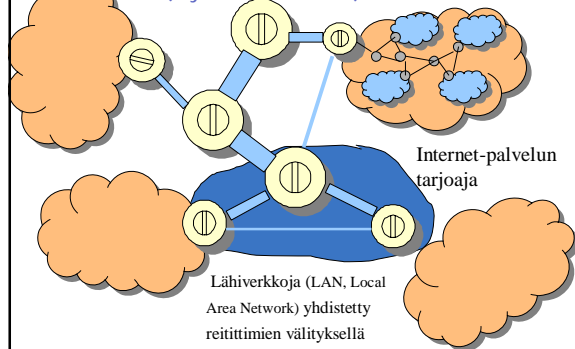


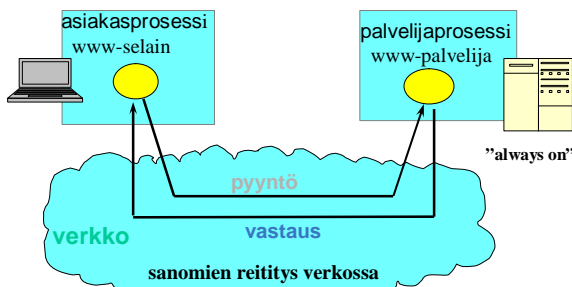
Tietoliikenteen perusteet

Vähän kertausta

Internet = verkkojen verkko (löyhää hierarkiaa)



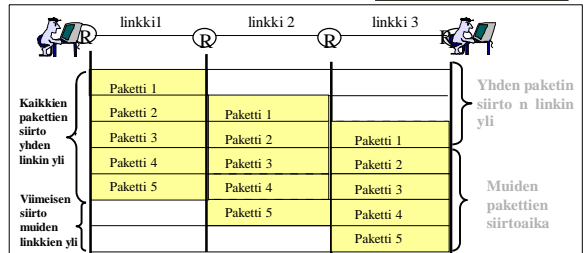
Asiakas-palvelija-malli /vertaistoimijamalli



Oikea kone, oikea prosessi

Pakettivälitys siirto-aika

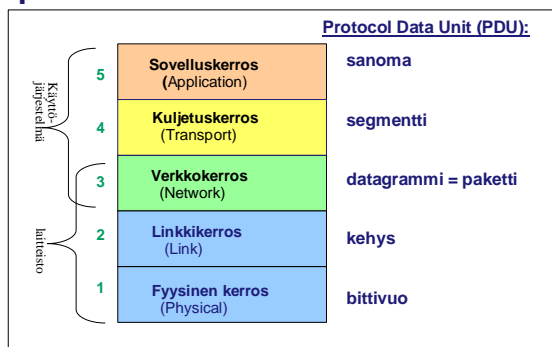
Olkoon siirtoaika a:
 a) $ka+(n-1)a = (k+n-1)a$
 b) $na+(k-1)a = (n+k-1)a$



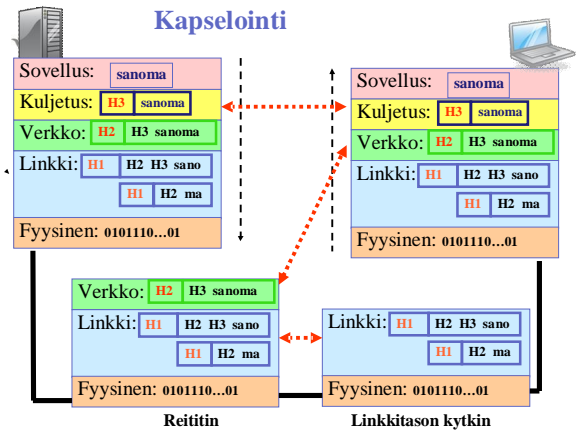
Sanoman siirtoaika, kun sanomassa on k pakettia ja linkejä on n kappaletta
 a) $k \cdot n$ paketin siirto 1. linkin yli + viimeisen paketin siirto n-1 linkin yli.
 b) $1 \cdot n$ paketin siirto n:n linkin yli + muiden k-1 paketin siirto yhden linkin li

Animaatio: http://wps.aw.com/aw_kurose_network_4/63/163034173750.cw/index.html

Internet-protokollapino



Kapselointi



HTTP (HyperText Transfer Protocol)

WWW:N sovellusprotokolla

Tekstimuotoiset sanomat
pyyntö – vastaus

Asiakas
Selain: **FireFox, Internet Explorer, Opera, Apple Safari, ...**
pyytää, noutaa ja näyttää objektit

Palvelija
etsii objektin (tiedoston) koneen hakemistosta ja lähettää sen vastauksena asiakkaalle

