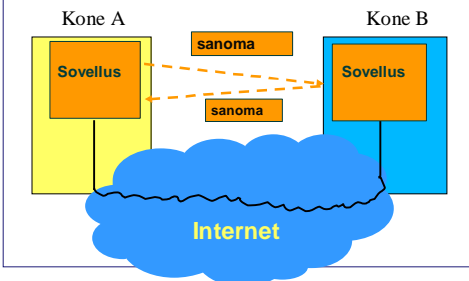


TIETOLIIKENTEEN PERUSTEET kevät 2009

Tässä on koottuna kalvot, joita käytettiin apuna kerrattaessa luentokerran alussa edellisen luentokerran pääkohtia.



Tietoliikenteen perusteet /2009/ Liisa Marttinen

1

Kertausta: termejä ja käsitteitä

internet – intranet - extranet
 palvelu - protokolla
 yhteydetön – yhteydellinen
 luotettava - epäluotettava
 isäntäkone - reititin - linkki
 asiakas-palvelin-malli - vertaistojimalli
 piirikytkentä - pakettikytkentä
 kanavointi: taajuusjako - aikajako

siirtonopeus – siirtoaika
 etenemisviive - jonotusviive

Tietoliikenteen perusteet /2009/ Liisa Marttinen

2

Kertaus jatkuu:

pakettivälityksen tehokkuus

sopii purskeiseen liikenteeseen
 etappivälitys – sanoma paketeiksi
 paketin otsake – yleisrasite

pääsy Internettiin: kaapelimodeemi, ADSL,
 lähiverkko, langaton yhteys

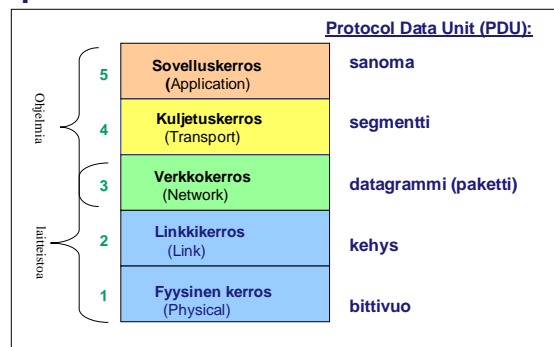
fyysinen siirtomedia: kierretty pari, koaksiaalikaapeli,
 valokaapeli, sähkömagneettinen aaltoliike

siirtovirhe: signaalin vaimeneminen ja
 vääristyminen, häiriöt => bittivirheitä
 analoginen signaali – digitaalinen signaali

Tietoliikenteen perusteet /2009/ Liisa Marttinen

3

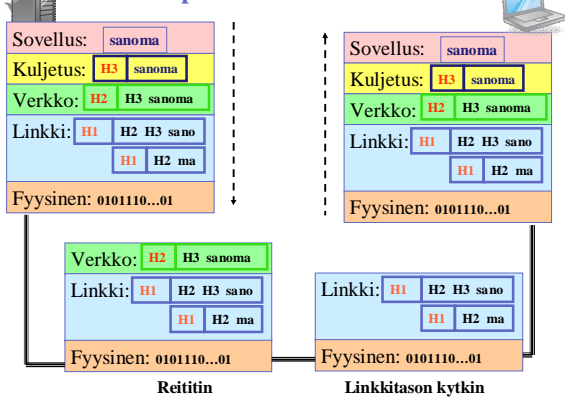
Internet-protokollapino



Tietoliikenteen perusteet /2009/ Liisa Marttinen

4

Kapselointi



Tietoliikenteen perusteet /2009/ Liisa Marttinen

5

Sovellusarkkitehtuuri

Asiakas-palvelija-malli (esim. selain ja www-palvelin)

- Aina toiminnassa oleva palvelinohjelma, jolla kiinteä, tunnettu **IP-osoite**
- Asiakasohjelmat ottavat yhteyttä palvelimeen ja pyytävät siltä palvelua

Google, e-Bay, Facebook, YouTube, Amazon, ..



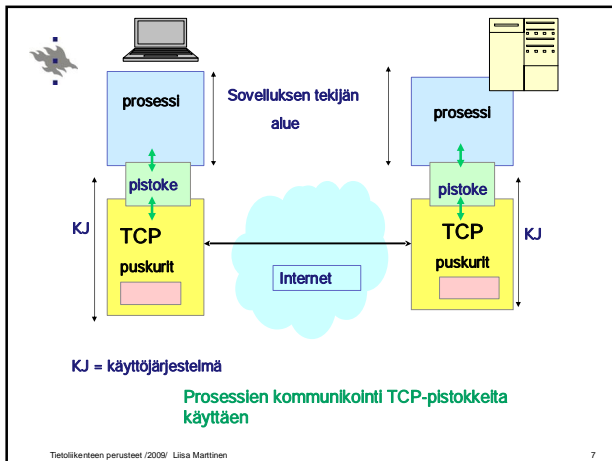
Vertaistojimalli (esim. BitTorrent, eMule, Skype)

- Vertaisisännät kommunikoiivat suoraan keskenään
- Ei tarvitse olla aina toiminnassa, IP-osoite voi muuttua
- Jokainen toimii sekä palvelijana että asiakkaana

Hybridimalli (esim. Napster, pikaviestimet)

Tietoliikenteen perusteet /2009/ Liisa Marttinen

6



Osoittaminen

- Sanomissa oltava lähettäjän ja vastaanottajan IP-osoite ja porttinumero
- IP-osoite $\hat{=}$ oikea kone
 - koneen (verkkokortin) yksilöivä 32-bittinen tunnistus
 - osoitteen verkko-osa yksilöi verkon
 - osoitteen koneosa yksilöi koneen verkossa
- porttinumero $\hat{=}$ oikea prosessi
 - Yleisillä palveluilla standardoidut tunnetut porttinumerot:
 - www-palvelin kuuntelee porttia 80,
 - Postipalvelin kuuntelee porttia 25
 - KJ osaa liittää porttinumeron prosessiin

Tietoliikenteen perusteet /2009/ Liisa Marttinen 8

HTTP -protokolla

- Request -sanoma
 - GET, HEAD, POST, PUT, DELETE
 - sanoman muoto
- Response- sanoma
- Tilaton protokolla => eväste (cookie)
- Käyttää TCP:tä
 - Säilyvä yhteys
 - Liukuhinna (pipeline)
 - Verkkovälimuisti (proxy server)
 - Ehdollinen GET

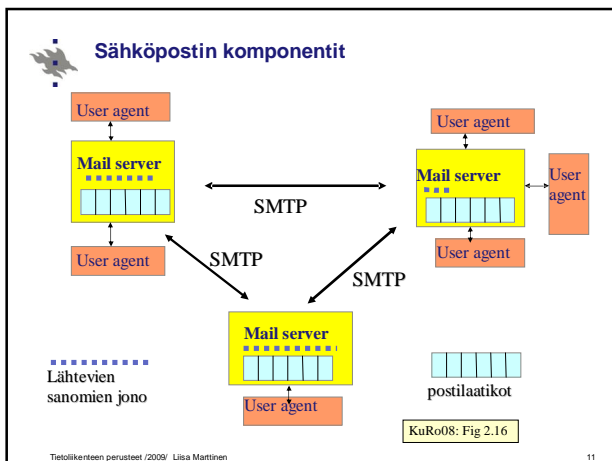
Tietoliikenteen perusteet /2009/ Liisa Marttinen 9

FTP: File Transfer Protocol (RFC 959)

