

5. Verkkokerros

- **sovelluskerros**
 - ‘asiakas’
- **kuljetuskerros**
 - ‘end-to-end’
- **verkkokerros**
 - ‘deliver packets given to it by its customers’
- **siirtoyhteyskerros**
- **peruskerros**

19.10.2000

1

Verkkokerroksen palvelut

- **tavoitteet**
 - palvelut riippumattomia aliverkkojen tekniikasta
 - kuljetuskerros eristettävä aliverkkojen ominaisuuksista
 - lukumäärä
 - tyypit
 - topologia
 - kuljetuskerroksen käyttämät verkko-osoitteet globaaleja

19.10.2000

2

connection-oriented ~ connectionless

- **yhteydetön (Internet, 30 vuoden kokemus)**
 - aliverkot ovat luonnostaan epäluotettavia
 - tehtävä: bittien kuljetus
 - operaatiot: send packet, receive packet
 - virheen tarkistus, vuonvalvonta isäntäkoneille
- **yhteydellinen (puhelin 100 vuoden kokemus)**
 - muodostetaan yhteys, neuvotellaan parametrit (palvelunlaatu (QOS), kustannus)
 - kaksisuuntainen kuljetus, paketit järjestyksessä

19.10.2000 vuonvalvonta virhevalvonta

3

Verkkokerroksen tärkein tehtävä: reititys

- **(hajautettu) päätöksenteko reitistä**
 - yhteydellinen: alussa
 - yhteydetön: jatkuvasti
- **jatkuvaa muutosta verkossa**
 - rikkoutuvat komponentit, muuttuva topologia
- **ristiriitaisia vaatimuksia reititykselle**
 - optimaalisuus /reiluus (fairness)
- **reitityksen suorituskyky**
 - mean packet delay, network throughput

19.10.2000

4

Reititysalgoritmi

- **Päittää, mikä reitti valitaan**
 - mihin paketti ohjataan seuraavaksi
- **dynaaminen verkkoympäristö => dynaaminen reititys**
 - jatkuvaan verkon tarkkailuun perustuva
 - Internetin reititys
 - muuttumaton ympäristö => käytetään kerran laskettuja reittejä tai sovitua lähetystapaa
 - tulvitus (flooding)
 - Dijkstran algoritmeilla lasketut lyhyimmät reitit

19.10.2000

5

Tulvitus

- jokainen saapunut paketti lähetetään kaikille muille ulosmenoille
 - => verkko täyttyy pian paketeista
- eri tapoja tulvituksen lopettamiseen
 - käsitellään harjoituksissa
- käyttö
 - tietyissä erityistilanteissa tilanteissa hyödyllinen
 - käsitellään harjoituksissa

19.10.2000

6

Dijkstran algoritmi

- 'lyhyin' reitti yhdestä solmusta muihin

- A -> {muut solmut}

- kaarihin liittyy kustannus

- kapasiteetti (bps)

- viive: hyppyjä, aikaa

- raha

- virhetodennäköisyys

19.10.2000

7

Algoritmi

- merkitään D_i on solmun i tähän asti tutkituista reiteistä solmuun A halvin kustannus eli lyhyin pituus

- verkko $G = (V, E)$, V on solmujen joukko, E kaarten joukko

- olkoon d_{ij} on kaaren (i,j) kustannus (> 0). Jos kaarta ei ole, d_{ij} on ääretön

- algoritmissa oletetaan, että kaikki kustannukset ovat ei-negatiivisia

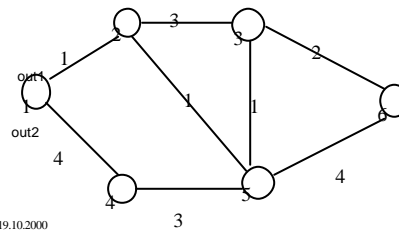
19.10.2000

8

1. $P := \{1\}$; $D_1 := 0$; $D_j := \infty$ ($j <> 1$);
2. while $P \neq V$ do
3. etsi solmu i , joka ei vielä ole joukossa P ja jolle $D_i = \min D_j$ P :hen kuulumattomista solmuista
4. $P := P \cup \{i\}$
5. kaikille muille P :hen kuulumattomille solmuille j $D_j := \min\{D_j, d_{ij} + D_i\}$
6. end while
7. end

Esimerkki

- Tarkastellaan esimerkkinä verkkoa



19.10.2000

10

1. $P = \{1\}$; $D_1 := 0$; $D_2 := 1$; $D_3 := \infty$; $D_4 := 4$; $D_5 := \infty$; $D_6 := \infty$

3. pienin D_i on solmulla 2 (=1)

