

Tero Höylä

OSPF-REITITYSPROTOKOLLAN OMINAISUUDET

Opinnäytetyö
Tietotekniikan koulutusohjelma


Marraskuu 2012




MIKKELIN AMMATTIKORKEAKOULU

Mikkeli University of Applied Sciences

KUVAILULEHTI

 MIKKELIN AMMATTIKORKEAKOULU <small>Mikkeli University of Applied Sciences</small>	Opinnäytetyön päivämäärä		
Tekijä(t) Tero Höylä	Koulutusohjelma ja suuntautuminen Tietoliikennetekniikka		
Nimeke OSPF-reititysprotokollan ominaisuudet			
Tiivistelmä Tämän työn tarkoituksena oli tutustua tarkemmin OSPF-reititysprotokollaan, joka on yksi laajimmin käytettyjä reititysprotokollia. Tutustuin OSPF-protokollaan aluksi teoriassa. Käytin yhtenä lähteenä John T. Moyn kirjaa OSPF - Anatomy of an Internet Routing Protocol. John T. Moy kuuluu työryhmään, joka kehitti OSPF-protokollan. Teoriaosuuden alussa tutustuin ensin tietoverkkoihin ja niissä tapahtuvaan tietoliikenteeseen. Tutustuin myös muutamiin muihin käytössä oleviin reititysprotokolliin ja niiden ominaisuuksiin. Selvitin miten reititysprotokollat luokitellaan ja miten ne eroavat toisistaan. RIP-protokollasta kerroin hieman tarkemmin, koska se on yksi vanhimpia nykyisiä reititysprotokollia ja koska OSPF kehitettiin sen korvaajaksi. Lopuksi keskityin OSPF-protokollaan ja sen ominaisuuksiin. Tutustuin OSPF-protokollaan myös käytännössä. Suunnittelin esimerkkiverkon, jota testasin luomalla siihen vikatilanteita. Testit suoritin Mikkelin ammattikorkeakoulun Mikpoli-rakennuksen luokassa MB316, mistä löytyy esimerkkiverkon käytännön toteutukseen tarvittava laitteisto.			
Asiasanat (avainsanat) Reititys, protokolla, tietoliikennetekniikka.			
Sivumäärä 43	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Kieli Suomi</td> <td style="width: 50%;">URN</td> </tr> </table>	Kieli Suomi	URN
Kieli Suomi	URN		
Huomautus (huomautukset liitteistä) Dijkstran algoritmi, esimerkkiverkon reitittimien asetukset ja reititystaulut			
Ohjaavan opettajan nimi Matti Juutilainen	Opinnäytetyön toimeksiantaja		

DESCRIPTION

 MIKKELIN AMMATTIKORKEAKOULU Mikkeli University of Applied Sciences		Date of the bachelor's thesis	
Author(s) Tero Höylä		Degree programme and option Information technology	
Name of the bachelor's thesis Features of OSPF routing protocol.			
Abstract <p>The purpose of this bachelor's thesis was to explore the OSPF routing protocol in more detail. OSPF is one of the most widely used routing protocols. At first, I explore the OSPF in theory. One source that I used, was a book called OSPF- Anatomy of an Internet Routing Protocol by John T. Moy. He is member of the working group that developed the OSPF.</p> <p>In addition to the OSPF protocol the theory part also explored networks and telecommunications solutions. I also introduced few other existing routing protocols and their features, how routing protocols are classified and how they differed from each other. I dealt with the RIP protocol more closely, because it is one of the oldest modern routing protocols and OSPF was developed to replace it.</p> <p>Finally I focused on the OSPF protocol and its properties and explored OSPF in practice I designed an example network, which I tested by creating fault conditions into it. These tests were performed in the classroom MB316 of the Mikpoli building of the Mikkeli University of Applied Sciences. This room had the hardware that I needed to put the example network into practice.</p>			
Subject headings, (keywords) Routing, protocol, information technology.			
Pages 43	Language Finnish	URN	
Remarks, notes on appendices Dijkstra's algorithm, settings and routing tables of the routers of an example network			
Tutor Matti Juutilainen		Bachelor's thesis assigned by	

SISÄLTÖ

1	JOHDANTO	1
2	TIETOVERKKO.....	2
2.1	Nykyisten tietoverkkojen historiaa	4
2.2	Viitemallit	5
2.3	Osoitteiden jakaminen ja selvittäminen tietoverkossa.....	7
3	IP-OSOITTEET JA REITITYS	9
3.1	IP-osoitteet.....	9
3.2	Reititys	12
4	REITITYSPROTOKOLLAT	14
4.1	RIP	15
4.2	EIGRP	17
4.3	IS-IS	19
4.4	BGP.....	20
5	OSPF-PROTOKOLLA	21
5.1	Linkkitilietokanta ja LSA	22
5.2	OSPF protokollapaketit	23
5.3	OSPF multiaccess-verkoissa.....	26
5.4	Alueet.....	27
5.5	Asetukset.....	28
6	OSPF-PROTOKOLLA KÄYTÄNNÖSSÄ	33
6.1	Esimerkkiverkko.....	34
6.2	Testit ja tulokset.....	35
7	YHTEENVETO	40
	LÄHTEET	42

LIITTEET

1 Dijkstran algoritmi

2 Esimerkkiverkon reitittimien asetukset ja reititystaulut

LYHENTEET

ABR	<i>Area Border Router</i> , alueiden rajalla toimiva reititin OSPF-protokollassa
AD	<i>Administrative Distance</i> , hallinnollinen etäisyys, reititysprotokollien ja siirtoyhteyksien hallinnollinen arvo
ARP	<i>Address Resolution Protocol</i> , protokolla, jolla selvitetään IP-osoitetta vastaava MAC-osoite
ARPANET	<i>Advanced Research Projects Agency Computer Network</i> , Internetin tekninen edeltäjä, Yhdysvaltain puolustusministeriön kehittämä tietoverkko
AS	<i>Autonomous System</i> , itsenäinen järjestelmä, Internetiin yhdistetty lähiverkko
ASCII	<i>American Standard Code for Information Interchange</i> , standardoitu kirjoitusmerkkikoodi
BDR	<i>Backup Designated Router</i> , varalla oleva hallitseva reititin OSPF-protokollaa käyttävässä multiaccess-verkossa
BGP	<i>Border Gateway Protocol</i> , operaattoreiden käyttämä tietoverkkojen yhdistämiseen tarkoitettu reititysprotokolla
CIDR	<i>Classless Interdomain Routing</i> , luokaton reititys
DHCP	<i>Dynamic Host Configuration Protocol</i> , IP-osoitteita dynaamisesti jakava protokolla
DNS	<i>Domain Name System</i> , www-palvelimien Domain-nimiä IP-osoitteiksi kääntävä palvelu

DR	<i>Designated Router</i> , hallitseva reititin OSPF-protokollaa käyttävässä multiaccess-verkossa
DROther	Reititin OSPF-protokollan <i>multiaccess</i> -verkossa
EGP	<i>Exterior Gateway Protocol</i> , itsenäisten järjestelmien yhdistämiseen tarkoitettu protokollatyyppi
EIGRP	<i>Enhanced Interior Gateway Routing Protocol</i> , Cisco Systemsin kehittämä reititysprotokolla, kehittyneempi versio IGRP-protokollasta
IETF	<i>Internet Engineering Task Force</i> , standardointijärjestö
IGP	<i>Interior Gateway Protocol</i> , itsenäisen järjestelmän (AS) sisäiseen reititykseen tarkoitettu protokollatyyppi
IGRP	<i>Interior Gateway Routing Protocol</i> , Cisco Systemsin kehittämä reititysprotokolla
IP	<i>Internet Protocol</i> , tiedonsiirtoprotokolla
IS-IS	<i>Intermediate System to Intermediate System</i> , reititysprotokolla
ISO	<i>International Organisation for Standardisation</i> , standardointijärjestö
JPEG	Kuvien tallennusformaatti
LAN	<i>Local Area Network</i> , lähiverkko
MAC-osoite	<i>Media Access Control</i> , tarkoittaa verkkokortin fyysistä osoitetta, 48-bittinen heksadesimaalikoodi
NAT	<i>Network Address Translation</i> , lähiverkon sisäisiä IP-osoitteita julkisiksi kääntävä palvelu

NSSA	<i>Not So Stubby Area</i> , aluetyyppi OSPF-reititysprotokollassa
OSI	<i>Open Systems Interconnection Reference Model</i> , OSI-malli, standardoitu tiedonsiirtokäytäntö tietoliikennetekniikassa
OSPF	<i>Open Shortest Path First</i> , reititysprotokolla
RFC	<i>Request For Comments</i> , IETF:n julkaisemia Internetiä koskevia standardeja
RIP	<i>Routing Information Protocol</i> , reititysprotokolla
SPF	<i>Shortest Path First</i> , Dijkstran algoritmin nimi
SPT	<i>Shortest Path Tree</i> , tietokanta, jonka perusteella OSPF-protokollaa käyttävä reititin tietää itsenäisen järjestelmän (tai alueensa) topologian
TCP	<i>Transmission Control Protocol</i> , varmennettu tiedonsiirtoprotokolla
TCP/IP	<i>Transmission Control Protocol / Internet Protocol</i> , standardoitu tiedonsiirtokäytäntö tietoliikennetekniikassa
UDP	<i>User Datagram Protocol</i> , varmentamaton tiedonsiirtoprotokolla, käytetään esimerkiksi videotiedostoissa
VLSM	<i>Variable Length Subnet Mask</i> , vaihtuvan pituinen aliverkon peite
VLAN	<i>Virtual Local Area Network</i> , virtuaalinen lähiverkko, samassa lähiverkossa olevien kytkimien portit konfiguroidaan ryhmiksi niin, että ryhmät toimivat kuin olisivat itsenäisiä lähiverkkoja
WAN	<i>Wide Area Network</i> , laajaverkko
WLAN	<i>Wireless Local Area Network</i> , langaton lähiverkko

1 JOHDANTO

Tämän työn tarkoituksena on tutustua tarkemmin OSPF-reititysprotokollaan, joka on yksi laajimmin käytettyjä reititysprotokollia. Tutustun OSPF-protokollaan sekä teoriassa että käytännössä. Teoriaosuudessa käytän yhtenä lähteenä John T. Moyn kirjaa *OSPF – Anatomy of an Internet Routing Protocol*. John T. Moy kuuluu työryhmään, joka kehitti OSPF-protokollan. Käytännön osuudessa suunnittelen esimerkkiverkon, jota testaan luomalla siihen vikatilanteita.

Käytän tässä työssä usein sanaa protokolla. Protokolla tarkoittaa yleisesti sovittua käytäntöä tai standardia. Tietoliikennetekniikassa protokollat jakautuvat reititettäviin protokolliin ja reititysprotokolliin. Selvitän tässä työssä mitä reititysprotokollat tekevät ja miten ne liittyvät reititettäviin protokolliin.

Käytän usein myös termejä reititys ja reititin. Reititys tarkoittaa reitittimien välityksellä tapahtuvaa tiedonsiirtoa. Reititin on eräänlainen tietokone, minkä ainoa tehtävä on reitittää datapaketteja. Se voi käyttää reititysprotokollia reitittäessään paketteja. Reitityksestä ja siinä käytettävistä osoitteista kerron tarkemmin työn teoriaosuudessa.

Tutustun tässä työssä tietoliikenteeseen sekä muutamiin yleisimpiin reititysprotokolliin keskittyen lopuksi OSPF-protokollaan. Tutustun OSPF:n ominaisuuksiin, kuten linkkitilietokantaan jonka avulla OSPF reitittää paketteja. Selvitän mitä linkkitilietokanta sisältää ja miten se muodostuu. Tutustun myös protokollapaketteihin, jotka kuljettavat OSPF:n reitityspäivityksiä. Kerron myös miten lähiverkko voidaan jakaa osiin OSPF:n avulla. Teoriaosuuden lopuksi kerron millä käskyillä OSPF otetaan käyttöön Cison reitittimissä.

Teoriaosuuden jälkeen esittelen OSPF-protokollan toimintaa käytännössä. Suunnittelen Cison *Packet Tracer*-ohjelmalla esimerkkiverkon, jolla esittelen OSPF:n toimintaa. Toteutan verkon käytännössä Mikkelin ammattikorkeakoulun Mikpoli-rakennuksen luokkaan MB316.

