Luento 11: Tietoturvasta ja kertausta

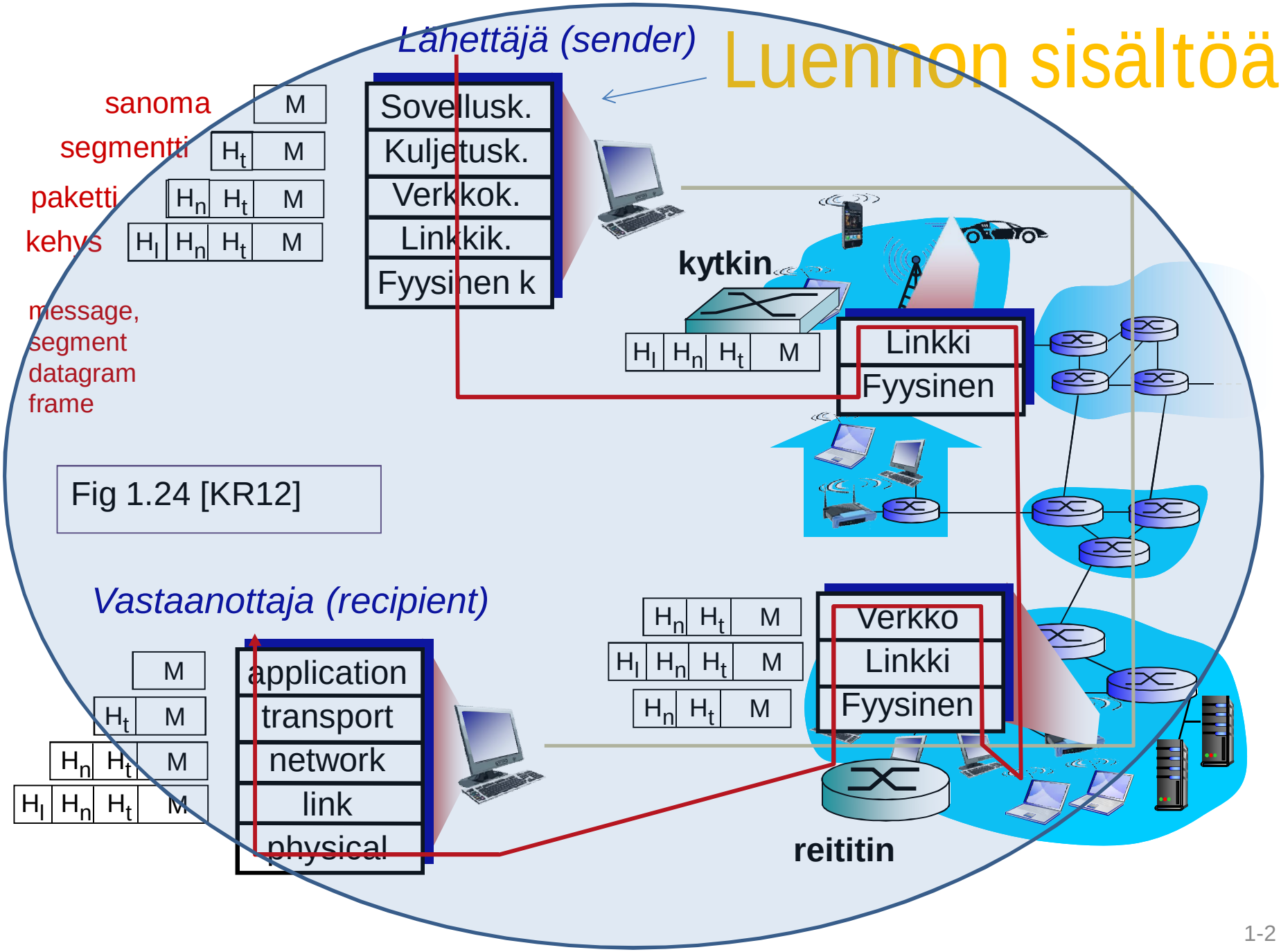
3.12.2012

Tiina Niklander

Kurose&Ross
Ch 1.6, Ch 8.1, Ch 8.9.1

Pääasiallisesti kuvien
© J.F Kurose and K.W. Ross,
All Rights Reserved

Luennon sisältöä



Sisältö

Tietoturva-kurssit:
- kryptografian perusteet
- IPSec

Turvavaatimukset

Uhkia

Palomuuri

Oppimistavoitteet:
- Osata kuvailla tietoliikenteeseen kohdistuvat riskitekijät ja turvallisuusuhat
- Osata selittää, kuinka palomuuri toimii
- Ymmärtää tietoturvasta sen verran, että osaa huolehtia oman koneen turvallisuudesta

Tietoturvasta

Turvavaatimukset

Ch 8.1

Turvavaatimukset

- Luottamuksellisuus (confidential, secrecy)
  - Vain lähettäjä ja vastaanottaja 'ymmärtävät' sanoman sisällön
  - Muu eivät saa välttämättä tietoa edes sen olemassaolosta (esim. Salakirjoitus)
- Autentikointi (authentication)
  - Lähettäjä ja vastaanottaja varmistuvat toistensa identiteeteistä
    - Oikeaksi todentaminen, salakirjoitus
- Eheys, koskemattomuus (message integrity)
  - Lähettäjä ja vastaanottaja varmoja siitä, ettei sanomaa ole muutettu siirron aikana tai myöhemmin (esim. Digitaalinen allekirjoitus)
- Palveluiden saatavuus ja suojaus
  - Palvelut ovat saatavilla käyttötarkoituksen mukaisesti
  - Vain niillä pääsy, joilla lupa käyttää käyttöoikeuksien mukaisesti
    - Käyttäjätunnus ja salasana, tiedostojen / objektien käyttöoikeudet, …
  - Suojautuminen ‘ulkoa’ tulevia hyökkäyksiä vastaan (haittaohjelmat, palvelunestohyökkäys) vastaan
    - palomuuri, havaitsemis- ja puhdistusohjelmat

Ystävä ja tunkeutuja

Tuttu asetelma reaalimaailmastakin

- Bob ja Alice kommunikoivat keskenään (salassa muilta?)
- Trudy (intruder) voi siepata sanomia: nuuskia, kerätä tietoa
- Trudy voi muunnella, tuhota ja lisätä sanomia

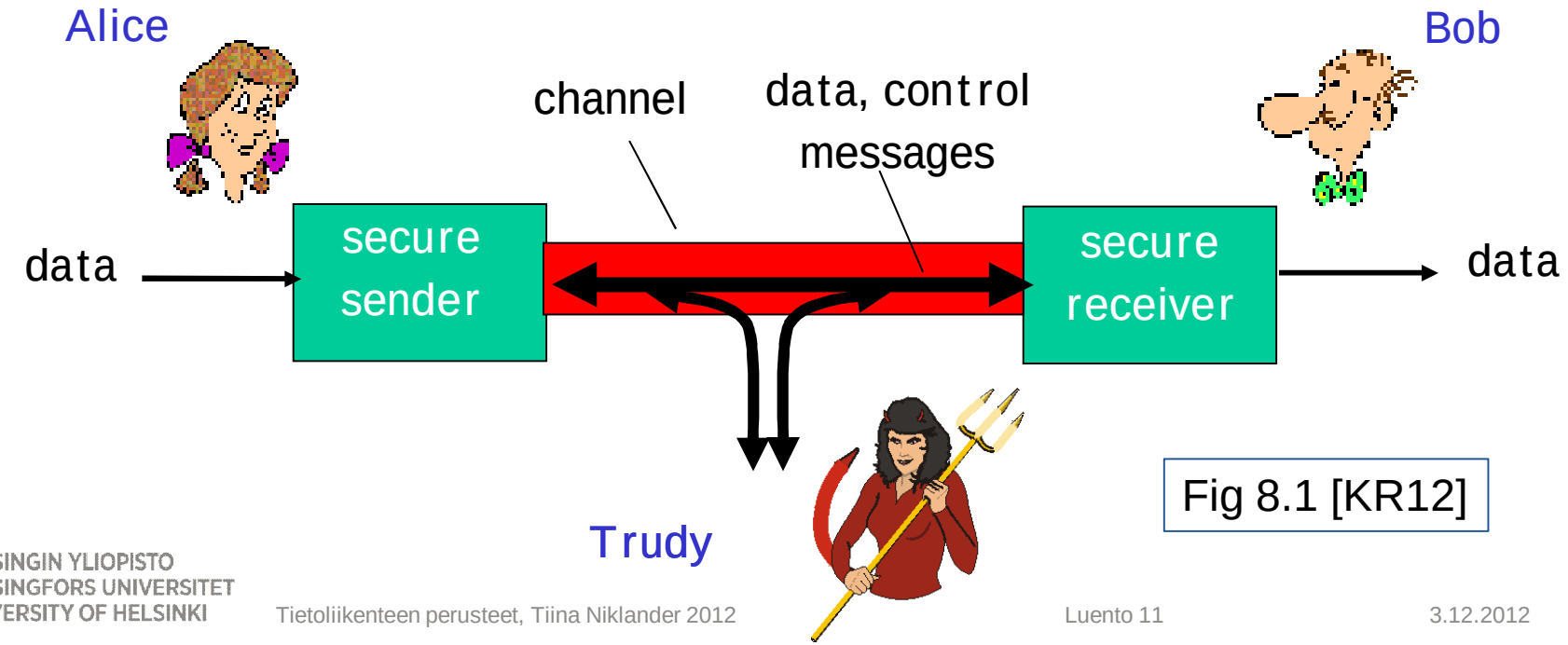


Fig 8.1 [KR12]

Kuka Alice, kuka Bob?

- Asiakasprosessi - palvelijaprosessi
  - Ihminen koneen ääressä ja palvelu palvelinkoneessa
- Web-selain ja -palvelija
  - Elektroninen kaupankäynti
  - On-line pankkipalvelu
  - .....
- DNS-kysely ja DNS-palvelu
- Reititystietoja vaihtavat reitittimet

....

Tietoturvasta

Järjestelmän tietoturvan määrittää sen tietoturvan kannalta heikoin elementti

Tietoturva pitää ottaa huomioon alusta asti järjestelmäsuunnittelussa

Protokollat pitää suunnitella niin että niitä voidaan päivittää (esim. SHA-1 → SHA-256)

Käytännössä tietoturvaratkaisuita tarvitaan useilla kerroksilla

- Linkki (WPA, WEP, EAP, 802.1X, 2G/3G)
- Verkko (IPSec); Välikerroksella HIP (Host Identity Protocol)
- Kuljetus (TLS)
- Sovellus (HTTPS, Radius, Diameter, S/MIME, XML Security…)

Salausmenetelmiä

Asymmetrinen

Julkinen avain (public key), salainen avain (private key)

Julkisten avainten jakelu

Salaus (A -> B):

- A salaa B:n julkisella avaimella
- B avaa salaisella avaimellaan

Allekirjoitus:

- A allekirjoittaa salaisella avaimellaan
- B tarkistaa allekirjoituksen A:n julkisella avaimella

Symmetrinen

Jaetut salasanat

Yleensä julkisen avaimen salausta käytetään symmetrisen avaimen muodostamiseen tiettyä yhteyttä varten

Tietoturvasta

Uhkia

Ch 1.6

Mitä Trudy puuhii?

- Koputtelee koneen portteja (mapping)
  - Turva-aukkojen löytämiseksi ja koneen valtaamiseksi
- Salakuuntelee (eavesdropping, sniffing)
  - Sieppaa sanoman matkalla ja tutkii sisällön
- Väärentää, “peukaloi” (impersonation, spoofing)
  - Vaihtaa paketin tietoja, esim. IP-osoitteen
- Tehtailee sanomia, “satuilee” (fabrication)
  - Tekee ja lisää liikenteeseen ylimääräisiä sanomia
- Kaappaa yhteyden (hijacking)
  - Vaihtaa oman IP-osoitteen lähettäjän / vastaanottajan tilalle
- Estää palvelun (DoS, Denial of Service)
  - Kuormittaa palvelinta, jotta se ei ehdi palvella oikeita käyttäjiä

Koputtelu ja kartoitus (mapping)

Kaivelee ensin taustatietoja

- IP-osoitteista, käyttöjärjestelmistä, verkko-ohjelmista

Hyödyntää sitten tunnettuja turva-aukkoja

Ping

- Lähettää kyselyjä valittuihin verkon IP-osoitteisiin
- Hengissä olevat koneet vastaavat

Koputtelu ja kartoitus (mapping)

Porttiselaus (port scanning)

- Kokeilee systemaattisesti TCP/UDP-yhteyttä koneen portteihin
- Vastauksista saa selville tarjotut palvelut
- Onko niissä tunnettuja turva-aukoja?
  - Firefox-selain 27.3.08, Facebook 25.3.08, Sampo Pankki, Applen Quicktime Player, FlashPlayer turva-aukkojen paikkausta
  - Internet Explorer 7, DNS, BGP, …
  - Linux-päivityksen turva-aukko => laitoksen salasanojen vaihto (pari vuotta sitten)

Salakuuntelu (packet sniffing)

Tutkii linkkikerroksen kehysten sisältöä

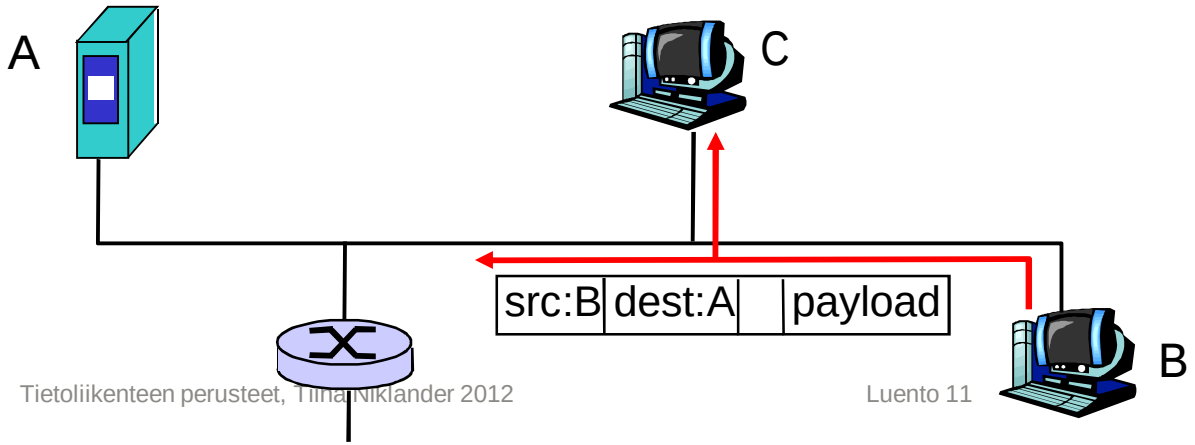
- Yleislähetys: kaikki kuulevat kaikki kehykset
- Valikoimattomassa moodissa (promiscuous) toimiva sovitinkortti myös kopioi kaikki kehykset itselleen
- Kuuntelevan koneen oltava samassa LAN:ssa

Ohjelmia, joilla paketit voidaan purkaa tekstimuotoon

- Hyödyllisiä verkon valvojalle, mutta ...