4. $P = \{1, 2\}$

5. $D_3 := 1 + 3 = 4$, $D_4 = 4$, $D_5 := 1 + 1 = 2$, $D_6 = \infty$

3. pienin D_i nyt solmulla 5 (=2)

4. $P = \{1, 2, 5\}$

5. $D_3 := 1 + 2 = 3$, $D_4 := 4$, $D_6 := 4 + 2 = 6$

3. pienin D_i solmulla 3 (=3)

4. $P = \{1, 2, 3, 5\}$

5. $D_4 := 4$, $D_6 := 2 + 3 = 5$;

3. Pienin D_i solmulla 4 (=4)

4. $P = \{1, 2, 3, 4, 5\}$

5. $D_6 = 5$

4. $P = \{1, 2, 3, 4, 5, 6\}$

Löydetyt reitit ja kustannukset

- 1-> 2 :1
- 1-> 2->5->3: 3
- 1-> 4: 4
- 1->2->5: 2
- 1->2->5->3->6: 5

Solmu	linkki	kustann.
2	2	1
3	2	3
4	1	4
5	2	2
6	2	5

Solmulle 1

19.10.2000

13

Reititystaulu

- **Kukin reititin pitää kirjata reititiedoista**

- minne paketti seuraavaksi lähetetään

Kohde	minne lähetetään
Abc	reititin D, ulosmeno 2
...
Xyz	reititin T, ulosmeno 3

- **reitittimien tietojen hankinta ja ylläpito?**

- erityisen nopeasti muuttuvassa hyvin isossa verkossa

19.10.2000

14

Reititystietojen keruu

- kukin reititin kerää 'kustannustietoja' omasta ympäristöstään
 - esim. viiveet naapurireitittimiin
- ja vaihtaa tietoja muiden reitittimien kanssa
 - tai lähettää tiedot reitittimelle, joka keskitetysti laskee parhaat reitit
- kukin laskee esim. Dijkstran algoritmilla parhaat reitit koko verkosta
 - tai saa tarvitsemansa reititiedot ne laskeneelta

19.10.2000

15

Etäisyysvektoreititys (distance vector)

- **Arpanetin alkuperäinen reititys**
 - vieläkin jonkin verran käytössä Internetissä
- **kullakin reitittimellä reititystaulu**
 - kullekin verkon reitittimelle
 - ulosmenolinja
 - aika/etäisyys kohteeseen
 - hyppyjen lkm
 - arvioitu viive
 - jononpituus
 - jokin mitattavissa oleva

19.10.2000

16

reititystaulun ylläpito

- **tietojen vaihto naapurireitittimien kanssa**
 - tietyin aikavälein
 - tilan vaihtuessa
- **lasketaan uudet reititaulut**
 - 'kustannus' naapuriin +
 - naapurin ilmoittama 'kustannus' kohteeseen
 - kullekin solmulle valitaan pienimmän 'kustannuksen' reitti

19.10.2000

17

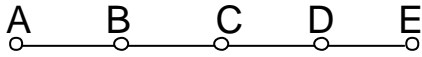
Ongelma: tietojen muuttumisnopeus

- **tietojen muuttamiseen kuluu aikaa**
- **reagoi nopeasti hyviin uutisiin**
 - uusi nopea reitti löytynyt/linkki jälleen pystyssä
 - tieto etenee joka vaihdossa yhden hypyn
- **reagoi hitaasti huonoihin uutisiin**
 - linkki nurin => etäisyys ääretön
 - joka vaihdossa 'paras arvio' huononee yhdellä
 - **count - to - infinity** -ongelma

19.10.2000

18

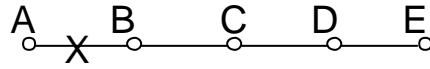
Hyvät uutiset etenevät nopeasti:



Aluksi yhteys A:han on poikki ja sitten linkki AB toimii taas:

B	C	D	E
ääretön	ääretön	ääretön	ääretön
1	ääretön	ääretön	ääretön
1	2	ääretön	ääretön
1	2	3	ääretön
1	2	3	4

Huonot uutiset etenevät hitaasti:



Toimiva linkki katkeaa välillä AB:

B	C	D	E
1	2	3	4
3	2	3	4
3	4	3	4
5	4	5	4
5	6	5	6
7	6	7	6
7	8	7	8

Linkkitilareititys (Link State Routing)

• reitittimen tehtävät

- selvitettävä naapurit ja niiden osoitteet
- mitattava etäisyys/ kustannus naapureihin
- koottava tietopaketti ko. tietoista
- lähetettävä tietopaketti kaikille reitittimille
- laskettava lyhin reitti kaikkiin muihin reitittimiin esim. Dijkstran algoritmilla