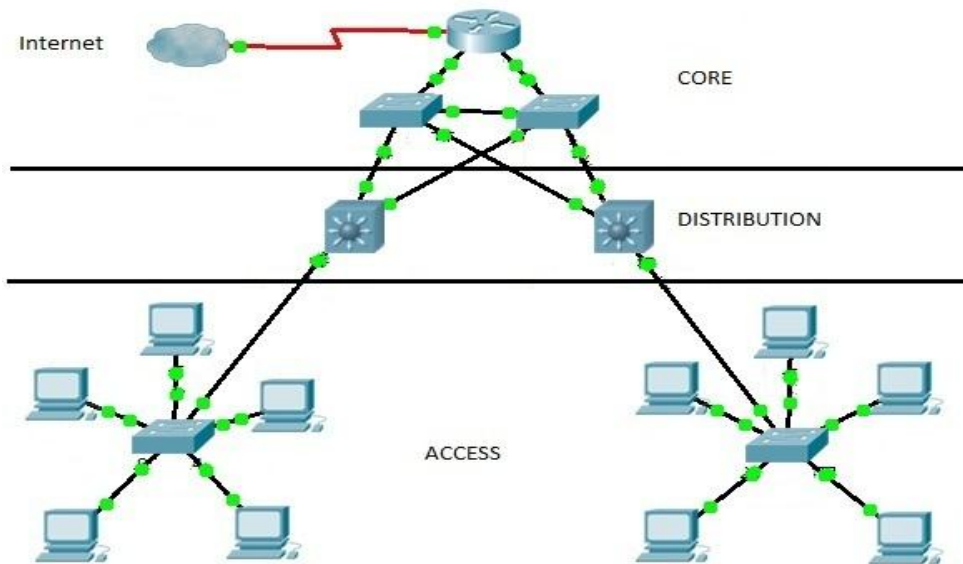
2 TIETOVERKKO

Internet on suuri pakettikytkentäinen tietoverkko (Moy 2000, 3). Sana pakettikytkentäinen tulee siitä, että tieto liikkuu verkossa sähköisinä datapaketteina. Lukuisat pienemmät tietoverkot yhdistyvät Internetiin ja sen kautta toisiinsa. Näitä pienempiä tietoverkkoja kutsutaan tietoliikennetekniikkaa käsittelevässä kirjallisuudessa nimityksellä *Autonomous System* (AS), mikä tarkoittaa itsenäistä järjestelmää. Termiä itsenäinen käytetään, koska jokaista verkkoa hallitsee sitä käyttävä organisaatio. Yleisempi nimitys näille järjestelmille on lähiverkko eli LAN (*Local Area Network*). Lähiverkkoja on yrityksissä, kouluissa ja yksityiskodeissakin. Lähiverkot yhdistyvät Internetiin palveluntarjoajan kautta. Palveluntarjoajia ovat yleensä puhelinoperaattorit. (Moy 2000, 32.)

Verkot koostuvat reitittimistä, kytkimistä, verkkokaapeleista ja päätelaitteista. Reitittimien tehtävä on välittää viestejä lähiverkkojen välillä (Moy 2000, 3). Ne välittävät viestejä myös lähiverkkojen sisällä olevien aliverkkojen välillä, esimerkiksi virtuaalisten lähiverkkojen (VLAN) välinen viestitys tapahtuu reitittimien kautta (Jaakohuhta 2002, 30). Kaikki viestit lähiverkkojen sisällä kulkevat kytkimien läpi. Kytkimet varmistavat, että viestit päätyvät oikeisiin kohteisiin. Päätelaitteet, eli tietokoneet, palvelimet ja verkkotulostimet liittyvät verkkokaapeleilla reitittimiin kytkimien kautta. (Jaakohuhta 2002, 209.)

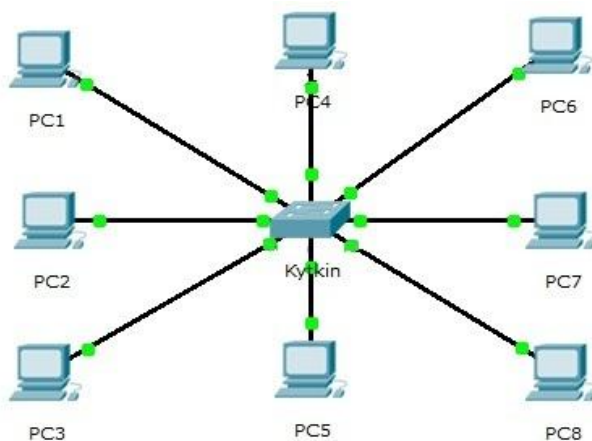
Kytкимиä on olemassa kahta kategoriaa. Peruskytkimet ovat toisen tason kytkimiä. Ne välittävät dataa MAC-osoitteiden perusteella. Kolmannen tason kytkimet osaavat huomioida myös IP-osoitteet. Nimitykset toinen ja kolmas taso tulevat OSI-mallin kerroksista (Khabardar ja Finn 1998, 7). OSI-mallin esittelen myöhemmin.

Ennen kuin reitittimet, kytkimet ja päätelaitteet liitetään toisiinsa, pitää suunnitella verkon rakenne, eli topologia. Suunnittelussa pitää ottaa huomioon laitteiden määrä, verkolle asetetut vaatimukset ja paikka, johon verkko rakennetaan. Verkon fyysinen topologia vaihtelee paikan mukaan, mutta loogisen topologian suunnittelua varten on olemassa malleja, kuten kuvissa 1 ja 2.



KUVA 1. Hierarkkinen topologia

Kuvassa 1 esittelen hierarkkisen topologian, mikä on yksi tapa kaapeloida lähiverkko. Siinä verkko jaetaan kolmeen kerrokseen. *Core*-kerroksella kytkimet ja reitittimet huolehtivat pakettien nopeasta välittämisestä verkon sisällä ja verkosta ulos. *Core*-kerroksella voidaan käyttää myös kolmannen tason kytkimiä. *Distribution*-kerroksella reitittimet tai kolmannen tason kytkimet (kuten kuvassa) välittävät tietoliikennettä eri aliverkkojen välillä. *Access*-kerroksessa käyttäjät yhdistyvät verkkoon. Kuvassa 2 on päätelaitteiden kytkemistapa lähiverkossa.



KUVA 2. Tähtitopologia

Kuvan 2 esimerkissä esittelen tähtitopologian, joka on nykyään yleisin tapa kytkeä tietokoneita lähiverkossa. Kuvassa 2 kahdeksan tietokonetta on liitetty kytkimen kautta toisiinsa verkkokaapeleilla. Käytännössä vastaava verkko toteutetaan niin, että kytkin on sijoitettuna kytkentäkaappiin ja siitä lähtevät kaapelit kulkevat esimerkiksi sei-

nään kiinnitetyssä kiskossa oleviin verkkopistorasioihin. Tietokoneet kytketään verkkokaapeleilla kiskon verkkopistorasioihin. Yleisin verkkokaapelityyppi nykyään on Ethernet-kaapeli.

2.1 Nykyisten tietoverkkojen historiaa

Internetin alkuperänä voidaan pitää ARPANETia. ARPANETin kehitystyö alkoi Yhdysvalloissa vuonna 1958 vastauksena Neuvostoliiton Sputnik-hankkeelle. Presidentti Eisenhower halusi Yhdysvaltojen olevan tieteen ja tekniikan kehityksen kärjessä. Vuonna 1962 Yhdysvaltain puolustusministeriö palkkasi J. C. R. Lickliderin suunnittelemaan tietoverkkoa. Lickliderin jälkeen projektia johtivat ensin Ivan Sutherland ja sitten Bob Taylor. Vuonna 1966 ARPANET yhdisti jo kolme tietokonetta. Yksi sijaitsi MIT:ssä, toinen Kalifornian yliopistossa, Berkleyssä, ja kolmas System Development Corporationin toimitiloissa Santa Monicassa. Vuoteen 1971 mennessä verkko yhdisti useita yliopistoja ja tutkimuslaboratorioita. (Doyle 2005, luku 1.2.)

Internet rakentui ARPANETin ympärille. ARPANETin laitteisto toimi Internetin ytimenä vuoteen 1985, jolloin NSFNET (*National Science Foundation NET*) korvasi sen. NSFNET toimi ytimenä vuoteen 1995, jolloin se poistui käytöstä. Nykyisen Internetin ytimenä toimivat kaupallisten Internet-palveluntarjoajien laitteet. (Moy 2000, 33.)

Lähiverkkotekniikoista yleisin on tällä hetkellä Ethernet. Se on myös yksi ensimmäisiä lähiverkkoja. Sen alkuperänä voidaan pitää ALOHA-nimistä radioverkkoa, joka kehitettiin Havaijilla 1960-luvulla. ALOHA-verkon kehitystyöstä vastasi Norman Abramson työryhmineen. ALOHA-verkko yhdisti Havaijin yliopiston tietokoneen Oahun saarella ympärillä olevien saarien ja merellä olevien laivojen päätteisiin. (Jaakohuhta 2002, 9.)

Ensimmäisen varsinaisen Ethernetin kehitystyö alkoi 1970-luvun alkupuolella Xeroxin PARC-tutkimuskeskuksessa (Palo Alto Research Center). Kehitystyöstä vastasi Robert Metcalfe, joka oli juuri palkattu tutkimuskeskukseen verkkotekniseksi asiantuntijaksi. Ensimmäisenä tehtävänä Metcalfen piti liittää Xeroxin ALTO-niminen tietokone ARPANETiin. (Jaakohuhta 2002, 11.)

Vuoden 1972 lopussa Metcalfe suunnitteli työoverinsa David R. Boggsin kanssa verkon, joka yhdisti joukon ALTO-tietokoneita Xeroxin kehittämään, maailman ensimmäiseen lasertulostimeen, nimeltä EARS. Tutkimustyössään Metcalfe oli tutustunut Abramsonin ALOHA-järjestelmään, ja siksi verkon työnimi oli ALTO ALOHA. Verkko saatiin toimimaan 22.5.1973. Metcalfe nimesi sen Ethernetiksi. Nimi perustuu vanhaan uskomukseen, jonka mukaan eetterin avulla voisi välittää sähkömagneettista säteilyä avaruuteen. (Jaakohuhta 2002, 11–12.)

2.2 Viitemallit

1980-luvulla tietoverkot alkoivat levitä liikeyrityksiin. Ensimmäiset yritysten tietoverkot eivät olleet keskenään yhteensopivia, koska ne oli rakennettu eri valmistajien laitteilla ja eri ohjelmistoja käyttäen. Tämä johtui verkkoyritysten kilpailusta. Jokainen yritys kehitti omaa tekniikkaansa. Tekniikka haluttiin yhdenmukaistaa ja siksi päätettiin luoda standardi, jonka mukaisesti valmistajien piti suunnitella tuotteensa. Laitteiden ja ohjelmistojen suunnittelua varten tarvittiin malli. (Jaakohuhta 2002, 13–18.)

Tietoliikennetekniikan opetuksessa lähdetään usein liikkeelle OSI-mallista (*Open Systems Interconnection*), jonka kehitti *International Organisation for Standardisation* (ISO). OSI-mallissa tiedonsiirto on jaettu seitsemään osa-alueeseen, eli kerrokseen, joista jokaisella on oma tehtävänsä. Jokaisella kerroksella toimii protokollia, jotka (fyysistä kerrosta lukuun ottamatta) lisäävät dataan kerroksissa käytettyjen protokollien *Header-* ja *Trailer-*tiedot (Cisco Press 1998, 20–21). *Header* eli otsikko lisätään datapaketin alkuun, ja se kertoo minkä protokollan dataa paketti sisältää. *Trailer* on kontrollitieto, jonka protokolla lisää datapaketin loppuun. (Cisco Systems 2002, 45.)

Nykypäivän käytetyin ja tunnetuin standardi on Yhdysvaltain puolustusministeriön kehittämä TCP/IP-malli. Mallin pohjalta suunnitellun verkon pitäisi kestää kaikki olosuhteet, myös ydinsota. Siinä on sama periaate kuin OSI-mallissa, mutta siinä sovellus-, esitys- ja istuntokerrokset on yhdistetty yhdeksi sovelluskerrokseksi ja siirtoyhteyskerros ja fyysinen kerros on yhdistetty verkkoyhteyskerrokseksi. Kuljetuskerros on molemmissa malleissa samanniminen. Verkkokerrosta vastaa Internet-kerros, jossa toimivat reititysprotokollat. Nimi Internet-kerros johtuu siitä, että Internet perustuu TCP/IP-malliin (Moy 2000, 4). Taulukossa 1 nähtävillä OSI- ja TCP/IP-mallit. (Cisco Systems 2002, 64–67.)

TAULUKKO 1. OSI-malli ja TCP/IP-malli (Cisco Systems 2002, 54)

7. APPLICATION = SOVELLUS	APPLICATION = SOVELLUS
6. PRESENTATION = ESITYS	
5. SESSION = ISTUNTO	
4. TRANSPORT = KULJETUS	TRANSPORT = KULJETUS
3. NETWORK = VERKKO	INTERNET
2. DATALINK = SIIRTOYHTEYS	NETWORK ACCESS = VERKKOYHTEYS
1. PHYSICAL = FYY SINEN	

Tiedonsiirto lähtee liikkeelle ylimmästä eli sovelluskerroksesta. Verkossa liikkuva data on lähtöisin sovelluskerrokselta. Siellä määritetään myös käyttöliittymä, esimerkiksi Internet-selain. Esityskerros määrittää datan esitystavan, kuten kirjoitusmerkkikoodi ASCII:n tai JPEG-kuvaformaatin. Esityskerros huolehtii myös datan mahdollisesta salauksesta ja salauksen purkamisesta (*encryption, decryption*). Istuntokerros huolehtii yhteyksien avaamisesta ja katkaisemisesta sekä tiedonsiirron ajoituksista. Nämä kolme ylintä kerrosta luovat ja muokkaavat datan binäärikoodiksi. TCP/IP-mallissa sovelluskerros hoitaa OSI-mallin kahden ylimmän kerroksen tehtävät ja kuljetuskerroksen TCP-protokolla huolehtii yhteyksien luomisesta. TCP/IP-mallissa istuntokerrosta ei siis käytetä. (Cisco Press 1998, 15.)