Tilaton protokolla
Palvelija ei muista mitään edellisistä pyynnöistä => evästeet (cookies)

PC, jossa on Explorer-selain

Palvelin, jossa on Apache-ww-palvelija

Linux-kone, jossa on Firefox-selain

GET /some-dir/page.html
HTTP/1.1
Host: www.herkkuto.com
User-Agent: Mozilla/4.0
Connection: close
Accept-Language: fr

HTTP/1.1 200 OK
Connection: close
Date: Thu, 22 Feb 2007 12:00:15 GMT
Server: Apache/1.3.3.0 (Unix)
Last-Modified: Mon, 29 Jan 2007 09:23:24 GMT
Content-Length: 6021
Content-Type: text/html
data data data data data ...

Tietoliikenteen perusteet /2009/ Liisa Marttinen 7

origin servers

public Internet

www.herkkuto.com

1.5 Mbps access link

institutional network

10 Mbps LAN

institutional cache

GET /fruit/kiwi.gif HTTP/1.1
Host: www.herkkuto.com

HTTP/1.1 304 Not Modified
Date: Thu, 14 Jul 2007 15:39:29

GET /fruit/kiwi.gif HTTP/1.1
Host: www.herkkuto.com

If-modified-since: Wed, 4 Jul 2007 09:23:24

Tietoliikenteen perusteet /2009/ Liisa Marttinen 8

Sähköpostin komponentit

Push-protokolla
yhteydellinen

User agent

Mail server

User agent

SMTP

User agent

Mail server

User agent

SMTP

Mail server

User agent

Lähtevien sanomien jono

postilaatikat

Tietoliikenteen perusteet /2009/ Liisa Marttinen 9

Hajautettu, hierarkinen tietokanta

Root DNS Servers

com DNS servers

org DNS servers

edu DNS servers

yahoo.com DNS servers

amazon.com DNS servers

pbs.org DNS servers

poly.edu DNS servers

umass.edu DNS servers

KuRo5: Fig 2.18

n 13 juuritason nimipalvelija
Replikoituja, kaikilla samat tiedot

n Yliätason palvelimet maa- ja yleistunnuksille (n. 265 kpl)
..., fi, fr, uk, ... edu, net, com, org, ...

n Autorisoidut aluepalvelimet (domain) (2-taso) www.iana.org
Isoilla yliopistoilla ja firmoilla omansa, pienet käyttävät jonkun muun ylläpitämää

Tietoliikenteen perusteet /2009/ Liisa Marttinen 10

Skaalautuvuus

Asiakas-palvelinmalli:
Palvelimen siirrettävä $n * F$ biittia => 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.

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

KuRo8: Fig. 2.24

Server

Internet

Client 1: u_1, d_1

Client 2: u_2, d_2

Client 3: u_3, d_3

Client 4: u_4, d_4

Client 5: u_5, d_5

Client 6: u_6, d_6

Client 7: u_7, d_7

Client 8: u_8, d_8

Client 9: u_9, d_9

Client 10: u_{10}, d_{10}

File F

aika

Summamerkki

Tietoliikenteen perusteet /2009/ Liisa Marttinen 11

Pistoke (socket)

n Kuljetuspalvelun ja sitä käyttävän sovelluksen rajapinta isäntäkoneessa
Sovelluksen tietoliikenne = KJ:n palvelupyynnöitä
Pistoke on "palveluluukku"

n Alunperin Berkeley UNIXin (BSD) mukana

Sovellus-ohjelmoija

process

socket

Käyttöjärjestelmä

TCP with buffers, variables

internet

host or server

host or server

Sovellus-ohjelmoija

process

socket

Käyttöjärjestelmä

TCP with buffers, variables

KuRo8: Fig 2.26

Tietoliikenteen perusteet /2009/ Liisa Marttinen 12

UDP: Tarkistussumma

32 bittia

Source port #	Dest. Port #
Length	Checksum

Application data (message)

UDP-otsake

Lähetys

- Summaa 16 bittin 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!

```

1 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
1 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 1 1
1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 0
Checksum 0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1
  
```

Tietoliikenteen perusteet /2009/ Liisa Marttinen 13

rdt3.0

KuRo08: Fig 3.15

Tietoliikenteen perusteet /2009/ Liisa Marttinen 14

Liukuhinnoitus: käyttöasteen kasvattaminen

KuRo05: Fig 3.18

$$U_{\text{sender}} = \frac{3 * L / R}{RTT + L / R} = \frac{.024}{30.008} = 0.0008$$

Increase utilization by a factor of 3!