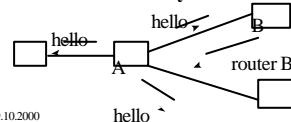
19.10.2000

21

Naapurien löytäminen

- reititin lähettää jokaiseen kaksipisteyhteyteen HELLO-paketin
- linjan toisessa päässä oleva reititin vastaa ja lähettää nimensä

- router ID
- nimien oltava yksikäsitteisiä koko verkossa



19.10.2000

22

Etäisyyden mittaaminen

• kaikille naapureille ECHO-paketti

- vastaanottajan palautettava paketti välittömästi

• => kiertoviive (round-trip-time)

- dynaaminen etäisyyssmitta

• pitäisikö ottaa kuormitus huomioon?

- kello käynnistetään, kun paketti viedään jonoon
- kello käynnistetään, kun paketti lähtee
- kuormitus mukana kuvaa todellista tilannetta
- jos kuormitus mukana => reititys muuttaa kuormitusta => reititys suosii huonoa reittiä

19.10.2000

23

Tietopakettien kokoaminen

• muodostus

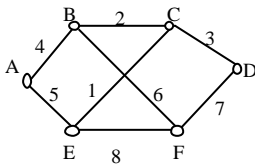
- tietyin aikavälein
- kun muutoksia havaittu

• sisältö

- reitittimen tunnus
- paketin järjestysnumero
- paketin ikä
- etäisyydet kuhunkin reitittimen naapuriin

19.10.2000

24



B	
seq	
age	
A	4
C	2
F	6

Tietopaketin jakelu

- **käytetään tulvitusta (n. 10 minuutin välein)**
 - pidetään kirjaa jo nähdyistä paketeista
 - reititin A, paketti 145
 - ⇒ paketti lähetetään korkeintaan kerran
 - paketissa elinaikalaskuri (age, time-to-live)
 - väärät ja vanhentuneet tiedot katoavat aikanaan vaikka reititin itse olisikin vikaantunut
- **tietopaketit kuitataan**
 - linjavirheiden takia

19.10.2000

26

Miksi elinaikalaskuri on tarpeen?

- **virheellinen järjestysnumero**
 - kaatunut reititin aloittaa väärästä numerosta
 - edennyt jo pakettiin 204 ja aloittaa uudestaan paketista 0 => kaikki seuraavat paketit hylätään duplikaatteina pakettiin 205 saakka
 - virhe tietopaketin seq-kentässä
 - 4 muuttuu virheellisesti 65540:ksi => seuraavat paketit hylätään pakettiin 65541 saakka

19.10.2000

27

elinaikalaskuri (TTL-laskuri)

- **laskuri vähenee ajan kuluessa**
 - vähenee yhdellä sekunnin välein
- **paketti tuhotaan, kun laskuri = 0**
 - vanhentunut (virheellinen) tieto poistetaan
 - pitkäkö elinaika >> päivitysten väli
 - tuhotaan vain jos reititin kaatunut
 - usea (6) paketti on jäänyt saapumatta reitittimeltä
- **käytössä myös tulvituksessa**
 - kukin reititin vähentää yhdellä

19.10.2000

28

Lisäparannuksia

- **paketteja ei lähetetä välittömästi eteenpäin**
 - ne jätetään odottamaan
 - jos samalta reitittimeltä tulee muita paketteja, niistä valitaan vain yksi, tuorein edelleenlähetettäväksi

19.10.2000

29

Reittitaulun laskeminen

- **kukin reititin laskee omat reittitaulunsa**
- **kaikki tarvittava tieto on saatu tietopakettien avulla**
 - kukin linkki molempiin suuntiin
- **laskeminen Dijkstran algoritmilla**
 - lyhyin reitti kuhunkin muuhun reitittimeen
 - isoissa verkoissa voi olla muisti- ja laskenta-aikaongelmia

19.10.2000

30

ongelmia

- **väärin toimiva reititin**
 - kertoo vääriä tietoja
 - ei välitä tietopaketteja
 - väärentää tietopaketteja
 - laskee reitit väärin
- **isossa verkossa aina joku toimii väärin**
 - tavoitteena rajata ongelmat pienelle alueelle

19.10.2000

31

Käyttö

- **paljon käytetty nykyisissä verkoissa**
 - Internetin OSPF-protokolla
 - ISO:n IS-IS -protokolla