Kuljetuskerros huolehtii yhteydestä lähteen ja kohteen välillä. Kuljetuskerroksella data muokataan joko segmenteiksi tai datagrammeiksi. TCP-protokolla (*Transmission Control Protocol*) segmentoi datan ja UDP-protokolla (*User Datagram Protocol*) muokkaa datan datagrammeiksi (Cisco Press 1998, 8). Protokollien ero on siinä, että TCP on varmennettu ja UDP ei ole. TCP-protokollaa käytetään yleisemmin tiedostojen siirtämisessä. TCP-protokolla havaitsee tiedonsiirrossa mahdollisesti tapahtuvat virheet. Jos lähettävä tietokone ei saa vastaanottajalta tiedonsiirron jälkeen ACK- viestiä (*Acknowledgement*), jolla vastaanottaja ilmoittaa saaneensa koko tiedoston, lähettää TCP-protokolla tiedoston uudelleen. (Parziale ym. 2006, 150–151.)

UDP-protokollan etuna TCP-protokollaan verrattuna on keveys. UDP-protokolla ei takaa, että datapaketit saapuvat perille oikeassa järjestyksessä, tai että data pysyy kokonaisena ja muuttumattomana (Moy 2000, 8). UDP-protokollaa käytetään reaaliaikaiseen tiedonsiirtoon, kuten videokuvan tai äänen siirtoon, joissa pienet häiriöt eivät

haittaa. Jos esimerkiksi videokuvasta häviää yksi datagrammi siirron aikana, ei ihmisen silmä edes huomaa sitä videota katsottaessa. (Cisco Press 1998, 20–21.)

Reititys tapahtuu verkkokerroksella. Internet-protokolla (IP) muokkaa kuljetuskerrokselta tulleen TCP-segmentin tai UDP-datagrammin IP-paketiksi. IP-protokollan otsikotiedot sisältävät lähteen ja kohteen IP-osoitteet, joita kutsutaan myös loogisiksi osoitteiksi. Loogisista osoitteista kerron tarkemmin alaluvussa 3.1. (Moy 2000, 8.)

Seuraavaksi paketti siirtyy siirtoyhteyskerrokselle, missä siihen lisätään kehys (*Frame*). Kehyksen otsikko sisältää lähteen ja kohteen fyysiset osoitteet, eli MAC-osoitteet. MAC-osoite on 48-bittinen heksadesimaalikoodi, joka on poltettu verkkokorttiin valmistusvaiheessa. MAC-osoite on jokaisessa verkkokortissa ainutlaatuinen. Kehyksen otsikko-osan alussa on myös *start*-kenttä, joka kertoo kehyksen alusta. Lisäksi otsikko-osan lopussa on verkkokerroksen protokollaa ilmaiseva kenttä. Kehyksen lopussa oleva kontrollitieto-osa sisältää virheentarkistuksen ja kehyksen loppua merkitsevän *stop frame* -kentän. Kehyksellä varustettu datapaketti siirretään fyysiselle kerrokselle, joka muuntaa sen signaaliksi ja lähettää tietoverkon läpi tietokoneelta toiselle. (Cisco Systems 2002, 56–59.) TCP/IP-mallissa vastaavat tehtävät hoitaa verkkoyhteyskerros.

2.3 Osoitteiden jakaminen ja selvittäminen tietoverkossa

Jotta tietoliikenne olisi mahdollista, verkkolaitteiden pitää tietää mihin lähettää paketteja. Tätä varten reitittimillä ja tietokoneilla on oltava osoitteet. MAC-osoitteet ovat fyysisiä osoitteita ja IP-osoitteet loogisia osoitteita. IP-osoitteita käytetään reitityksessä. IP-osoitteiden selvittämiseen verkossa toimivat laitteet käyttävät ARP-protokollaa (*Address Resolution Protocol*). ARP selvittää mitä MAC-osoitetta IP-osoite vastaa. ARP lähettää viestejä, joilla se tiedustelee IP-paketin vastaanottajan MAC-osoitetta. Vastaanottaja on joko reititin, joka välittää pakettia eteenpäin, tai paketin kohteeksi määritelty laite. (Moy 2000, 10.)

Loogisen IP-osoitteen voi määrittää tietokoneeseen manuaalisesti tai sen voi saada palvelun tarjoajalta verkon kautta. Osoitteita jakava protokolla on DHCP (*Dynamic Host Configuration Protocol*). DHCP-palvelin, joka on joko erillinen palvelinkone tai reititin, jakaa IP-osoitteet, aliverkon peitteet, yhdyskäytävät ja DNS-palvelimien osoit-

teet lähiverkossa oleville tietokoneille. Käynnistyessään tietokone, joka on määritelty hakemaan IP-osoitteensa DHCP-palvelimelta, lähettää lähiverkkoon *DHCPDISCOVER*-viestin yleislähetystenä (*broadcast*). DHCP-palvelin vastaa pyyntöön tarjoamalla IP-osoitetta *DHCPOFFER*-viestillä. Jaettavissa oleva IP-osoitealue on palvelimella *DHCP-POOL*-nimisessä tietokannassa. Jos tietokone hyväksyy tarjotun IP-osoitteen, se vastaa palvelimelle *DHCPREQUEST*-viestillä, eli DHCP-pyyntöllä. Palvelin vastaa pyyntöön *DHCPACK*-viestillä, joka määrittää tietokoneelle sen tarvitsemat tiedot (IP-osoite, aliverkon peite, yhdyskäytävä ja DNS-palvelin). Annettu IP-osoite on tietokoneen käytössä joko pysyvästi tai määräajan, esimerkiksi kunnes tietokone sammutetaan. Verkon ylläpitäjä on voitu myös määrätä tietylle tietokoneelle tietty IP-osoitteen DHCP:n kautta. (Parziale ym. 2006, 130–135.)

Jokaisella Internetiä käyttävällä laitteella on oltava yksilöllinen julkinen IP-osoite, jotta reititys olisi mahdollista. Laitteiden määrän kasvaessa alettiin epäillä, että IPv4-osoitteet eivät riittäisi. Tekniikka nimeltä NAT (*Network Address Translation*) kehitettiin säästämään IPv4-osoitteita. NAT:n ansiosta monet Internetiin kytketyt tietokoneet, jotka eivät kuulu samaan lähiverkkoon, voivat käyttää samaa IP-osoitetta. Tämä on mahdollista siksi, että reititin, jonka kautta tietokone on kytketty Internetiin, muuntaa lähiverkon sisällä käytetyn lokaalin IP-osoitteen globaaliksi IP-osoitteeksi, joka näkyy lähiverkon ulkopuolella, Internetissä. NAT:n ansiosta lähiverkon sisällä voidaan käyttää osoitteita, jotka on määritelty yksityisiksi, eli niitä ei reititettäisi julkisessa verkossa. (Parziale ym. 2006, 89–94.)

Myös WWW-palvelimilla on IP-osoitteet. Koska ihminen on huono muistamaan pitkiä numerosarjoja, keksittiin tapa nimetä palvelimien osoitteet kirjaimilla. Käännös-palvelimia kutsutaan DNS-palvelimiksi (*Domain Name System*). DNS-palvelimia käytetään kääntämään verkossa olevien palvelimien Domain-nimiä IP-osoitteiksi ja päinvastoin. Kun Internetin käyttäjä kirjoittaa Internet-selaimen haluamansa sivuston nimen (esim. www.google.fi), lähtee kysely DNS-palvelimelle. Palvelin etsii tietokannastaan nimeä vastaavan IP-osoitteen ja lähettää sen selaimelle. (Parziale ym. 2006, 432.)

3 IP-OSOITTEET JA REITITYS

Reititys tarkoittaa reitittimen tai reitittimien välityksellä tapahtuvaa sähköisten datapakettien välittämistä tietoverkossa paketin lähettäjän ja vastaanottajan välillä (Moy 2000, 3). Paketteja lähettävät reitittimet, kytkimet, palvelimet ja tietokoneet. Lähetysmuotoja on kolme. Yhdelle laitteelle lähetys (*unicast*), kahdelle tai useammalle laitteelle lähetys, eli ryhmälähetys (*multicast*) ja kaikille lähiverkon laitteille lähetys, eli yleislähetys (*broadcast*). Kytkimet välittävät paketteja MAC-osoitteiden perusteella. Reitittimet käyttävät pakettien välittämiseen IP-osoitteita. (Moy 2000, 10.)

3.1 IP-osoitteet

IP-protokollasta on kaksi versiota, IPv4 ja IPv6. IPv4-osoitteet ovat 32-bittisiä, ja ne muodostuvat neljästä kahdeksanbittisestä luvusta, eli oktetista (esimerkiksi 192.168.0.0). Bittien merkityksen selitän myöhemmin tässä luvussa. IPv6-osoitteet ovat 128-bittisiä, ja ne muodostuvat kahdeksasta 16:n bitin heksadesimaalikoodista (esimerkiksi 4722:OC62:0:0:2:1298:OCC:A096), joka erotellaan kaksoispisteillä. IPv6-protokolla kehitettiin, kun huomattiin, että IPv4-osoitteet olivat vähenemässä. IPv6-osoitteita on vähitellen otettu käyttöön. (Moy 2000, 23.) Vuoden 2012 kesäkuussa muutamat palvelimet siirtyivät käyttämään IPv6-osoitteita.

IP-osoite koostuu kahdesta osasta, verkko-osasta ja päätelaiteosasta. Verkko-osa kertoo verkon osoitteen ja päätelaiteosa kertoo päätelaitteen, eli esimerkiksi tietokoneen osoitteen. Esimerkiksi IPv4-osoitteessa 192.168.10.2 aliverkon peitteellä 255.255.255.0 ensimmäiset kolme lukua, 192, 168 ja 10 ovat verkko-osaa ja viimeinen luku, 2, on päätelaiteosa. Aliverkon peite esitetään kirjallisuudessa usein myös prefiksinä IP-osoitteen yhteydessä, esimerkiksi 192.168.10.0/24. Esimerkin prefiksi /24 vastaa maskia 255.255.255.0. (Parziale ym. 2006, 68.)

Ennen vaihtuvan pituisen aliverkon peitteen (*Variable Length Subnet Mask*, VLSM) kehittämistä IPv4-osoitteet jaettiin IP-osoiteluokkiin A, B, C, D ja E. A-luokan osoitteet käyttivät prefiksiä /8, eli verkko-osaa oli IP-osoitteen ensimmäinen oktetti. Oktetin arvo oli välillä 0–127. Muut oktetit olivat päätelaiteosaa. B-luokan osoitteet käyttivät prefiksiä /16, eli verkko-osaa oli kaksi ensimmäistä oktettia. Ensimmäisen oktetin arvo oli välillä 128–191 ja toisen oktetin arvo välillä 0–255. C-luokan osoitteissa käy-

tettiin prefiksiä /24, eli verkko-osaa oli kolmea ensimmäistä oktettia. Ensimmäisen oktetin arvo oli välillä 192–223 ja kahden muun arvot välillä 0–255. D-luokan osoitteet varattiin *multicast*-lähetyksiin. Niissä ensimmäisen oktetin arvo oli välillä 224–239 ja muut oktetit välillä 0–255. E-luokan osoitteet varattiin myöhempää käyttöä varten. Niissä ensimmäisen oktetin arvo on välillä 240–255. (Moy 2000, 19–21.)

Luokaton reititys (*Classless Inter Domain Routing, CIDR*), käyttää vaihtuvan pituista aliverkon peitettä (VLSM). Aliverkon peitteen tarkoitus IP-osoitteen ohessa avautuu selkeämmin binääriluvuilla esitettynä. Luku 255 tarkoittaa, että kaikki oktetin bitit ovat ykkösiä (1111 1111). Arvo 255 on bittejä vastaavien lukujen yhteen laskettu summa. Bittien arvot ovat järjestyksessä vasemmalta oikealle 128, 64, 32, 16, 8, 4, 2 ja 1. Esimerkiksi luku 192 on binäärisenä 1100 0000 (128 + 64 = 192). IP-osoitteeseen liitetyn aliverkon peitteen binäärioktetin ykköset määräävät vastaavan IP-osoitteen binäärioktetin bittien muuttumattomuuden. Muuttumattomuudella tarkoitetaan, että bittejä ei voi muuttaa muuttamatta aliverkkoa. Taulukossa 2 on nähtävillä esimerkki.