- Tiedostojen kopioiminen koneelta koneelle
 - Asiakas voi selata etäkoneen hakemistoissa FTP-sanomilla, voi noutaa tai tallettaa haluamansa tiedoston (download/upload)
- FTP-palvelin kuuntelee porttia 21
 - yhteys kontrollitiedon välitystä varten
- Asiakas kuuntelee porttia 20
 - palvelija avaa tiedoston siirtoa varten
- FTP-palvelin **ylläpitää tilatietoa** mm. työhakemiston polku, autentikointi

2 TCP-yhteyttä

Tietoliikenteen perusteet /2009/ Liisa Marttinen 10



Esimerkki

```

S: 220 helsinki.fi
C: HELO princeton.edu
S: 250 Hello princeton.edu
C: MAIL FROM:<Bob@princeton.edu>
S: 250 <Bob@princeton.edu> OK
C: RCPT TO:<pekka.puupaa@cs.helsinki.fi>
S: 250 <pekka.puupaa@cs.helsinki.fi> OK
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: dataa ... dataa
C: dataa ... dataa
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 princeton.edu closing connection
  
```

SMTP:n kättely tai EHLO

Viesti(t)

SMTP:n lopetus

Tietoliikenteen perusteet /2009/ Liisa Marttinen 12

Sähköpostiviestin rakenne

Eri asia kuin SMTP: eri standardit (RFC 822)

Esim.

To:
CC:
From:
Message-Id:
Received:
Date:
Reply-To:
Subject:
....

otsakerivit

tyhjä rivi

runko
varsinainen viesti,
jossa vain ASCII-
merkkejä ja lopussa
piste omalla rivillä

Tietoliikenteen perusteet /2009/ Liisa Marttinen 13

MIME

- MIME-sisältötyyppejä
 - text/plain; charset=us-ascii
 - text/html
 - image/gif, image/jpeg, video/mpeg
 - application/postscript
 - application/msword
 - application/octetstream
 - multipart/mixed
- Base-64-koodaus
 - Sanoma 24 bitin ryhmät on jaettu 6 bitin osiksi, jotka kukin on koodattu ASCII-merkeiksi. 64 eri vaihtoehtoa

MIME-versio:
Content-Transfer-Encoding:
Content-Type:

Tietoliikenteen perusteet /2009/ Liisa Marttinen 14

Postinoutoprotokollat (mail access protocols)

Koska SMTP on 'PUSH'-protokolla, sitä ei voi käyttää sanomia haettaessa ('PULL').

Posti omalta postipalvelimelta postiohjelmaan

- POP3: Post Office Protocol versio 3
Viestien lataamiseen omalle koneelle, ei postikansioita
- IMAP: Internet Mail Access Protocol
Monipuolisempi: postikansiot (folders), lataa vain otsikot, viestien säilytys postipalvelimelle
- HTTP: Esim. TKTL:lla käytettävä IlohaMail, Hotmail, ...
Web-palvelija käyttää IMAP-palvelijaa

KuRo08: Fig 2.18

Tietoliikenteen perusteet /2009/ Liisa Marttinen 15

DNS (Domain Name System)

- Hakemistopalvelu ja sovelluskerroksen protokolla
Isännät ja nimipalvelimet käyttävät
Käyttää itse UDP-kuljetuspalvelua DNS-sanomien kuljettamiseen
- Nimien muuttaminen IP-osoitteiksi (ja päinvastoin)
Posix: gethostbyname(hydra.cs.helsinki.fi) 218.214.4.29
Kone = hydra =29, verkko= cs.helsinki.fi = 218.214.4.0
- Sallii aliasnimet, palvelijan replikoinnin
Esim. WWW.cs.helsinki.fi ja cs.helsinki.fi ovat aliasnimiä
Esim. www-palvelijaan voi liittyä useita IP-osoitteita, rotaatio
- Hajautettu, hierarkinen tietokanta (hakemisto)
Toteutettu useiden replikoitujen nimipalvelimien yhteistyönä
skaalautuvuus, kuormantasaus, ylläpito, vikasietoisuus, ..
Jos oma nimipalvelija ei tunne, se kysyy muilta.

Tietoliikenteen perusteet /2009/ Liisa Marttinen 16

Hajautettu, hierarkinen tietokanta

Standarditoteutus Unix + BIND-ohjelma

Root DNS Servers

- com DNS servers
 - yahoo.com DNS servers
 - amazon.com DNS servers
- org DNS servers
 - pbs.org DNS servers
- edu DNS servers
 - poly.edu DNS servers
 - umass.edu DNS servers

KuRo08: Fig 2.19

- 13 juuritason nimipalvelija
Replikoituja, kaikilla samat tiedot
- Yliätason palvelimet maa- ja yleistunnuksille (n. 265 kpl)
..., fi, fr, uk, ... edu, net, com, org, ...
- Autorisoidut aluepalvelimet (domain) (2-taso)
Isoilla yliopistoilla ja firmoilla omansa, pienet käyttävät jonkun muun ylläpitämää
www.iana.org

Tietoliikenteen perusteet /2009/ Liisa Marttinen 17

Domain -nimiavaruus

ICANN
The Internet Corporation for Assigned Names and Numbers

COM, NET, ORG, EDU, INT, MIL, GOV, ARPA ja maakoodit

fi, com, org, edu

helsinki, CS

Uusia: .biz, .info, .name

Tietoliikenteen perusteet /2009/ Liisa Marttinen 18

Skaalautuvuus

KuRo08: Fig. 2.24

Asiakas-palvelinmalli:
 Palvelimen siirrettävä $n \cdot F$ bittä => siirtoaika = nF/u_s
 Hitain asiakas d_{min} saa tiedoston ajassa F/d_{min}

Siirtoaika =
 $\max(nF/u_s, F/d_{min})$

Kun n kasvaa, palvelimen kuorma kasvaa ja siirtoaika kasvaa.

Vertaistomijamalli (alussa tiedosto on palvelinella)
Siirtoaika = $\max(F/u_s, F/d_{min}, nF/(u_s + V u_s))$

Summamerkki

Tietoliikenteen perusteet /2009/ Liisa Marttinen 19

TCP-kuljetuspalvelu

Luo 'pysyvä' pistoke

server: SOCKET, BIND, LISTEN, ACCEPT, RECEIVE, SEND(DATA), CLOSE

client: SOCKET, CONNECT(SERVER), SEND(DATA), RECEIVE, CLOSE

Handshakes: conn.req, conn.ack, discon.req, discon.ack

Actions: Ota yhteys, lähetä, vastaanota dataa, sulje yhteys

Tietoliikenteen perusteet /2009/ Liisa Marttinen 20

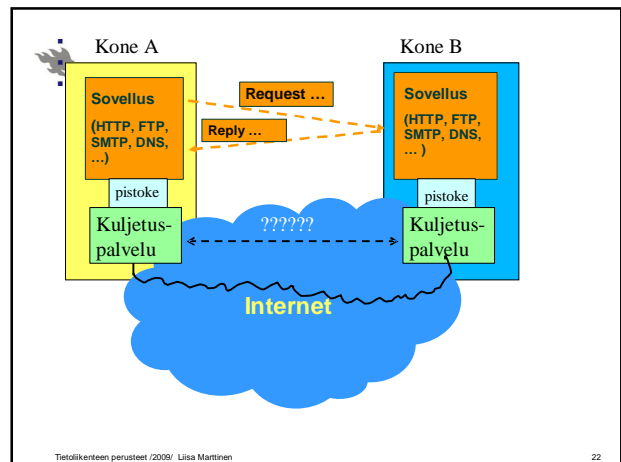
Esimerkki: TCP-asiakas (Java)

```

import java.io.*; import java.net.*;
class TCPClient {
  public static void main(String argv[]) throws Exception {
    String sentence;
    String modifiedSentence;
    BufferedReader inFromUser =
      new BufferedReader(new InputStreamReader(System.in));
    Socket clientSocket = new Socket("hostname", 6789); yhteyspyyntö
    DataOutputStream outToServer =
      new DataOutputStream(clientSocket.getOutputStream());
    BufferedReader inFromServer =
      new BufferedReader(new InputStreamReader(
        clientSocket.getInputStream()));
    sentence = inFromUser.readLine();
    outToServer.writeBytes(sentence + '\n');
    modifiedSentence = inFromServer.readLine();
    System.out.println("FROM SERVER: " + modifiedSentence);
    clientSocket.close(); Sulkee myös TCP-yhteyden
  }
}

```

Tietoliikenteen perusteet /2009/ Liisa Marttinen 21



Mikä kone /Mikä prosessi?