19.10.2000

32

Hierarkkinen reititys

- **reitityksen skaalautuvuus**
 - isossa verkossa runsaasti reitittimiä (Internet: miljoonia)
 - reititystaulut suuria
 - reittien laskeminen raskasta
 - tietopaketit kuluttavat linjakapasiteettia
- **hallinta-autonomia => autonominen järjestelmä AS**
 - organisaatio päättää omista asioistaan
 - myös reitityksestä
 - oma sisäinen reititystapa

19.10.2000

33

Reitityshierarkia

- **Ylimmällä tasolla AS**
 - sama reititys AS:n sisällä
 - tehokkuus tärkeää
 - reititys AS:ien välillä
 - 'poliittinen asia'
- **AS:n sisällä alueita**
 - jaetaan reitittimet ryhmiin (alueet, regions)
 - kukin reititin tuntee kaikki alueensa sisällä
 - tietää mikä reititin hoitaa liikenteen muihin alueisiin

19.10.2000

34

Hierarkkisen reitityksen ongelmat

- **reititin pituus kasvaa**
 - aina ei voida käyttää optimaalista reittiä
 - yleensä siedettävä
- **hierarkiatasojen määrä**
 - suorituskyky
 - hallinto

19.10.2000

35

5.4 Internetworking

- **verkot erilaisia**
 - nyt ja aina
- **verkkoja yhdistävät**
 - **toistin**: bittien kopiointia
 - **silta**: kehys, store-and-forward
 - **reititin**: paketti, store-and-forward, erilliset verkot
 - kulj.kerr. yhdyskäyt.: tavuvirta kuljetuskerroksessa
 - sovelluskerr. yhdysk.: sovelluksen tietoyksiköitä

19.10.2000

36

• **silta**

- tunnettava kehysotsikot
- ei tarpeen tietää hyötykuorman takana olevaa protokollaa

• **reititin**

- tunnettava verkkoprotokolla
- alakerran toimintatavoista ei väliä

• **käytännössä termejä käytetään vapaammin!**

Verkkojen erot

• **palvelu**

- yhteydellinen / yhteydetön

• **protokolla**

• **osoittaminen**

- yksitasoinen / hierarkkinen

• **monilähetys/yleislähetys**

- on / ei

• **paketin koko**

lisää eroja:

- palvelulaatu
- virheiden käsittely
- vuon valvonta
- ruuhkan valvonta
- turvaus
- parametrit
- laskutus

• **ongelmana on erilaisten toiminnallisuuden yhteensopivuus**

- luotettavuus
- ruuhkan valvonta
- kuittaukset
- toimitusaikatakuut

Yhteydettömien verkkojen yhdistäminen

• **verkkokerroksen protokollien oltava (lähes) samoja**

• **osoittaminen**

- IP: 32-bittinen osoite
- OSI: puhelinnumeron kaltainen osoite
- osoitteiden yhteensovittaminen?
- globaaliosoiteavaruus? standardi?

Yhdistetyn verkon reititys

• **kahden tason reititysalgoritmi**

- kunkin verkon sisällä (intranet routing)

• **interior gateway protocol**

- verkkojen välillä (internet routing)

• **exterior gateway protocol**

- gateway tässä vanhempi termi reitittimelle!

• **eroja**

- EGP eri maiden välillä

- EGP: erilaiset verkkokustannukset, erilainen QoS

Pakettien paloittelu (fragmentation)

- **kaikissa verkoissa paketilla jokin maksimikoko**
 - laitteisto (TDM-viipaleen pituus)
 - käyttöjärjestelmä (käytetty puskurinkoko)
 - protokolla (pituuskentän bittien lukumäärä)
 - standardinmukaisuus
 - virheistä johtuvan uudelleenlähetyksen vähentäminen
 - tasapuolisuuden tavoite
- **48 tavua (atm) => 65515 tavua (IP)**

19.10.2000

43

Liian iso paketti verkkoon

- **liian iso paketti paloitellaan yhdyskäytävässä**
- **missä paketti kootaan?**
 - samassa verkossa, missä paloiteltiin
 - kaikki paketit ohjattava samaan yhdyskäytävään
 - jatkuvaa pilkkomista ja kokoamista!
 - vasta määränpäässä
 - pieni pakettikoko => lisää yleisrasitetta
 - kaikkien solmujen kyettävä kokoamaan paketteja