Tietoliikenteen perusteet /2009/ Liisa Marttinen 15

Go-Back-N: Esimerkki

Ikkunankoko = 4

KuRo08: Fig 3.22

Tietoliikenteen perusteet /2009/ Liisa Marttinen 16

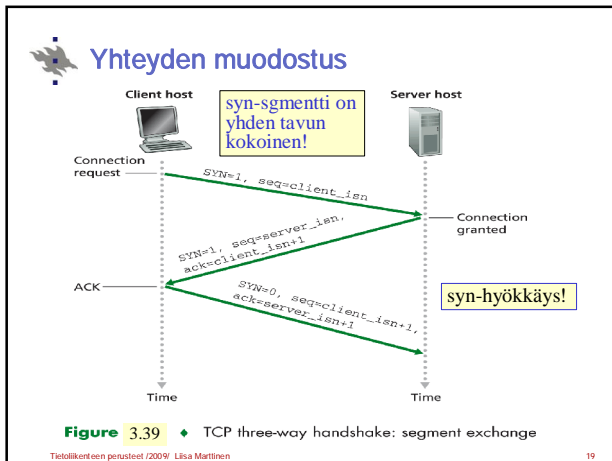
TCP: prosessilta prosessille -tavuvirta

Tietoliikenteen perusteet /2009/ Liisa Marttinen 17

TCP:

- Luotettava, järjestyksen säilyttävä tavuvirta
 - Ei sanomaraajoja
 - Tavunumerointi
 - Checksum-tarkistus
 - Puskurointi uudelleenlähetystä varten
 - Kumulatiiviset kuitaukset
- Yhteydellinen
 - Kolminkertainen käsittely, yhteyden purku
- Vuonvalvonta, ruuhkanhallinta (-valvonta)
 - Lähetäjä ei saa tukahduttaa vastaanottajaa eikä reitittimä
 - Vuonvalvonta: Receive window
 - Ruuhkanhallinta

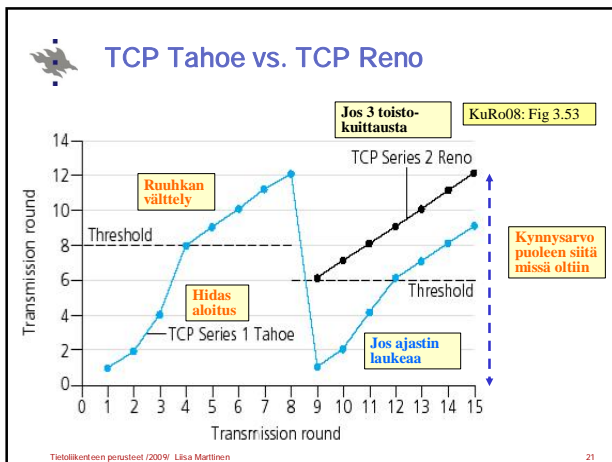
Tietoliikenteen perusteet /2009/ Liisa Marttinen 18



TCP Reno: Hidas aloitus (slow start) ja ruuhkanvälttely (congestion avoidance)

- Aluksi ruuhkalkkuna = yksi segmentti
- Alussa hidas siirtonopeus = MSS/RTT
- Kukin kuitaus kasvattaa yhdellä ruuhkalkkunan kokoa
 - Eksponentiaalinen kasvu
 - Ikkuna kaksinkertaistuu yhden RTT:n aikana
- Jos uudelleenlähetyks, puolita ruuhkalkkunan koko
 - Multiplicative decrease
- Sen jälkeen kasvata ikkunaa yksi segmentti/RTT
 - Lineaarinen kasvu (Additive increase)
- Ruuhkan välttely (congestion avoidance)
- Siirtonopeus = $CognWin / RTT$ tavua/sek