- n Kuljetuskeros tarjoaa päästä-päähän yhteyden
 - n Prosessilta prosessille (= pistokkeesta pistokkeeseen)
 - n Prosessi lukee ja kirjoittaa sanomia halutessaan
- n Datan lisäksi on välitettävä osoitietoja
 - n Vastaanottajan ja lähettäjän tiedot
 - n Eri koneiden prosessit voivat käyttää samaa palvelua
 - n Saman koneen prosessit voivat käyttää eri palveluita
- n Kuljetuskeros: mikä prosessi = mikä portti
- n Verkkokerros: mikä kone = mikä IP-osoite
- n Porttinumero
 - n 16-bittinen: 0 – 65535
 - n Portit 0 – 1024 on varattu kukin tietylle palvelulle (well known ports)
 - Esim. www-palvelulle portti 80, SMTP-postipalvelulle portti 25

Tietoliikenteen perusteet /2009/ Liisa Marttinen 23

Mikä kone /Mikä prosessi?

Lähetys (asiakas)

- n Kuljetuskeros
 - n Segmentin otsakkeessa lähde- ja kohdeprosessin porttinumero
 - n Antaa segmentin verkkokerroksen välitettäväksi
 - n TCP: huolehtii myös luotettavuudesta
 - n UDP: tarjoaa pelkän välityspalvelun
- n Verkkokerros
 - n Paketin otsakkeessa lähde- ja kohdekonen IP-osoite → reitittimet osaavat ohjata oikealle koneelle

32 bits

source port # dest port #

other header fields

application data (message)

TCP/UDP-segmentti

Source IP

Destination IP

Other IP header fields

TCP/UDP-segmentti

IP-paketti

Tietoliikenteen perusteet /2009/ Liisa Marttinen 24

UDP-segmentin rakenne

- Porttinumerot**
 - Koska on prosessien välinen palvelu
- Length**
 - Segmentin kokonaispituus otsake (8 B) mukaanluettuna
- Checksum (optionaalinen)**
 - Bittivirheen havaitsemiseen
 - UDP ei yritä toipua, hävittää segmentin
- Data**
 - Pitkä sanoma pilkottuna useaksi segmentiksi
- IP-osoitteet vasta verkkokerroksen otsakeessa**
 - Näitä tarvitaan reitityksessä

UDP-otsake

32 bittia

UDP-tarkistussumma

- Lähetys**
 - Summaa 16 bitin kokonaisuudet (otsake + pseudo-otsake mukana), ylivuotobitit lasketaan mukaan, talleta **yhden komplementtina**
- Vastaanotto**
 - Summaa 16 b kokonaisuudet (myös tarkistussumma).
 - Jos tuloksena on 16 ykköstä, niin OK!

UDP-otsake

32 bittia

Epäluotettava kanava

Verkkokerros

Bittivirheitä, segmenttejä katoaa, segmentit väärässä järjestyksessä

Kuinka saada luotettavaksi?

- Tarkastellaan yleisesti luotettavan tiedonsiirron ongelmia ja erilaisia ratkaisuyrityksiä
- Edeten ideaalitalanteesta yhä ongelmaisempaan
- Käyttämällä äärellisiä tila-automaatteja lähettäjän ja vastaanottajan toiminnan kuvaamiseksi

rdt3.0

Sender

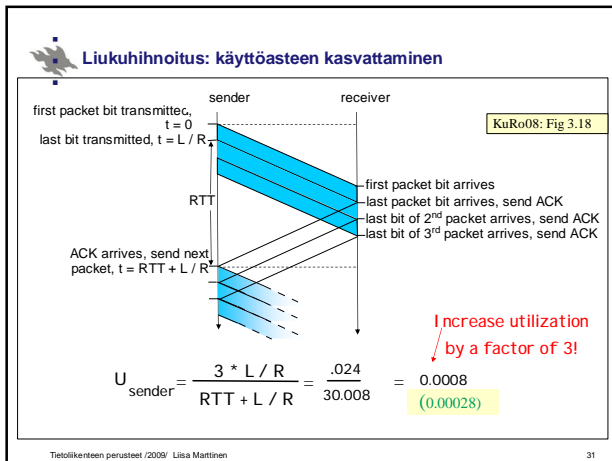
KuRo08: Fig 3.15

rdt3.0 toiminnassa

(a) operation with no loss

(b) lost packet

KuRo08: Fig 3.16



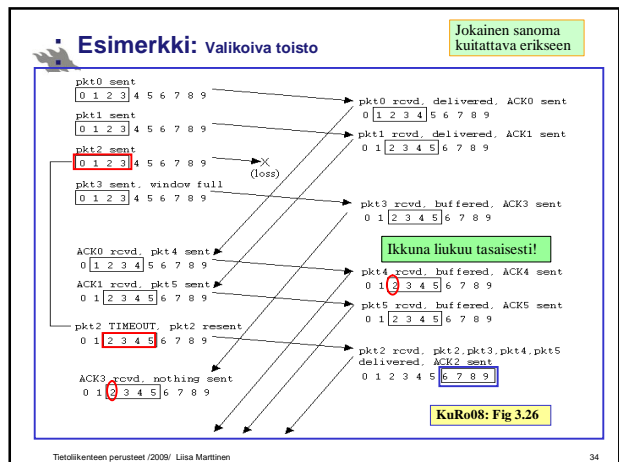
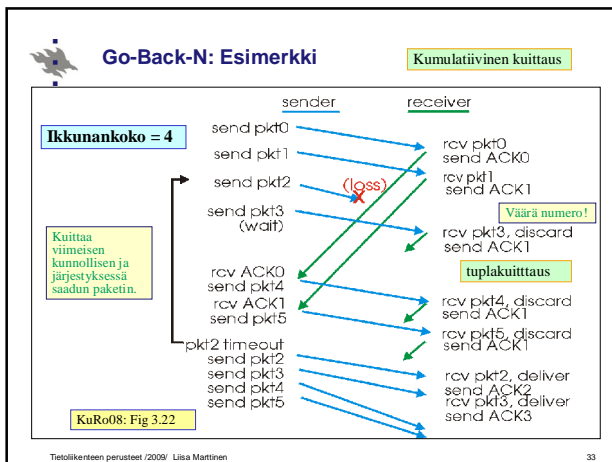
Liukuva ikkuna (sliding window)

- Säätelee pakettien lähettämistä ja vastaanottoa
- Kertoo millä pakettinumeroilla on lähetetty/vastaanotettu, mistä saatu/lähetetty kuittaukset ja millä numeroilla voi vielä lähettää/vastaanottaa paketteja
- Ikkunan koko riippuu yhteyden tyypistä ja puskurien koosta

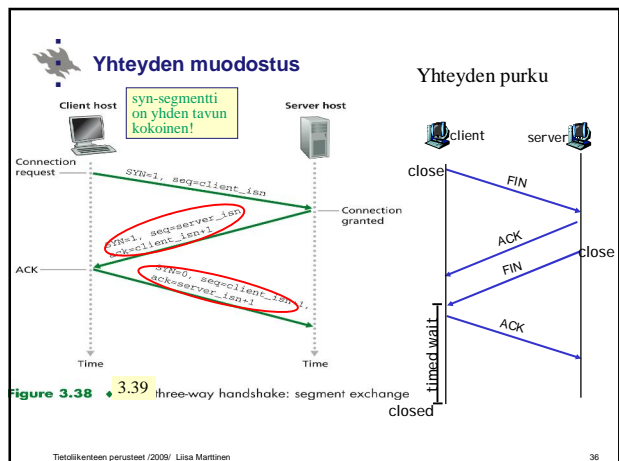
...10 11 12 13 5 16 17 18 19 20 21 22 23 24 ...