19.10.2000

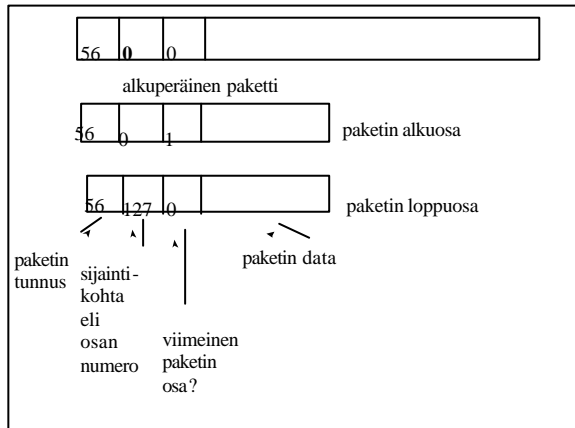
44

Pakettien kokoaminen

- **edellyttää palojen 'numerointia'**
 - on tiedettävä, minkä paketin mikä osa on kyseessä
- **kaikissa paloissa alkuperäisen paketin tunniste + sijainti paketissa**
 - sijainti: pakettiin kuuluvan ensimmäisen tavun sijainti alkuperäisessä paketissa
- **lisäksi tieto, onko pala paketin viimeinen**

19.10.2000 • tai tiedettävä paketin pituus

45



5.5 Internetin verkkokerros

- **Internet**
 - on kokoelma 'itsenäisiä' aliverkkoja eli autonomisia järjestelmiä (AS, Autonomous Subsystem)
 - joita yhdistää runkolinjat
- **IP-protokolla**
 - verkkotason protokolla, joka pitää Internetin koossa
 - tavoite: kuljettaa paketti (datasähke, datagram) lähteestä kohteeseen yli kaikkien välissä olevien erilaisten verkkojen

19.10.2000

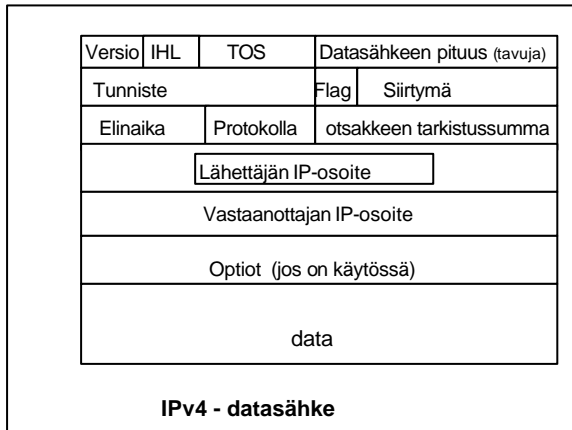
47

5.5.1 IP-protokolla

- **IP-datasähke**
 - otsake
 - dataosa
- **otsake**
 - 20 tavun kiinteä osa
 - tunnistetiedot, pituustiedot, tarkistusbitit (-summa)
 - osoitteet, minkä kuljetusprotokollan sanoma
 - liian pitkän paketin paloittelu ja kokoaminen
 - erilaisen palvelun tarjoaminen eri sovelluksille
 - vaihtelevan mittainen valinnainen osuus
 - lisäoptioita

19.10.2000

48



IP-otsakkeen kentät

- **Versio IPv4 (IPv6)**
- **IHL**
 - otsakkeen pituus vähintään viisi 32 bitin sanaa (20-60 tavua)
- **type of service**
 - kertoo halutun palvelun
 - nopeus, luotettavuus, kapasiteetti
 - ääni <-> tiedostonsiirto
 - yleensä ei käytössä (käytössä uusissa Cisco-reitittimissä)

19.10.2000

Type of service -bitit:

- **presedence-kenttä** (3 bittiä)
 - sanoman **prioriteetti** 0-7
 - 0 normaali
 - 7 verkon valvontapaketti
- **D-bitti, T-bitti, R-bitti**
 - mikä on tärkeää yhteydessä
 - D: viive (Delay),
 - T: läpimeno (Throughput)
 - R: luotettavuus (Reliability)
- lisäksi vielä 2 käyttämätöntä bittiä

19.10.2000

IP-otsakkeen kentät jatkuvat

- **Datagram length**
 - koko datasähkeen pituus
 - maksimi 65535 tavua
 - maksimipituus vielä riittävä, mutta tulevaisuuden nopeille verkoille jo ongelma
 - yleensä koko 576 -1500 taavua
- **Identification**
 - datasähkeen numero
 - kaikissa saman datasähkeen osissa sama tunnus

19.10.2000

IP-otsakkeen kentät jatkuvat: liput

- **DF- bitti (Don't fragment)**
 - kieltää paloittelun
 - esim. jos vastaanottaja ei kykene kokoamaan datasähkettä
- **MF- bitti (More fragments)**
 - ilmoittaa, onko datasähkeen viimeinen osio vai tuleeeko vielä lisää

19.10.2000

IP-otsakkeen kentät jatkuvat

- **Fragment offset**
 - osion paikka datasähkeessä
 - osioiden oltava 8 tavun monikertoja (paitsi viimeisen)
 - 13 bittiä => korkeintaan 8192 osiota yhdessä datasähkeessä
- **lisäksi 1 käyttämätön bitti**

