DIS - Name Services & TCP/ UDP

TCP

1. I want to establish a session
2. Okay, send me some data
3. Sends data.
4. Receiver confirms the data that is received
5. Sender resends data that wasnt confirmed

- Syn, Syn / Ack, Ack

Hej server jeg vil gerne snakke med dig, serveren svarer tilbage ”jeg hører, at du gerne vil snakke med mig”. Så siger clienten ”jeg er klar til at snakke med dig”.

Både klient og server checker om man får alle pakkerne – checksum tjekker man får hele pakken.

Datasikker, idet man ved, at data kommer frem, ellers kommer der en fejlmeddelelse

Use of ports (BEGGE)
The first characteristic to note is that, whereas IP supports communication between pairs of computers (identified by their IP addresses), TCP and UDP, as transport protocols, must provide process-to-process communication. This is accomplished by the use of ports. Port numbers are used for addressing messages to processes within a particular computer and are valid only within that computer. A port number is a 16-bit integer. Once an IP packet has been delivered to the destination host, the TCP- or UDP-layer software dispatches it to a process via a specific port at that host

TCP provides a much more sophisticated transport service. It provides reliable delivery of arbitrarily long sequences of bytes via stream-based programming abstraction. The reliability guarantee entails the delivery to the receiving process of all of the data presented to the TCP software by the sending process, in the same order. TCP is connection-oriented. Before any data is transferred, the sending and receiving processes must cooperate in the establishment of a bidirectional communication channel. The connection is simply an end-to-end agreement to perform reliable data transmission; intermediate nodes such as routers have no knowledge of TCP connections, and the IP packets that transfer the data in a TCP transmission do not necessarily all follow the same route

- Sequencing
- Flow control
- Retransmission
- Buffering
- Checksum

sequence number, der sørger for at holde styr på pakkens nummer i rækken

DNS - Domain Name Service

Lokal → Name Server → TLD(top level domain (fx .com)) → Root

- En DNS fungerer i forskellige trin – her kommer cachen virkelig ind
- Først spørger vi vores lokale DNS server – altså vores lokale cache – vores computer kan måske huske hvor facebook.dk er.
- Dernæst spørger vi Name server – den ved vi ikke hvor er. Det første jeg vil kigge på, er .com hvis jeg vil kigge på facebook.
- Root serverne er dem der indeholder toplevel domains – fx .dk
- Hvis jeg skal finde et dk domæne, så vil jeg starte med at spørge hvem indeholder dk domæner
- Så spørger man DK hostmaster – DK Hostmaster vil så spørge ”hvem holder styr på dk domæner er henne?”
- Man kan skifte sit Name Server – det gør man hos DK Host Master. Hvad fortæller DK hostmaster mig om, hvis jeg spørger om cbs.dk – de fortæller hvem der holder styr på.
- Kender jeg ikke adressen så spørger jeg internettets root server, så går jeg ned på top level domæne, hvem holder styr på det her domæne.
- TDC ville have sine egne DNS service der holder styr på nogle af de her ting…..

UDP

Buffering – så står vi og henter nogle pakker, så vi har et lager.

Vi er ligeglade med om vi viser hvor varmt det var på ét sekund. Derimod et SOS signal mere vigtigt, og man ikke kan misse.

UDP er ikke sikker

Computerspil, telefonsamtaler, går hurtigere

UDP is almost a transport-level replica of IP. A UDP datagram is encapsulated inside an IP packet. It has a short header that includes the source and destination port numbers (the corresponding host addresses are present in the IP header), a length field and a checksum. UDP offers no guarantee of delivery. We have already noted that IP packets may be dropped because of congestion or network error. UDP adds no additional reliability mechanisms except the checksum, which is optional. If the checksum field is non-zero, the receiving host computes a check value from the packet contents and compares it with the received checksum; packets for which they do not match are dropped

Thus UDP provides a means of transmitting messages of up to 64 kbytes in size (the maximum packet size permitted by IP) between pairs of processes (or from one process to several in the case of datagrams addressed to IP multicast addresses), with minimal additional costs or transmission delays above those due to IP transmission. It incurs no setup costs and it requires no administrative acknowledgement messages. But its use is restricted to those applications and services that do not require reliable delivery of single or multiple messages

Use of UDP • For some applications, it is acceptable to use a service that is liable to occasional omission failures. For example, the Domain Name System, which looks up DNS names in the Internet, is implemented over UDP. Voice over IP (VOIP) also runs over UDP. UDP datagrams are sometimes an attractive choice because they do not suffer from the **overheads** associated with guaranteed message delivery. There are three main sources of overhead:
• the need to store state information at the source and destination;
• the transmission of extra messages;
• latency for the sender

URI & URL

URI
A Uniform Resource Identifier (URI) is a compact sequence of characters that identifies an abstract or physical resource.

URL
A URI can be further classified as a locator, a name, or both. The term “Uniform Resource Locator” (URL) refers to the subset of URIs that, in addition to identifying a resource, provide a means of locating the resource by describing its primary access mechanism (e.g., its network “location”)

Se billede for URL dele

Protokol □ HTTP (port 80), HTTPS (port 443), FTP Subdomain □ CBS har lejet cbs.dk – man kan ikke købe danske domænenavne, man kan leje dem. .dk □ Toplevel domain – hver af disse har en organisation bag sig, som sørger for at fordele disse navn 80 □ Port – det behøves vi normalt ikke taste i vores navne, fordi den ud fra protokollen kan identificere porten

Path har ikke længere relevans for dns men har relevans for hvor jeg finder min ressource.

Query = id=2 – en måde at sende parametre på. Det betyder i dette tilfælde at det er kursus nummer 2.

Fragment - # - typisk brugt til et internt link på siden – fx wiki

Det forventes at de ud fra en URL hvilke elementer der indgår i dem. Der kan stå flere ting i query – der kan man blive ved med at skrive ting. På learn vil man også kunne skrive &name=hopsa fx.

Firewalls

The purpose of a firewall is to monitor and control all communication into and out of an intranet. A firewall is implemented by a set of processes that act as a gateway to an intranet (Figure 3.21a), applying a security policy determined by the organization.
The aims of a firewall security policy may include any or all of the following:

- Service control
- Behaviour control
- User control

Der findes to typer firewalls, henholdsvis **network level** og **application level** firewalls. Den ene er en boks, man sætter stik ind i, den anden er noget, som man installerer på sin computer.

Typisk så kan de spørge – hvordan virker en firewall? En firewall virker sådan at man kan blokere på forskellige ting. Selvfølgelig kan man sætte en firwall op der siger at man ikke kan modtage trafik fra denne her specifikke ip adresse. Man kan også sige som bank ”vi er en skandi bank, og vi vil ikke modtage trafik fra rusland”
Eller vælge mellem porte

Packet inspection

Man kan lave en packet inspection hvor man går ned i pakker og ser hvad de indeholder. Her ser man om pakken indeholder noget man ikke vil have, gør de der så smider man pakken ud.