TAULUKKO 2. Esimerkki aliverkon peitteen vaikutuksesta IP-osoitteeseen

1. oktetti	2. oktetti	3. oktetti	4. oktetti	IP/aliverkon peite
1100 0000	1010 1000	0000 1010	0000 0010	192.168.10.2
1111 1111	1111 1111	1111 1111	0000 0000	255.255.255.0

Taulukon 2 esimerkissä aliverkon peitteen 255.255.255.0 ykköset määräävät IP-osoitteen ensimmäisen, toisen ja kolmannen oktetin muuttumattomiksi. Aliverkon peitteen nollat ovat IP-osoitteessa muuttuvaa päätelaiteosaa, johon tässä esimerkissä kuuluu koko neljäs oktetti.

VLSM:n avulla IP-osoitteita voidaan jakaa minimissään neljän osoitteen aliverkkoihin käyttämällä maskia 255.255.255.252, joka vastaa prefiksiä /30. Esimerkiksi lisäämällä maski 255.255.255.252 osoitteeseen 192.168.1.0 saadaan neljän osoitteen aliverkko, jossa ensimmäinen osoite on verkon osoite (192.168.1.0) ja viimeinen on aliverkon *broadcast*-osoite (192.168.1.3), kuten kaikissa IP-osoitteissa. Toinen ja kolmas osoite (192.168.1.1 ja 192.168.1.2) ovat laitteiden käytössä. (Parziale ym. 2006, 74.)

Seuraavaksi esittelen miten IP-osoitteita määritetään reitittimen portteihin. Käytän esimerkkinä Ciscon reitittimiä. Ciscon reitittimiin tehdään asetuksia *global configuration* -tilassa. Jos reititin on vasta käynnistetty, ollaan *user exec* -tilassa ja täytyy mennä ensin *privileged exec* -tilaan kirjoittamalla reitittimen komentoriville:

```
enable
```

Privileged exec -tilassa voidaan esimerkiksi tutkia reitittimelle tehtyjä asetuksia *show*-komentoilla, joista reitityksen kannalta oleellisimpia esittelen alaluvussa 5.5. *Global configuration* -tilaan pääsee *privileged exec* -tilasta komennolla:

```
configure terminal
```

Tämän jälkeen voidaan määrittää IP-osoitteet haluttuihin portteihin. Esimerkiksi reitittimen FastEthernet 0/1 -porttiin määritetään IP-osoite komentoilla:

```
interface fa0/1
ip address 192.168.1.1 255.255.255.0
no shutdown
```

IP-osoitteen jälkeen tuleva *no shutdown* -käsky aktivoi portin toimimaan. (Cisco Systems 1996.) Käskyä käytetään kaikissa, paitsi virtuaalisessa *Loopback*-portissa. Ne menevät automaattisesti päälle. (InformIT 2012.) *Loopback*-portin voi määrittää esimerkiksi käskyllä:

```
interface lo0
ip address 200.200.200.1 255.255.255.0
```

Muita *interface*-käskyllä määritettäviä portteja ovat *GigabitEthernet*-portit ja *Serial*-portit. *GigabitEthernet*-portit ovat nopeampia kuin *FastEthernet*-portit, mutta niihin tehdään asetuksia samalla tavalla. Myös *Serial*-portteihin tehdään asetukset samalla tavalla, paitsi vanhemmissa reitittimissä niihin pitää määrittää lisäksi kellotaajuus (*clock rate*). *Serial*-portteja käytetään kahden reitittimen yhdistämiseen, jolloin kellotaajuus määritetään vain toiseen reitittimeen. (Cisco Systems 2012a.) *Serial*-portin määrittäminen tapahtuu esimerkiksi käskyillä:

```
interface serial0/1
ip address 192.168.1.0 255.255.255.252
clock rate 9600
no shutdown
```

Esimerkissä reitittimen *Serial*-portin 0/1 kellotaajuudeksi on asetettu 9600 bittiä sekunnissa. Kun IP-osoitteet on määritetty reitittimien portteihin, voidaan seuraavaksi määrittää reitit. Reitittämisestä ja reittien määrittämisestä kerron seuraavaksi.

3.2 Reititys

IP-osoitteet mahdollistavat reitityksen. Kaikkiin verkossa toimiviin laitteisiin on määritetty IP-osoitteet, joiden perusteella reitittimet reitittävät datapaketteja. Reittien määrittämiseen reitittimille on olemassa kaksi tapaa, staattinen ja dynaaminen. Staattinen reititys tarkoittaa, että jokaiselle verkon reitittimelle konfiguroidaan kaikki verkon sisällä olevat ja verkosta ulos menevät reitit käsin. Pienissä verkoissa tämä on toimiva vaihtoehto, mutta suurissa verkoissa erittäin epäkäytännöllistä. (Parziale ym. 2006, 175.)

Reittien määrittäminen staattisesti Ciscon reitittimissä tapahtuu *global configuration* -tilassa komennolla:

```
ip route [lähteen IP-osoite][lähteen aliverkon peite][kohteen IP-osoite]
```

Toinen vaihtoehto on käyttää kohdeosoitteen sijasta porttia, josta on siirtoyhteys kohdeosoitteeseen. Staattinen reititys on hyvä tapa määrittää verkosta ulos vievä oletusreitit. Tämä määritetään esimerkiksi komennolla:

```
ip route 0.0.0.0 0.0.0.0 200.100.100.2
```

Komennossa lähdeosoitetta vastaavat nollaosoitteet merkitsevät Internetiä ja osoite 200.100.100.2 merkitsee Internet-palveluntarjoajan osoitetta. Staattisia oletusreittejä käytetään usein niissäkin verkoissa, joissa käytetään reititysprotokollia. (Cisco Systems 1996.)

Dynaaminen reititys tarkoittaa reititysprotokollien käyttämistä. Reititysprotokollat kehitettiin helpottamaan tietoverkkojen laajentamista ja ylläpitoa. Reititysprotokollia käytettäessä riittää, että jokaiselle verkon reitittimelle aktivoidaan sama protokolla ja konfiguroidaan reitittimiin itseensä suoraan kytketyt verkot. Loput hoitaa reititysprotokolla. Kaikki saman itsenäisen järjestelmän (AS) reitittimet, joissa on sama protokolla, lähettävät toisilleen päivityksinä tiedot tietämistään reiteistä ja itseensä konfiguroiduista verkko-osoitteista. Saamiensa reititietojen perusteella reititin laskee reititysprotokollan algoritmilla nopeimmat reitit itsensä ja tietämiensä aliverkkojen välillä. Reititysprotokollat levittävät myös tietoja verkon muutoksista, kuten katkenneista reiteistä. (Parziale ym. 2006, 174.) Reititysprotokollien osalta esittelen vain OSPF-protokollaa koskevia komentoja alaluvussa 5.5.

Kaikille reititysprotokollille ja siirtoyhteyksille on määritetty hallinnollinen etäisyys, eli *Administrative Distance* (AD). Etäisyyksien oletusarvot ovat yksi peruste reitin valinnalle silloin, kun käytössä on useita reititysprotokollia. Se reitti, jolla on pienin AD-arvo, otetaan käyttöön. Taulukossa 3 esittelen reititysprotokollien ja siirtoyhteyksien hallinnolliset oletusarvot Ciscon reitittimien käyttöjärjestelmässä (Cisco IOS). Jos reitin oletusarvo on välillä 0 – 9 reitti on verkon sisäinen. Reittiä, jonka arvoksi laskeaan 255, ei oteta reititystauluun. (Cisco Press 1998, 305.)

TAULUKKO 3. Reititysprotokollien ja siirtoyhteyksien hallinnolliset arvot Ciscon verkkolaitteissa (Cisco Press 1998, 305)

Protokolla tai siirtoyhteys	AD
Suoraan kytketty	0
Staattinen reitti	1
EIGRP (koontireitti)	5
BGP (verkosta ulos)	20
EIGRP (verkon sisällä)	90
OSPF	110
RIP	120
EIGRP (muista protokollista opitut)	170
IS-IS	115
BGP (verkon sisällä)	200
Tuntematon	255

Kun IP-paketti saapuu reitittimeen, se tutkii ensin mihin osoitteeseen paketti on menossa. Seuraavaksi se tutkii reititystaulustaan tietääkö se reitin kohteeseen. Jos kohteeseen on useampi reitti, valikoi reititin pienimmällä AD-arvolla olevan protokollan tai siirtoyhteyden ja lähettää paketin eteenpäin sitä pitkin. Ciscon reitittimissä AD-arvot määräytyvät taulukon 3 mukaisesti.

4 REITITYSPROTOKOLLAT

Reititysprotokollia voidaan jakaa ryhmiin käyttötarkoituksen ja toimintaperiaatteen perusteella. Käyttötarkoituksen perusteella protokollat voidaan jakaa lähiverkon sisällä käytettäviin (*Interior Gateway Protocol*, IGP) ja lähiverkkojen yhdistämiseen käytettäviin (*Exterior Gateway Protocol*, EGP). Toimintaperiaatteen perusteella jaettuna protokollat ovat joko *Distance vector*, eli etäisyysvektori-protokollia, tai *Link State*, eli linkkitilaprotokollia. On myös olemassa kolmas luokka, ns. hybridiprotokollat, joissa yhdistyy etäisyysvektori- ja linkkitilaprotokollien ominaisuuksia. (Parziale ym. 2006, 173–174.) Hybridiprotokollista esimerkkinä on EIGRP, josta kerron alaluvussa 4.2. Taulukossa 4 on nähtävillä reititysprotokollien luokittelu.

TAULUKKO 4. Reititysprotokollien luokittelu

	Etäisyysvektori-protokollat	Linkkitilaprotokollat
IGP	RIPv1, RIPv2, RIPv6	OSPF
	IGRP	IS-IS
	EIGRP	
EGP	BGP	

Sekä etäisyysvektori-, että linkkitilaprotokollia käytettiin jo Internetin edeltäjän, ARPANETIN, aikana. Ensimmäisiä olivat etäisyysvektori-protokollat. Linkkitilaprotokollaan siirryttiin vuonna 1979, kun siirtoyhteyksien nopeudet alkoivat kasvaa. Etäisyysvektori-protokollat eivät osanneet ottaa huomioon siirtoyhteyksien nopeutta. (Tanenbaum 1996, 359.)

EGP-protokollista käsitellen tässä työssä vain BGP:tä. EGP-protokollia käyttävät yleensä Internet-palveluntarjoajat, jotka yhdistävät niillä lähiverkkoja (AS) Internetiin.

Ne yhdistävät myös palveluntarjoajien verkkoja toisiinsa. EGP-protokollissa verkossa käytettävien reitittimien määrä voi olla moninkertainen verrattuna IGP-protokoliin. (Moy 2000, 34.)

IGP-protokollista etäisyysvektori-protokollia ovat RIPv1, RIPv2, RIPng ja IGRP. Näistä RIPv1 ja IGRP ovat käytännössä jo historiaa, eikä niitä käytetä. RIP-protokolla tekee reitinvalinnan hyppyjen (*hop count*) perusteella. IGRP ottaa huomioon myös siirtoyhteyden nopeuden (Moy 2000, 298). Useimpien etäisyysvektori-protokollien reitinvalinta perustuu *Bellman–Ford* -algoritmiin. Algoritmia käyttävä reititin laskee kaikki mahdolliset reitit itsensä ja tietämiensä verkkojen ja aliverkkojen välillä erikseen. Valittuaan omat nopeimmat reittinsä reitittimet vertaavat reititystietojaan keskenään. Jos jokin reititin tietää nopeamman reitin, sen naapurit muokkaavat omia reititystaulujaan tiedon mukaisesti. (Moy 2000, 35–36.) Linkkitilaprotokollia ovat IS-IS ja OSPF. Niiden yhtäläisyyksistä ja eroista kerron IS-IS:n yhteydessä. Linkkitilaprotokollien toimintaa esittelen OSPF:n kautta.

Etäisyysvektori-protokollat ovat rakenteeltaan yksinkertaisempia kuin linkkitilaprotokollat ja kuluttavat vähemmän reitittimen muistia ja prosessointitehoa. Toisaalta etäisyysvektori-protokollat vievät enemmän kaistanleveyttä ajoitettujen päivitysten takia. Jotkut linkkitilaprotokollatkin (kuten OSPF ja IS-IS) lähettävät ajoitettuja viestejä, mutta eivät kuitenkaan koko reititystauluaan, kuten etäisyysvektori-protokollat. Etäisyysvektori-protokollat (RIP, IGRP) saavuttavat konvergenssin hitaammin, koska tiedot verkon muutoksista liikkuvat hitaammin. Konvergenssilla tarkoitetaan tässä tapauksessa sitä, että kaikilla lähiverkon reitittimillä on tiedossa reitit toistensa välillä ja verkosta ulos. (Ballew 1998, 124–128.) Seuraavaksi kerron lyhyesti nykyisin käytössä olevista reititysprotokollista. RIP-protokollasta kerron hieman tarkemmin, koska siinä esiintyneet ongelmat antoivat aikanaan aiheen uusien reititysprotokollien, kuten OSPF:n suunnittelulle.