Hyökkääjä etsii erityisesti salasanoja

- Salasanat verkkoon vain salakirjoitettuina
- Älä käytä telnet:iä etäyhteyksiin, käytä ssh:ta (leap of faith security)



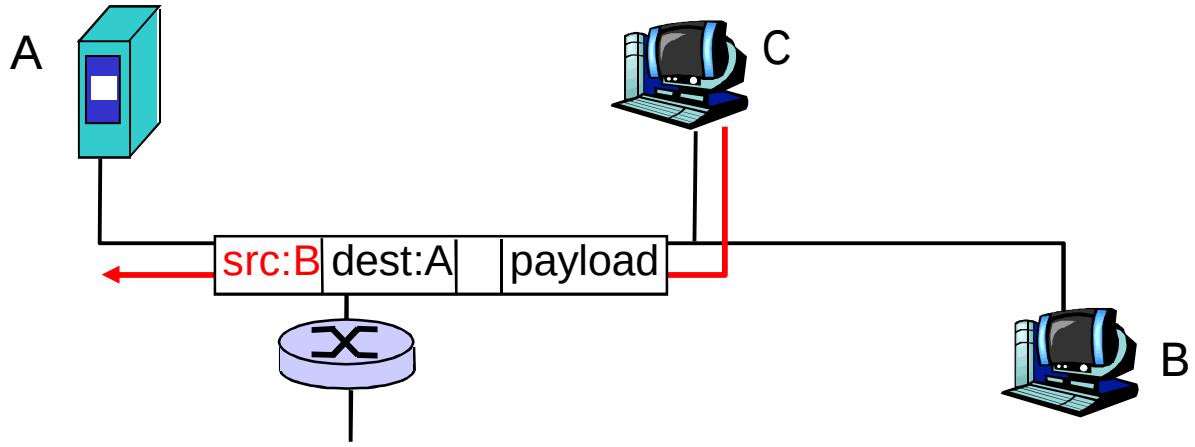
Väärentäminen (spoofing)

Vastaanottaja ei voi tietää, kuka on todellinen lähettäjä

Jokainen, joka kontrolloi koneensa ohjelmistoa

(erityisesti KJ:tä) voi väärentää mm. IP-osoitteen

Sovellus voi tehdä itse IP-paketin ja ohittaa KJ:n pakettia lähettäessä ('raw' mode)



Palvelunestohyökkäys (DoS)

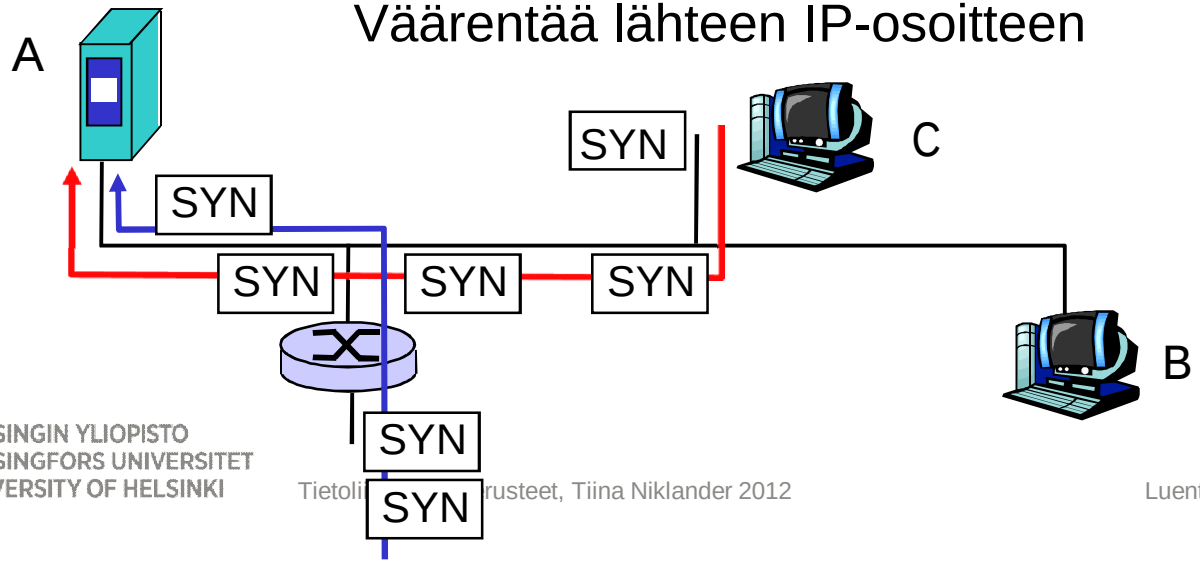
Kuormittaa palvelua, jotta oikeat käyttäjät eivät pääse lainkaan käyttämään

SYN-tulvitus

Pakottaa uhrin suuriin määriin TCP-yhteydenmuodostuksia

- Lähettää SYN-segmenttejä, mutta ei ACK-segmenttejä
- Uhri varaa puskuritilaa, muisti voi loppua

Väärentää lähteen IP-osoitteen



Palvelunestohyökkäys (jatkuu)

IPv4-paloittelu

Lähettää runsaasti IP-pakettien osia (M=1), mutta ei lainkaan sitä viimeistä palaa (M=0).

Vastaanottaja puskuroi ja jää odottamaan puuttuvia paloja

– Muisti loppuu

Smurf-hyökkäys

Lähettää suurelle määrälle koneita uhrin IP-osoitteella varustettuja ICMP Echo request -paketteja ja niihin tulevat vastaukset tukkivat uhrin koneen.

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/kellonaikaa

Kaapatut koneet aloittavat samaan aikaan hyökkäyksen uhrin kimppuun hajautetusti

- IP-osoitteet peukaloituina (harvoin)

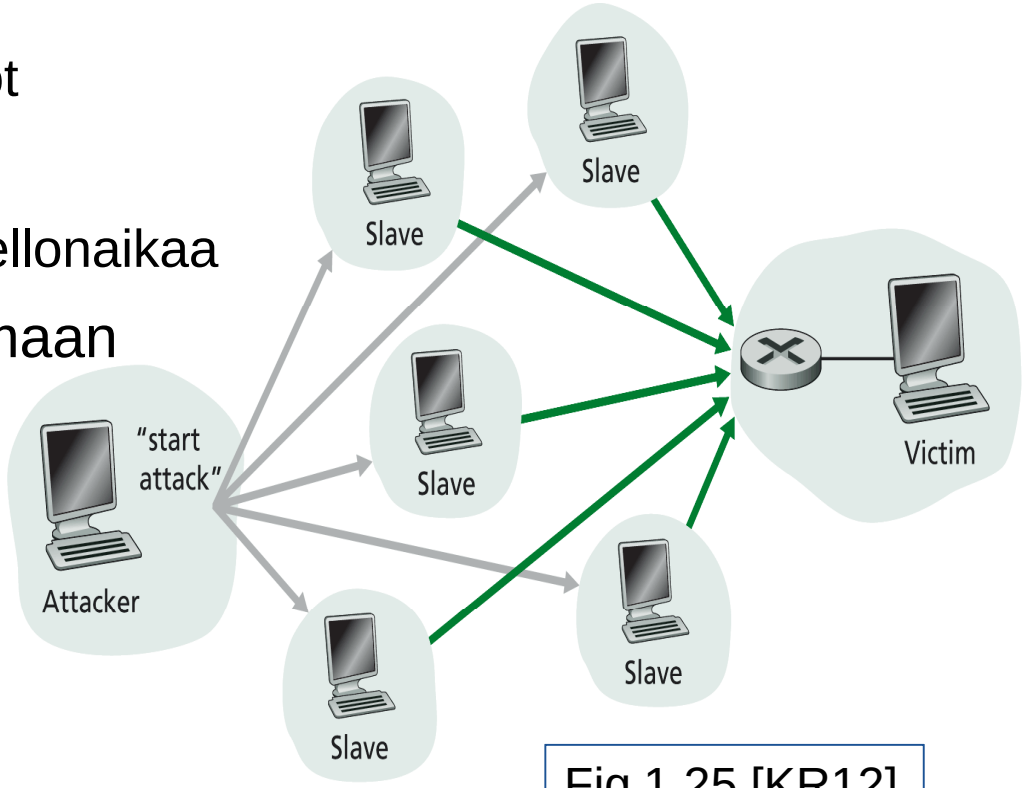


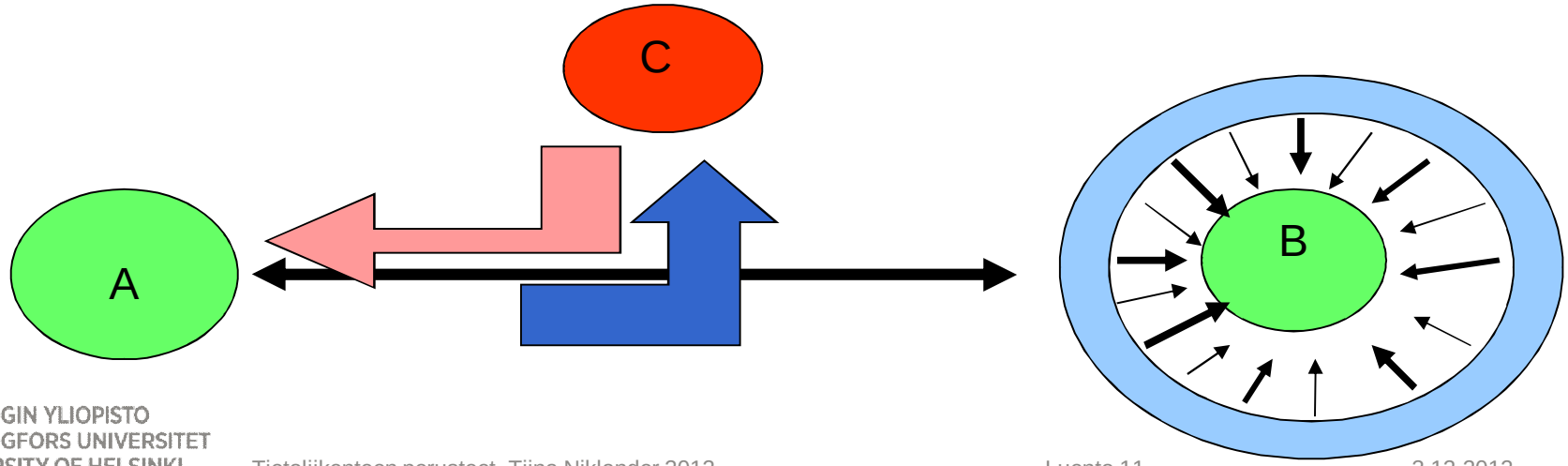
Fig 1.25 [KR12]

Figure 8.26 ♦ A DDoS attack

Yhteyden kaappaus (hijacking)

Hyökkääjä C kaappaa itselleen A:n ja B:n välisen yhteyden

- Kuuntelee ensin yhteyttä ja selvittää mm. tavunumeroinnin, kuittausnumeroinnin, ikkunan koon, ...
- Poistaa B:n pelistä palvelunestohyökkäyksellä
- Tekeytyy itse B:ksi
- Oltava fyysisesti kytkettynä linkkiin



Haittaohjelma (malware) (1)

Itseään monistava: kun on saastuttanut yhden koneen, pyrkii levittämään kopioitaan muihin koneisiin

Virus

Tarvitsee isännän levitäkseen ja vaatii yleensä käyttäjän toimintoa

Sähköpostin liitetiedosto, joka avataan

Mato

Tulee tietoturva-aukosta ja leviää automaattisesti (Sasser) Slammer (2003 kaatoi 5 nimipalvelijaa))

Levinneimmät madot kyllä kulkivat sähköpostin liitetiedostoina

- Morrisin mato (1988), Melissa (1999), Nimda (2001),Sobig (2003), ILoveYou,

Downadup (2007-2008): hyödyntää Microsoftin Windows-käyttöjärjestelmässä joulukuussa löytynyttä turvareikää, arvaa verkon salasanoja ja tartuttaa USB-muistitikkuja

Haittaohjelma (2)

Troijalainen

on ohjelma, joka sisältää myös jotakin muuta kuin käyttäjä uskoo sen sisältävän. Suorittaa kyllä jonkun hyödyllisen toiminnon

Mutta lisäksi se voi

- käynnistää viruksen, madon,
- avata takaportin tai muun haavoittuvuuden tietojärjestelmään
- tehdä tiedonhakua, tietojen tuhoamista tai vastaavaa jopa jättämättä mitään jälkiä.

Vastatoimet? (1)

Pidä KJ:n turvapäivitykset ajan tasalla!

- Koputtelu
  - Käytä palomuuria
  - Seuraa liikennettä, reagoi, jos normaalista poikkeavaa
  - Seuraa aktiviteettia (IP-osoite, porttien koputtelu)
- Salakuuntelu
  - Käytä kaksipisteyhteyksiä; Ethernet-kytkin keskittimen sijasta
  - Salakirjoitus
  - Tarkista, ettei verkkokortti ole promiscuous-moodissa
- IP-osoitteen väärentäminen
  - Lähetysverkossa helppo havaita ja estää
  - Yhdyskäytäväreititin voi tarkistaa, että lähettäjän IP-osoite kuuluu lähettävään verkkoon (ingress filtering)
  - Tutkimista ei voi tehdä pakolliseksi

Vastatoimet (2)

- Palvelunesto
  - Vaikea todeta / estää
  - Miloin SYN on oikea yhteyspyyntö, millloin osa hyökkäystä?
  - Hyökkäyksen havaitsemis- ja estämisjärjestelmät
  - SYN cookie (seuraava kalvo)
  - ISP Hotline
- Haittaohjelmat
  - Turva-aukkopäivitysten asentaminen heti
  - Varovaisuus sähköpostiliitteiden kanssa
  - Älä asenna tai käytä ’tuntemattomia’ ohjelmia
  - Käytä palomuuria ja virustorjuntaohjelmia

Tietoturvasta

Palomuuri

Ch 8.9.1

Palomuuri (firewall)

Ohjelmisto + laitteisto

Suodattaa (filteroi) liikennettä organisaation oman verkon (intranet) ja julkisen Internetin välillä