Tietoliikenteen perusteet / 2009/ Lissa Marttinen 20

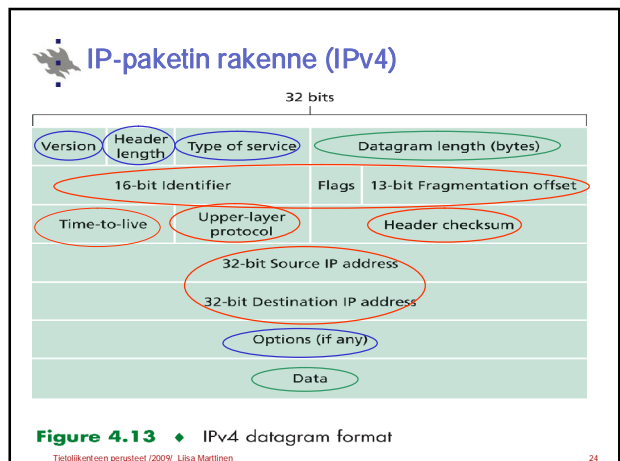
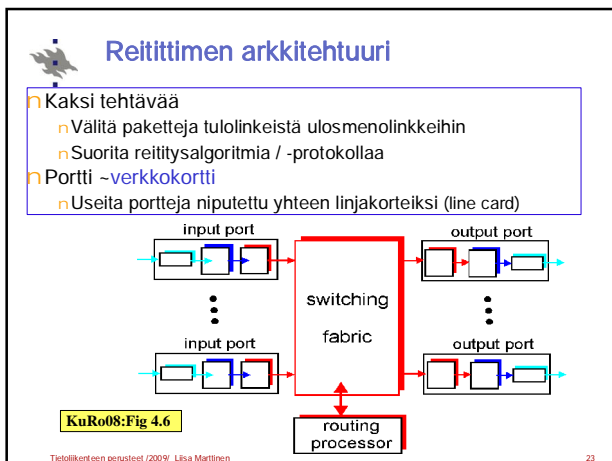


Verkkokerros

- Toimittaa kuljetuskerroksen segmentit vastaanottajalle
- Lähetäjä
 - luo segmenteistä verkkokerroksen IP-paketteja
 - Lisää otsaketietoja: mm. IP-osoitteet
- Reitittäminen
 - Isäntä - reititin ... reititin - Isäntä
- Vastaanotto
 - Polsta otsake
 - Anna segmentti kuljetuskerrokselle
- Verkkokerros toimii etenkin reitityksessä
 - Reititin tutkii IP-paketin otsakkeen ja päättää, mihin linkkiin se lähetetään seuraavaksi

KuRo08: Fig 4.1

Tietoliikenteen perusteet / 2009/ Lissa Marttinen 22



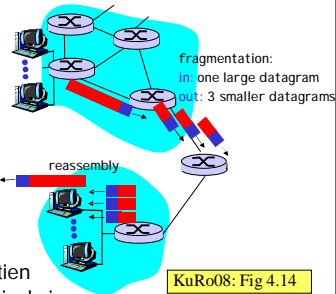
IP-pakettien paloittelu (fragmentointi)

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 ja kokoamiseksi



KuRo08: Fig 4.14

Tietoliikenteen perusteet /2009/ Liisa Marttinen

25

Esimerkki

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

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

Yhdestä IP-paketista tulee 3 pienempää IP-pakettia

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

1480 B dataa
20 B IP-otsaketta

offset = 1480/8

0	1480	2860
1. Pala: 1480 tavua	2. Pala: 1480 tavua	3. Pala: 1020 tavua

Tietoliikenteen perusteet /2009/ Liisa Marttinen

26

CIDR: Classless InterDomain Routing

Verko-osa voi olla minkä tahansa kokoinen

Vanha luokallinen osoite: A-luokka 8 b, B-luokka 16 b, C-luokka 24 b

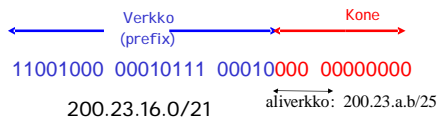
Formaatti: a.b.c.d/x

x ilmoittaa verko-osan bittien lukumäärän (prefix)

Esim. Organisaatio, jolla 2000 konetta varaa 2024 = 2¹¹

konenumeroa, jolloin verkko-osaa varten jää 21 bittiä

Yritys voi vielä itse jakaa viimeiset 11 bittiä aliverkko-osoiteeksi ja koneosioiteeksi. Tämä jako ei näy ulkopuolelle.



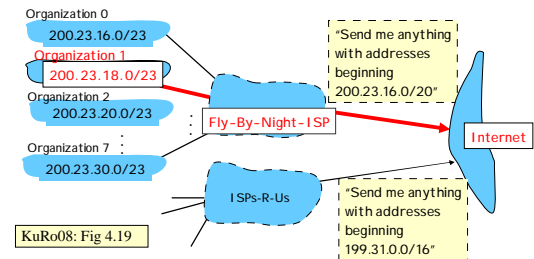
Tietoliikenteen perusteet /2009/ Liisa Marttinen