19.10.2000

IP-otsakkeen kentät jatkuvat

• Time to live

- rajoittaa paketin elinaikaa
- maksimi 255 sekuntia
- vähenee
 - joka hypyllä reitittimestä toiseen
 - myös odottaessaan reitittimessä (ei yleensä)
 - paketti hävitetään kun laskuri menee nolliille

• Protocol

- mille kuljetuskerrokselle kuuluu
- esim. TCP- tai UDP-siirtoon kuuluva

19.10.2000

55

IP-otsakkeen kentät jatkuvat

• Header checksum

- tarkistussumma lasketaan vain otsakkeelle
- 16-bitin sanat lasketaan yhteen yhden komplementin aritmetiikalla
- laskettava uudestaan joka reitittimessä

• Source address, Destination address

- kohteen ja lähettäjän osoitteet muodossa
 - verkon numero ja isäntäkoneen numero
 - = IP-osoite

19.10.2000

56

IP-otsakkeen kentät jatkuvat

• Options

- vaihtelevan mittaisia
 - 1. tavu kertoo option koodin
 - voi seurata pituuskenttä
 - datakenttiä
 - täytetty jotta 4 tavun monikertoja
- käytössä 5 optiota
 - mutta reitittimet eivät välttämättä ymmärrä

19.10.2000

57

Optiot

– Security

- datasihteen luottamuksellisuus ja salassapidetävyys

– Strict source routing

- datasihteen kuljettava tarkalleen annettua reittiä

– Loose source routing

- kuljettava ainakin annettujen reitittimien kautta

– Record route

- reitin varrella olevat reitittimet liittävät tunnuksensa

– Timestamp

- tunnuksen lisäksi liitettävä myös aikaleima

19.10.2000

58

5.5.2 IP-osoitteet

• jokaisella verkon isäntäkoneella ja reitittimellä on oma yksikäsitteinen osoite muotoa

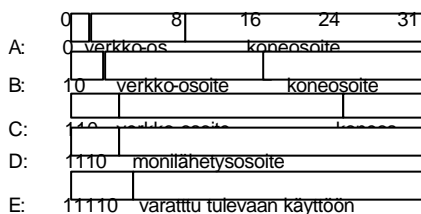
- verkon numero
 - isäntäkoneen numero
- ### • osoite on 32-bittinen
- osoitteen luokasta riippuen bitit jaetaan verkon numeroon ja isäntäkoneen numeroon eri tavoin

• osoitteet palveluntarjoajille jakaa ICANN (The Internet Corporation for Assigned Names and Numbers)

- nämä puolestaan jakavat muille

19.10.2000

59



IP-osoitteiden muodot

(alkuperäinen luokallinen osoitus)

IP-osoitteiden luokat

• A-luokka

- hyvin isoille verkoille, joissa paljon koneita:
 - 1 bitti tunnukseen
 - 7 bittiä verkko-osoitteeseen, 24 bittiä isäntäkoneille
 - **126 verkkoa, 16 miljoonaa konetta/verkko**

• B-luokka

- keskikokoisille verkoille
 - 2 bittiä tunnukseen, 14 bittiä verkoille, 16 bittiä koneille
 - **16382 verkkoa, 65528 konetta/verkko**

19.10.2000

61

• C-luokka

- esim. LANeille
 - 3 bittiä tunnisteeseen, 21 bittiä verkoille, 8 bittiä verkon koneille
 - **noin 2 miljoonaa verkkoa, kussakin korkeintaan 254 konetta**

• D-luokka

- monilähetysosoite
 - tunnus: 1110

• E-luokka

- varattu tulevaan käyttöön
 - tunnus: 11110

• osoitteet merkitään usein desimaalimuodossa

- kukin osoitteen neljästä tavusta kirjoitetaan desimaalilukuna (0-255)
- luvut erotetaan pisteellä
- esim.
 - heksadesimaaliosoitte C0 29 06 14 on 192.41.6.20 eli C0 => 192, 29 => 41, 06 => 6, 14 => 20
- pienin osoite on 0.0.0.0 ja suurin 255.255.255.255

19.10.2000

63

Osoiteluokkien ongelmia

• verkon kasvu => ongelmia

- C-luokan verkossa max 256 osoitetta
 - liian vähän useimmille yrityksille => tarvitsevat B-luokan osoitteen tai monta C-luokan verkko-osoitetta
- B-luokan verkkoja liian vähän (max 16382) ja niissä liian paljon osoitteita (max 65536)
 - 100000 verkkoja jo 1996!
 - useassa B-verkossa alle 50 konetta