Osa IP-paketeista pääsee palomuurin läpi, osa ei

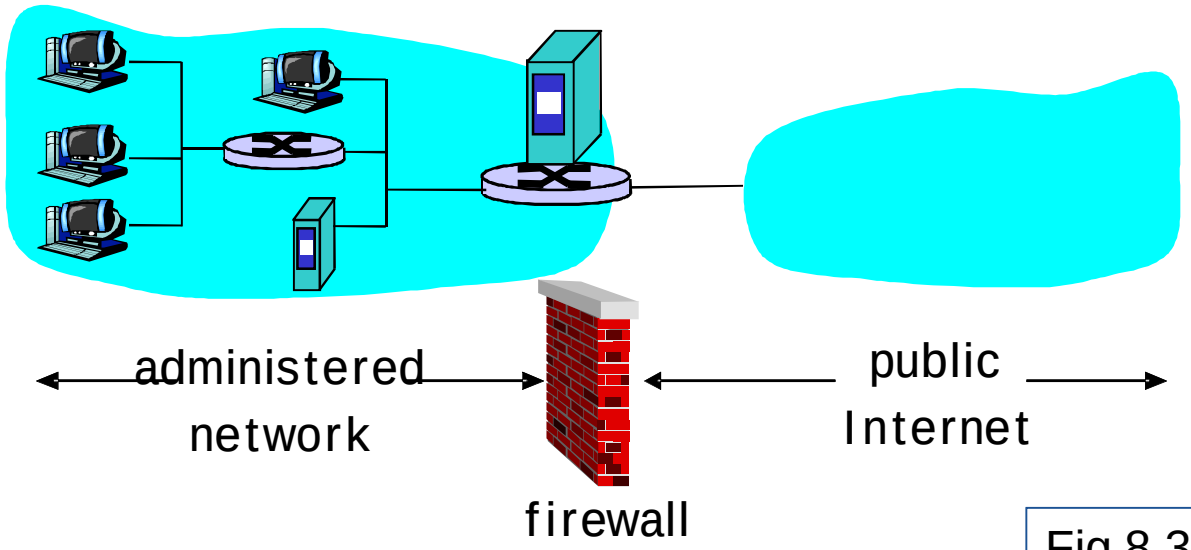
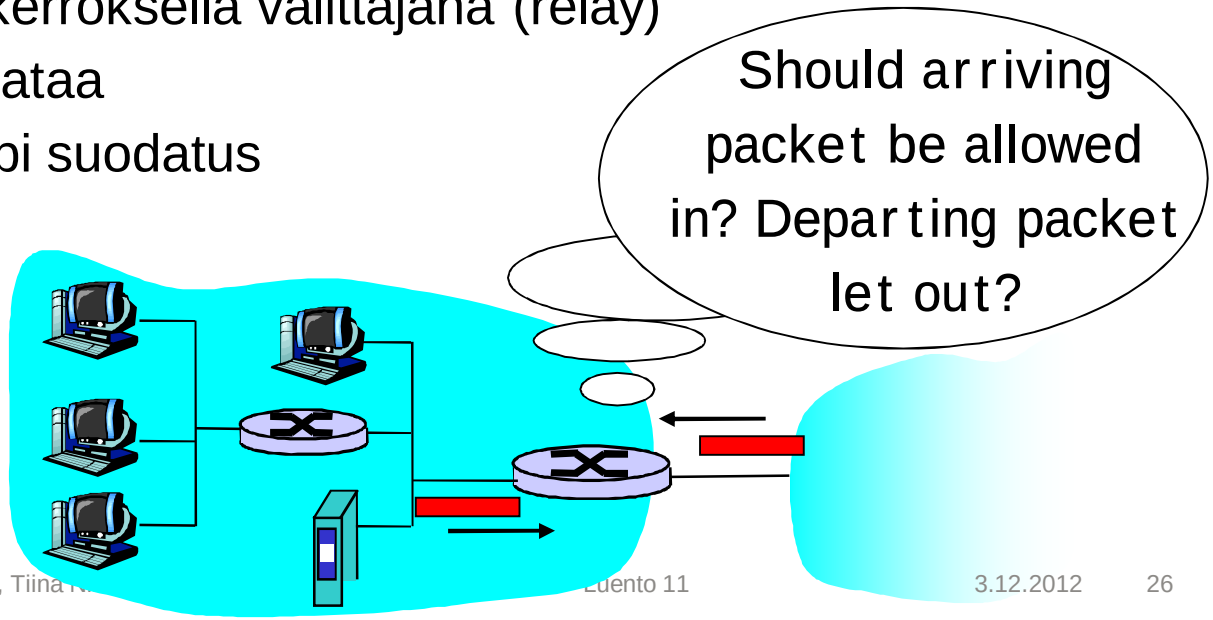


Fig 8.33 [KR12]

Kaksi erilaista palomuuria

Paketteja suodattava palomuuri (packet filtering firewall)
- Toimii verkkotasolla (reititys)
- Tutkii pakettien IP- ja TCP/UDP-otsakkeita
- Karkea suodatus

Sovellustason yhdyskäytävä (application-level gateway)
- Toimii sovelluskerroksella välittäjänä (relay)
- Tutkii sovellusdataa
- Hienojakoisempi suodatus



Palomuuri ja suodatus

- Ennalta annetut säännöt suodatukselle
  - Salliiko vai kieltääkö paketin etenemisen
- Säännöt otsakekenttien perusteella
  - Lähettäjän ja vastaanottajan IP-osoite
  - Protokollan tyyppi
  - TCP- ja UDP-porttinumerot
  - Kontrollisanoman (ICMP) tyyppi
  - TCP:n kättelysegmenttien SYN / ACK-bitit
- Eri säännöt lähteville ja tuleville paketeille
- Eri säännöt eri linkeille

Palomuuri ja suodatus (jatkuu)

- Esim 1: Estä IP-pakettien liikenne (sisään/ulos), jos protokolla = 17 tai portti = 23
  - Palomuuri hävittää kaikki UDP-paketit ja estää telnet-yhteydet
- Esim 2: Estä sellaisten tulevien TCP-pakettien liikenne, joissa ACK = 0
  - Vain ensimmäisessä segmentissä SYN = 1, ACK = 0
  - Palomuuri hävittää kaikki ulkoa tulevat TCP-yhteyspyyntöpaketit
  - Oman verkon koneet voivat silti ottaa yhteyttä organisaation ulkopuolisiin palveluihin

www.cert.org/tech_tips/packet_filtering.html

Tilallinen pakettien suodatus
(Stateful packet filter)

Säännöillä on hankala toteuttaa monimutkaisia estopolitiikkoja

Sääntöjä tarvitaan helposti paljon, jopa tuhansia

Niitä käydään läpi jossain järjestyksessä => väärä järjestys voi aiheuttaa ongelmia / virheitä paketin käsittelyssä

Suodatus kohdistuu yksittäiseen pakettiin

Tilallinen pakettien suodatus

Suodatin tietää, mitkä TCP-yhteydet ovat käytössä

- SYN, SYNACK ja ACK => yhteys muodostetaan
- FIN-paketit => yhteys puretaan / poistetaan, jos ei käytetä (60 s)
- Taulukko voimassa olevista TCP-yhteyksistä

Esim. intranetistä lähetetty web-kysely => päästetään vastaus läpi

Sovellustason yhdyskäytävä
(Application gateway)

Kun halutaan hienojakoisempaa suodatusta

Esim. Telnet-yhteyden salliminen tunnetuille käyttäjille, mutta näiden identiteetti on ensin todettava (autentikointi)

Tähän pelkkä IP/TCP/UDP-otsakkeiden tutkiminen ei riitä

Toimii välittävänä koneena (relay)

sisäverkon ja Internetin välissä

Eri sovelluksilla oma yhdyskäytäväpros.

Esim. IMAP, SMTP, HTTP

Ulkoa yhteys ensin yhdyskäytäväkoneeseen

Autentikoi tarvittaessa

Muodostaa yhteyden sisäverkon koneeseen (palomuuri sallii vain sille)

Välittää sanomat sisään/ulos

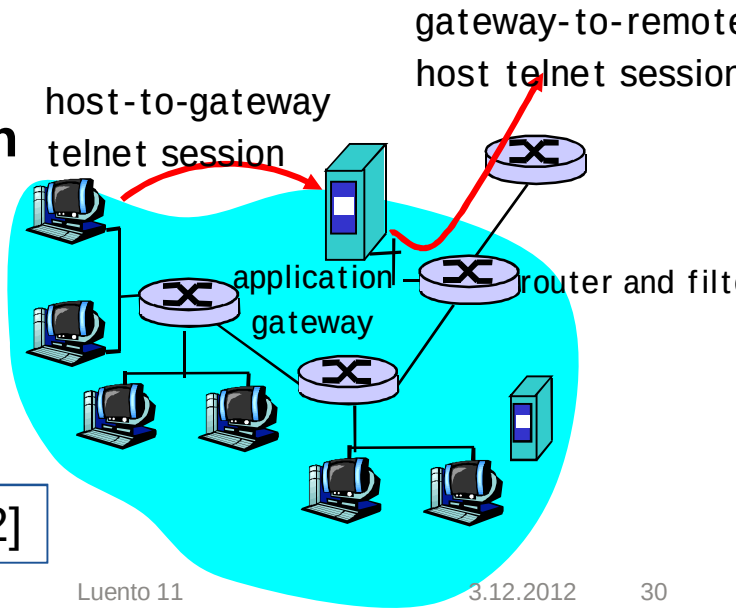


Fig 8.34 [KR12]

Palomuuri / Yhdyskäytävä

Yhteyttä haluavan on osattava ottaa yhteyttä yhdyskäytävään

- Esim. Web-selaajalle on kerrottava proxy-palvelimen osoite

Ei auta kaikkiin turvaongelmiin

- IP-osoitteiden ja porttinumeroiden väärentäminen
- Yhdyskäytäväohjelmissa voi olla turva-aukkoja
- Langattomat yhteydet ja soittoyhteydet

Myös hyvin ylläpidetyt järjestelmät kärsivät hyökkäyksistä!

Käytännön ohjeita

Käytä palomuuria
Huolehdi KJ:n päivityksistä
Käytä virustorjuntaa
Hävitä haittaohjelmat

Uusi kone

- Älä kytke verkkoon ennenkuin olet ottanut palomuurin käyttöön
- Päivitä käyttöjärjestelmä heti

Yliopiston lisenssillä saat koneellesi F-Securen ja Symantecin virustorjunta- ja palomuuriohjelmat

- https://ohjelma.it.helsinki.fi/

Muitakin ilmaisia ohjelmia löytyy

Lue lisää esim. “Jokakodin tietoturvaopas”

- www.tietoturvaopas.fi tai www.tietoturvakoulu.fi

Kertauskysymyksiä

Mitä ominaisuuksia halutaan turvalliselta

yhteydeltä?

Millaisia uhkia verkkoihin (koneisiin, tietoliikenteeseen ja palveluihin) kohdistuu?

Miten eri uhkiin pyritään varautumaan?

Mitä ovat haittaohjelmat?

Mikä on DoS? Entä DDoS?

Miten palomuuri toimii? Mihin sitä käytetään?

Tietoliikenteen perusteet

Kertausta

Kurssin karkea sisältö

1. **Tietokoneverkot ja Internet**

   Internetin rakenne, terminologiaa
2. **Verkkosovelluksia ja sovellusprotokollia**

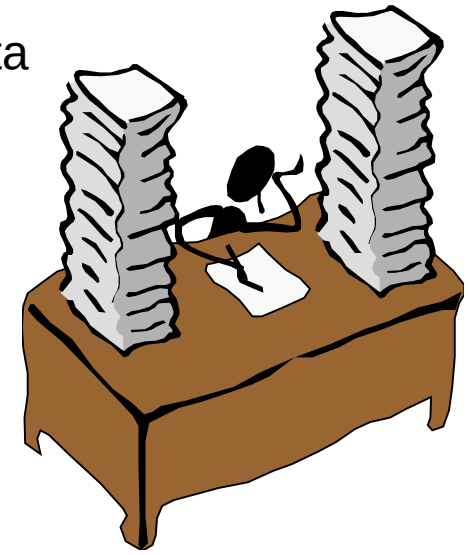
   Web, sähköposti, nimipalvelu, tiedostopalvelu, pistokerajapinta
3. **Kuljetuskerros: TCP, UDP**

   yhteydellinen / yhteydetön, ruuhkanhallinta
4. **Verkkokerros: IP**

   reitittimet ja reititys
5. **Linkkikerros, lähiverkot**

   Ethernet, kytkimet
6. **Tietoturvasta**

   Uhkat, palomuuri



Yleistä

Oppimistavoitteet:
-Perusterminologiaa tutuksi
-Yleiskuva Internetistä
- rakenne
- toiminnallisuus
-Internetin protokollapino ja sen eri kerrosten tehtävät

Internet
Verkon reunalla:
- **asiakkaat ja palvelimet,**
- **yhteydetön ja yhteydellinen palvelu**

Pääsy Internetiin, fyysinen media

Viivytykset ja katoamiset siirrossa
- **Mitä viipeitä? Miksi dataa katoaa**

Protokolla ja protokollapino
- **Kerrosarkkitehtuuri**
- **Internet-protokollapino: kerrokset ja sanomat**

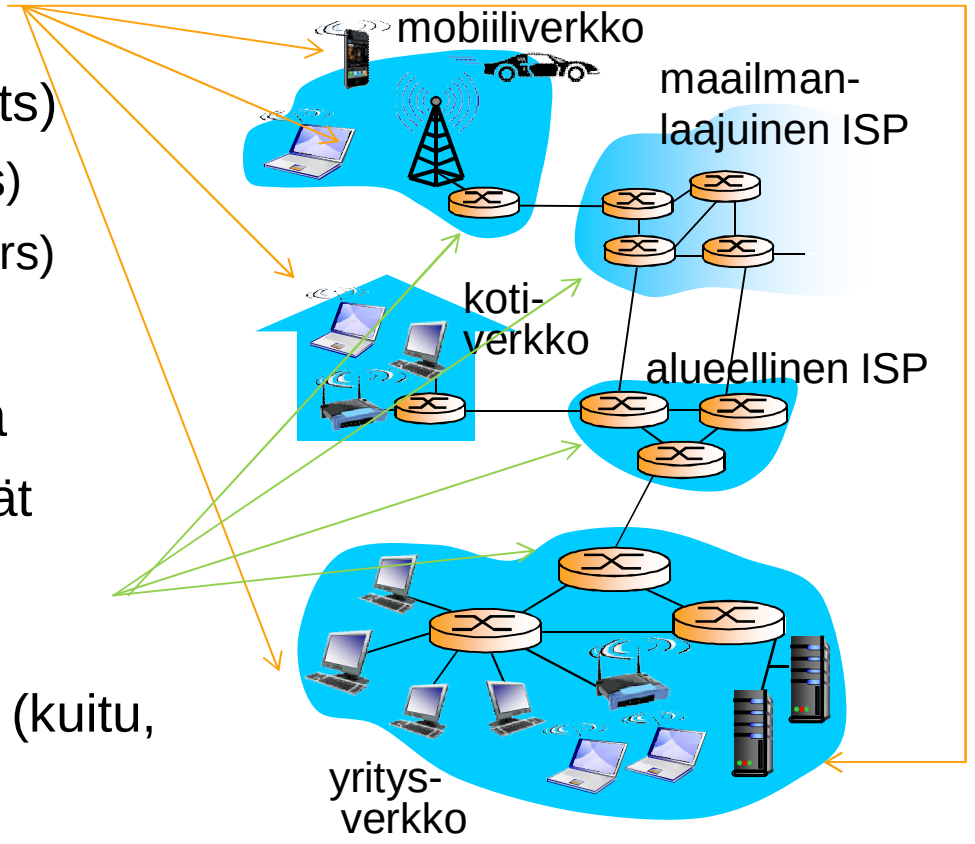
Verkon sisällä
- **Piirikytkentäinen, pakettikytkentäinen verkko**
- **Datasähkeverkko, virtuaalipiiriverkko**

Internetin uhista

Tietoliikenneverkon osat

Fig 1.1 [KR12]

- Verkon reunoilla
  - Isäntäkoneet (hosts)
    - Asiakkaat (clients)
    - Palvelimet (servers)
- Pääsy Internetiin
- Verkon syövereissä
  - Verkkoja yhdistävät reitittimet
  - Verkkojen verkko
  - Verkkoteknologiat (kuitu, kupari, langaton)



Internet

Julkinen Internet vs. rajattu **intranet** ja **extranet**

Päästä-päähän suunnittelumalli: tila ja toiminnot reunoilla

Sovellukset voivat lähettää sanomia verkon välityksellä toisilleen

yhteydellinen (connection-oriented) **palvelu** **vai**

yhteydetön (connectionless) **palvelu**

luotettava (reliable) (= pyrkii estämään, havaitsemaan ja paikkaamaan virheet) / **epäluotettava** (unreliable) (= 'hälläväliä')

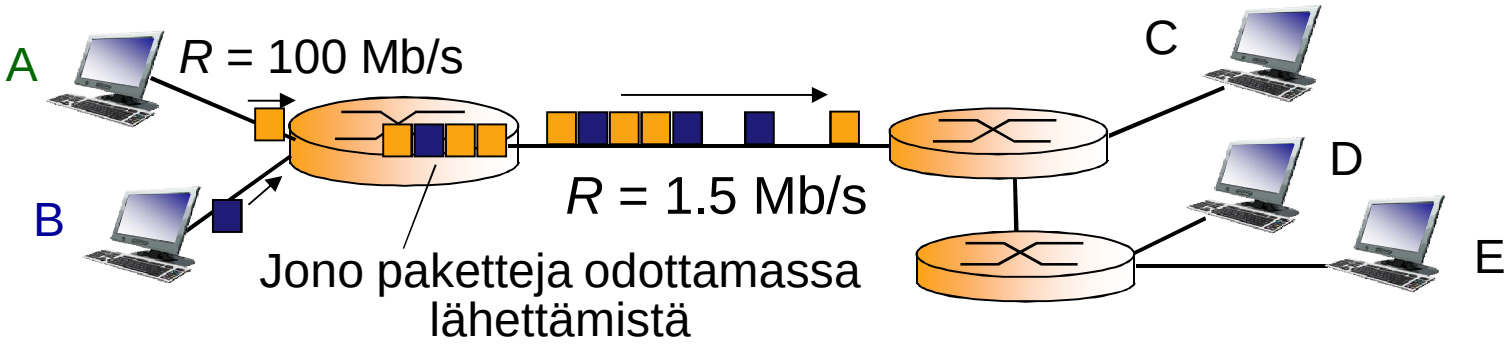
Internetissä: yhteydellinen=luotettava, yhteydetön=epäluotettava

