



IT för personligt arbete F2

Nätverk och Kommunikation

DSV Peter Mozelius



Kommunikation i nätverk

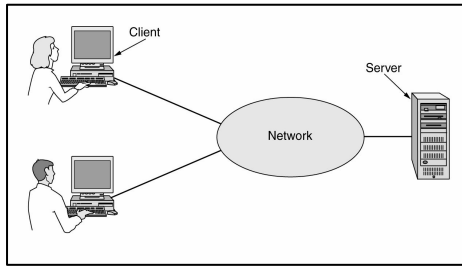
- *The Network is the Computer*
- Allt fler datorer är sammankopplade i olika typer av nätverk
- En dators funktionalitet är beroende av dess förmåga att kommunicera med andra datorer



Några definitioner

- Nätverk
 - En samling enheter (t ex datorer) som samarbetar och utbyter information
- Protokoll
 - En samling regler för kommunikation mellan olika enheter (t ex datorer)
- Klient - Server
 - En klient är en dator som använder de tjänster som erbjuds av en server

Klienter och servrar



Olika typer av kommunikation

Man kan klassificera kommunikation efter en tidskala:

- **Synkron kommunikation.** Både sändare och mottagare är aktiva samtidigt, t ex telefonsamtal
- **Asynkron kommunikation.** Sändare och mottagare är aktiva vid olika tidpunkter, t ex en telefonsvarare.

Olika typer av kommunikation

Man kan också klassificera kommunikation efter hur mottagarna kontaktas:

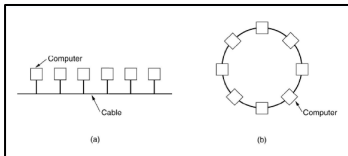
- **point-to-point.** En sändare skickar information till en mottagare, t ex ett telefonsamtal
- **broadcast.** En sändare skickar till många (okända) mottagare, t ex radioutsändning
- **multicast.** En sändare skickar till många (utvalda) mottagare, t ex pay-per-view

Olika typer av nätverk

- o LAN **Local Area Network**
 - När datorerna befinner sig så nära varandra att de kan kopplas ihop med en tunn kabel
 - Den vanligaste tekniken här är **Ethernet**
- o WAN **Wide Area Networks**
 - Nätverk som sträcker sig över stora områden
 - **Internet** är en samling WAN

LAN-topologier

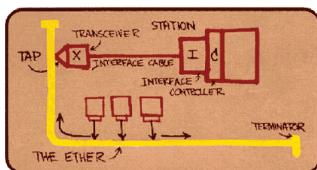
- o Det viktiga är att alla kan prata med alla



Vanligast är bussnätverk (a) med Ethernet

LAN med Ethernet

- o Robert Metcalfes ursprungsskiss (1973)



En vidareutveckling av ALOHANET

LAN med Ethernet

- Ethernets "partyprotokoll"
 - En grupp av människor på ett party turas om att berätta roliga historier
 - När någon berättar en historia så lyssnar de övriga i gruppen utan att avbryta
 - En kort paus efter varje klar historia och sedan går ordet över till nästa berättare
 - Om två personer börjar berätta exakt samtidigt så körs slumpad omstart

Att hantera kollisioner

- Låt alla sända när de vill och skicka om det som kolliderar (Ethernet)
- Ge alla en viss tid när de får sända
- Låt datorerna skicka runt rätten att sända till varandra
- Låt datorerna innan sändning reservera den gemensamma kanalen

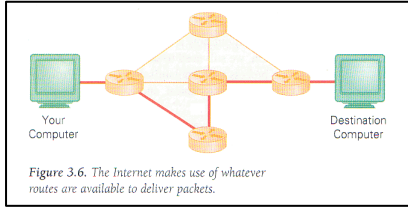
Rast 15 min?

Internet

- En uppfinning som har förändrat vår infrastruktur på ett flertal sätt
- En fantastisk teknik som kan användas på både bra och mindre bra sätt
- Vår globala gemensamma kunskapsbank
- Ursprunget finns i ARPA-net, det första paketförmedlande datornätverket

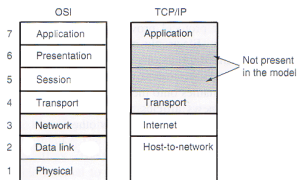
Internet

- o Point-to-Point med alternativa vägval

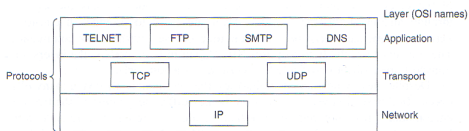


Internet och nätverkslager

Internet jobbar med en TCP/IP-stack



Protokoll i TCP/IP-stacken



TCP och UDP, två olika sätt att skicka paket

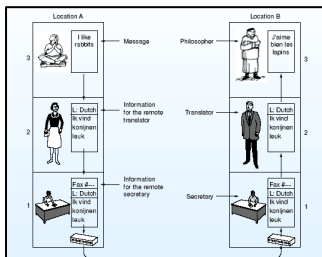
Nätverkslager

- Nätverkskommunikation kan ses som en packe lager
- Varje lager erbjuder en viss tjänst till lagret ovanför, och lovar att utföra den
- Varje lager döljer för lagret ovanför hur den exakt löser sin uppgift.