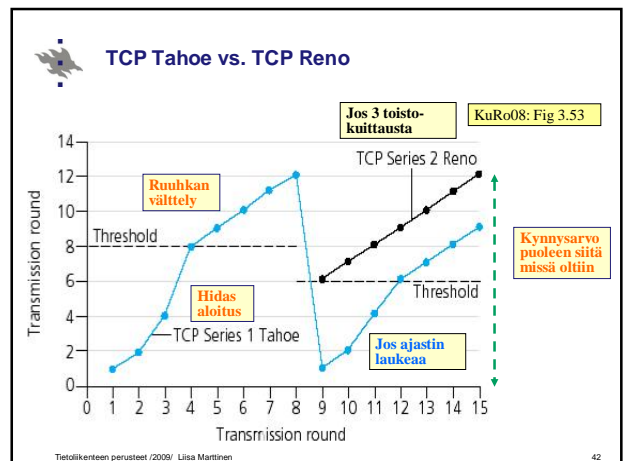
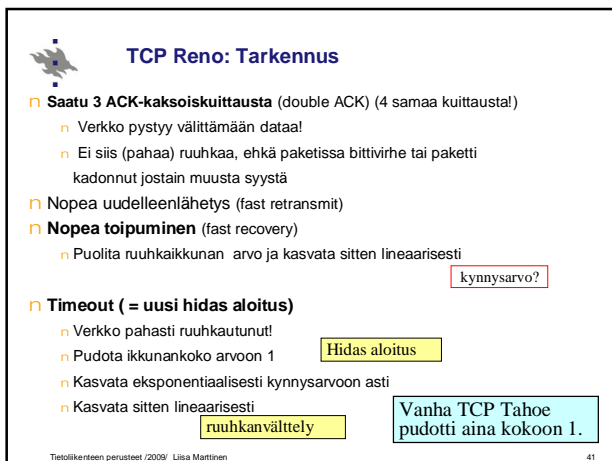
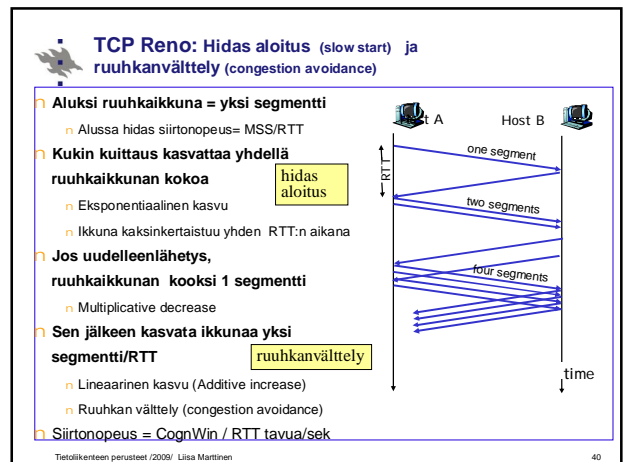
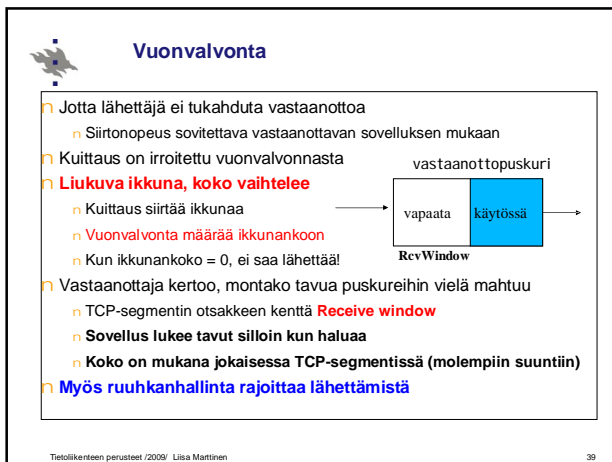
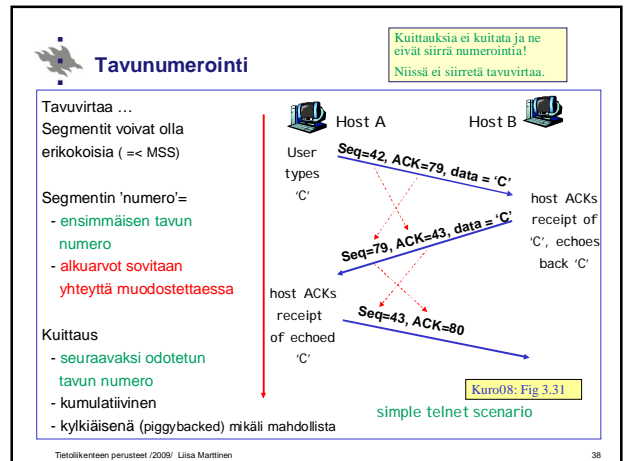
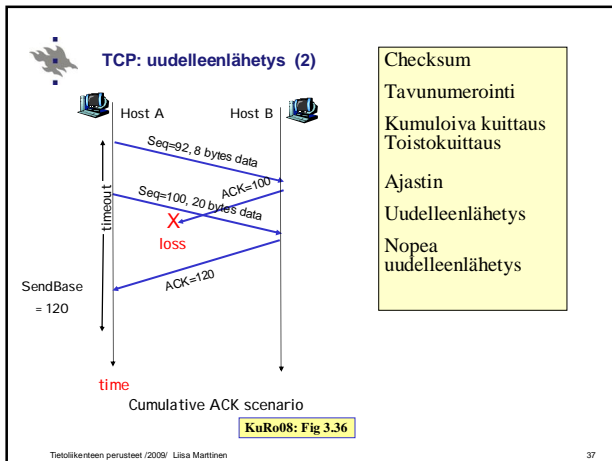
- Lähetysikkuna (sender window)
- Ikkunan koko = montako pakettia saa olla kuittaamatta
- Mitkä pakettinumerot on käytetty, mutta kuittaamatta
- Mitä pakettinumeroita voi vielä käyttää
- Lähetäjän on odotettava, jos kaikki ikkunan numerot on käytetty
- Kun kuittaus saapuu, ikkuna liukuu
- Seuraavat numerot tulevat luvalleiksi

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 32



- ### Yhteenvedo menetelmistä
- Ks. KuRo08 Table 3.1
 - Tarkistussumma
 - Ajastin
 - Järjestysnumero
 - Kuittaukset
 - Positiiviset ACK, tuplakuittaukset
 - Negatiiviset NAK
 - Ikkunat, liukuhhnoitus
- Tietoliikenteen perusteet / 2009/ Liisa Marttinen 35





Ajastimen arvo?