TCP-protokolla => yhteydellinen ja luotettava

UDP-protokolla =>yhteydetön ja epäluotettava

Pakettikytkentä

Fig 1.12 [KR12]



Etappivälitys (store and forward) = paketti vastaanotetaan kokonaan ja vasta sitten lähetetään eteenpäin

Koko linkin kapasiteetti siirrettävälle paketille

Yhteenlaskettu siirtotarve voi ylittää lähtevän linjan siirtonopeuden

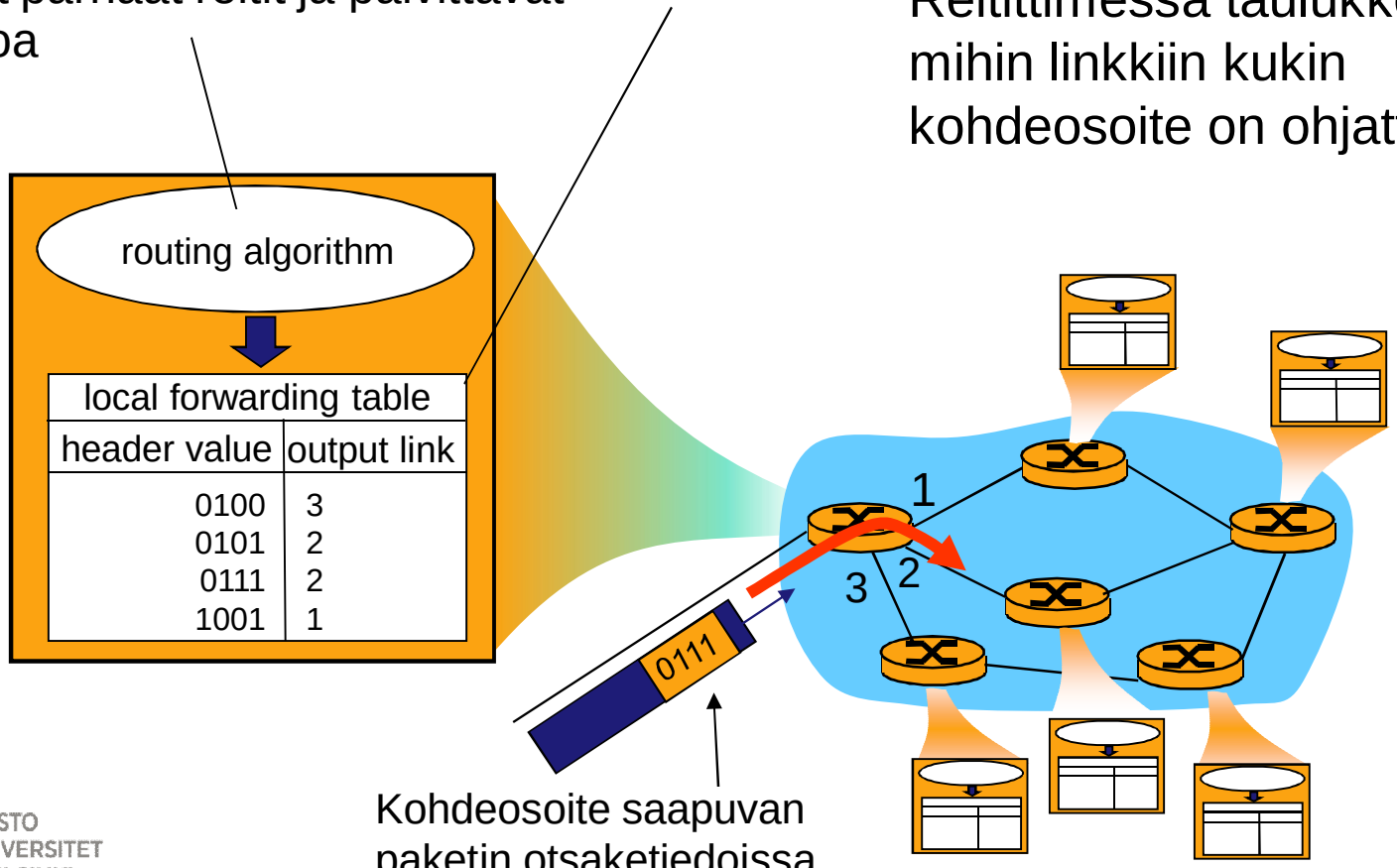
- Paketti joutuu odottamaan vuoroaan reitittimen muistissa
- Ruuhka (congestion) => jopa paketin häviäminen

Reititys

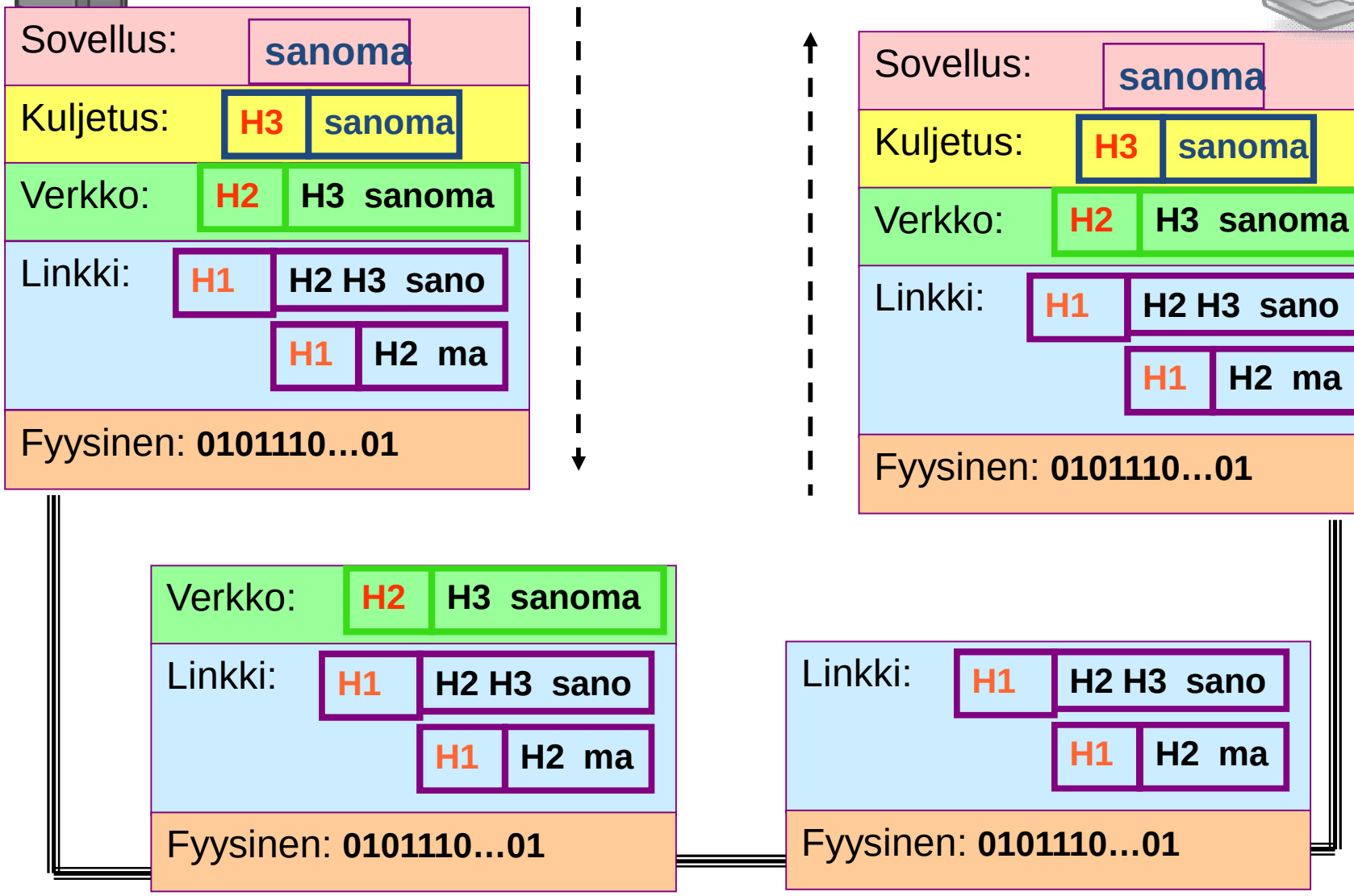
Fig 4.2 [KR12]

Reititys (routing): Reititysalgoritmit laskevat parhaat reitit ja päivittävät taulukkoa

Edelleenlähetys (forwarding): Reitittimessä taulukko => mihin linkkiin kukin kohdeosoite on ohjattava



Protokollapino ja kapselointi



Reititin **Linkkitason kytkin**

Kertauskysymyksiä

Isäntäkone vs. reititin?

Protokolla vs. palvelu?

Vertaisverkkomalli vs. asiakas-palvelin malli?

Fyysinen siirtomedia?

Piiri- ja pakettikytkentä? Hyödyt ja haitat?

Viipeet ja pakettien katoamiset

Internet-protokollakerrokset ja niiden tehtävät?

Miksi kerrosrakenne?

Mitä protokollakerroksia eri laitteissa tarvitaan?

Ks . myös kurssikirja ss. 94-96.

Sovellus-kerros

Verkkosovellusten periaatteet
World Wide Web ja HTTP
Pistoke ja sen käyttö

Nimipalvelu ja DNS
Tiedostonsiirto ja FTP
Sähköposti ja SMTP, IMAP, POP3
Vertaistoimijat (peer-to-peer)

Oppimistavoitteet:

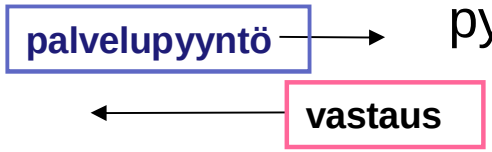
- Osaa selittää asiakaspalvelija–malliin perustuvien verkkosovellusten toimintaperiaatteet
-Tuntee sovellusprotokollien syntaksia ja semantiikkaa
- Osaa selittää www:n ja sähköpostin toimintaideat

Sovellusarkkitehtuuri

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

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

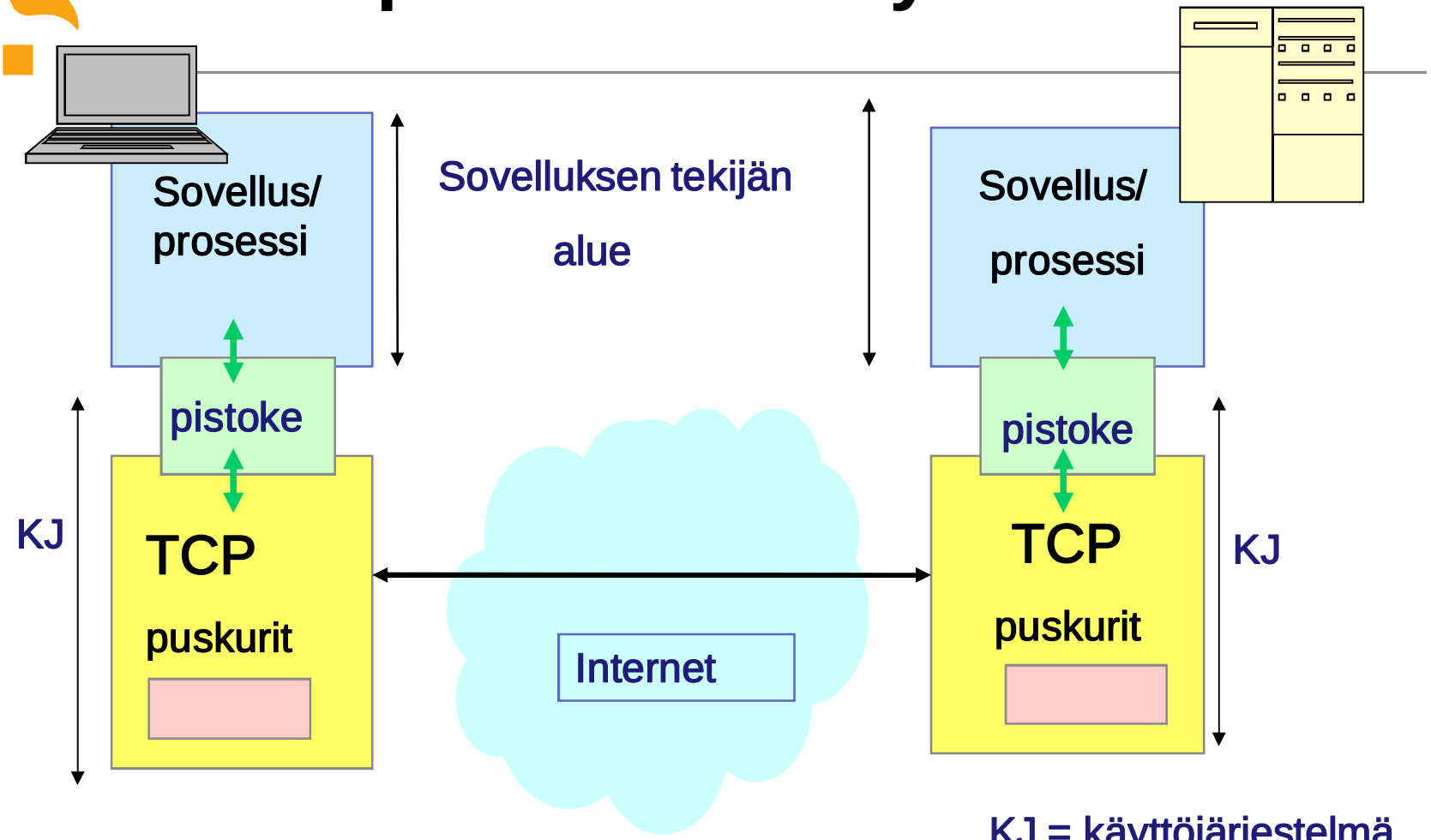


Vertaistoimijamalli (esim. BitTorrent, eMule, Skype)

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

Hybridimalli (esim. Napster, pikaviestimet)

Prosessien kommunikointi TCP-pistokkeita käyttäen



Osoittaminen

- Sanomissa oltava lähettäjän ja vastaanottajan IP-osoite ja porttinumero
- IP-osoite → oikea kone

www.iana.org

  - koneen (verkkokortin) yksilöivä 32-bittinen tunniste
  - osoitteen verkko-osa yksilöi verkon
  - osoitteen koneosa yksilöi koneen verkossa
- Porttinumero → oikea prosessi
  - Yleisillä palveluilla standardoidut tunnetut porttinumerot:
    - www-palvelin kuuntelee porttia 80,
    - Postipalvelin kuuntelee porttia 25
  - KJ osaa liittää porttinumeron prosessiin

Sovellukset ja niiden käyttämät kuljetusprotokollat

Applications	Application-Layer Protocol	Underlying Transport Protocol
Electronic mail	SMTP [RFC 2821]	TCP
Remote terminal access	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
File transfer	FTP [RFC 959]	TCP
Remote file server	NFS [McKusik 1996]]	UDP or TCP
Streaming multimedia	Often proprietary (e.g., Real Networks)	UDP or TCP
Internet telephony	Often proprietary (e.g., Net2phone)	Typically UDP

Figure 2.5 ♦ Popular Internet applications, their application-layer protocols, and their underlying transport protocols

Domain Name System (DNS)

Hakemistopalvelu ja sovelluskerroksen protokolla

- Isäntäkoneet ja nimipalvelimet käyttävät
- Käyttää UDP-kuljetuspalvelua DNS-sanomien kuljettamiseen

Hajautettu, hierarkinen tietokanta (hakemisto)

- Toteutettu useiden replikoitujen nimipalvelimien yhteistyönä
- skaalautuvuus, kuormantasaus, ylläpito, vikasietoisuus, ..
- Jos oma nimipalvelija ei tunne, se kysyy muilta.