4.1 RIP

Lyhenne RIP tulee sanoista *Routing Information Protocol*. RIP-protokollasta on kolme versiota, RIPv1, RIPv2 ja uusin RIP *new generation* eli RIPng. Versiot 1 ja 2 eroavat toisistaan niin, että versio 1 käyttää luokallista reititystä ja versio 2 luokatonta

reititystä. (Parziale ym. 2006, 189.) Uusin versio, RIPng on kehitetty IPv6 osoitteita varten (Malkin ja Minnear 1997, 3).

Kaikissa RIP-protokollan versiossa verkon koko rajoittuu 16 reitittimeen. Tämä johtuu siitä, että RIP käyttää metriikkana, eli reitinvalinnan perusteena, *hop count* -arvoa ja 15 on *hop countin* maksimiarvo (Cisco Systems 2002, 445). *Hop count* tarkoittaa siis reitittimien määrää lähetettävän tietopaketin lähteen ja kohteen välissä. Jokainen siirtymä reitittimeltä toiselle on yksi hyppy, eli *hop*. Jos tietopaketti pitäisi lähettää esimerkiksi 16 hypyn päähän, RIP ei saisi pakettia perille. Tämä johtuu siitä, että 16 hyppyä olisi RIP-protokollan mielestä saavuttamaton. (Parziale ym. 2006, 189.)

Yksi RIP-protokollan ongelma on *Count to Infinity*, eli lasku äärettömään. Tämä ongelma aiheuttaa hitaan konvergenssin vikatilanteissa. Jos jokin verkon reitittimistä kaatuu tai jokin verkkoyhteys katkeaa, RIP reagoi asiaan hitaasti. Muut verkon reitittimet eivät merkitse reittiä saavuttamattomaksi heti, vaan ne kasvattavat reitin *hop count* -arvoa jokaisessa ajoitetussa päivityksessä sen mukaan, minkä tiedon reitistä ne saavat toisiltaan. Kaikki verkon reitittimet tietävät reitin kaatuneelle reitittimelle ja päivitysten mukana ilmoittavat tiedostaan naapureilleen. *Hop count* nousee lopulta kaikilla arvoon 16, mutta tämä tapahtuu hitaasti. Tällä tavalla muodostuu *routing-loop*, eli reitityssilmukka, jossa kaatuneelle reitittimelle suunnattu viesti kulkee edestakaisin reitittimeltä toiselle, kunnes viestin TTL-laskuri, eli *time-to-live*-laskuri saavuttaa raja-arvonsa ja viesti hävitetään. Kun kaatunut reititin saadaan lopulta taas toimintaan, tieto reitin toimivuudesta lähtee leviämään yhtä hitaasti, kuin tieto reitin saavuttamattomuudesta. (Tanenbaum 1996, 357-359.) Kuvassa 3 on esimerkki katkenneesta reitistä ja taulukossa 5 reitittimien lasku äärettömään.



KUVA 3. Katkennut reitti

Kuvassa 3 neljä reitintä on yhdistetty toisiinsa *Point-to-Point*-yhteyksillä ja niihin on konfiguroitu RIP. Verkko toimii, kunnes yhteys reitittimien A ja B välillä katkeaa. Ilman reititysprotokollia reitittimet C ja D eivät tietäisi reitin katkeamisesta. RIP:n

ansiosta tieto reitin katkeamisesta leviää verkon sisällä. Taulukossa 5 nähdään miten reitittimien *hop count* -arvo kasvaa vaiheittain reittipäivityksiä vaihdettaessa.

TAULUKKO 5. Lasku äärettömään

Reititin A	Reititin B	Reititin C	Reititin D
0	1	2	3
0	2	3	4
0	3	4	5
0
0	16	16	16

Taulukossa 5 nähdään kuvan 3 tilanteesta seuraava lasku äärettömään. Reititin A:n ja reititin B:n välinen yhteys on katkennut. Reititin B saa reittipäivityksen reitittimeltä C, että se tietää toisen reitin reitittimelle A. Tämän tiedon reititin C on saanut katkeamisilmoituksen jälkeen reitittimeltä D, joka on saanut reititiedon aikaisemmin C:ltä. Eli vaikka yhdelläkään reitittimellä ei ole yhteyttä A:lle, ne luulevat että niillä on. Jokaisessa ajoitetussa päivityksessä ne kasvattavat *hop count* -arvoa yhdellä, kunnes saavuttavat arvon 16.

Split Horizon -tekniikka kehitettiin estämään värien reititystietojen leviäminen verkossa ja estämään reitityskierteen muodostumista. *Split Horizon* toimii niin, että reititin ei lähetä tietoa reitistä takaisin sille reitittimelle, jolta reitin oppii. Näin väärä reititystieto ei periaatteessa pääse leviämään. Valitettavasti tämäkin ratkaisu ei ole immuuni virheille. Vaikka tekniikka onkin teoriassa toimiva, käytännössä toimivuus riippuu verkon topologiasta. (Tanenbaum 1996, 358–359.)

Poison reverse -tekniikka kehitettiin nopeuttamaan äärettömään laskemista. Jos reitti huomataan katkenneeksi, reittiin suoraan kytketyt reitittimet nostavat reitin *hop count* -arvon suoraan saavuttamattomaksi. Yhdessä *Split Horizon* -tekniikan kanssa tämä ehkäisee reitityskierteiden muodostumista. (Cisco Press 1998, 103.)

4.2 EIGRP

EIGRP (*Enhanced Interior Gateway Routing Protocol*) on hybridiprotokolla. Siinä yhdistyy sekä etäisyysvektori-protokollan että linkkitilaprotokollan ominaisuuksia. Se

4.3 IS-IS

IS-IS (*Intermediate System-to-Intermediate System*) on linkkitilaprotokolla, kuten OSPF. IS-IS ja OSPF ovat kahden eri organisaation luomuksia. OSPF-protokollan standardoi *Internet Engineering Task Force* (IETF) ja IS-IS-protokollan *International Organisation for Standardisation* (ISO). Kumpikin protokolla perustuu Edsger W. Dijkstran kehittämään *Shortest Path* -algoritmiin, joka tunnetaan nimellä Dijkstra-algoritmi. (Moy 2000, 300–301.) Hän kehitti algoritmin vuonna 1956 ARMAC -nimisen tietokoneen toimintaa esittelemään (Thomas 2003, 64).

IS-IS-protokollan alkuperäiset versiot kehitettiin OSI-mallin mukaisiin verkkoihin. TCP/IP-mallin mukaisten verkkojen yleistyessä IS-IS-protokollasta muokattiin yhteensopiva. Yhteensopivaa versiota kutsutaan nimillä Dual IS-IS ja Integrated IS-IS. (Moy 2000, 300.)

IS-IS- ja OSPF-protokollissa käytetään erilaisia termejä samoista asioista, mikä johtuu eri suunnitteluorganisaatioiden nimeämistavoista. ISO käyttää reitittimistä nimitystä *Intermediate System* eli protokollan nimi IS-IS tarkoittaa reitittimien välistä yhteydenpitoa. IS-IS-protokollan käyttämistä datapaketeista käytetään nimitystä *Protocol Data Unit* (PDU). Esimerkiksi reititystietokannan tieto on *Link State PDU* (LSP). (Moy 2000, 301.)

IS-IS-protokollan ja OSPF-protokollan välisiä yhtenäisyyksiä ovat linkkitilaprotokollille ominainen tietokantamainen reititystaulu ja mahdollisuus jakaa lähiverkko alueisiin. IS-IS-protokollan tapauksessa alueisiin jakaminen tarkoittaa lähiverkon organisoitua kaksitasoiseksi hierarkiaksi. Alueen sisällä tapahtuvaa reititystä kutsutaan tason 1 reitittämiseksi (*Level 1*) ja alueiden välistä reititystä tason 2 reititykseksi (*Level 2*). Tasolla 1 toimivat reitittimet huolehtivat vain alueen sisäisestä reitityksestä ja tasolla 2 toimivat reitittimet hoitavat reitityksen alueiden välillä ja lähiverkosta ulos. (Oran 1990, 1.)

IS-IS-protokolla käyttää myös, OSPF- ja EIGRP-protokollien tapaan, *Hello*-protokollaa naapureidensa selvittämiseen ja siirtoyhteyksien ylläpitoon. IS-IS-protokollassa Hello-protokolla tunnetaan nimellä *IS-IS Hello* (IIH) ja sen lähettämät viestit nimellä IIH PDU. (Moy 2000, 301.)

4.4 BGP

BGP, eli *Border Gateway Protocol*, kuuluu EGP-protokolliin. Sitä käyttävät yleensä Internetin palveluntarjoajat. BGP reitittää Internetin palveluntarjoajien välistä liikennettä. Yksityisissä lähiverkoissa BGP on harvinainen. Se saattaa olla aktivoituna reititimeen, joka yhdistää lähiverkon Internetiin. (Moy 2000, 284–286.)

BGP on etäisyysvektori-protokolla. Se tuntee naapurinsa vertaisreitittiminä (*Peer*). BGP ei valikoi vertaisreitittämiä automaattisesti, vaan verkon ylläpitäjä määrää ne. Muodostaessaan konvergenssia kaksi vertaisreitittäjä lähettää toisilleen aluksi *Open*-viestejä. Niiden jälkeen ne lähettävät toisilleen reititystaulunsa *Update*-viestien muodossa. Erotten muista etäisyysvektori-protokollista, BGP-protokolla lähettää reititystaulunsa naapurilleen vain kerran. Vaihdon jälkeen BGP lähettää ainoastaan reititystauluissa tapahtuneet muutokset, jotka lähetetään *Update*-viestien muodossa. Jos muutoksia tapahtuu vain harvoin, niin vertaisreitittimet lähettävät ajoittain toisilleen *Keepalive*-viestejä pitääkseen yllä yhteyttä. Yhteyden katkaisemisesta ilmoitetaan *Notification*-viestillä. Yhteyden katkaisemisen jälkeen vertaisreitittimet poistavat reititystaulustaan toisiltaan oppimansa reitit. (Moy 2000, 286–287.)

Toinen eroavaisuus BGP:n ja muiden etäisyysvektori-protokollien välillä on se, että BGP, kuten EIGRP, osaa valita aina reitin jossa ei esiinny reitityssilmukoita. Tämä johtuu siitä, että ilmoittaessaan tuntemastaan verkosta BGP kertoo myös täydellisen reitin verkkoon. Reititietoon sisältyy reitin varrella olevien itsenäisten lähiverkkojen (AS) määrä. Näiden lähiverkkojen (AS) määrän perusteella BGP päättää mitkä reitit se hyväksyy reititystauluunsa. Se reitti, jonka varrella on vähiten itsenäisiä lähiverkkoja (AS), pääsee reititystauluun. Toinen peruste reitin valinnalle on itsenäisten lähiverkkojen (AS) arvot. Verkon ylläpitäjä voi määrätä lähiverkoille (AS) arvot. Pienimmän arvon omaava lähiverkko (AS) on ensisijainen. (Moy 2000, 286–288.)

BGP:llä on reititystaulunsa lisäksi tietokanta, nimeltä RIB-In (*Routing Information Base-Inbound*). Tähän tietokantaan BGP tallentaa vaihtoehtoiset reitit. Jos jokin ensisijaisista reiteistä katkeaa, BGP etsii tietokannastaan parhaan vaihtoehtoisen reitin, jota se käyttää kunnes ensisijainen reitti palaa toimintaan. (Moy 2000, 286.)

5 OSPF-PROTOKOLLA

OSPF-protokollan version 1 standardi valmistui vuonna 1990. Sen oli kehittänyt *Internet Engineering Task Forcen* kokoama ryhmä. Standardin kehittäminen oli alkanut kolme vuotta aikaisemmin 1987. Versio 1 oli testiversio, jota kehitettiin. Versio 2 (rfc 1247) valmistui vuonna 1991 ja se oli ensimmäinen yleiseen käyttöön tullut OSPF:n versio. Versiota 2 on sittemmin kehitetty eteenpäin vuosien varrella. (Moy 2000, 44.) Siitä on esimerkiksi muokattu versio 3, mikä toimii IPv6-osoitteilla (Moy 2000, 69).

OSPF-protokollalle asetettiin suunnitteluvaiheessa vaatimuksia. Suunnittelun aikoihin, 1980-luvun lopulla, RIP oli käytetyin reititysprotokolla lähiverkoissa. Lähiverkkojen koon kasvaessa RIP:n ongelmat alkoivat korostua. OSPF-protokollasta haluttiin tehdä tehokkaampi kuin RIP:sta. OSPF:n haluttiin kuormittavan vähemmän verkkoa ja säästävän nopeammin konvergenssin. OSPF-protokollaan haluttiin myös parempi metriikka, kuin RIP-protokollaan. RIP ei huomioi reittivaihtoehtojen nopeutta, eikä reiteille ole mahdollista määrätä arvoja (*cost*). Nämä ominaisuudet haluttiin OSPF-protokollaan. (Moy 2000, 45.)

Uutena ominaisuutena silloisiin protokolliin verrattuna, OSPF:n haluttiin pystyvän hyödyntämään IP-protokollan datapakettiin sisältyvää palvelun tyyppi- eli *Type of Service*-ominaisuutta (TOS). Yksikään silloinen protokolla ei vielä osannut hyödyntää sitä. *Type of Service* -ominaisuudessa on viisi vaihtoehtoa: *normal service*, *minimize monetary cost*, *maximize reliability*, *maximize throughput* ja *minimize delay*. Näiden vaihtoehtojen avulla voidaan vaikuttaa IP-paketin käsittely- ja reititystapaan. (Moy 2000, 46.)