• => B-luokan osoitteita tuhlaantuu ja osoitteista pulaa

19.10.2000

64

CIDR (Classless InterDomain Routing)

- verkko-osa voi olla minkä tahansa kokoinen (ei vain 8,16,24 bittiä)
 - a.b.c.d/x, jossa x ilmoittaa verkko-osan bittien lukumäärän
- esim. yritykselle, jolla 2000 konetta varataan 2048 = 2^{11} koneosoitetta, jolloin verkko-osaa varten jää 21 bittiä
 - C-luokan verkkoja
- yritys voi itse vielä jakaa koneosoitteen 11 bittiä aliverkko-osoitteeksi ja koneosoitteeksi

19.10.2000

65

CIDR-idea jatkuu

- jaetaan osoitteet neljään osaan, kukin osa varataan yhdelle maanosalle (Eurooppa, Pohjois-Amerikka, Etelä-Amerikka, Aasia+Pasific)
 - kullekin noin 32 miljoonaa osoitetta
 - 320 miljoonaa jää vielä varastoon
- reititetään myös maanosien mukaan
 - osoitteet 194.0.0.0 - 195.255.255.255 Eurooppaan
- => pienemmät reititustaulut

19.10.2000

66

Muita Internet-protokollia

- ICMP (Internet Control Message Protocol)
- ARP (Address Resolution Protocol)
- OSPF (Open Shortest Path First)
- BGP (Border Gateway Protocol)

- IPv6
 - käsitellään Tietoliikenne II -kursilla

19.10.2000

67

5.5.4 ICMP (Internet Control Message Protocol)

- reitittimet ilmoittavat verkon ongelmista toisilleen
- käytetään yleensä testaukseen
- ICMP-sanomat kapseloidaan IP-paketteihin
- 12 erilaista sanomaa määritelty
 - Destination unreachable, Time exceeded, Parameter problem
 - Source quench, Redirect
 - Echo request, Echo reply (ping)
 - Timestamp request, Timestamp reply

19.10.2000

68

5.5.5. OSPF (Open Shortest Path First)

- linkkitilaprotokolla reitittämään yhden AS:n sisällä
 - linkkikustannukset naapureihin
 - tiedot muille tulvittamalla
 - kukin laskee parhaat reitit Dijkstran algoritmilla
- parannuksia
 - turvallisuus: reitittimien autentikointi
 - useita yhtä hyviä reittejä
 - eri kustannusmittoja eri tyyppiselle liikenteelle
 - hierarkkinen reititys

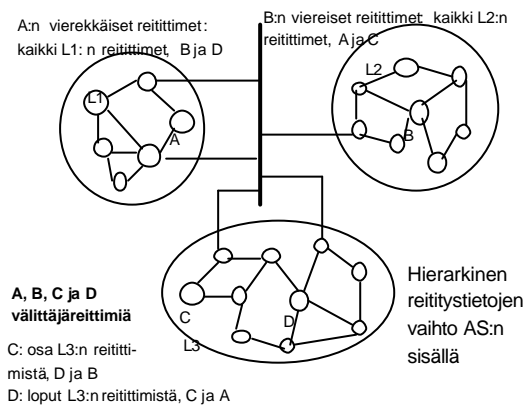
19.10.2000

– AS jaettu alueiksi (area)

69

- AS (autonomous system)
 - näistä muodostuu Internet
 - yritysten ja organisaatioiden omat verkot
- AS:t ovat usein hyvin laajoja
 - => voidaan jakaa alueiksi
 - verkko tai verkkojoukko
 - alueen ulkopuolella sen topologia ei näy
- jokaisessa AS:ssä runkolinja-alue
 - alue 0
 - kaikki alueet kiinni runkolinjassa

- alueen sisällä kaikilla reitittimillä
 - sama linkkitilatietokanta
 - sama lyhimmän polun algoritmi
- reititin laskee lyhimmän polun kaikkiin muihin alueen reitittämiin
- alueiden välillä oleva reititin
 - tuntee molempien alueiden tietokannat ja lyhimmän polun algoritmit
- reititysinformaation vaihtoa ei kaikkien alueen reitittimien kesken, vaan 'vierekkäisten' (= vaihtavat tietoa keskenään)



- **kullekin eri etäisyysmitalle lasketaan omat kustannukset**

- eri reitit optimoitaessa viivettä, läpimenoa ja luotettavuutta

- **toiminnassa tarvitaan kolmenlaisia reittejä**

- alueen sisäisiä
 - reititin itse tietää lyhyimmän reitin
- alueiden välisiä
 - alueiden väliset reitit kulkevat **aina runkolinjaa pitkin**
 - reititin tietää lyhyimmän reitin runkolinjaan
- AS:ien välisiä
 - hoidetaan BGP-protokollalla