Nimien muuttaminen IP-osoitteiksi (ja päinvastoin)

- POSIX: gethostbyname

gethostbyname (hydra.cs.helsinki.fi)
218.214.4.29

- Kone = hydra =29, verkko= cs.helsinki.fi = 218.214.4.0

Sallii aliasnimet, palvelijan replikoinnin/toisintamisen

- Esim. WWW.cs.helsinki.fi ja cs.helsinki.fi ovat aliasnimiä
- Esim. www-palvelijaan voi liittyä useita IP-osoitteita, rotaatio

Kertauskysymyksiä

Asiakas-palvelija-malli? Vertaisverkkomalli?

Kuinka asiakas löytää palvelimen?

Miten KJ osaa antaa bitit oikealle sovellukselle?

Miten koneen nimestä saadaan selville sen IP-osoite?

Miten HTTP-protokolla toimii?

Miksi SMTP ei riitä, vaan tarvitaan POP3 tai IMAP?

Mitä hyötyä on proxy-palvelimesta?

Miksi käytetään evästeitä?

Mikä on pistoke ja missä sitä käytetään?

Ks. myös kurssikirja s.195-197.

Kuljetuskerros

Kuljetuspalvelut
Luotettavan kuljetuspalvelun periaatteet
Yhteydetön kuljetuspalvelu, UDP
Yhteydellinen kuljetuspalvelu, TCP
Ruuhkanhallinta TCP:ssä

Oppimistavoitteet:
- Tuntea Internetin kuljetusprotokollien (UDP/TCP) toiminnallisuus ja periaatteet
- Osata luotettavan kuljetuspalvelun ja vuonvalvonnan periaatteet ja toteutukset
- Osata TCP-ruuhkanhallinnan

Kuljetuskerros

Tarjoaa kuljetuspalvelun prosessien välille

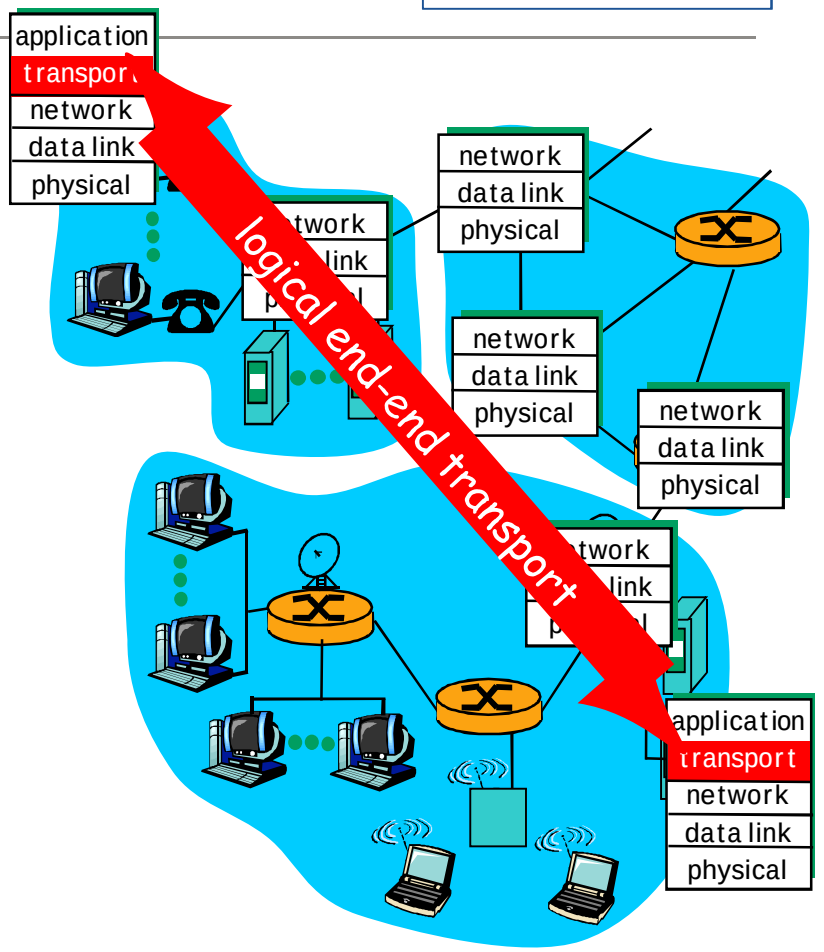
Vain isäntäkoneissa

- Lähetys: Pilko sovelluskerroksen sanoma pienemmiksi segmenteiksi, jotka verkkokerros toimittaa perille.
- Vastaanotto: Kokoa segmentit sanomaksi, jonka sovellus lukee.

Verkkokerros reitittää koneesta koneelle

Segmentin koko s.e. verkkokerros pystyisi välittämään sellaisena

Fig 3.1 [KR12]



rdt3.0: vuorottelevan bitin protokolla

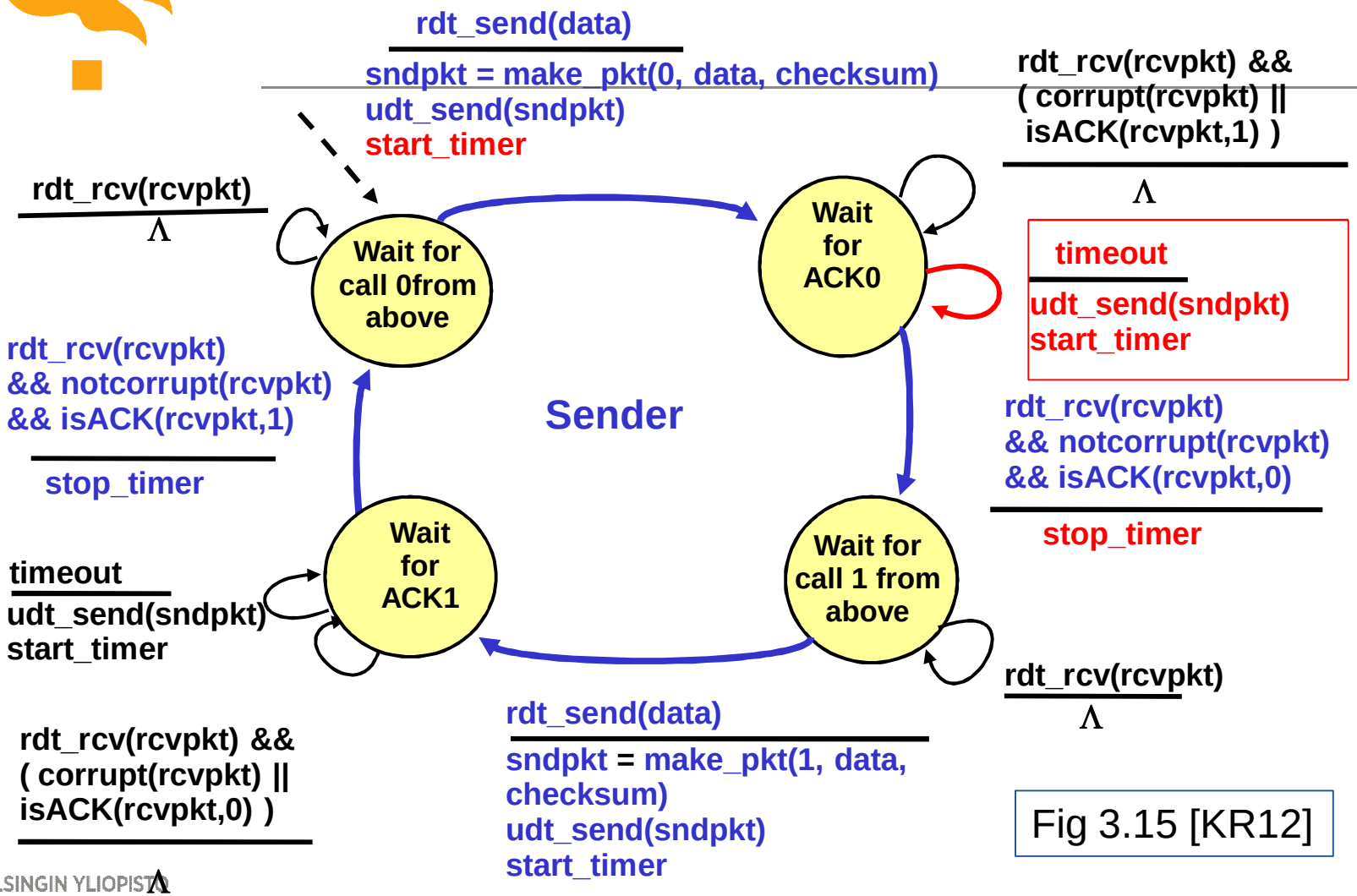


Fig 3.15 [KR12]

Internetin kuljetusprotokollat

TCP: luotettava, järjestyksen säilyttävä tavujen kuljetuspalvelu

- **Virheenvalvonta** (error control): Huomaa ja korjaa virheet, hylkää kaksoiskappaleet
- **Vuonvalvonta** (flow control): Älä ylikuormita vastaanottajaa
- **Ruuhkanhallinta** (congestion control): Älä ylikuormita verkkoa
- **Yhteyden** muodostaminen ja purku

UDP: Ei-luotettava, ei-järjestyksen säilyttävä sanomien kuljetuspalvelu

- Välittää vain sanomia, ei pyri mitenkään parantamaan verkkokerroksen tarjoamaa palvelun laatua
- Luotettavuus jää sovelluskerroksen hoidettavaksi

Kumpikaan kuljetuspalvelu ei anna takuita viiveelle tai siirtonopeudelle (“best effort”)

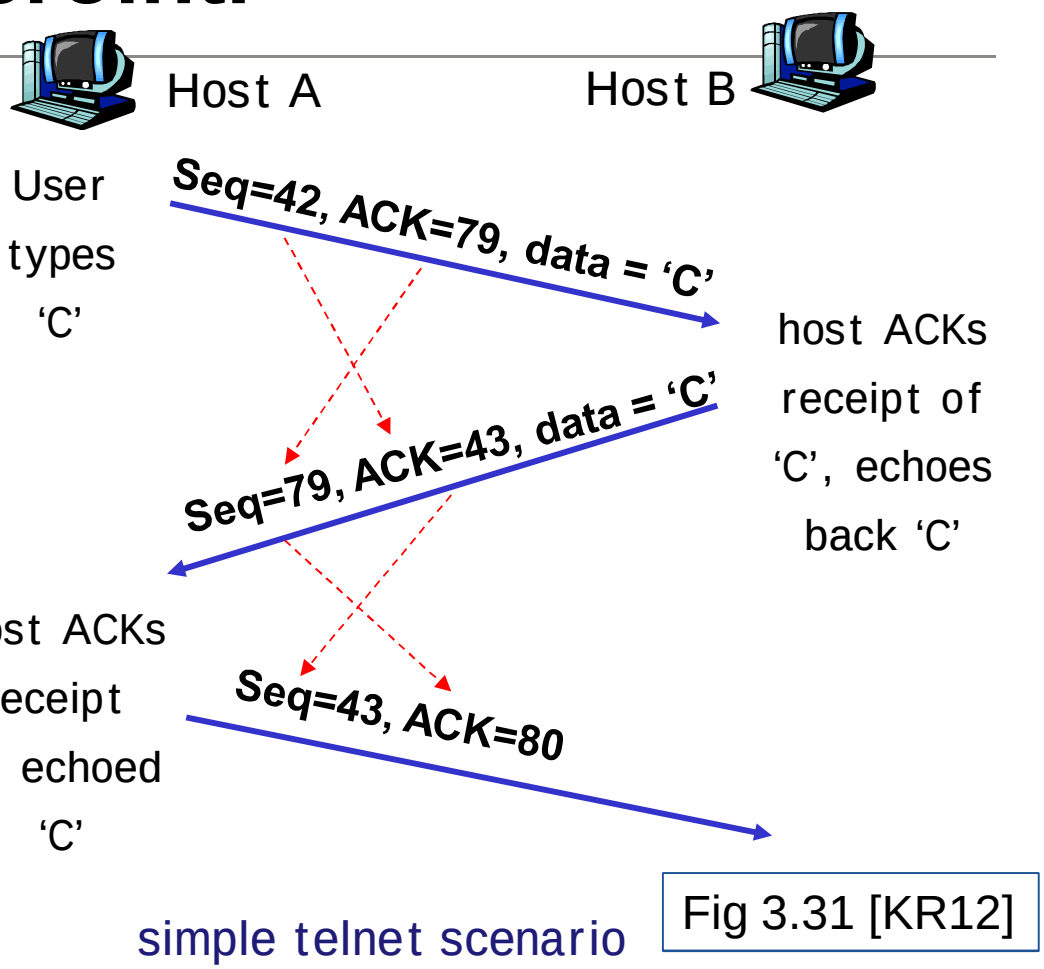
Tavunumerointi

Kuittauksia ei kuitata ja ne eivät siirrä numerointia!

Niissä ei siirretä tavuvirtaa.