27

Hierarkkinen osoite

CIDR luo reititystä helpottavan hierarkian

Aggregointi (yhdistäminen): yhteinen alkuosa => samaan suuntaan

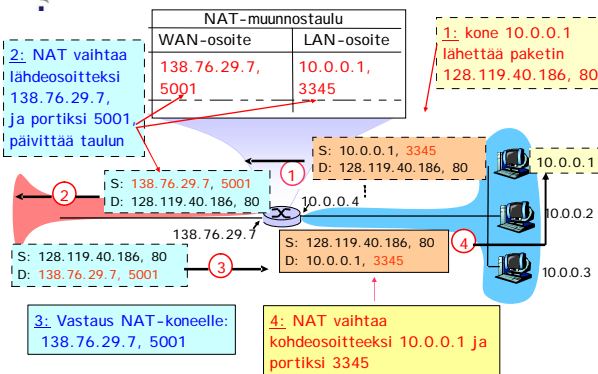


KuRo08: Fig 4.19

Tietoliikenteen perusteet /2009/ Liisa Marttinen

28

NAT: Esimerkki



Tietoliikenteen perusteet /2009/ Liisa Marttinen

29

Reititysalgoritmi

Etsii edullisimmat reitit lähdekoneelta kohdekoneelle

Käytetään reititystaulun muodostamiseen

Mille linkille paketti seuraavaksi siirretään tältä reitittimeltä

Reititysalgoritmi, joka tarvitsee täydellisen tiedon verkosta

Ennen laskentaa käytössä koko kuva verkosta:

Kaikki linkkiyttyvät solmujen välillä ja niiden kustannukset

Käytännössä vain tietyistä autonomisista alueista

Parhaat reitit lasketaan joko keskitetysti tai hajautetusti

Linkktila-algoritmi (link-state algorithm)

Reititysalgoritmi, jolle riittää epätäydellinen kuva verkosta

Aluksi reititin tietää vain niistä koneista, joihin itse on yhdistetty

Iteratiivinen algoritmi: reititin vaihtaa tietoja naapuriensa kanssa ja saa tietoa muusta verkosta

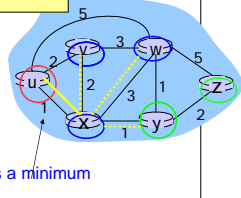
Etäisyysvektorialgoritmi (distance vector algorithm)

Tietoliikenteen perusteet /2009/ Liisa Marttinen

30

Dijkstran algoritmi

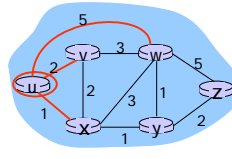
1 **Initialization:**
 2 $N' = \{u\}$
 3 for all nodes v
 4 if v adjacent to u
 5 then $D(v) = c(u,v)$
 6 else $D(v) = \infty$
 7
 8 **Loop**
 9 find w not in N' such that $D(w)$ is a minimum
 10 add w to N'
 11 update $D(v)$ for all v adjacent to w and not in N' :
 12 $D(v) = \min(D(v), D(w) + c(w,v))$
 13 /* new cost to v is either old cost to v or known
 14 shortest path cost to w plus cost from w to v */
 15 until all nodes in N'



Etäisyysvektoreireititys:

Esimerkki 1

Jos on jo saatu selville
 (= naapurit kertoneet), että
 $D_v(z) = 5, D_x(z) = 3, D_w(z) = 3$



Kohde | kust. linkki

Z 4 X:ään

$$D_u(z) = \min \{ c(u,v) + d_v(z), c(u,x) + d_x(z), c(u,w) + d_w(z) \}$$

$$= \min \{ 2 + 5, 1 + 3, 5 + 3 \} = 4$$

Kun paketti on matkalla solmusta u solmuun z, se tulee seuraavaksi lähettää solmuun x, joka tuotti tuon minimin => talleta tieto omaan etäisyysvektoriin (= reititystauluun)

Huono uutinen etenee hitaasti!

Linkki AB katkeaa => etäisyys äärettömäksi	$D_B(A)$	$D_C(A)$	$D_D(A)$	$D_E(A)$
äärettömän	2	3	4	4
3	2	3	4	4
4	3	4	5	6
5	4	5	6	7
6	5	6	7	8
7	6	7	8	
8	7	8		

Joka vaihdossa 'paras arvio' huononee vain yhdellä = reitityssilmukka

Count-to-infinity -ongelma

Etäisyys A:han

Linkkikerros

- in Laittoimintoa
- in Siirtää paketin fyysisestä linkistä pitkin koneelta (solmulta (node)) toiselle
 - langallinen / langaton
 - bitit sisään, bitit ulos
- in Kapseloi paketin siirtoon sopivaan muotoon
 - in Siirtokehys (frame)
- in Lähiverkossa linkkejä voi yhdistää keskittimillä tai kytkimillä
 - in Käytetään fyysisiä osioitelta
 - in 'reititystä' ilman IP-osioitelta