OSPF-protokollaan haluttiin lisäksi kuorman tasaus ominaisuus. Lähetettävät paketit piti siis voida siirtää kohteeseen useampaa kuin yhtä reittiä pitkin, jos vain mahdollista. Kuorman tasauksella haluttiin nopeuttaa tiedonsiirtoa ja keventää verkon kuormitusta. (Tanenbaum 1996, 424.)

5.1 Linkkitilatietokanta ja LSA

OSPF käyttää muiden linkkitilaprotokollien tapaan linkkitilatietokantaa reititystaulunsa lisäksi. Jokaisella OSPF-protokollaa käyttävällä reitittimellä on muistissaan linkkitilatietokanta, joka kattaa lähiverkon tai sen alueen (*area*, katso tarkemmin alueita käsittelevä alaluku 5.4), johon reititin kuuluu (Moy 1997, 21). Tietokantaan tallentuvat kaikki verkon tai alueen OSPF-protokollaa käyttävät reitittimet ja niiden väliset reitit. (Moy 2000, 83.)

Tietokannan perusteella reititin laskee itselleen oman reititystaulunsa. Reititystaulussa ovat ensisijaiset reitit joita reititin käyttää lähiverkossa. Algoritmia, jota OSPF käyttää reittien laskemiseen, kutsutaan Dijkstran lyhimmän reitin algoritmiksi (*Shortest Path First*, SPF). (Moy 1997, 4–5.) SPF-algoritmin pohjalta tehty C-kielinen ohjelma on liitteessä 1.

Linkkitilatietokannan reititystietoja kutsutaan nimellä *Link State Advertisement* (LSA). Kun OSPF aktivoidaan reitittimeen, se lähettää aluksi *Hello*-paketteja, joilla se tutustuu suoraan kytkettyihin naapureihinsa. Kun yhteys on saavutettu, se alkaa lähettää LSA-tietoja naapureilleen. Aluksi se lähettää LSA-tiedoista vain otsikot, joista esimerkki kuvassa 4. Otsikot lähetetään *Database Description* -nimisissä protokollapaketeissa, joista kerron tarkemmin alaluvussa 5.2. Naapuri päättää otsikkojen perusteella mistä LSA-tiedoista se on kiinnostunut. (Moy 2000, 89.)

LS Age	Options	LS type
Link State ID		
Advertising Router		
LS Sequence Number		
LS Checksum	Lenght	

KUVA 4. LSA-otsikko (Moy 2000, 74)

LSA-otsikon *LS Age* -kenttä kertoo LSA:n iän sekunteina. Oletusarvona LSA:t voidaan päivittää aikaisintaan viiden sekunnin välein. Itse lähettämänsä LSA:t reitittimen pitää päivittää vähintään 30 minuutin välein. Jos LSA:ta ei ole päivitetty tuntiin, se

poistetaan tietokannasta. *Options*-kenttä kertoo pitääkö LSA huomioida eri tavalla esimerkiksi reittiä laskettaessa. (Moy 2000, 79–80.)

LS type -kenttä luokittelee LSA:n toiminnan mukaisesti. Tyyppi 1 tarkoittaa *router-LSA*:ta, mitä käytetään kahden reitittimen välisissä *Point-to-Point*-yhteyksissä. Tyyppi 2 tarkoittaa *network-LSA*:ta, mitä käytetään *multiaccess*-verkoissa. Tyyppiä 3 (*network-summary-LSA*) ja 4 (*ASBR-summary-LSA*) käytetään hierarkkisessa reitityksessä jakamaan reititystietoja OSPF-alueiden välillä (katso alueita käsittelevä alaluku 5.4) ja tyyppiä 5 (*AS-external-LSA*) käytetään levittämään alueiden ulkopuolisia reititietoja. Tyyppiä 6 (*group-membership-LSA*) käytetään *multicast*-reitityksessä. Tyyppiä 7 käytetään NSSA-alueiden (*Not-So-Stubby Area*) yhteydessä ja tyyppiä 8 käytetään välittämään BGP-protokollan reititystietoja. (Moy 2000, 74–75.)

Link State ID -kentän tiedon avulla reititin erottaa lähettämänsä saman LS-tyypin LSA:t toisistaan. *Advertising Router* -kenttä kertoo miltä reitittimeltä LSA on lähtöisin. Tieto on sama kuin *Router ID*, mikä yleensä on suurin reitittimeen määritetty IP-osoite, ellei arvoa ole erikseen määrätty. *LS Sequence number* -kenttä kertoo LSA-tietojen järjestyksen. (Moy 2000, 75–76.)

LS Checksum -kenttä ilmaisee LSA:n muuttumattomuuden siirron aikana (Moy 2000, 78). Jos vastaanotetun LSA:n *Checksum*-arvo ei vastaa lähettäjän antamaa arvoa, LSA:ta ei merkitä tietokantaan. *Length*-kenttä ilmaisee LSA:n koon tavuina (Moy 2000, 80).

Otsikkoa seuraava LSA sisältää tiedon reitittimen tuntemista verkko-osoitteista ja reiteistä. LSA kertoo myös reiteille lasketut arvot (*cost*). (Moy 2000, 82.) LSA-tiedot kulkevat OSPF:n protokollapaketeissa, joista kerron seuraavaksi.

5.2 OSPF protokollapaketit

OSPF-protokolla käyttää reitittimien väliseen kommunikointiin protokollapaketteja. OSPF:n protokollapaketeilla on oma IP-protokollanumero (89), millä reitittimet erottavat ne muista datapaketeista. Protokollapaketteja on viittä tyyppiä, jotka ovat luetteloituna taulukossa 6. Pakettityypit ilmaistaan numerolla protokollapaketin OSPF-otsikossa, joka nähtävillä kuvassa 5. (Moy 2000, 85.)

TAULUKKO 6. Protokollapaketit (Thomas 2003, 122)

Numero	Pakettityyppi
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Ack

Lyhyesti kerrottuna *Hello*-paketteja käytetään naapureihin tutustumiseen, *Database Description* -paketteja käytetään linkkitilietokannan LSA-otsikoiden jakamiseen, *Link State Request* -paketeilla pyydetään reittitietoja, *Link State Update* -paketeilla levitetään reittitietoja ja *Link State Ack* -paketeilla ilmoitetaan paketit vastaanotetuiksi. Pakettityypit 2–4 muodostuvat LSA-otsikosta ja tiedoista. *Hello*- ja *Database Description* -pakettien rakenteen esitän tarkemmin kuvissa 6 ja 7 tässä luvussa.

OSPF version	Packet type number	Packet length
RouterID		
AreaID		
Checksum	Authentication type	
Authentication		
Authentication		

KUVA 5. OSPF-otsikko (Thomas 2003, 122)

Jokainen OSPF:n protokollapaketti alkaa OSPF-otsikolla. *OSPF version* -kenttä kertoo mikä OSPF:n versio on kyseessä. *Packet type* -kenttä ilmaisee numerolla mikä protokollapaketti on kyseessä. *Packet length* -kenttä kertoo paketin koon. *Router ID* kertoo suurimman reitittimelle konfiguroidun IP-osoitteen, ellei arvoa ole erikseen määritelty. *Area ID* kertoo mihin alueeseen (katso alueista kertova alaluku 5.4) reititin kuuluu. *Checksum*-kenttä ilmaisee onko viesti saapunut perille muuttumattomana. *Authentication type* -kenttä kertoo paketissa mahdollisesti käytetyn salauksen tyyppin ja seuraavat *authentication*-kentät sisältävät salausavaimet. (Moy 2000, 85–86.)

Otsikkoa seuraa protokollapaketti, joista tyyppi 1 on *Hello*-paketti. *Hello*-paketteja käytetään naapurireitittimien välisten yhteyksien luomiseen ja ylläpitoon. Kuvassa 6 *Hello*-paketin rakenne. *Hello*-paketteja käytetään myös reitittimien arvojärjestyksen selvittämiseen (katso alaluku 5.3). (Moy 2000, 85–87.)

Network mask		
HelloInterval	Options	Router priority
RouterDeadInterval		
Designated Router		
Backup Designated Router		
Neighbor		

KUVA 6. Hello-paketti (Moy 2000, 88)

Hello-paketti alkaa *Network mask* -kentällä, joka kertoo aliverkonpeitteen siltä reitittimen portilta, josta viesti lähti. *HelloInterval*-kenttä kertoo kuinka usein *Hello*-paketteja lähetetään. Oletuksena *Hello*-paketteja lähetetään 10 sekunnin välein. *Options*-kenttä ilmaisee mahdolliset erityisominaisuudet. *Router priority* -kenttä kertoo reitittimen lähettävän portin prioriteetin. *RouterDeadInterval*-kenttä kertoo kuinka kauan *Hello*-paketteja lähetetään, ennen kuin naapurireititin todetaan kaatuneeksi. Oletusarvo on neljä *Hello*-pakettia eli 40 sekuntia. *Designated Router*- ja *Backup Designated Router* -kentät kertovat DR:n ja BDR:n IP-osoitteet, jos ne on jo valittu (katso alaluku 5.3). *Neighbor*-kenttä kertoo kaikkien niiden verkon/alueen reitittimien ID:t, jotka ovat lähettäneet *Hello*-paketin. (Thomas 2003, 126.)

Tyyppi 2 protokollapaketti on *Database Description* -paketti. Niitä käytetään linkkitietokannan rakentamiseen ja synkronointiin. Paketit sisältävät LSA-otsikon, joilla reitittimet kertovat naapureilleen tietämistään verkoista. *Database Description* -paketin rakenne on esitettyä kuvassa 7. (Thomas 2003, 122.)

Interface MTU	Options	0	0	0	0	0	I	M	MS
Database Description Sequence Number									
LSA-otsikko									

KUVA 7. Database Description-paketti (Thomas 2003, 121)

Interface MTU -kenttä kertoo mikä on suurin sallittu paketin koko tavuina ko. reitittimen portissa. *Options*-kenttä on sama kuin *Hello*-paketissa. I-bitti kertoo onko paketti järjestyksessä ensimmäinen, M-bitti kertoo kuinka monta pakettia kulkee siirron aikana ja MS-bitti kertoo onko lähetävä reititin *master* vai *slave*. *Database Description Sequence Number* kertoo pakettien määrän tietokannassa. Paketin lopussa on LSA-otsikoita. (Moy 1997, 168.)

Protokollapakettien tyypit 3 ja 4 eli *Link State Request* ja *Link State Update* koostuvat LSA-otsikosta tutuista *LS type*-, *Link State ID*- ja *Advertising Router* -kentistä. Tyypin 3 paketeilla reititin pyytää reittitietoja reiteistä, mitä sille on mainostettu *Database Description* -paketeilla. Tyypin 4 paketeilla lähetetään ja päivitetään pyydettyjä reittitietoja. Tyypin 5 protokollapaketti on *Link State Ack* -paketti. Sillä ilmoitetaan lähettäjälle, että LSA-paketti on tullut perille ja se on ymmärretty.

5.3 OSPF multiaccess-verkoissa

Multiaccess tarkoittaa verkkoa, jossa enemmän kuin kaksi reititintä on yhdistettynä toisiinsa samassa aliverkossa esimerkiksi kytkimen kautta. Yhdistetyt reitittimet muodostavat konvergenssin, missä ne synkronoivat tietokantansa. Kun tietokantojen synkronointi on valmis, ovat naapurireitittimet muodostaneet tilan nimeltä *adjacency*. *Adjacency* muodostuu myös kahden reitittimen välisissä *Poin-to-Point*-yhteyksissä (Halabi 1996, 21). *Multiaccess*-verkoissa *adjacency*-prosessiin kuuluu, että OSPF-protokolla valikoi yhden reitittimistä hallitsevaksi reitittimeksi (*Designated Router*, DR), mikä muodostaa *adjacency*n muiden naapureiden kanssa. Tämä yksinkertaistaa verkon toimintaa. Koska DR voi lakata toimimasta, valitaan myös varalla oleva hallitseva reititin (*Backup Designated Router*, BDR). Jäljelle jäävät reitittimet tunnetaan nimellä DROther. (Thomas 2003, 83–85.)

Reitittimien arvojärjestys (DR, BDR, DROther) selvitetään *Hello*-paketeilla. Jos reitittimet liitetään *multiaccess*-verkkoon yksi kerrallaan, tulee ensimmäisenä verkkoon liitetystä reitittimestä DR ja toisesta BDR. Jos reitittimet liitetään verkkoon samanaikaisesti, se reititin jolla on suurin IP-osoitteen arvo valitaan DR-reitittimeksi. Seuraavaksi suurimman IP-osoitteen omaava on BDR ja loput ovat DROther-reitittimiä. Näin tapahtuu oletuksena jos reitittimien prioriteettiarvoja ei ole muutettu. (Moy 2000, 107.)