Tavuvirtaa …
Segmentit voivat olla erikokoisia (=< MSS)
Järjestysnumero=
- ensimmäisen tavun numero
- alkuarvot sovitaan yhteyttä muodostettaessa

Kuittaus
- seuraavaksi odotetun tavun numero
- kumulatiivinen
- kylkiäisenä (piggybacked) mikäli mahdollista



simple telnet scenario

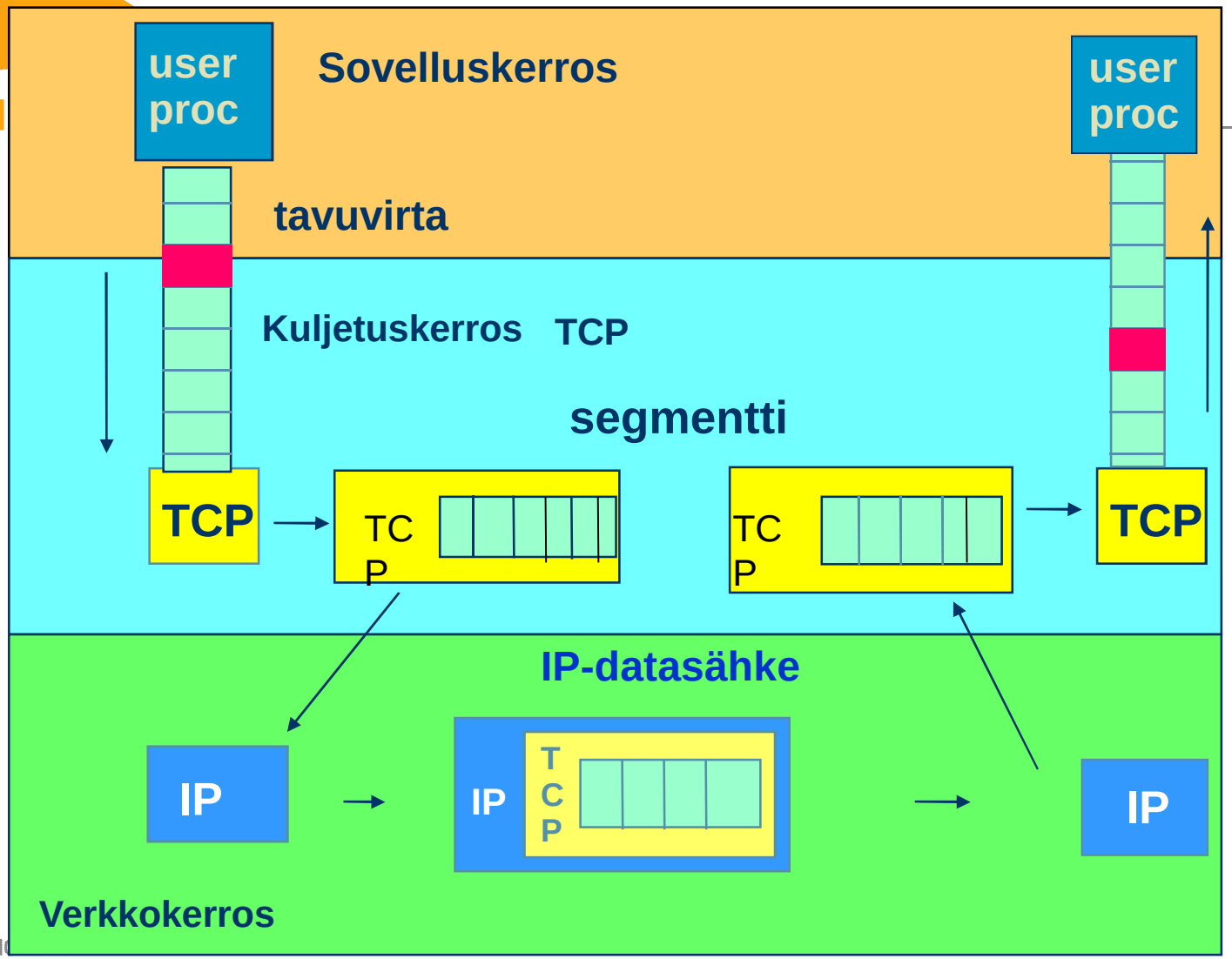
Fig 3.31 [KR12]

TCP-protokolla

- **Päästä-päähän kuljetuspalvelu**
  - Yksi lähettäjä, yksi vastaanottaja
  - Reitittimet eivät ole kiinnostuneita kuljetustason protokollasta
- **Yhteydellinen** (connection-oriented)
  - Yhteydenmuodostus
  - Isäntäkoneissa: puskuritila, ikkunakoko, tavunumerointi
  - Yhteyden purku
- **Kaksisuuntainen** (full duplex)
  - Yksi yhteys, jossa yhtä aikaa liikennettä molempiin suuntiin
- **Luotettava, järjestyksen säilyttävä tavuvirta**
  - Ei sanomarajoja
  - Tavunumerointi
  - Kumulatiiviset kuittaukset

Vuonvalvonta
Ruuhkanvalvonta

TCP: prosessilta prosessille -tavuvirta



Ruuhkaikkuna (congestion window)

- Internetin hetkellinen kuormitus vaihtelee
- Ruuhkaikkuna
  - Paljonko lähettäjä saa tietyllä hetkellä kuormittaa verkkoa
  - Paljonko lähettäjällä saa olla kuittaamattomia segmenttejä
- Lähettäjän pääteltävä itse sopiva ikkunankoko
  - Jos uudelleenlähetysajastin laukeaa, on ruuhkaa
  - Jos kuittaukset tulevat tasaisesti, ei ole ruuhkaa
- Dynaaminen ruuhkaikkunan koko
  - Kasvata ikkunaa ensin nopeasti, kunnes törmätään ruuhkaan
  - Pienennä sitten ikkunaa reilusti ja kasvata varovasti
- Lähetysikkunan raja voi tulla vastaan ensin
  - Kuittamatta saa olla min(lähetysikkuna, ruuhkaikkuna)

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

Aluksi ruuhkaikkuna = yksi segmentti
- Alussa hidas siirtonopeus= MSS/RTT

Kukin kuittaus kasvattaa yhdellä ruuhkaikkunan kokoa
- Eksponentiaalinen kasvu
- Ikkuna kaksinkertaistuu yhden RTT:n aikana

hidas aloitus

Jos uudelleenlähetys, ruuhkaikkunan kooksi 1 segmentti
- Multiplicative decrease

Sen jälkeen kasvata ikkunaa yksi segmentti/RTT
- Lineaarinen kasvu (Additive increase)
- Ruuhkan välttely (congestion avoidance)

ruuhkan-välttely

Siirtonopeus = CognWin / RTT tavua/sek

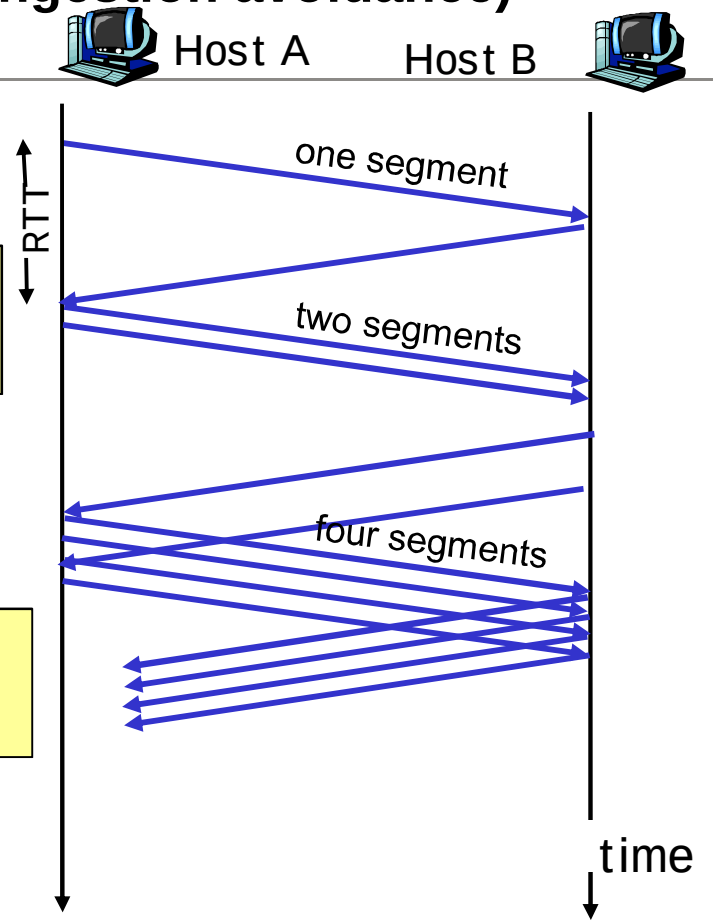
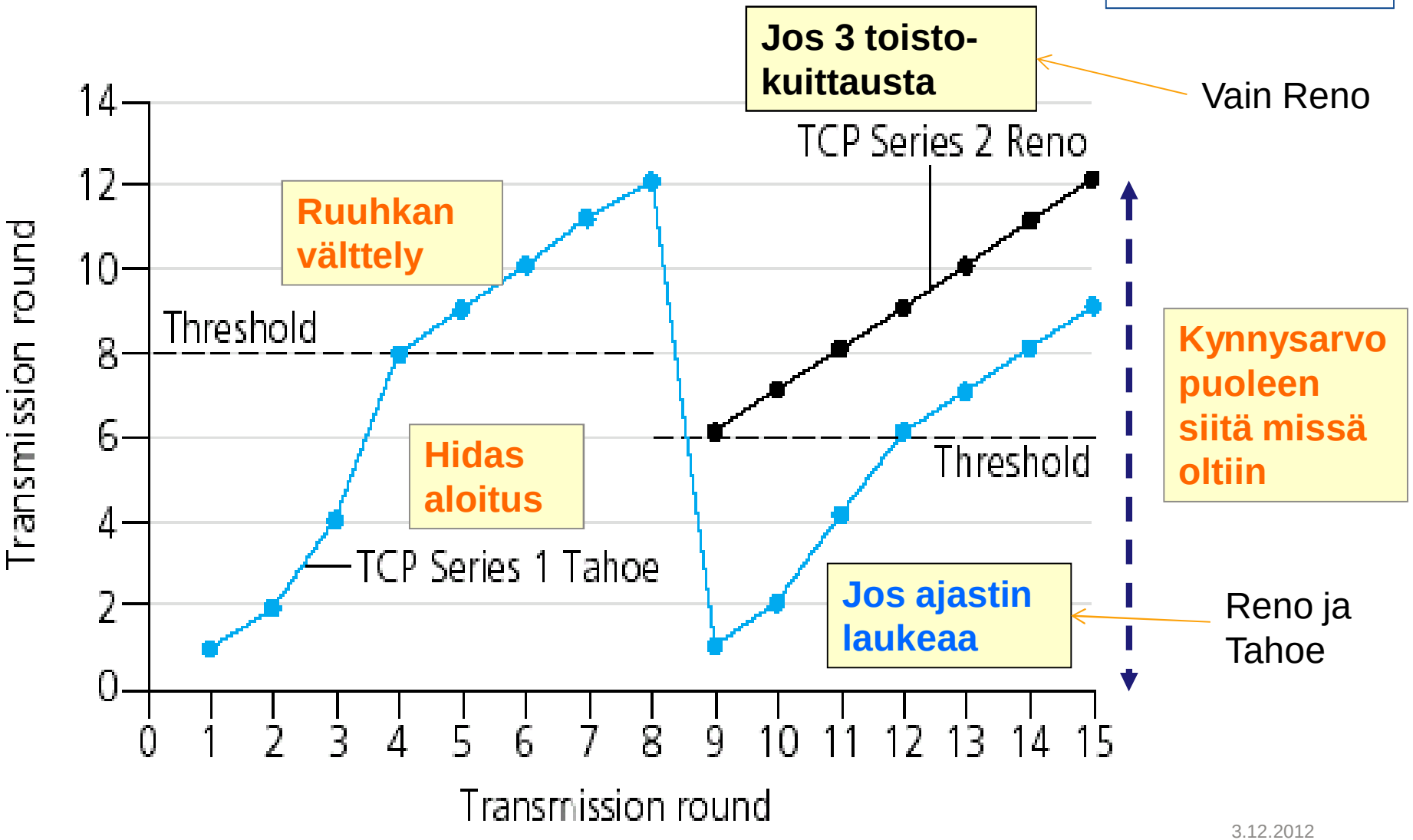


Fig 3.51 [KR12]

TCP Tahoe vs. TCP Reno

Fig 3.53 [KR12]



Verkkokerros

Verkkokerros
Reititin
IP-protokolla
IP-osoitteet, DHCP, NAT
Reititysalgoritmit

Oppimistavoitteet:
- Osaa selittää, kuinka IP-paketteja välitetään verkossa
- Tietää, mitä tietoja sisältyy IP-pakettiin (ja miksi)
- Osaa selittää reitittimen rakenteen ja toiminnan
- Osaa kuvailla, kuinka reitittimet kokoavat reititystietonsa
= linkkitila- ja etäisyysvektorialgoritmien toimintaideat

Verkkokerros: Toimittaa kuljetus-kerroksen segmentit vastaanottajalle

Lähetys

- Luo kuljetuskerroksen segmenteistä verkkokerroksen IP-paketteja
- Lisää otsaketietoja: mm. IP-osoitteet

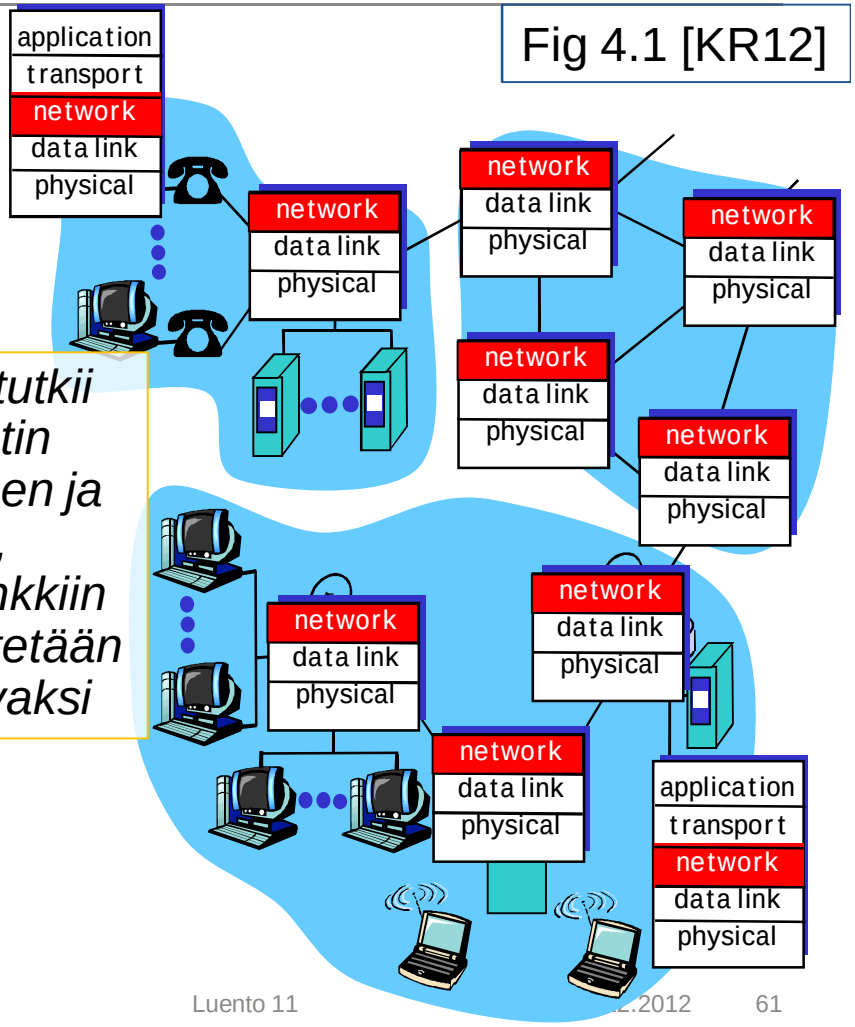
Pakettien kulku verkossa

- Isäntä (lähde) – reititin - ...- reititin – isäntä (kohde)

Vastaanotto

- Poista otsake
- Anna segmentti kuljetuskerrokselle

Fig 4.1 [KR12]