- n Aseta ajastin, kun segmentti on lähetetty
- n Liian lyhyt aika
 - n Ennenaikainen timeout, turha uudelleenlähetyks
 - n Turhat ruuhkatoiminnot
- n Liian pitkä aika
 - n Turhan hidas reagointi segmentin katoamiseen
 - n Ei huomata ruuhkaa ajoissa
- n Alkujaan: $Timeoutinterval = 2 * RTT$
- n Kuittausaika vaihtelee suuresti ja nopeasti => käytössä dynaaminen arvo
 - n Saadaan jatkuvien mittausten perusteella
- n Jos ajastin laukeaa, tuplaa Timeout
 - n Exponential backoff

TCP-segmentti

Otsake aina vähintään 20 B
Optio-osa tarvittaessa

Segmentti- ja kuittausnumerot **tavunumeroina**

lkkunankoko: paljonko tilaa vastaanottopuskurissa (tavua)

ACK= kuittausnumero validi,
RST (reset),
SYN yhteydenmuodostus
FIN yhteydenpurku
URG, PSH ei yleensä käytetty

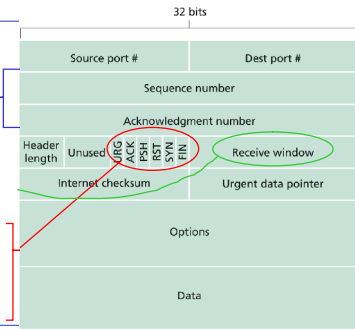
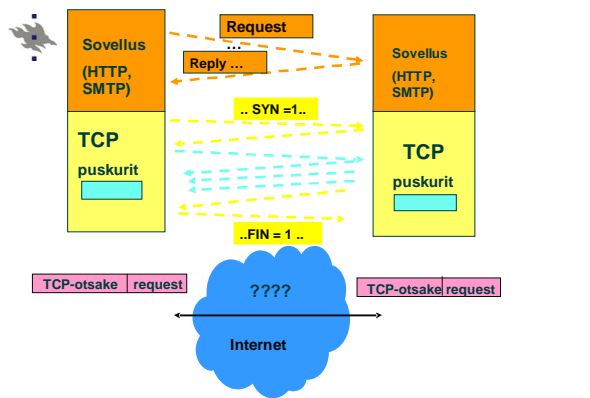


Figure 3.29 TCP segment structure



Verkkokerros

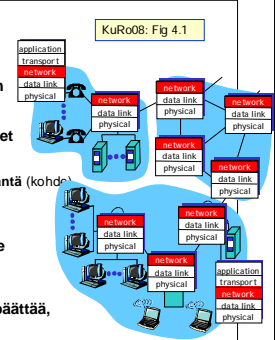
Toimittaa kuljetuskerroksen segmentit vastaanottajalle

Lähetys

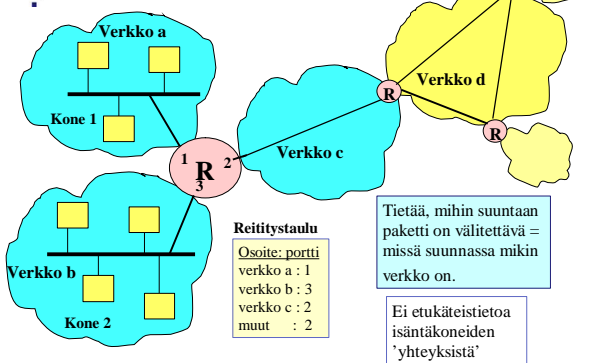
- Luo segmenteistä verkkokerroksen IP-paketteja
 - Lisää otsaketietoja: mm. IP-osoitteet
- Pakettien kulku verkossa
- Isäntä (lähde) - reititin - ... - reititin - isäntä (kohde)
- Vastaanotto
- Poista otsake
 - Anna segmentti kuljetuskerrokselle

Toimii etenkin reitityksessä

- n Reititin tutkii IP-paketin otsakkeen ja päättää, linkkiin se lähetetään seuraavaksi



Reititys: datagrammiverkko



Reititys: Virtuaaliipiiriverkko

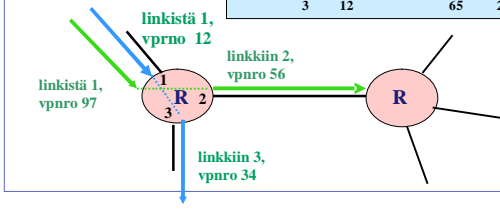
1. paketti muodostaa reitin, muut paketit kulkevat samaa reittiä

otsakkeessa kohdeosoite

reititin ylläpitää tietoa piirirun

Reititys = selvitä vprno

Sisään: linkki / vprno	Ulos: vprno / linkki
1 12	34 3
1 97	56 2
2 42	101 3
2 10	78 1
3 12	65 2



Reitittimen arkkitehtuuri

- Kaksi tehtävää
 - Välitä paketteja tulolinkeistä ulosmenolinkkeihin
 - Suorita reititys algoritmia / -protokollaa
- Portti ~ verkkokortti
 - Useita portteja niputettu yhteen linjakortiksi (line card)

KuRo08: Fig 4.6

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 49

Kolme erilaista kytkentätapaa:

KuRo08: Fig. 4.8

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 50

Paketin hylkäys

- Kun puskuritila ei riitä
 - Hylkää saapuva paketti (drop-tail) tai joku muu ..
 - Se kummassa jonossa paketit hylätään, riippuu kytkennän ja linjan nopeuden suhteista
 - RED (Random Early Detection): hylkää jo ennenkuin puskurit täyttyy
- Siirtovirhe
 - Linkkikerros saa hylätä virheellisen
 - Verkkokerros saa hylätä virheellisen (ICMP-protokolla)
- Paketin elinaika (Time-to-live, TTL)

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 51

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 52

IP-paketin rakenne (IPv4)

32 bits			
Version	Header length	Type of service	Datagram length (bytes)
16-bit Identifier		Flags	13-bit Fragmentation offset
Time-to-live	Upper-layer protocol	Header checksum	
32-bit Source IP address			
32-bit Destination IP address			
Options (if any)			
Data			

Figure 4.13 ♦ IPv4 datagram format

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 53

IP-pakettien paloittelu (fragmentointi)

Maximum transfer Unit (MTU)
 suurin mahdollinen IP-paketti eri linkeillä eri koko
 Esim. Ethernet 1500 B

Liian iso paketti pilkottava reitittimessä pienemmiksi paketeiksi (fragmenteiksi), jotka kohdekone kokoaa voivat kukin kulkea eri reittiä

IP-otsakkeessa kentät yhteenkuuluvien fragmenttien tunnistamiseksi

KuRo08: Fig 4.14

Tietoliikenteen perusteet / 2009/ Liisa Marttinen 54

Esimerkki

length =4000	ID =x	fragflag =0	offset =0
-----------------	----------	----------------	--------------

4000 tavun IP-paketti:
dataa 3980 B
MTU 1500 B

1480 B dataa
20 B IP-otsaketta

offset = 1480/8

length =1500	ID =x	fragflag =1	offset =0
length =1500	ID =x	fragflag =1	offset =185
length =1040	ID =x	fragflag =0	offset =370

Yhdestä IP-paketista tulee
3 pienempää IP-pakettia

0	1480	2960
1. Pala: 1480 tavua	2. Pala: 1480 tavua	3. Pala: 1020 tavua

Tietoliikenteen perusteet /2009/ Liisa Marttinen 55

IP-osoitteet

32 bittinen tunniste isäntäkoneille ja reitittimien linkeille

- verkkoliittymän tunniste
- kullakin oma IP-osoite
- Myös isäntäkone voi olla liitettynä useaan verkkoon

ICANN Internet Corporation for Assigned names and Numbers
verkkonumerot palvelun tarjoajille, nämä edelleen aliverkoiksi

KuRo08:Fig 4.15

Tietoliikenteen perusteet /2009/ Liisa Marttinen 56

Aliverkot

Osoitteen osat
aliverkon numero (alkuosa)
koneen numero (loppuosa)