Nätverkslager

- Det lager som tagit på sig en uppgift, delar i sin tur upp den, och ger sedan deluppgifter i uppdrag till underliggande lager
- Varje lager kan implementeras som ett eget program, men kan också vara olika delar i samma program
- I moderna datorer sköts de flesta av lagren av operativsystemet

Kommunikation mellan olika lager

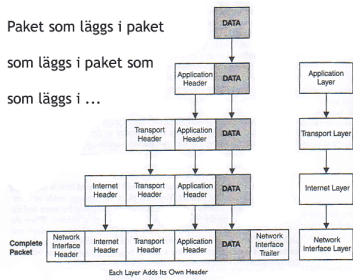


Inkapsling av data

Paket som läggs i paket

som läggs i paket som

som läggs i ...



Network Interface Layer

- Lagret längst ner som sköter den direkta kontakten med datorerna
- Mjukvara för kommunikation i det lokala nätverket
- Hårdvara i form av ett nätverkskort
- Fungerar med felkontroll i ett antal olika media som t ex: koaxialkabel, tvinnad parkabel, fiberoptik mm

Internet Layer

- Lagret som jobbar med IP-paket
- IP = **I**nternet **P**rotocol
- Ett IP-paket kallas också datagram
- Innehåller bl a information om:
 - avsändaradress
 - destinationsadress
 - typ av service (t ex FTP eller HTTP)
 - TTL (Time To Live)

Transport layer

- Hanterar **hur** paketflödet sker
- TCP **T**ransmission **C**ontrol **P**rotocol
 - ett pålitligt protokoll där en uppkoppling etableras mellan sändare och adressat
 - garanti för att paketen kommer fram
- UDP **U**ser **D**atagram **P**rotocol
 - etablerar ingen uppkoppling och passar för broadcast till okända adressater
 - snabbt men utan leveransgaranti

Application layer

- Här utförs de tjänster som användaren vill utnyttja med olika protokoll som:
 - FTP, SMTP, POP, NNTP, HTTP ...
- Vi ska efter en rast titta på ett exempel där vi hämtar en hemsida m h a HTTP

Rast 15 min?

IP-adresser

- Datorer har en unik numerisk adress ett **IP-nummer**
- En IP-adress består av fyra grupper av nummer mellan 0 och 255, som t ex kan se som i: 130.237.161.107
- Adressen kan läsas bakifrån och de tre sista siffrorna är den specifika **datorns adress**
- De tre andra tregrupperna anger **nätverkets adress**
- En organisation, som exempelvis ett universitet eller ett företag, kan tilldelas en grupp nätverksadresser och bestämmer sedan själva vilken adress varje dator ska ha inom nätverket

Domain Name System

- Hur kan dessa svärmemorerade adresser omformas till något mer användarvänligt?
- **Domain Name Service** som använder sig av
- **Domain Name Servers**

Domain Name System

Namnadressen **www.dsv.su.se** översätts av en DNS-server till Sifferadressen **66.35.250.151**

- DNS består av hierarkiskt ordnade servrar som svarar på frågor om adresser
- Om den först tillfrågade servern inte vet svaret, frågar den en annan server

URL

- **Uniform Resource Locator**
- URL-syntaxen finns specificerad för en rad protokoll i RFC 1738
- Vi tittar här på hur den ser ut för HTTP som är protokollet för hemsidor/webben

`http://host[:port]/path/filename[#section]`
(+ eventuell följande ?query_string)

URL

- Vi tittar på den URL som leder fram till en hemsida med lite material till denna kurs:

<http://www.dsv.su.se/~mozelius/GKITP/>

`http://host[:port]/path/filename[#section]`

- protocol/protokoll är `http://`
- host/värddator är www.dsv.su.se
- port är HTTP:s defaultport `:80`
- path/sökväg är `~mozelius/GKITP/`
- filename/filnamn blir `index.html`

HTTP

- Hyper Text Transfer Protocol

- GET
- POST
- HEAD
- TRACE
- PUT
- DELETE

Vi tittar på hur en sida hämtas med [GET](#)

HTTP - GET

En riktigt enkel HTML-sida skickas efter begäran från servern till webb-läsaren

Web Browser: GET /index.html HTTP/1.0

Web Server: HTTP/1.0 200 OK
Server: Apache/1.3.9 (Win32)
Content-type: text/html
Content-length: 70

```
<HTML><HEAD><TITLE>Example</TITLE></HEAD>  
<BODY>It works!</BODY></HTML>
```

En första titt på HTML

HyperText Markup Language

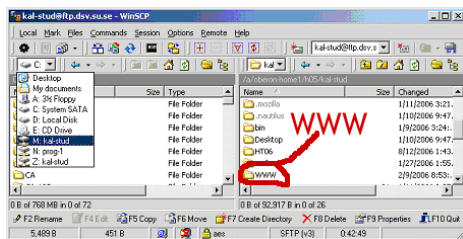
```
<html>
  <head>
    <title> Ett exempel </title>
  </head>
  <body>
    <h2> Det fungerar! </h2>
  </body>
</html>
```

FTP

- o Ett protokoll för att flytta filer via Internet
- o Används inte lika mycket nu som för 10 eller 20 år sedan
- o Här på GKITP ska ni lägga upp era lösningar av Uppgift1 på valfri server med valfri FTP-klient

SFTP och WinSCP

SFTP = **Secure File Transfer Protocol**





Tentatips 1

Läs på lite om **Klient - Server**

- Webbserver - HTTP
- Filserver - FTP
- HTTP-klient
- FTP-klienter
- Epostserver - SMTP och POP



Tack för idag!

Tack för idag!

Det blir mera om HTML och
Hur en dator fungerar när vi
träffas nästa gång på F5!