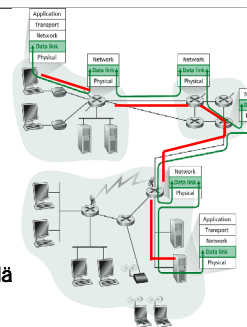


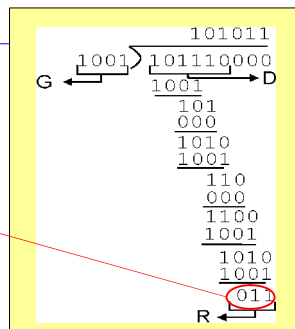
Figure 5.1 • The link layer

CRC-esimerkki

Data: 101110
 G: 1001, polynomina
 $1*x^3 + 0*x^2 + 0*x^1 + 1*x^0$
 $\langle D, R \rangle$: 101110???

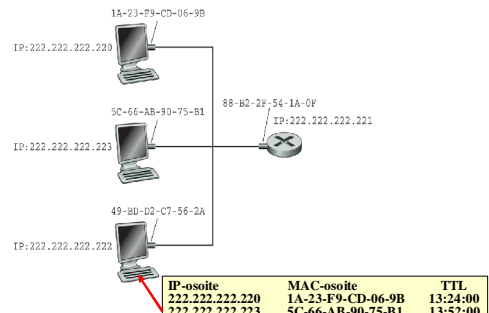
Lähetä: 101110011

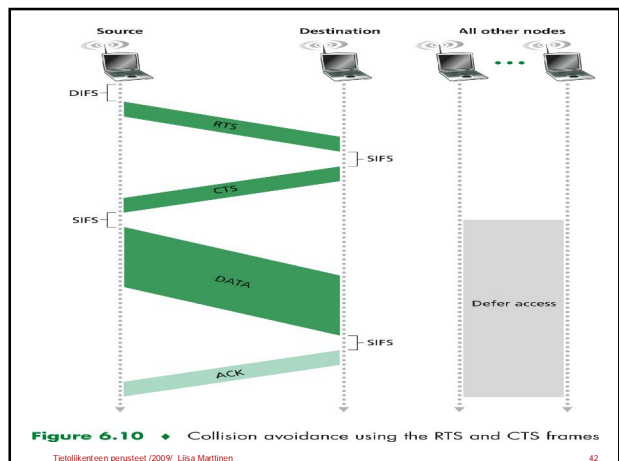
