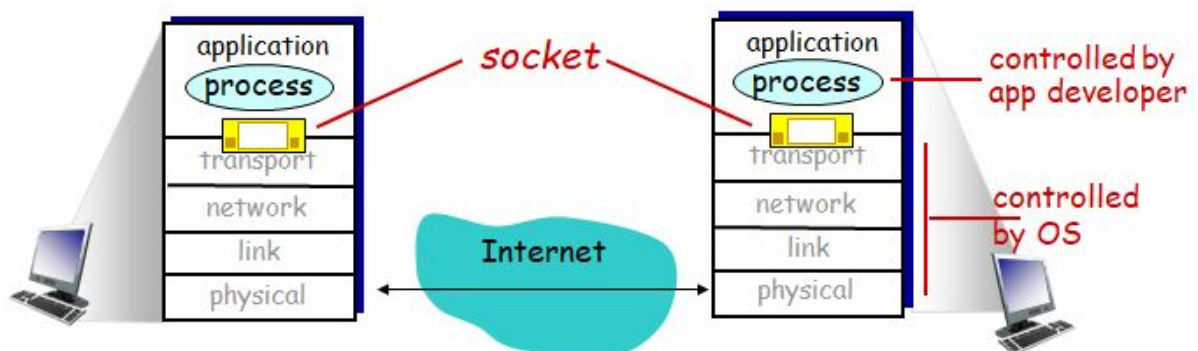
Processes communicating

Process adalah program yang berjalan dalam suatu host

- Client process: menginisiasi komunikasi
- Server process: menunggu untuk dikontak

Socket

Process mengirimkan/menerima pesan ke/dari socket.



Untuk menerima pesan dibutuhkan identifier yang menginclude IP address dan port numbers.

Contoh port numbers:

- HTTP server: 80
- mail server: 25

Beberapa hal yang harus diperhatikan/requirements saat transport service:

1. Data integrity
2. Timing
3. Throughput
4. Security

TCP vs UDP

	<u>application</u>	<u>application layer protocol</u>	<u>underlying transport protocol</u>
remote terminal access	e-mail	SMTP [RFC 2821]	TCP
	Web	Telnet [RFC 854]	TCP
	file transfer	HTTP [RFC 2616]	TCP
streaming multimedia		FTP [RFC 959]	TCP
		HTTP (e.g., YouTube), RTP [RFC 1889]	TCP or UDP
Internet telephony		SIP, RTP, proprietary (e.g., Skype)	TCP or UDP

Kesimpulan: TCP sebaiknya digunakan jika kita menginginkan data integrity, reliable, dan tidak time sensitive, sebaliknya, gunakan UDP untuk file transfer yang menitikberatkan pada kecepatan transfer, bukan pada kualitas file.

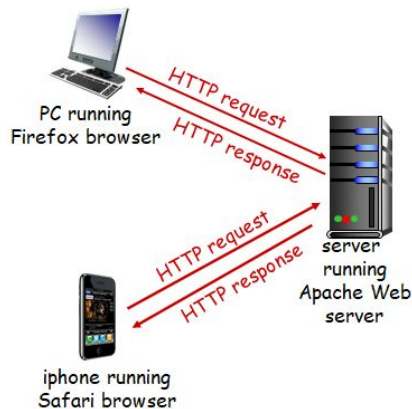
SSL

SSL dapat mengenkripsi TCP, memproviden data integrity, berjalan pada application layer.

Web and HTTP

HTTP: hypertext transfer protocol

Merupakan web's application layer protocol



Jalan kerjanya menggunakan TCP:

Client menginisiasi TCP (membuat socket), server menerima TCP connection dari client, terjadi pertukaran HTTP message antara browser dan web server, kemudian TCP ditutup.

Sifat dari HTTP adalah stateless, yang artinya tidak ada client requests yang disimpan pada server.