Verkon ylläpitäjä voi myös määrätä reitittimien arvojärjestyksen manuaalisesti. Ensimmäinen vaihtoehto on muuttaa DR:ksi halutun reitittimen *multiaccess*-verkkoon yhdistävän siirtoyhteyden prioriteetti suuremmaksi kuin muilla reitittimillä. Prioriteetin oletusarvo on 1. Jos prioriteetiksi määrätään nolla, reitittimestä ei voi tulla DR eikä BDR. (Halabi 1996, 17.) Toinen vaihtoehto on asettaa DR:ksi halutulle reitittimelle mahdollisimman suuri IP-osoitteen arvo. Verkossa käytettävistä osoitteista riippumatta tämä voidaan tehdä esimerkiksi määrittämällä reitittimelle erikseen *Router ID* tai konfiguroimalla reitittimeen *Loopback*-portti. (Thomas 2003, 80–85.)

DR:n tehtävä on pitää muiden samassa *multiaccess*-verkossa olevien OSPF-protokollaa käyttävien reitittimien linkkitilietokannat ajan tasalla. Jos DR-reititin kaatuu, tulee BDR-reitittimestä uusi DR ja DROther-reitittimien keskuudesta valitaan uusi BDR. Kun alkuperäinen DR palaa toimintaan, siitä ei tule uudestaan DR-reititintä, vaan DROther. DROther valitaan DR:ksi jos sekä DR että BDR lakkaavat toimimasta. (Moy 2000, 106–107.)

5.4 Alueet

OSPF tukee hierarkkista reititystä. Tämä tarkoittaa, että lähiverkko (AS) voidaan jakaa alueisiin (*area*). Esimerkiksi rakennuksen jokainen kerros voidaan valita omaksi alueekseen. OSPF-protokollassa hierarkia on kaksitasoista. Alueita yhdistää toisiinsa ja Internetiin *Backbone*-alue, eli runko. Backbone-alue merkitään nolla-alueeksi (*area 0*). Muut alueet verkon ylläpitäjä voi numeroida haluamallaan tavalla. (Thomas 2003, 58.)

Backbone-alueeseen liittyvistä alueista pienempiä kutsutaan *Stub*-alueiksi ja NSSA-alueiksi. Nimitys *Stub*-alue tulee siitä, että alue liittyy *Backbone*-alueeseen vain yhden

siirtoyhteyden kautta (Thomas 2003, 75). *Stub*-alueiden sisällä voidaan käyttää vähäisemmällä muistin määrällä ja laskentateholla varustettuja reitittimiä (Moy 2000, 139).

Stub-alueiden sisällä olevat reitittimet eivät tiedä suoria reittejä alueen ulkopuolella oleville laitteille. Kaikki *Stub*-alueelta ulos lähtevät paketit kulkevat alueiden rajalla olevan reitittimen (*Area Border Router*, ABR) kautta. ABR ilmoittaa *Stub*-alueen reitittimille tietävänsä oletusreitit verkosta ulos. ABR myös estää ulkoisten reitittitietojen leviämisen *Stub*-alueen reitittimille. (Thomas 2003, 75.)

NSSA-alueet ovat *Stub*-alueiden laajempia versioita. Molemmissa käytetään vähäisemmällä resursseilla varustettuja reitittimiä. Erona *Stub*-alueisiin NSSA-alueet ovat joustavampia. NSSA-alueilla sallitaan joitakin alueen ulkopuolelta tulevia reittipäivityksiä. (Moy 2000, 143.)

Hierarkkinen reititys keventää sekä verkon että reitittimien kuormitusta. Koska jokainen alue on oma *broadcast*-alueensa, protokollapaketit liikkuvat vain alueen sisällä, eivätkä leviä muille alueille. Tämä keventää verkon kuormitusta, varsinkin suurten verkkojen kohdalla. Reitittimien laskentatehon ja muistin kuormitus kevenee, koska reitittimien ei tarvitse tietää kuin oman alueensa topologia ja laskea reitit alueen sisällä ja alueelta ulos. (Moy 2000, 119.)

5.5 Asetukset

Tässä luvussa esittelen muutamia OSPF:n kannalta oleellisia komentoja, mitä käytetään Ciscon reitittimiä konfiguroitaessa. OSPF aktivoidaan reitittimeen *global configuration* -tilassa käskyllä:

```
router ospf 1
```

Esimerkissä numero 1 tarkoittaa reititysprosessin numeroa. Prosessinumerolla voidaan erotella eri reititysprosesseja toisistaan reitittimellä (Halabi 1996, 10). Monien eri prosessien käyttäminen reitittimellä kuluttaa reitittimen resursseja, joten sitä kannattaa välttää. Prosessinumero valitaan väliltä 1-65535 (Thomas 2003, 271).

Seuraavaksi määritetään verkko-osoitteet esimerkiksi käskyllä:

```
network 192.168.10.0 0.0.0.255 area 0
```

Network-sanan jälkeen tulee siis ensin verkon IP-osoite, sitten *wildcard*-maski ja lopuksi *area*-sanon jälkeen alueen numero. *Wildcard*-maski on *subnet*-maskin käänteinen muoto, eli esimerkiksi 255.255.255.0 kääntyy muotoon 0.0.0.255.

Alueiden rajalla oleville reitittimille (ABR) verkot voidaan konfiguroida myös suoraan koontireittinä. Jos alueeseen 1 esimerkiksi kuuluisivat verkot 192.168.1.0/24–192.168.3.0/24, niin nämä merkitään koontireitiksi komennolla:

```
area 1 range 192.168.0.0 255.255.252.0
```

Tämä asetus tehdään siis *global configuration* -tilassa, *router ospf 1* -komennon jälkeen. Kaikki alueen 1 sisällä olevat IP-osoitteet sisältyvät tähän koontireittiin ja tätä koontireittiä rajalla oleva reititin mainostaa muulle verkolle. Reititin osaa muodostaa koontireitin myös automaattisesti saamiensa reitityspäivitysten perusteella, mutta näin verkon ylläpitäjä voi konfiguroida alueen kaikki aliverkot kerralla. Jos alueeseen 1 kuuluisi jokin muu IP-osoitealue, joka ei sopisi koontireittiin, se ilmoitettaisiin erikseen *network*-komennolla, mutta vain jos se on suoraan kytketty alueiden rajalla olevaan reitittimeen (ABR). Muussa tapauksessa ABR saisi tiedon reitistä reitityspäivityksen mukana. (Ballew 1998, 136–137.)

Staattisten tai muiden samassa lähiverkossa toimivien reititysprotokollien reititietojen jakaminen OSPF-protokollassa tapahtuu *redistribute*-komennolla *router ospf 1* -komennon jälkeen, esimerkiksi:

```
redistribute static 192.168.1.1 255.255.255.252 192.168.1.2
```

Muista reititysprotokollista esimerkiksi RIP-protokollan reititystietojen jakaminen OSPF-protokollassa tapahtuu komennolla:

```
redistribute rip subnets
```

Lähiverkosta ulos, Internetiin menevästä oletusreitistä ilmoitetaan *default-information originate* -käskyllä (Cisco Systems 2012b). Tämä käsky voidaan kirjoittaa esimerkiksi staattisesti asetetun oletusreitien komennon jälkeen, esimerkiksi:

```
ip route 0.0.0.0 0.0.0.0 203.250.16.2
default-information originate
```

OSPF-protokollassa reiteille voidaan manuaalisesti määrittää kokonaiskustannusarvot (metriikka). Tämä tapahtuu *ip ospf cost* -komennolla, joka määritetään jokaiseen reitittimen porttiin erikseen. Esimerkiksi:

```
interface serial0/1
ip ospf cost 100
```

Käsky määrittää reitittimen sarjaportista 0/1 lähtevän reitin arvoksi 100 (Halabi 1996, 6–7). Mitä pienempi numeroarvo on, sitä parempi on reitti. Jos kustannusta ei määritetä, OSPF laskee siirtoyhteydelle kustannuksen kaavan 1 mukaisesti (Halabi 1996, 6):

$$\text{cost} = 100\,000\,000 / [\text{kaistanleveys bitteinä sekunnissa}] \quad (1)$$

Myös prioriteetti, mitä käytetään esimerkiksi *multiaccess*-verkoissa DR/BDR-valintaan, määritetään porttikohtaisesti. Prioriteetti määritetään *ip ospf priority* -käskyllä (Halabi 1996, 17). Esimerkiksi:

```
interface serial0/1
ip ospf priority 5
```

Kun reitittimeen on tehty asetukset, niiden tutkiminen tapahtuu *show*-komennoilla. Yksi useimmin käytettävä komento on *show ip route*, jolla saa näkyviin reitittimen reititustaulun:

```
show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
       B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
       OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
       NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
       external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
```

L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate default, U - per-user static route, o - ODR, P - periodic downloaded static route

Gateway of last resort is 10.0.0.2 to network 0.0.0.0

```

10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C    10.0.0.0/29 is directly connected, FastEthernet0/0
C    10.0.1.0/24 is directly connected, FastEthernet0/1
O IA 10.0.2.0/24 [110/2] via 10.0.0.4, 00:00:16, FastEthernet0/0
O IA 10.0.3.0/24 [110/2] via 10.0.0.5, 00:00:16, FastEthernet0/0
    200.100.100.0/30 is subnetted, 2 subnets
O    200.100.100.0 [110/2] via 10.0.0.1, 00:00:06, FastEthernet0/0
O    200.100.100.4 [110/2] via 10.0.0.2, 00:00:16, FastEthernet0/0
O*E2 0.0.0.0/0 [110/1] via 10.0.0.2, 00:00:16, FastEthernet0/0

```

Reititystaulussa *Codes*-kohta kertoo kirjainlyhenteiden merkityksen, esimerkiksi C tarkoittaa *connected*, eli reitittimeen itseensä suoraan kytketyt reitit ja O tarkoittaa OSPF:n tietämiä reittejä. *Gateway of last resort* tarkoittaa oletusreittiä lähiverkosta ulos, esimerkissä osoite 10.0.0.2 vie ulos lähiverkosta eli osoitteeseen 0.0.0.0. *Gateway of last resort* -tekstin alla ovat reititystaulun reitit. Esimerkiksi aliverkko 10.0.2.0 on toisessa OSPF:n alueessa kuin esimerkin reitin ja sinne pääsee osoitteen 10.0.0.4 kautta, yhteys on ollut toiminnassa 16 sekuntia ja yhteys on portissa FastEthernet0/0 (fa0/0).

OSPF-protokollaa koskevista show-komennoista yksi on *show ip ospf*:

```

show ip ospf
Routing Process "ospf 1" with ID 200.100.100.6
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
Area BACKBONE(0)

```

```

Number of interfaces in this area is 2
Area has no authentication
SPF algorithm executed 1 times
Area ranges are
Number of LSA 1. Checksum Sum 0x0001f0
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

```

Show ip ospf kertoo OSPF:n reititysprosessien numerot, esimerkissä vain 1 ja reitittimen ID:n, mikä esimerkin tapauksessa on suurin reitittimelle määritetty IP-osoite. Lisäksi tiedot kertovat mitä *Type of Service* -tyyppiä reititin tukee ja millaiset ovat sen LSA-vaihdon asetukset ja montako kertaa SPF-algoritmi on suoritettu. Alueiden osalta tiedot kertovat moneenko alueeseen reititin kuuluu ja millainen alue on kyseessä. (Cisco Systems 2012b.)

OSPF:n linkkitilatietokannan saa näkyviin komennolla *show ip ospf database*:

```

show ip ospf database
      OSPF Router with ID (200.100.100.6) (Process ID 1)

          Router Link States (Area 0)

Link ID          ADV Router      Age   Seq#           Checksum Link count
200.100.100.5   200.100.100.5   99    0x80000005    0x0003c8  2
10.0.2.1        10.0.2.1        103   0x80000003    0x00da43  1
200.100.100.6   200.100.100.6   99    0x80000004    0x00f3c8  2
200.100.100.1   200.100.100.1   99    0x80000005    0x00c814  2
10.0.3.1        10.0.3.1        99    0x80000003    0x00d644  1
10.0.1.1        10.0.1.1        99    0x80000003    0x00de42  1

          Net Link States (Area 0)

Link ID          ADV Router      Age   Seq#           Checksum
200.100.100.2   200.100.100.6   100   0x80000001    0x00f2c7
200.100.100.6   200.100.100.6   99    0x80000002    0x00e0d0
10.0.0.2        200.100.100.5   99    0x80000004    0x007a2e

          Summary Net Link States (Area 0)

Link ID          ADV Router      Age   Seq#           Checksum

```

10.0.2.0	10.0.2.1	134	0x80000001	0x008cb8
10.0.3.0	10.0.3.1	134	0x80000001	0x007ac8
10.0.1.0	10.0.1.1	140	0x80000001	0x009ea8

Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum	Tag
0.0.0.0	200.100.100.1	144	0x80000001	0x00261a	1
0.0.0.0	200.100.100.5	143	0x80000001	0x000e2e	1

Linkkitilietokanta kertoo kaikki reitittimen tietämät reitit lähiverkossa. Alussa ovat reitittimen ID ja reititysprosessin ID. Seuraavaksi ovat alue, johon reititin kuuluu ja lista lähiverkon reitittimien osoitteista (*Link ID*) ja niistä ilmoittaneiden reitittimien osoitteet (*ADV Router*). (Cisco Systems 2012b.)