Aliverkon koneet voivat kommunikoida ilman reititystä
Linkkikerros osaa lähettää koneelta toiselle
Esim. Ethernet

Aliverkkoa merkitään notaatiolla, jossa lopussa on verkko-osan pituus
Esim. 223.1.1.0 /24 subnet mask
eli verkko-osoite 24 bittiä ja koneosoite 8 bittiä

network consisting of 3 subnets

KuRo08: Fig 4.15

Tietoliikenteen perusteet /2009/ Liisa Marttinen 57

CIDR: Classless InterDomain Routing

Verkko-osa voi olla minkä tahansa kokoinen
Vanha luokallinen osoite: A-luokka 8 b, B-luokka 16 b, C-luokka 24 b

Formaatti: a.b.c.d/x
x ilmoittaa verkko-osan bittien lukumäärän (prefix)

Esim. Organisaatio, jolla 2000 konetta varaa 2048 = 2¹¹ konenumeroa, jolloin verkko-osaa varten jää 21 bittiä
Yritys voi vielä itse jakaa viimeiset 11 bittiä aliverkko-osoitteeksi ja koneosoitteeksi. Tämä jako ei näy ulkopuolelle.

Verkko (prefix) Kone

11001000 00010111 00010000 00000000

200.23.16.0/23 aliverkko aliverkkomaski

Tietoliikenteen perusteet /2009/ Liisa Marttinen 58

NAT: Network Address Translation

Vain ~ 4 miljardia osoitetta!

rest of Internet local network (esim. kotiverkko) 10.0.0/24

138.76.29.7 10.0.0.4

10.0.0.1
10.0.0.2
10.0.0.3

Kaikkilla ulosmenevillä ja sisääntulevilla paketeilla sama IP-osoite 138.76.29.7 mutta eri porttinumeroita.

Kotiverkossa käytössä sisäiset IP-osoitteet 10.0.0/24 (esim. DHCP:llä)

Tietoliikenteen perusteet /2009/ Liisa Marttinen 59

1) Linkkitila: Dijkstran algoritmi

Aluksi kaikilla reitittimillä on tiedossa verkon rakenne ja kaikkien linkkien kustannukset

- Kaikki reitittimet lähettävät tietonsa naapureistaan ja linkkikustannuksista naapureihin (mitatut/havaitut) joko kaikille muille tai jollekin keskussolmulle, joka välittää tiedon muille

Reititin laskee Dijkstran algoritmilla edullisimman kustannuksen kaikkiin muihin kohteisiin

- Kokoaa näistä oman reititystaulunsa

Merkinnät

C(x,y) linkin x,y kustannus; jos eivät naapureita = ∞

D(v) toistaiseksi edullisin kustannus solmuun v

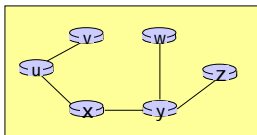
p(v) solmun v edeltäjä reitillä

N = solmujen joukko, N' = jo käsitellyt solmujen joukko

Tietoliikenteen perusteet /2009/ Liisa Marttinen 60

Lyhyimmät reitit ja reititustaulukko

Resulting shortest-path tree from u:



KuRo08: Fig. 4.28

Resulting forwarding table in u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Tietoliikenteen perusteet /2009/ Liisa Marttinen

61

2) Etäisyysvektorireititys (distance vector)

Arpanet-verkon alkuperäinen reititys algoritmi

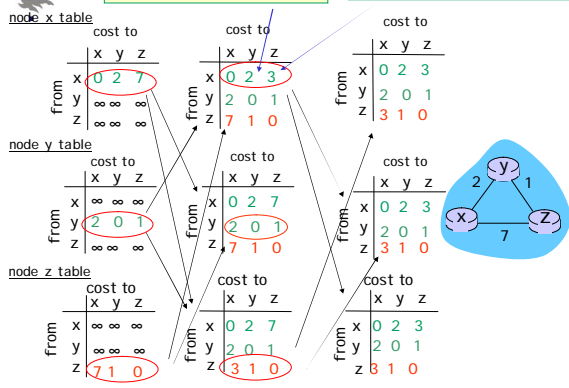
- n Käytössä useissa Internetin reititysprotokollissa
RIP, BGP, Novell IPX, ISO IDR
- n Interaktiivinen, hajautettu ja asynkroninen
- n Tiedot tarkentuvat asteittain, iteratiivisesti
 - n Tietyin väliajoin, linkin tilan vaihtuessa, naapurin tietojen muuttuessa, ..
- n Kukin solmu laskee itsenäisesti, mutta saa tietoa naapureiltaan
 - n Tietää / arvioi kustannuksen omiin naapureihinsa
 - n Kuulee naapureiden kustannukset muihin kohdesolmuihin, jotka nämä puolestaan ovat kuulleet omilta naapureiltaan
 - n Valitsee kullekin kohdesolmulle kuulemansa edullisimman reitin

Tietoliikenteen perusteet /2009/ Liisa Marttinen

62

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} = \min\{2+0, 7+1\} = 2$$

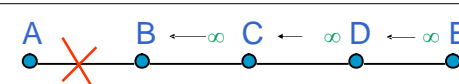
$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} = \min\{2+1, 7+0\} = 3$$



Tietoliikenteen perusteet /2009/ Liisa Marttinen

63

Huono uutinen etenee nopeasti: "poisoned reverse"



Ratkaisu count-to-infinity-ongelmaan!

Ilmoita etäisyys äärettömäksi naapurille, jonka kautta linkki kulkee. Kerro muille oikea etäisyys.

Etäisyys A:han

$D_B(A)$	$D_C(A)$	$D_D(A)$	$D_E(A)$
∞	2	3	4
∞	∞	3	4
∞	∞	∞	4
∞	∞	∞	∞

Tieto etenee joka vaihdossa yhden linkin yli

Tietoliikenteen perusteet /2009/ Liisa Marttinen

64

3) Hierarkkinen reititys

Reitityksen skaalautuus?

- n Isossa verkossa runsaasti reitittämiä
 - Kaikki eivät voi tuntea kaikkia muita
 - Reititustaulut suuria, reitien laskeminen raskasta netstat -r
 - Reititustietojen vaihtaminen kuluttaa linjakapasiteettia

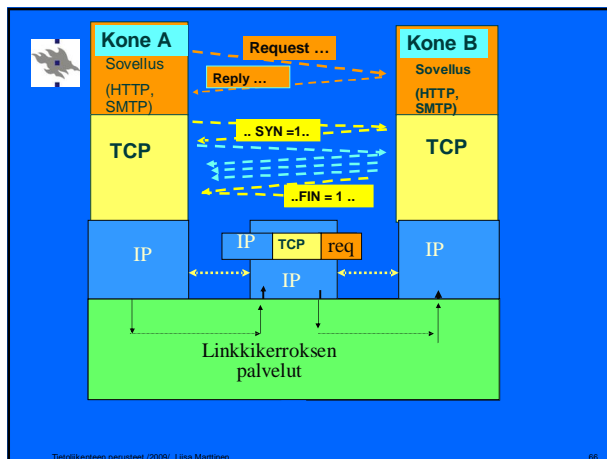
Jo CIDR ja IP-osoitteiden jakaminen lohkoina pienentää reititustauluja!