HTTP connections

- Non-persistent HTTP

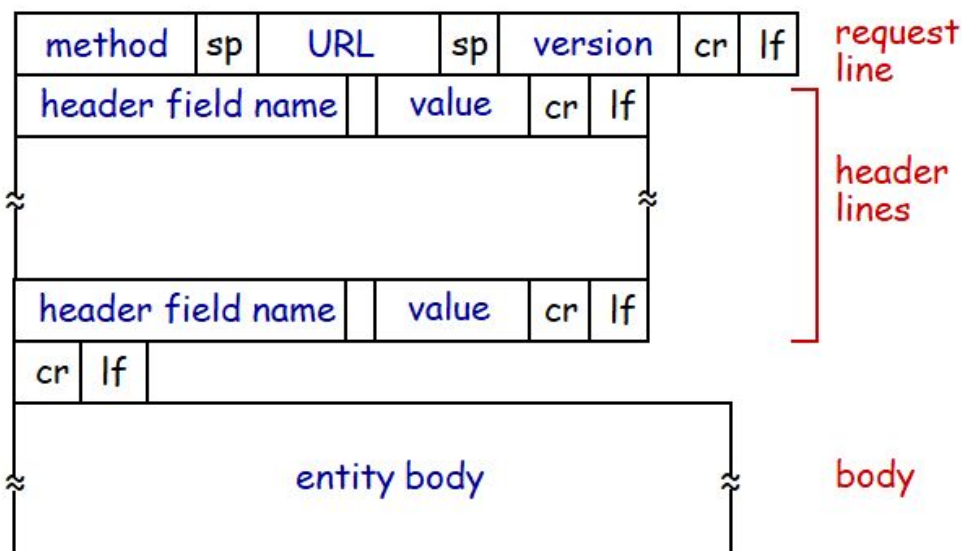
Hanya satu object yang dikirimkan pada satu TCP connection. Untuk mendownload banyak object, dibutuhkan multiple connection.

non-persistent HTTP response time = 2RTT+ file transmission time

- Persistent HTTP

Multiple objects bisa dikirimkan dengan single TCP connection antara client dengan server.

General format dari HTTP



HTTP response status code

- 200 OK
request succeeded, requested object later in this msg
- 301 Moved Permanently
requested object moved, new location specified later in this msg (Location:)
- 400 Bad Request
request msg not understood by server
- 404 Not Found
requested document not found on this server
- 505 HTTP Version Not Supported

Cookies

Keeping “state”

When initial HTTP requests arrives at site, site creates: unique ID, and entry in backend database for ID

Cookies bisa digunakan untuk:

- authorization
- shopping carts
- recommendations
- user session state (Web e-mail)

Web caches

Web caching bisa mengurangi client request response time

FTP: the file transfer protocol

transfer file dari/ke remote host

client: menginitiates transfer

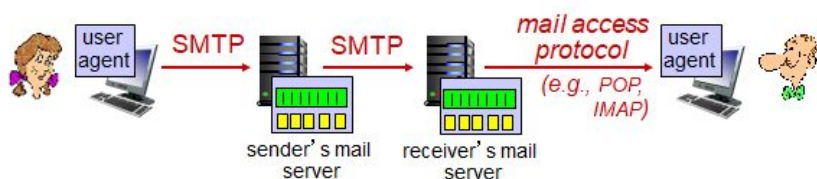
server: remote host

Electronic mail

Three major components:

- user agents
composing, editing, reading mail messages
- mail servers
SMTP protocol between mail servers to send email messages
client: sending mail server
server: receiving mail server
- simple mail transfer protocol: SMTP
multiple objects sent in one msg

Mail Access Protocol



DNS: domain name system

Manusia menggunakan nama, komputer menggunakan angka, dan DNS berada di antara mereka untuk menyesuaikan nama dengan angka dalam daftar tertentu

(<https://www.hostinger.co.id/tutorial/apa-itu-dns/>)