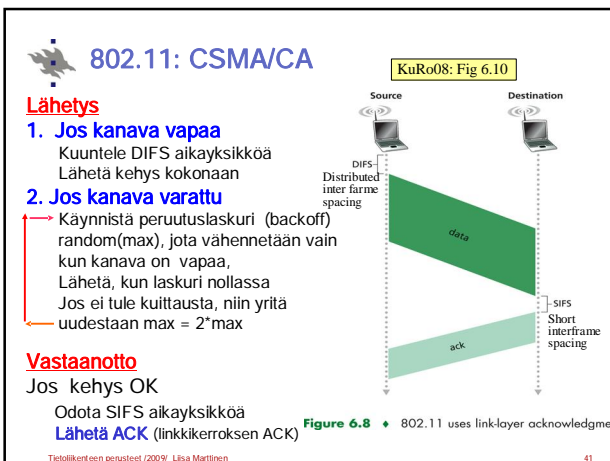
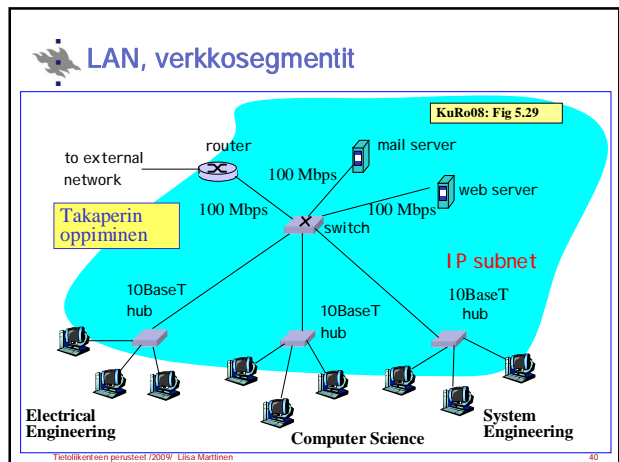
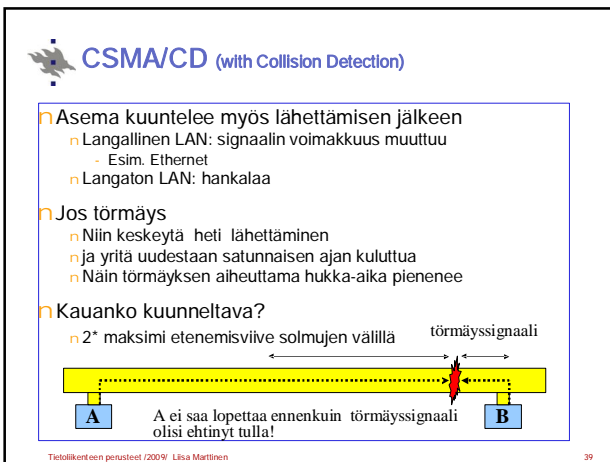
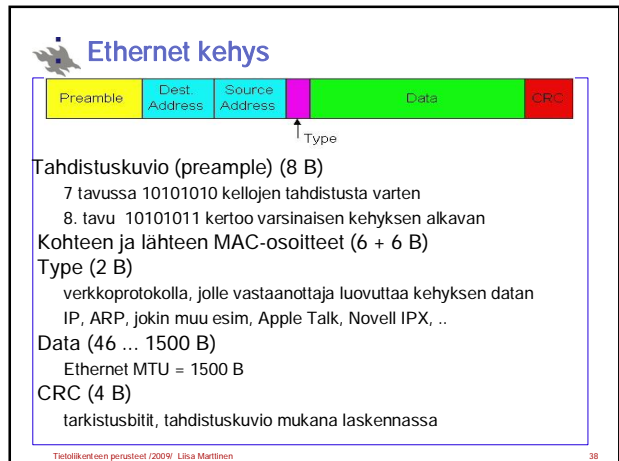
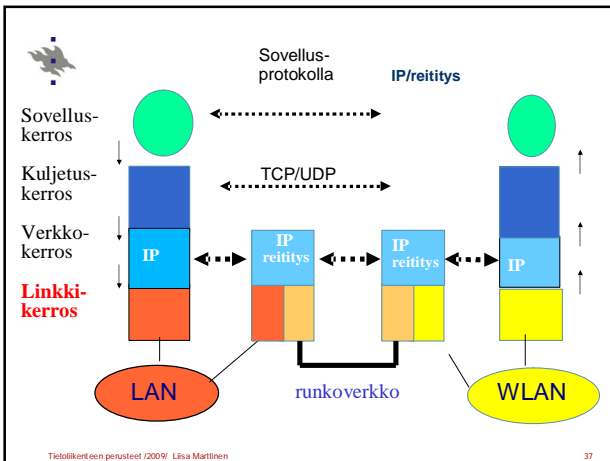
Modulo 2-aritmetiikka
 vähennyslasku yhteenlaskuja ei lainaamista, ei muistinumeroita = bittitason XOR
 $1+1=0, 1+0=0+1=1, 0+0=0$