Autonomiset järjestelmät AS (Autonomous Systems)

- n Internet – verkkojen verkko
- n Intra-AS routing
 - n Kukin verkko päättää itse sisäisestä reitityksestään
 - n RIP, OSPF
- n Inter-AS routing
 - n AS:t ilmoittelevat toisilleen, mihin muihin AS:iin niistä pääsee
 - n BGP (Border Gateway Protocol)

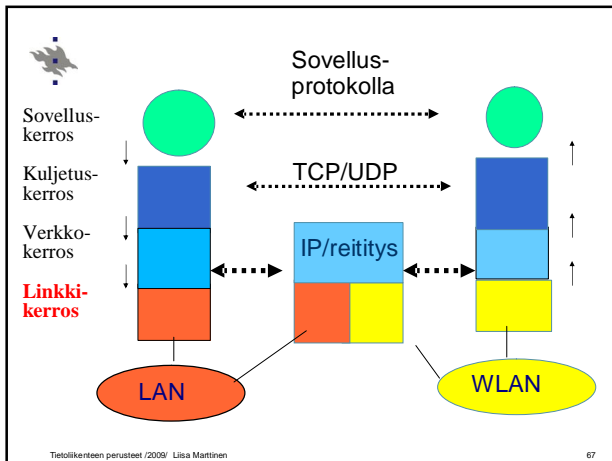
Tietoliikenteen perusteet /2009/ Liisa Marttinen

65



Tietoliikenteen perusteet /2009/ Liisa Marttinen

66



Linkkerroksen tehtäviä (2)

NIC (Network Interface Card) linkki- ja fyysinen kerros

- Vuonvalvonta, puskuroidi**
Kytkimessä on useita erinopeuksisia linkkejä
- Virhevalvonta**
signaali vaimenee, taustakohina häiritsee, ...
Kehyksessä on tarkistustietoa (error detection and correction bits)
Vastaanottava solmu korjaa, jos pystyy
Jos ei pysty, pyytää uudelleen tai hävittää
- Yksisuuntainen /kaksisuuntainen liikenne**
Yksisuuntainen: lähetysvuorojen hallinta

Esim. reititin

Tietoliikenteen perusteet /2009/ Liisa Marttinen 68

Linkkerroksen tehtäviä

- Kehystys (framing)**
Kehyksen rakenne ja koko riippuu siitä, millainen linkki on kyseessä
Otsake, data, lopuke otsake data lopuke
- Kohteen ja lähteen osoittaminen**
Yhteiseen linkkiin voi olla liitettyä useita laitteita
Käytössä laitetason MAC-osoite (Medium access control)
- Yhteisen linkin varaus ja käyttö (link access)**
Esim. langaton linkki, keskitettiin yhdistetyt linkit
- Luotettava siirto**
Langattomilla linkeillä suuri virhetodennäköisyys
Linkkitasoa huolehtii oikeellisuudesta
Miksi tästä täytyy huolehtia vielä kuljetuskerroksella?
Jotkut linkkityypit eivät huolehdi lainkaan!
Jos kehys hävitettävä...

Tietoliikenteen perusteet /2009/ Liisa Marttinen 69

Pariteettitarkistus

- Pariteettibitti**
Parillinen vs. pariton pariteetti
Virheröyryssä jopa 50% voi jäädä huomaamatta

0111000110101011 0

- Kaksikulotteinen pariteetti**
Eriksen horisontaalinen (parillinen) ja vertikaalinen (pariton) pariteetti
Pysty korjaamaan yhden bitin virheen.
- Hamming-koodi**
Korjaa yhden bitin virheen

Figure 5.6 ♦ Two-dimensional even parity

Tietoliikenteen perusteet /2009/ Liisa Marttinen 70

CRC-esimerkki

Data: 101110
G: 1001, polynomina
 $1*x^3 + 0*x^2 + 0*x^1 + 1*x^0$
<D,R>: 101110???

Lähetä: 101110011

Modulo 2-aritmetiikka
vähennyslasku yhteenlaskuna
ei lainaamista, ei muistinumeroita
= bittitason XOR
 $1+1=0, 1+0=0+1=1, 0+0=0$

KuRo08:Fig.5.8

Tietoliikenteen perusteet /2009/ Liisa Marttinen 71

Yksi kanava

- Kaksipisteyhteys (point-to-point)**
 - PPP-protokolla, puhelin-yhteys (dial-up access)
 - Ethernet-piuhia kytkimen ja isäntäkoneen välissä
- Yleislähetysyhteys (broadcast)**
 - Alkuperäinen Ethernet, Ethernet keskitimen ja isäntäkoneen välissä, kaapelimodeemiyhteys (upstream), WLAN, satelliitti,

shared wire (e.g. Ethernet) shared wireless (e.g. Wavelan) satellite cocktail party

KuRo08: Fig. 5.9

Tietoliikenteen perusteet /2009/ Liisa Marttinen 72



Lähetysvuorojen jakelu

1) Kanavanjakoprotokollat (channel partitioning protocol)

Jaa kanavan käyttö 'viipaleisiin' (time slots, frequency, code)
 Kukin solmu saa oman viipaleensa
 TDMA, FDMA, CDMA
 "käytä sinä tätä puolta, minä tätä toista"

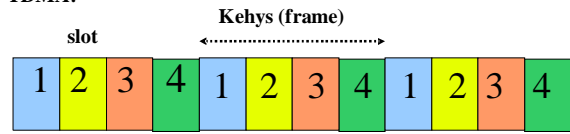
2) Kilpailuprotokollat (random access protocols)

"Se ottaa, joka ehtii."
 Jos sattuu törmäys, yritä myöhemmin uudelleen.
 Aloha, CSMA, CSMA/CD

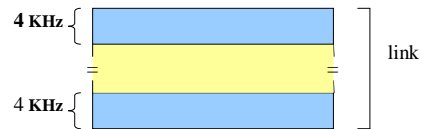
3) Vuoronantoprotokollat (taking-turns protocols)

Jaa käyttövuorot jollakin sovitulla tavalla:
 pollaus, vuoromerkki, ...
 "Minä ensin, sinä sitten."

TDMA:



FDMA:



CDMA: Eri lähettäjiillä omat keskenään ortogonaaliset sirukoodit biteille => yhteislähetyksestä kyetään erottamaan eri lähettäjät



Aloha

n Hawaijilla, 70-luvulla radiotietä varten

n Lähetä heti, kun on lähetettävää

n Ei mitään kuuntelua ennen lähetystä

n Kuuntele sitten, onnistuiko lähetys

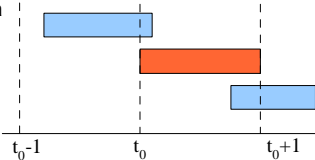
n Lähiverkossa törmäys havaitaan 'heti', sillä siirtoviive on pieni (toisin kuin satelliitilla)

n Jos törmäys, niin odota satunnainen aika ja yritä uudelleen

n Yksinkertainen

n Törmäyksen td. suuri

n Max tehokkuus ~ 18%



CSMA/CD (with Collision Detection)

n Asema kuuntelee myös lähettämisen jälkeen

n Langallinen LAN: törmäys => signaalin voimakkuus muuttuu
 - Esim. Ethernet
 n Langaton LAN: hankalaa

n Jos törmäys

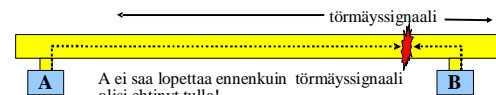
n Niin keskeytä heti lähettäminen

n ja yritä uudestaan satunnaisen ajan kuluttua

n Näin törmäyksen aiheuttama hukka-aika pienenee

n Kauanko kuunneltava?

n 2* maksimi etenemisviive solmujen välillä



Linkkikerroksen fyysinen osoite

n 32 bitin IP-osoite verkkokerroksella

n Reitityksen tapa viitata koneeseen

n Erilaisilla linkkikerroksilla omat tapansa osoittaa oikea linkki (- verkkokortti)

n Siirtokehys on kuljetettava fyysisen linkin yli jollekin toiselle samaan verkkoon (LAN) kytketyistä laitteista

n **MAC-osoite** (Media Access Control Address)

n Käytetään myös nimiä LAN-osoite, fyysinen osoite, laiteosoite, Ethernet-osoite, ...

n Liitetty valmistusvaiheessa kiinteästi laitteeseen

Analogia:

IP-osoite - katuosoite MAC-osoite - henkilötunnus



ARP-protokolla (Address Resolution Protocol)

n Ratkaisuna ARP-protokolla ja ARP-taulu

n **ARP-protokolla** lähettää yleislähetysosoitteella kyselyn, jonka kaikki vastaanottavat.