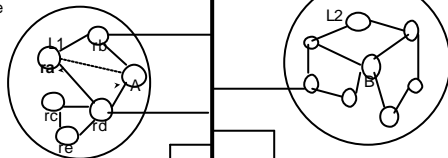
OSPF-sanomat

- **hello**
 - naapurien selville saaminen
- **link state update**
 - omien linkkikustannusten lähettäminen
- **link state ack**
 - vastaanotettujen linkkikustannusten kuittaus
- **database description**
 - tietokannan ajantasaisuuden selvittäminen
- **link state request**
 - toisen linkkikustannusten kysyminen

19.10.2000

74

ra mittaa etäisyydet naapureihinsa
rb:hen ja rc:hen ja lähettää tiedot
A:lle



A saa tiedot etäisyyksistä
kaikilta L1:n reitittimiltä
ja välittää tiedot
muille =>
ra osaa laskea
etäisyydet muihin
L1:n reitittimiin

A saa myös tiedot
muiden alueiden
etäisyyksistä B:ltä
ja D:ltä
=> rb:lle ja rd:lle =>
ra tietää kumpaa
runkoreitittintä rb vai
rd tulee kulloinkin
käyttää

- **eri reitit voivat olla 'yhtä pitkiä'**
 - => liikenne voidaan reitittää usean reitin yli
 - => kuormituksen tasapainoitus
- **eikä välttämättä kaikkia paketteja lähetetä samaa reittiä**
 - osa parasta reittiä
 - osa toiseksi parasta
- **lopputulokset voi olla parempi**

19.10.2000

76

5.5.6. BGP (Border Gateway Protocol)

- **AS:ien välillä**

- otettava huomioon eri AS:ien politiikat
 - AS:ien sisällä tärkeintä **tehokkuus**
 - AS:ien välillä kieltoja reitittää tiettyjen AS:ien kautta
- politiikat manuaalisesti BGR-reitittimiin
 - hyvin erilaisia sääntöjä

19.10.2000

77

BGP (jatkuu)

- **pohjimmiltaan etäisyysvektori-protokolla**

- tallettaa kunkin reitin koko polun
- kertoo naapureilleen käyttämänsä reitin
 - hylkää itsensä kautta kulkevat reitit

- **'reitikkustannuksen' laskeminen**

- eri tapoja laskea arviot
- 'kielleyille' reiteille etäisyys on ääretön

19.10.2000

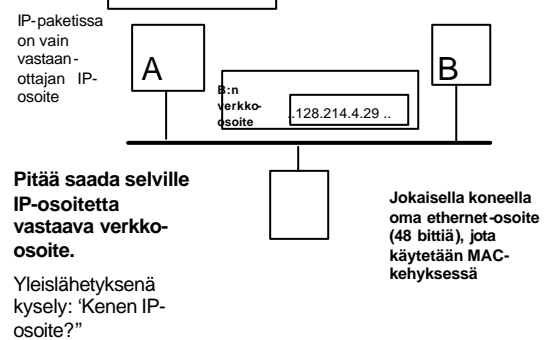
78

ARP (Address Resolution Protocol) (5.5.4.)

- **muuttaa IP-osoitteen siirtoyhteyskerroksen osoitteeksi**
 - lähiverkkoon liitetyt laitteet ymmärtävät vain LAN-osoitteita
 - esim. eetteriverkon 48-bittisiä osoitteita
- **yleislähetys lähiverkkoon**
 - “Kenellä on IP-osoite vv.xx.yy.zz ?”
 - vastauksena osoitteen omistavan laitteen lähiverkko-osoite

19.10.2000

79



• optimointia:

- kyselyn tulos välimuistiin
 - talletetaan muutaman minuutin ajan
- kyselijä liittää omat osoitteensa kyselyyn
- alustettaessa jokainen laite ilmoittaa osoitteensa muille
 - kysyy omaa osoitettaan
 - jos tulee vastaus, niin konfigurointivirhe

19.10.2000

81

Yhteenveto

- **Reititystä**
 - reititysalgoritmeja
 - etäisyysvektorireititys
 - linkkitilareititys
 - Dijkstran algoritmi
- **Verkkojen yhdistäminen**
 - pakettiverkot
 - pakettien paloittelu / kokoaminen

19.10.2000

82

• Internetin verkkokerroksen protokollia

- IPv4 -protokolla, IP-osoite
- ICMP
- reititysprotokollia
 - OSPF
 - BGP
 - ARP

19.10.2000

88