KuRo08:Fig 5.8

MAC-osoitteet ja ARP-taulu, ARP-protokolla





Hajautettu DoS-hyökkäys (DDoS)

- Hyökkääjä ottaa ensin haltuun ison joukon koneita niiden omistajien huomaamatta
 - Koputtelee ja löytää turva-aukot
 - Asentaa hyökkäysohjelman, joka vain odottelee käskyä /kelloyömiää
- Kaapatut koneet aloittavat samaan aikaan hyökkäyksen uhrin kimppuun
 - Hajautetusti
 - IP-osoitteet peukaloituina

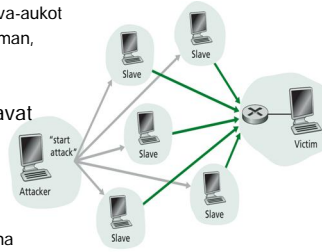
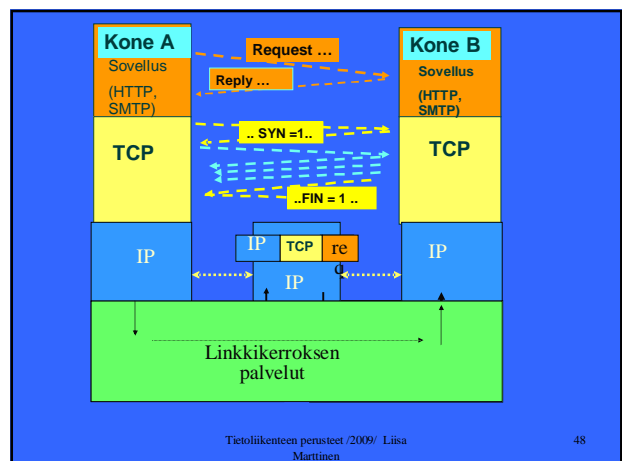
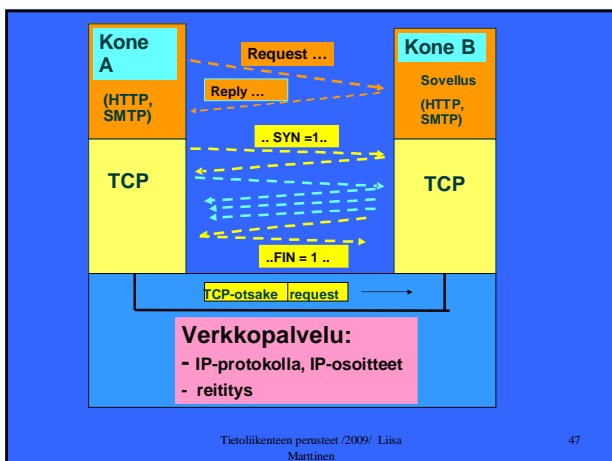
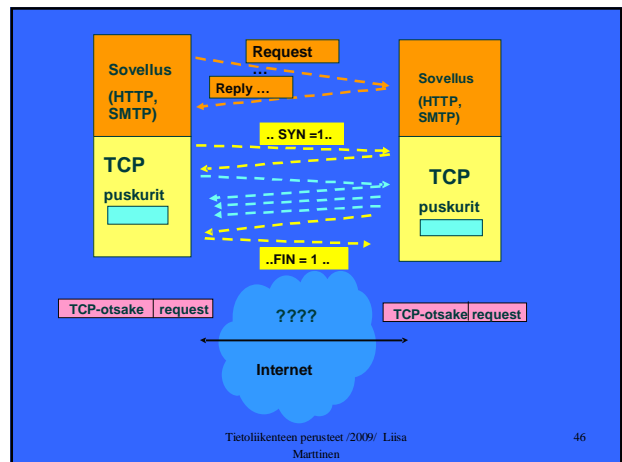
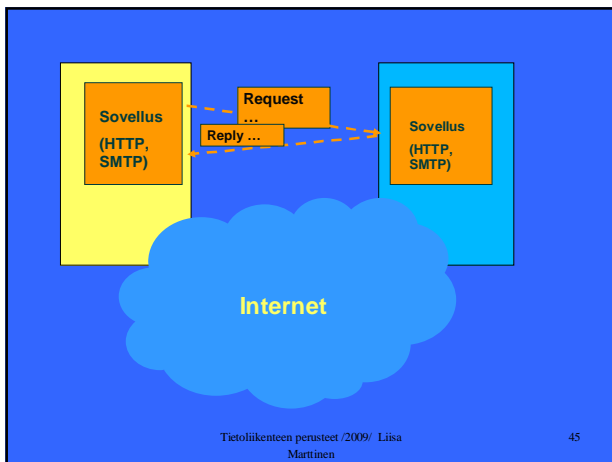
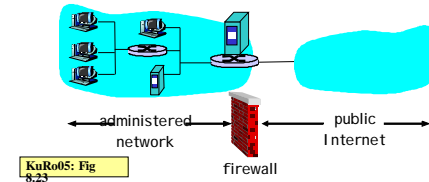
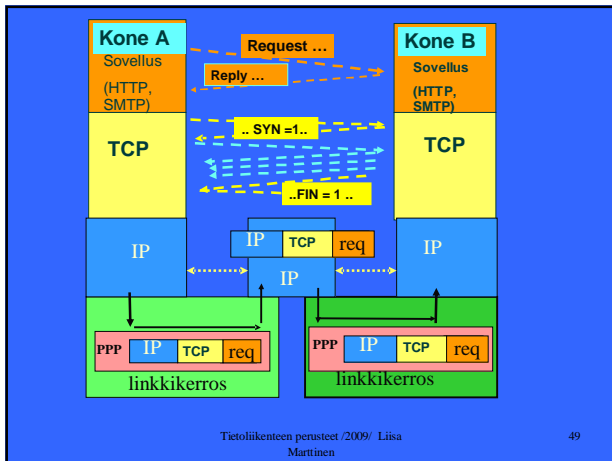


Figure 8.26 A DDoS attack

Palomuri (firewall)

- Ohjelmisto + laitteisto
- Suodattaa (filteroi) liikennettä organisaation oman verkon (intranet) ja julkisen Interbetin välillä
 - Osa IP-paketeista pääsee palomuurin läpi, osa ei





Tietoliikenteen perusteet

A cartoon illustration featuring several children engaged in various activities: one is jumping joyfully, another is reading a book, one is pushing a toy cart, and another is lying on the floor reading. The text "Siinäpä se!" (There it is!) is written in the center of the scene.

Tietoliikenteen perusteet /2009/ Liisa Marttinen 50