Oman osoitteensa tunnistava laite vastaa kyselijän **MAC-osoitteeseen** ja kertoo oman MAC-osoitteensa

"aa-bb-cc-dd-ee-ff", "FF-FF-FF-FF-FF-FF"
 "Kenen IP-osoite on "xx:yy:zz:vv"?"

MAC-
yleislähetysosoite:
FF-FF-FF-FF-FF-FF

"kk-ll-mm-nn-oo-pp", "aa-bb-cc-dd-ee-ff"

n **ARP-taulu** pitää tallella kyselyjen vastauksia: IP-osoite, MAC-osoite, TTL)

Kussakin koneessa (myös reitittimessä) jokaiselle aliverkolle oma taulunsa

Tiedot vanhenevat n. 20 minuutissa (time-to-live)

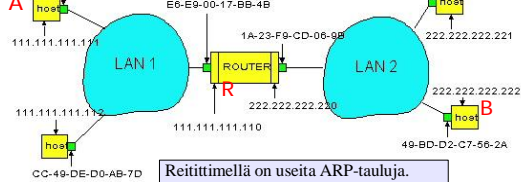
Lähtettäminen toiseen verkkoon (1)

Ensin omalle reitittimelle sen MAC-osoitteella ja reititin ohjaa eteenpäin

Reititystaulussa on verkko-osoite, jonne paketti seuraavaksi ohjattava

Katso kohdeverkon ARP-taulusta kohteen MAC-osoite

Jos ei ole taulussa, tee ARP-kysely kohdeverkon koneille



Tietoliikenteen perusteet (2009) Lissa Marttinen

79

10BaseT ja 100BaseT

10 Mbps tai 100Mbps (Fast Ethernet, FE)

T = Twisted Pair eli kierretty parikaapeli
Maks. etäisyys keskitimeen 100 m

Koaksiaali-
kaapeli
max. 500 m

Keskitin (hub) toistaa bitit heti sellaisenaan muille

Fyysisen tason toistin (repeater);

Yleislähetys, Signaalin vahvistus

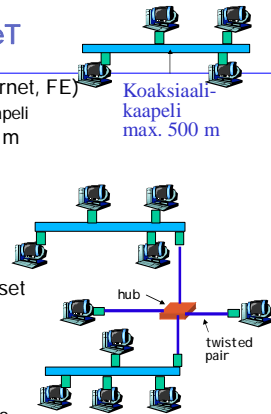
Verkkokortit käsittelevät törmäykset

Maks. 30 konetta / keskitin

Keskitin osaa jättää huomiotta vikaantuneen kortin

Kerää myös tietoa liikenteestä

Törmäysten lkm, keskim. kehyskoko, ...



Tietoliikenteen perusteet (2009) Lissa Marttinen

80

Gigabitin Ethernet (GE)

1 Gbps tai 10 Gbps

Edelleen sama kehysformaatti

Taaksepäin yhteensopiva

Yhteiskäyttöiset linkit edelleen OK

Koneiden yhdistely keskitimen välityksellä

CSMA/CD

Kaksipisteyhteydet

ei törmäyksiä

koneet yhdistetty **kytkimien** kautta

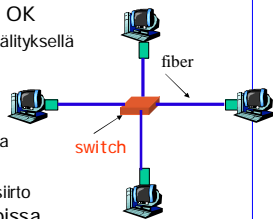
pitkät välimatkat mahdollisia

kaksisuuntainen täysivauhtinen siirto

Käytetään yleisesti runkoverkoissa

verkkojen yhdistely (reititin -> reititin)

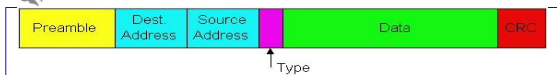
valokaapeli, myös cat5/cat6 parikaapeli



Tietoliikenteen perusteet (2009) Lissa Marttinen

81

Ethernet-kehys



Tahdistuskuvio (preamble) (8 B)

7 tavussa 10101010 kellojen tahdistusta varten

8. tavu 10101011 kertoo varsinaisen kehyksen alkavan

Kohteen ja lähteen MAC-osoitteet (6 + 6 B)

Type (2 B)

verkkoprotokolla, jolle vastaanottaja luovuttaa kehyksen datan

IP, ARP, jokin muu esim. Apple Talk, Novell IPX, ..

Data (46 ... 1500 B)

Ethernet MTU = 1500 B

CRC (4 B eli 32 bittiä)

tarkistusbitit, tahdistuskuvio mukana laskennassa

Tietoliikenteen perusteet (2009) Lissa Marttinen

82

Ethernet varaus: CSMA/CD

(klassinen Ethernet-verkko on yleislähetysverkko)

Carrier Sense

Kuuntele, onko väylä vapaa (96 b:n ajan)

Jos vapaa, lähetä heti

Muuten odota ja lähetä, kun linja vapautuu

Collision Detection

Kun lähetetty, kuuntele onnistuiko

Törmäys?

Huomaa signaalin voimakkuudesta

Lopeta kehyksen lähetys heti

Lähetä 48 bitin sotkusignaali (jam): muutkin huomaavat varmasti

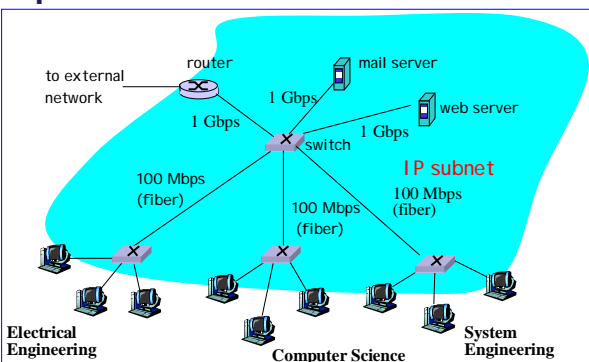
Random Access

Odota törmäyksen jälkeen satunnainen aika

Tietoliikenteen perusteet (2009) Lissa Marttinen

83

LAN, verkosegmentit



Tietoliikenteen perusteet (2009) Lissa Marttinen

84

• Takaperin oppiminen: Kytentäulu (switching table)

- n Aluksi taulu on tyhjä
- n Saapuva kehys
 - n **Lähteen MAC-osoite** x, kohteen MAC-osoite y, tuloportti p, yms
- n Lähde X ei ole taulussa
 - n Lisää (X, p, TTL) tauluun eli **kytkin oppii, että osoite X on saavutettavissa portin p kautta**
- n Lähde X on taulussa => päivitä TTL
- n Kohde Y ei ole taulussa
 - n Lähetetään kehys kaikkiin muihin portteihin = **tulvitus** (flooding)
 - n Opitaan myöhemmin Y:n oikea portti jostain sen lähettämästä kehuksesta
- n Lähde X ja kohde Y ovat jo taulussa
 - n X ja Y samassa portissa => hylkää kehys (**on jo oikeassa alverkossa**)
 - n X ja Y eri porteissa => lähetä kehys Y:n porttiin

Tietoliikenteen perusteet (2009) Liisa Mattinen

85