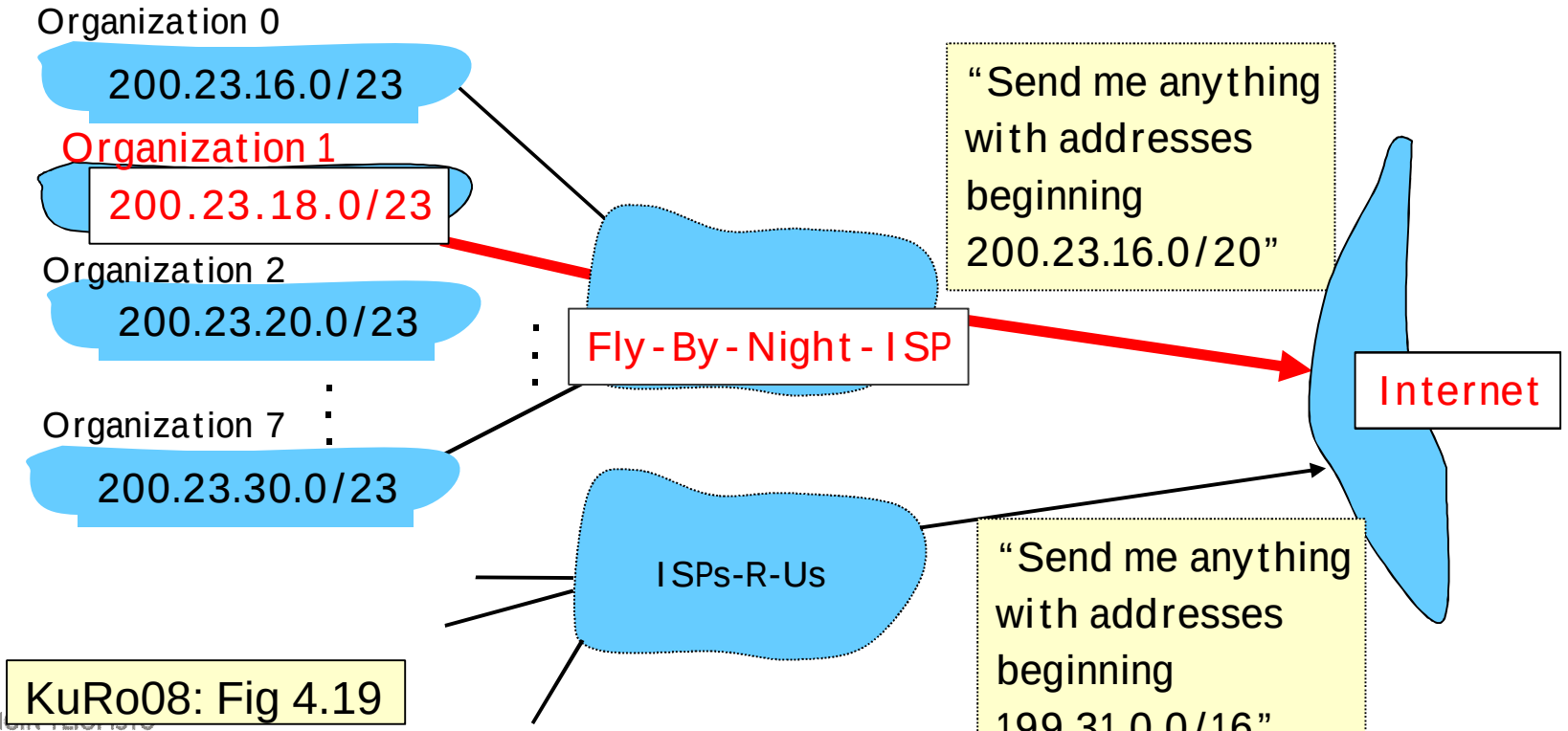
Reititin tutkii IP-paketin otsakkeen ja päättää, mihin linkkiin se lähetetään seuraavaksi

Hierarkkinen osoite

Fig 4.18 [KR12]

CIDR luo reititystä helpottavan hierarkian

Aggregointi (yhdistäminen): yhteinen alkuosa => samaan suuntaan

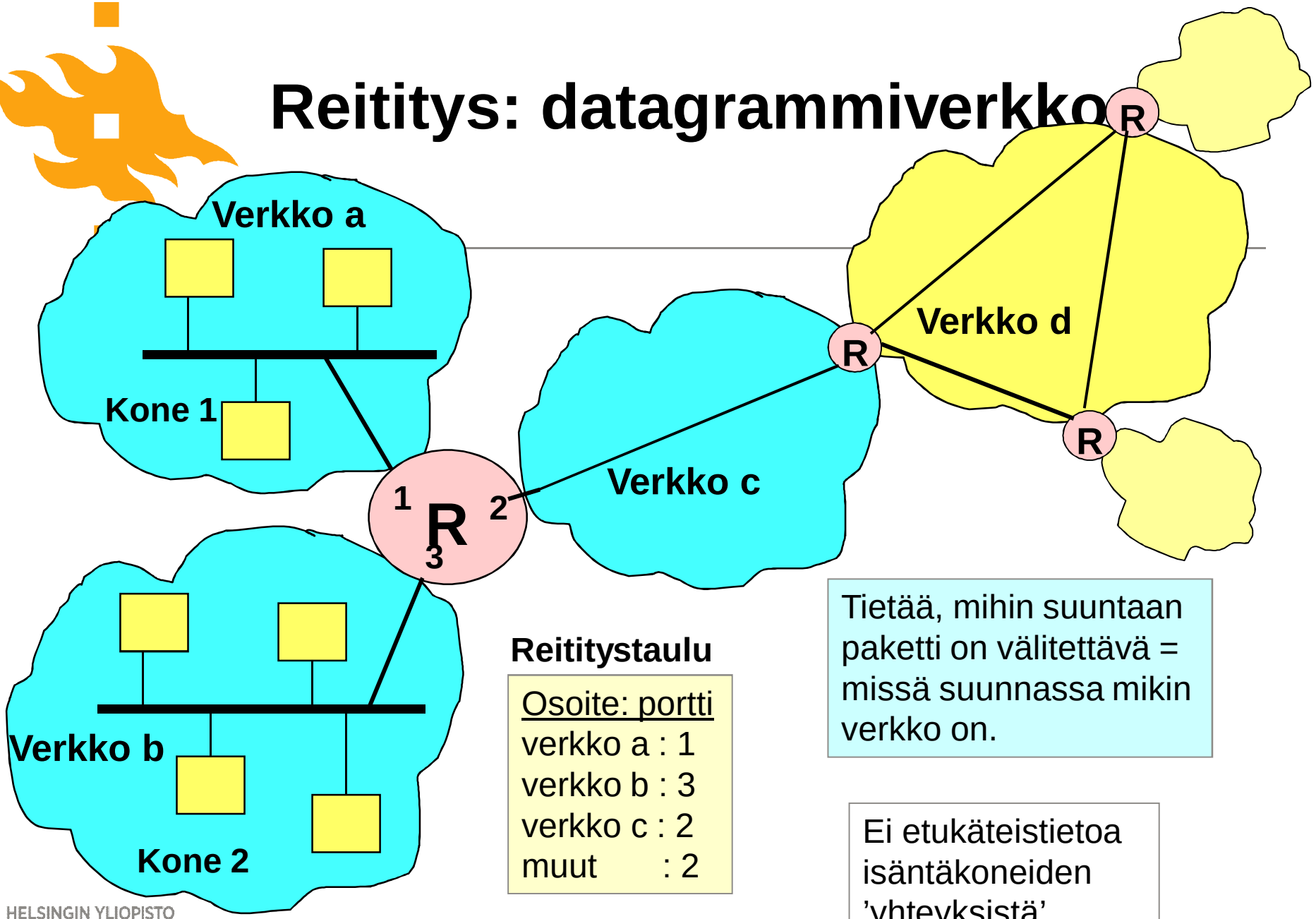


KuRo08: Fig 4.19

Koneen IP-osoite

- Koneen IP-osoite konfiguroidaan joko käsin koneelle tai
- saadaan automaattisesti käyttäen **DHCP**:tä (Dynamic Host Configuration Protocol)
  - Eri osoite eri kerroilla tai pysyvämpi osoite
  - DHCP-palvelija vastaa
    - antaa koneen käyttöön IP-osoitteen (rajallinen elinaika)
    - antaa DNS-tiedot, yms.
  - Palvelun tarjoaja: pienempi numeromäärä riittää
    - Toteutus UDP monilähetys (broadcast)
    - “wash-and-go”, “plug-and-play”

Reititys: datagrammiverkko



Reititystaulu

Osoite: portti
verkko a : 1
verkko b : 3
verkko c : 2
muut : 2

Tietää, mihin suuntaan paketti on välitettävä = missä suunnassa mikin verkko on.

Ei etukäteistietoa isäntäkoneiden 'yhteyksistä'

Reititysalgoritmeja

Reititysalgoritmi, joka tarvitsee täydellisen tiedon verkosta

- Ennen laskentaa käytössä koko kuva verkosta:
  - Kaikki linkkiyhteydet solmujen välillä ja niiden kustannukset
    - Käytännössä vain tietystä autonomisesta alueesta
- Parhaat reitit lasketaan joko keskitetysti tai hajautetusti
- **Linkkitila-algoritmi** (link-state algorithm)

Reititysalgoritmi, jolle riittää epätäydellinen kuva verkosta

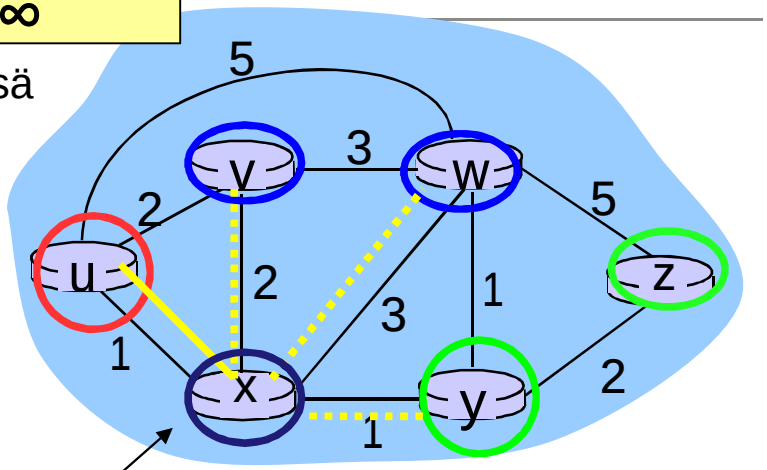
- Aluksi reititin tietää vain niistä koneista, joihin itse on yhdistetty
- Iteratiivinen algoritmi: reititin vaihtaa tietoja naapuriensa kanssa ja saa tietoa muusta verkosta
- **Etäisyysvektorialgoritmi** (distance vector algorithm)

Dijkstran algoritmi

D(v)=2, D(w) = 5, **D(x)=1**
D(y) = ∞ , D(z)= ∞

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

1. Eli jos u:n vieressä



2. Aina valitaan käsittelemätön, jonka etäisyys u:sta on pienin

3. Päivitetään etäisyys b:n naapureille, joita ei vielä ole käsitelty

Etäisyysvektorialgoritmi

Alussa kukin solmu tuntee vain etäisyydet naapureihinsa itsensä kautta:

x:

from \ cost to	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

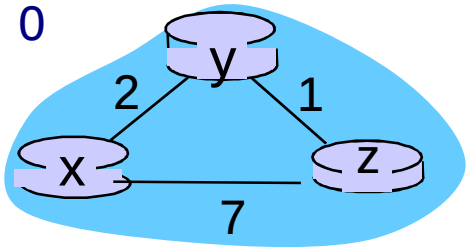
y:

from \ cost to	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

z:

from \ cost to	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

Sitten solmut lähettävät omat reittinsä toisilleen ja laskevat uudet parhaat reitit.



Esimerkiksi solmu x:

	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

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

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

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

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

x:

cost to

from	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

cost to

from	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

cost to

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

y:

cost to

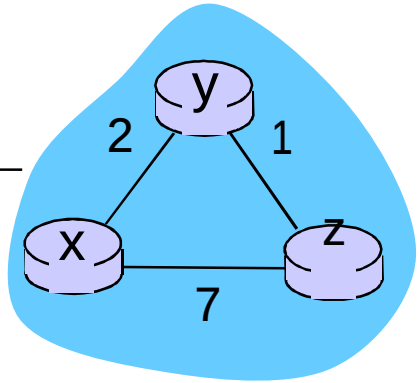
from	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

cost to

from	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

cost to

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0



z:

cost to

from	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

cost to

from	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

cost to

from	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Kertauskysymyksiä

ks. kurssikirja s. 439-441

Keskeisimmät IP-otsakkeen tiedot?

Paketin paloittelu

Millainen on IP-osoite?

Reitittimen arkkitehtuuri?

Longest prefix match?

Aliverkon peite (mask)

NAT:n toiminta

Miten reititin saa reititystiedot?

Linkkitila-algoritmi, Dijkstran algoritmi

Etäisyysvektorialgoritmi, count-to-infinity-ongelma

Sisältö

Linkkikerroksen tehtävät
Virheiden havaitseminen ja korjaaminen
Yhteiskäyttöisen kanavan varaus
Osoittaminen linkkikerroksella
Ethernet

Oppimistavoitteet:
- Osata selittää linkkikerroksen toiminnallisuus (MAC-osoitteet, bittivirheiden havaitseminen) ja ARP-protokollan käyttö.
- Osata selittää yhteiskäyttöisen siirtokanavan varaus ja käyttö
- Osata selittää, kuinka koneita voi yhdistellä lähiverkoiksi
- Osata selittää reitittimen, kytkimen ja keskittimen erot

Linkkikerros

Laitetoimintoa

Siirtää paketin fyysistä linkkiä pitkin koneelta (solmulta (node)) toiselle

- langallinen / langaton
- bitit sisään, bitit ulos

Kapseloi paketin siirtoon sopivaan muotoon

- Siirtokehys (frame)

Lähiverkossa linkkejä voi yhdistää keskittimillä tai kytkimillä

- Käytetään fyysisiä osoitteita
- 'reititystä' ilman IP-osoitteita

Fig 5.1 [KR12]

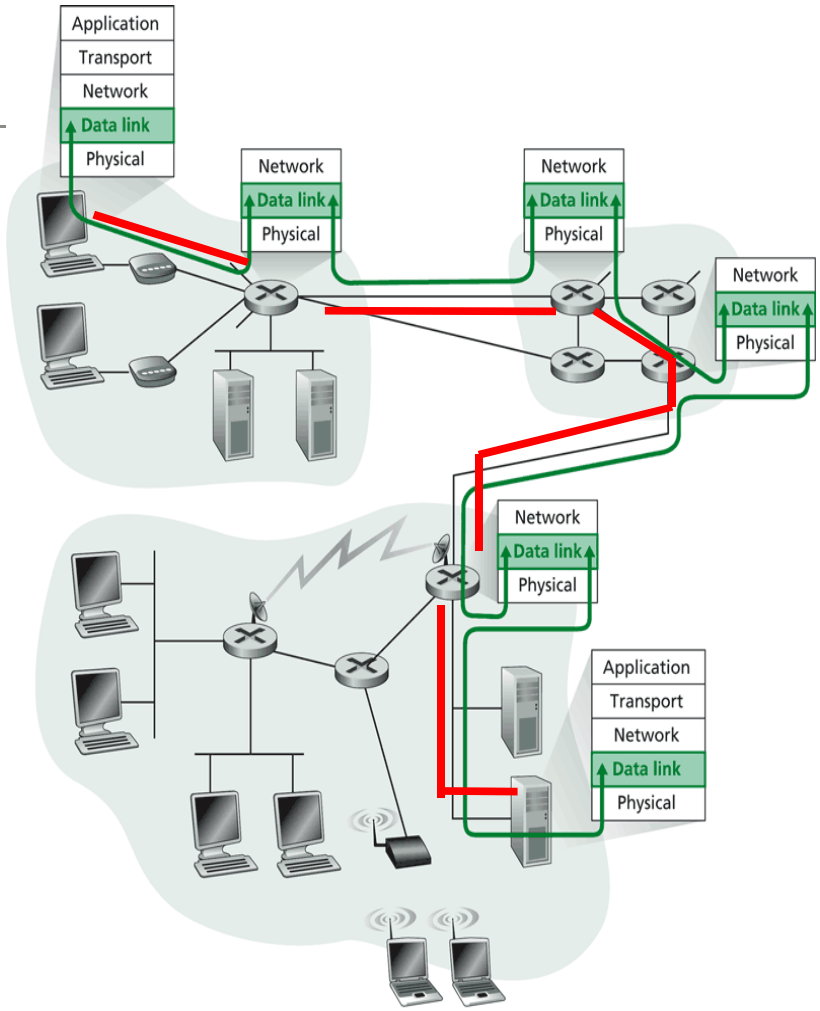


Figure 5.1 ♦ The link layer

Linkkikerroksen tehtäviä

Kehystys (framing)

- Kehyksen rakenne ja koko riippuu siitä, millainen linkki on kyseessä
- Otsake, data, lopuke

Kohteen ja lähteen osoittaminen

- Yhteiseen linkkiin voi olla liitettynä useita laitteita
- Käytössä laitetason MAC-osoite (Medium access control)

Yhteisen linkin varaus ja käyttö (link access)

- Esim. langaton linkki, keskittimiin yhdistetyt linkit

Luotettava siirto

- Langattomilla linkeillä suuri virhetodennäköisyys
- Linkkitaso huolehtii oikeellisuudesta
- Miksi tästä täytyy huolehtia vielä kuljetuskerroksella?
  - Jotkut linkkityypit eivät huolehdi lainkaan!
  - Jos kehys hävitettävä ..

Tarkistussumma

Internet-checksum

- Yhteenlasketaan 16 bitin kokonaisuuksia, yhden komplementti
- Kuljetuskerros laskee ja tarkastaa UDP- ja TCP-protokollissa
- Huom. IP sekä UDP/TCP ja UDP optionaalinen
- Ei kovin tehokas; linkkikerros ei käytä

CRC (cyclic redundance check)

- Linkkikerroksella paljon käytetty virheenpaljastusmenetelmä,
  - helppo toteuttaa laitteistotasolla, luotettava
- Perustuu polynomien aritmetiikkaan
  - tunnetaan myös nimellä polynomikoodi (polynomial code)
- Useita tarkistusbittejä; havaitsee usean bittivirheen ryöpyn.

Lähetysvuorojen jakelu

- Yksi yhteinen kanava lähettäjille
  - Lähetys onnistuu vain, jos yksi kerrallaan lähettää
- Jos useampi lähettää yhtäaikaa, syntyy yhteentörmäys
  - Kaikki solmut saavat useita signaaleja, “bittimössöä”
  - Törmänneet sanomat tuhoutuvat ja ne on lähetettävä uudelleen
- Multiple Access Protocol
  - Tapa, jolla solmu päättelee, voiko se lähettää
  - Kuinka solmun on toimittava törmäystilanteessa
  - Neuvottelu samassa kanavassa!

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:
   vuorokysely (polling), vuoromerkki, ...
   “Minä ensin, sinä sitten.”

Kilpailuprotokollat: Lähetyskanavan kuuntelu (CSMA)

Kuuntele ennen kuin lähetät

- Asema tutkii, onko kanava jo käytössä (carrier sense)
- Jos siirtotie on vapaa, saa lähettää
- Jos siirtotie on varattu, odota satunnainen aika ja yritä uudelleen

CSMA (Carrier Sense Multiple Access)
Useita variaatioita

- Ei aina paljasta jo alkanutta lähetystä
- Aina huomaaminen ei ole mahdollista
  - Esim. satelliittikanavan kuuntelu ei paljasta, onko jokin muu maa-asema jo aloittanut lähetyksen
  - Langattomassa lähiverkossa lähettäjän ympäristön kuuntelu ei kerro, onko vastaanottaja saamassa sanomia muilta

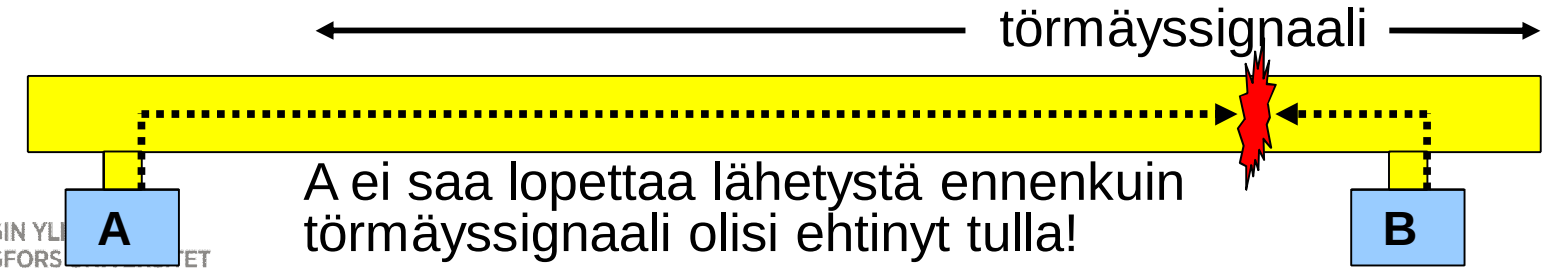
CSMA/CD

(Carrier Sense Multiple Access with Collision Detection)

Asema kuuntelee myös lähettämisen jälkeen
- Langallinen LAN: törmäys => signaalin voimakkuus muuttuu
  - Esim. Ethernet
- Langaton LAN: hankalaa

Jos törmäys, keskeytä lähettäminen **heti**
- ja yritä uudestaan satunnaisen ajan kuluttua
- Näin törmäyksen aiheuttama hukka-aika pienenee

Kauanko kuunneltava?
- 2* maksimi etenemisviive solmujen välillä



Ethernet

Ethernet Timeline (ennuste 2003)
- *** 10 Megabit Ethernet 1990**
- *** 100 Megabit Ethernet 1995**
- *** 1 Gigabit Ethernet 1998**
- *** 10 Gigabit Ethernet 2002**
- *** 100 Gigabit Ethernet 2006****
- *** 1 Terabit Ethernet 2008****
- *** 10 Terabit Ethernet 2010****

Yleisin lähiverkkoteknologia

Yksinkertainen, edullinen, helppo laajentaa

Lähiverkko syntyy kytkemällä koneet keskittimeen tai kytkimeen

IEEE:n standardoima LAN-verkko

Klassinen Ethernet (10 Mbps): CSMA/CD (kuulosteluväylä)

Fast Ethernet (FE, 100 Mbps), Gigabit Ethernet (GE), 10 Gigabit Ethernet, 100 Gb Ethernet (pian??), 400Gb Ethernet (tulossa??)
1 TB Ethernet (joskus vai ei ollenkaan??!)

- Yleensä kytkentäisiä kaksipisteyhteyksiä

Muita lähiverkkostandardeja

- Token Ring (vuororengas)
- FDDI (Fiber Distributed Data Interface)
- WLAN (langaton lähiverkko)

April 24, 2008
Terabit Ethernet around 2015

Bob Metcalfe (ethernet coinventor)
gave a keynote speech,
"Toward Terabit Ethernet."

MAC-osoite

Lähes 300 biljoonaa erilaista osoitetta.

Lähes 17 miljoonaa valmistajanumeroa, kuhunkin mahdollista lähes 17 miljoonaa osoitetta.

MAC spoofing

48-bittinen (6 tavua)
- 24 b kertoo valmistajan ja 24 b identifioi ohjainkortin (adapter)
- IEEE jakaa valmistajanumerot

Kiinteä - Liitetty mukaan valmistuksessa
- Säilyy, vaikka laite toiseen verkkoon (toisin kuin IP-osoite)

Ohjain
- Kuulee kaikki kanavalla kulkevat kehykset
- Välittää omalle koneelle vain sen MAC-osoitteella tai yleislähetysosoitteella FF-FF-FF-FF-FF-FF merkityt lähetykset

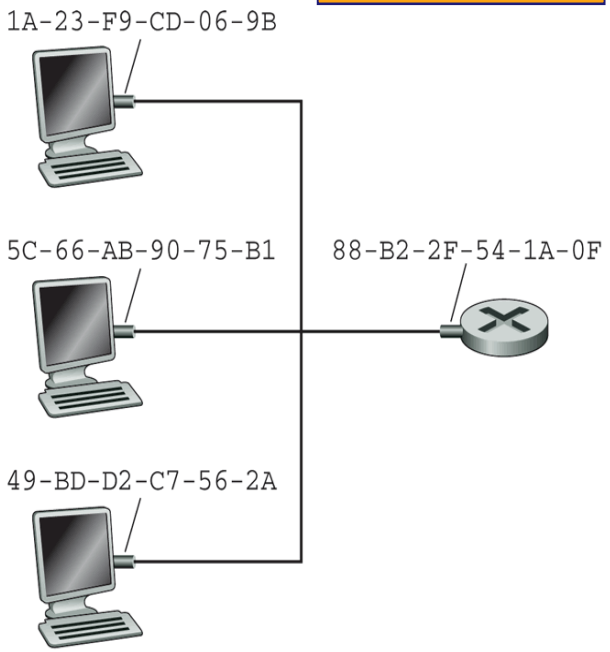


Fig 5.17 [KR12]

mm. Ethernet, Bluetooth, IEEE 802.11 langattomat verkot käyttävät

ARP-protokolla

(Address Resolution Protocol)

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

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"?

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

ARP-taulun sisältö kootaan suorituksen aikana, aluksi tyhjä

IPv6:ssa Neighbour Discovery Protocol (NDP)

Kytkimen kytkentätaulu

-tietojen keruu saapuvista kehyksistä

Aluksi taulu on tyhjä

Saapuva kehys

- **Lähteen MAC-osoite** x, kohteen MAC-osoite y, tuloportti p, yms

Lähde X ei ole taulussa =>

- Lisää (X, p ,TTL) tauluun eli **kytkin oppii, että osoite X on saavutettavissa portin p kautta**

Lähde X on taulussa => päivitä TTL

Kohde Y ei ole taulussa =>

- Lähetetään kehys kaikkiin muihin portteihin = **tulvitus** (flooding)
- Opitaan myöhemmin Y:n oikea portti jostain sen lähettämästä kehyksestä

Lähde X ja kohde Y jo taulussa

- X ja Y samassa portissa => hylkää kehys **(on jo oikeassa aliverkossa)**
- X ja Y eri porteissa => lähetä kehys Y:n porttiin

Reititin, kytkin ja keskitin

Reititin (router) - Verkkokerros

Kytkin (switch) - Linkkikerros

Keskitin (hub) - Fyysinen kerros

Vertailua

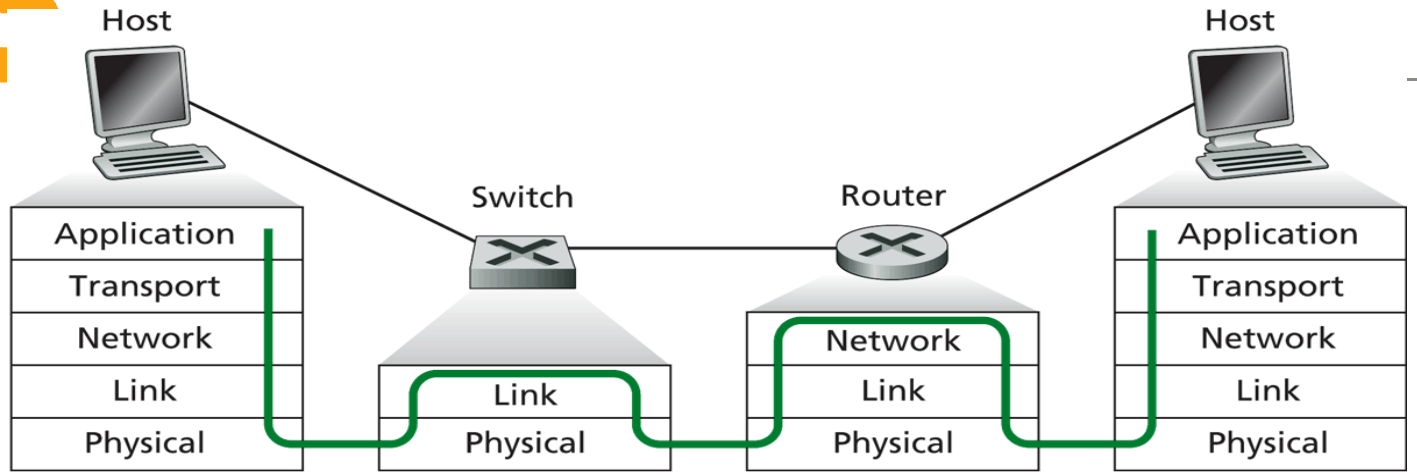


Fig 5.24 [KR12] ♦ Packet processing in switches, routers, and hosts

Table 5.1 [KR12]

	Keskitin (hub)	Kytkin (switch)	Reititin (router)
Traffic isolation	no	yes	yes
Plug and play	yes	yes	no
Optimal routing	no	no	yes
Cut through	yes	yes	no

Kertauskysymyksiä

Miten lähiverkko rakennetaan?

Reititin vs. kytkin vs. keskitin?

IP-osoite vs. MAC-osoite?

ARP-protokolla ja ARP-taulu?

Takaperinoppiminen ja kytkentätaulu?

Bittivirheiden havaitseminen?

CRC?

Lähetyskanavanjako?

CSMA/CD?

ks. kurssikirja s. 501

Langattoman verkon komponentit

Fig 6.1 [KR12]

- Tukiasema
  - LAN-yhteys
  - pääsy Internetiin
- Langattomat linkit
  - koneesta tukiasemaan
  - koneesta koneeseen
  - Rajattu kuuluvuusalue
- Isäntäkoneet
  - Laptop, PDA, IP-puhelin
  - Suorittaa sovelluksia
  - kiinteä tai liikkuva
- Haasteet
  - virhealtis linkki
  - liikkuva työasema

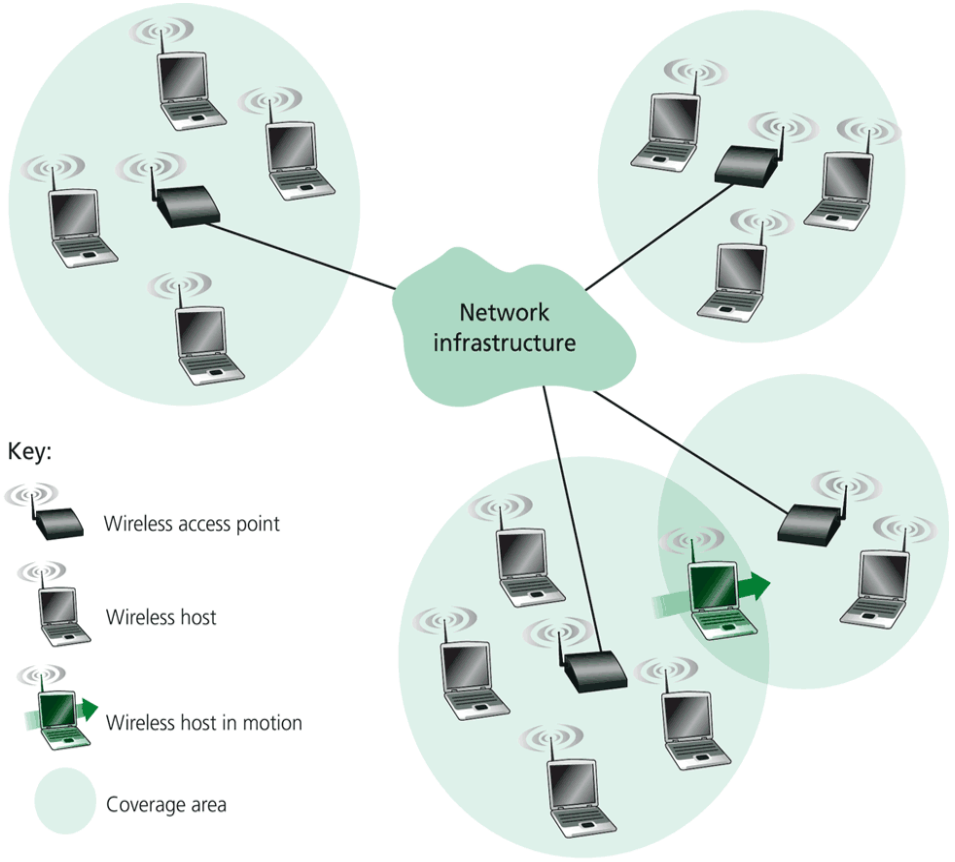


Figure 6.1 ♦ Elements of a wireless network

Kurssikoe

Kurssikoe 11.12.2012 16.00-19 A111

Tänään luennon jälkeen voin esitellä
Nodes laboratoriota 1. krs:n aulassa

Haluatteko vielä yhden kertauskerran
- maanantaina 10.12. klo 12 tai
- keskiviikkona 5.12. klo 14 ?