Komennolla *show ip ospf neighbor* nähdään reitittimen naapurireitittimet:

```
show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
10.0.3.1	1	2WAY/DROTHER	00:00:30	10.0.0.5	Fa0/0
10.0.2.1	1	2WAY/DROTHER	00:00:30	10.0.0.4	Fa0/0
200.100.100.1	1	FULL/BDR	00:00:30	10.0.0.1	Fa0/0
200.100.100.5	1	FULL/DR	00:00:30	10.0.0.2	Fa0/0

Esimerkissä *Neighbor ID* on sama kuin *Router ID*. Prioriteetit reitittimien välisissä siirtoyhteyksissä ovat oletusarvoja eli jokaisen arvo on 1. *State* kertoo *adjacencyn* tilan ja reitittimien arvojärjestyksen (DR, BDR tai DROther). *Dead Time* kertoo missä ajassa yhteys katsotaan katkenneeksi jos naapuri ei lähetä Hello-viestiä. *Address* kertoo osoitteet reitittimen ja sen naapureiden välillä. *Interface* kertoo minkä portin kautta reititin on yhteydessä mihinkin naapuriinsa. (Cisco Systems 2012b.)

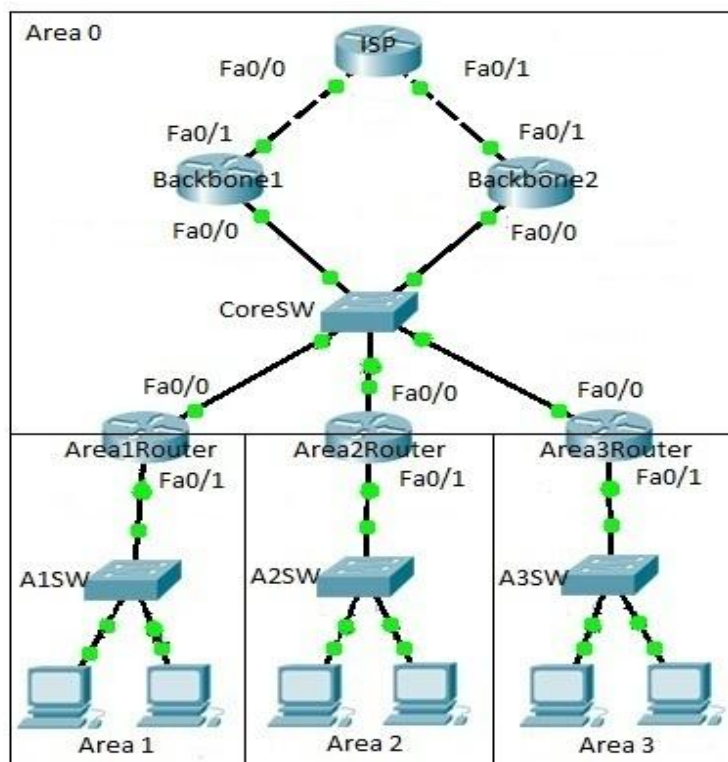
6 OSPF-PROTOKOLLA KÄYTÄNNÖSSÄ

Seuraavaksi esittelen OSPF-protokollan toimintaa käytännössä kuvassa 8 näkyvän esimerkiverkon avulla. Suunnittelin esimerkiverkon Ciscon *Packet Tracer*-ohjelmalla, millä verkon toimintaa voi myös simuloida. Toteutin verkon myös oikeilla laitteilla Mikkelin ammattikorkeakoulun Mikpoli-rakennuksen luokkaan MB316. Tes-

tasin verkon avulla OSPF:n toimintaa vikatilanteissa. Luotettavien tulosten saamiseksi suoritin testit oikeilla laitteilla.

6.1 Esimerkkiverkko

Esimerkkiverkko koostuu kuudesta reitittimestä ja neljästä kytkimestä. Verkko jakautuu neljään alueeseen (0, 1, 2 ja 3). Kuvassa alueilla 1, 2 ja 3 on jokaisella kaksi tietokonetta esimerkin vuoksi. Nolla-alue yhdistää alueet 1, 2 ja 3 toisiinsa ja Internetiin. ISP-reititin vastaa verkon Internet-palveluntarjoajaa, jonka kautta Internet-yhteys toimii. ISP-reititin toimii myös verkon DHCP-palvelimena ja se jakaa IP-osoitteet alueiden 1, 2 ja 3 päätelaitteille. Area1Routerin, Area2Routerin ja Area3Routerin fa0/1-portteihin on määritetty *ip helper-address*, minkä tarkoitus on ilmaista reitittimelle mihin osoitteeseen sen tulee reitittää päätelaitteilta tulevat DHCP-pyyntöt.



KUVA 8. Esimerkkiverkko

Verkon kaikkiin reitittimiin on konfiguroitu OSPF. ISP-, Backbone1- ja Backbone2-reitittimet kuuluvat pelkästään nolla-alueeseen. Muut reitittimet (Area1Router, Area2Router ja Area3Router) kuuluvat puoliksi 0 alueeseen ja puoliksi omiin alueisiinsa (1, 2 ja 3). CoreSW, A1SW, A2SW ja A3SW ovat kytkimiä. Reitittimien porttien IP-osoitteet ovat taulukossa 7. Reitittimien asetukset liitteissä 2(1-8).

TAULUKKO 7. Esimerkkiverkon reitittimien porttien IP-osoitteet

Reititin	Portti	IP
ISP	Fa0/0	200.100.100.2 255.255.255.252
	Fa0/1	200.100.100.6 255.255.255.252
Backbone1	Fa0/0	10.0.0.1 255.255.255.248
	Fa0/1	200.100.100.1 255.255.255.252
Backbone2	Fa0/0	10.0.0.2 255.255.255.248
	Fa0/1	200.100.100.5 255.255.255.252
Area1Router	Fa0/0	10.0.0.3 255.255.255.248
	Fa0/1	10.0.1.1 255.255.255.0
Area2Router	Fa0/0	10.0.0.4 255.255.255.248
	Fa0/1	10.0.2.1 255.255.255.0
Area3Router	Fa0/0	10.0.0.5 255.255.255.248
	Fa0/1	10.0.3.1 255.255.255.0

Taulukossa 7 ovat esimerkkiverkon IP-osoitteet. ISP-reitittimen IP-osoitteet ovat globaaleja osoitteita, koska se vastaa Internet-palveluntarjoajaa. Muut verkon osoitteet ovat lokaaleja. Lokaalien osoitteiden vuoksi Backbone1- ja Backbone2-reitittimiin on konfiguroitu NAT.

6.2 Testit ja tulokset

Teen esimerkkiverkolle testejä, joilla kokeilen miten OSPF toimii vikatilanteissa. Testit ovat seuraavat:

1. Yhteyden katkaisu CoreSW-kytkimen ja Area3Routerin välillä alueella 0
2. Yhteyden katkaisu Area1Routerin ja A1SW-kytkimen välillä alueella 1
3. Yhteyden katkaisu Backbone2:n ja CoreSW-kytkimen välillä alueella 0

Testi 1

Ensimmäisessä testissä alue 3 erotetaan muusta verkosta irrottamalla verkkokaapeli Area3Routerin fa0/0-portista. Reititin itse huomaa heti, että verkko on poikki, mutta muut reitittimet, jotka ISP:tä lukuun ottamatta olivat yhdistettynä Area3Routeriin kytkimen kautta, merkitsevät yhteyden katkenneeksi vasta 40 sekunnin kuluttua kun neljään *Hello*-pakettiin ei tule vastausta. ISP saa tiedon reitin katkeamisesta Backbone1:ltä ja Backbone2:lta. Kaikki reitittimet paitsi Area3Router poistavat 10.0.3.0-verkon reititystauluistaan. Area3Router vastaavasti poistaa kaikki muut reitit paitsi 10.0.3.0-verkon reititystaulustaan. Esimerkkinä reititystaulut ISP:ltä ja Area3Routerilta.

```
ISP#show ip route
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
        B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
        OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
        NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
        external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
        L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
        default, U - per-user static route, o - ODR, P - periodic
        downloaded static route
```

```
Gateway of last resort is 200.100.100.5 to network 0.0.0.0
```

```

    10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O       10.0.0.0/29 [110/2] via 200.100.100.1, 00:06:31, Fa0/0
          [110/2] via 200.100.100.5, 00:06:31, Fa0/1
O IA    10.0.1.0/24 [110/3] via 200.100.100.1, 00:06:31, Fa0/0
          [110/3] via 200.100.100.5, 00:06:31, Fa0/1
O IA    10.0.2.0/24 [110/3] via 200.100.100.1, 00:06:31, Fa0/0
          [110/3] via 200.100.100.5, 00:06:31, Fa0/1
    200.100.100.0/30 is subnetted, 2 subnets
C       200.100.100.0 is directly connected, Fa0/0
C       200.100.100.4 is directly connected, Fa0/1
O*E2 0.0.0.0/0 [110/1] via 200.100.100.5, 00:06:31, Fa0/1
```

```
Area3Router#show ip route
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
        B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
```

```
OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
default, U - per-user static route, o - ODR, P - periodic
downloaded static route
```

```
Gateway of last resort is not set
```

```
10.0.0.0/24 is subnetted, 1 subnets
```

```
C      10.0.3.0 is directly connected, Fa0/1
```

Kun verkkokaapeli kytketään uudelleen Area3Routeriin, se alkaa lähettää paketteja, joilla se ottaa selvää verkon rakenteesta. Saamiensa tietojen perusteella se muodostaa uudelleen reititystaulunsa. Testissä yhteyden katkaisu oli lyhytaikainen, joten mikään verkon reitittimistä ei poistanut yhtään reititietoa linkkitilatietokannastaan.

Testi 2

Toisessa testissä yhteys Area1Routerin ja alueella 1 olevan A1SW-kytkimen välillä katkaistaan irrottamalla verkkokaapeli reitittimestä. Reititin huomaa heti reitin katkenneen ja ilmoittaa siitä muille verkon reitittimille. Kaikki verkon reitittimet poistavat 10.0.1.0-verkon reititystauluistaan. Kun verkkokaapeli kytketään uudelleen reititimeen, se ilmoittaa heti muille reitittimille 10.0.1.0-verkosta ja kaikki reitittimet lisäävät verkon reititystauluihinsa. Esimerkkinä reititystaulu ISP:ltä kun reitin katkaaminen on huomattu.

```
ISP#show ip route
```

```
Codes:      C - connected, S - static, I - IGRP, R - RIP, M - mobile,
            B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
            OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
            NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
            external type 2, E - EGP, i - IS-IS, su - IS-IS summary,
            L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
            area, * - candidate default, U - per-user static route,
            o - ODR, P - periodic downloaded static route
```

```
Gateway of last resort is not set
```

```

200.100.100.0/30 is subnetted, 2 subnets
C      200.100.100.0 is directly connected, Fa0/0
C      200.100.100.4 is directly connected, Fa0/1
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O IA   10.0.2.0/24 [110/3] via 200.100.100.5, 00:02:03, Fa0/1
        [110/3] via 200.100.100.1, 00:02:03, Fa0/0
O IA   10.0.3.0/24 [110/3] via 200.100.100.5, 00:02:03, Fa0/1
        [110/3] via 200.100.100.1, 00:02:03, Fa0/0
O      10.0.0.0/29 [110/2] via 200.100.100.1, 00:02:04, Fa0/0
        [110/2] via 200.100.100.5, 00:02:04, Fa0/1

```

Testi 3

Kolmannessa testissä yhteys Backbone2-reitittimen ja alueella 0 olevan CoreSW-kytkimen välillä katkaistaan irrottamalla verkkokaapeli reitittimestä. Backbone2 huomaa heti verkon katkeamisen ja ilmoittaa siitä ISP:lle. ISP poistaa reititystaulustaan kaikki vaihtoehtoiset reitit, jotka kulkivat Backbone2:n kautta. Backbone2 laskee uudet reitit muille reitittimille ISP:n kautta. Area1Router, Area2Router ja Area3Router laskevat uudet reittinsä Backbone2:lle Backbone1:n kautta ja Backbone1 ISP:n kautta. Esimerkkinä reititystaulut ISP:ltä ja Backbone2:lta.

```

ISP#show ip route
Codes:      C - connected, S - static, I - IGRP, R - RIP, M - mobile,
           B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
           OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
           NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
           external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
           L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
           default, U - per-user static route, o - ODR, P - periodic
           downloaded static route

```

```

Gateway of last resort is 200.100.100.5 to network 0.0.0.0

```

```

10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
O      10.0.0.0/29 [110/2] via 200.100.100.1, 00:01:12, Fa0/0
O IA   10.0.1.0/24 [110/3] via 200.100.100.1, 00:01:02, Fa0/0
O IA   10.0.2.0/24 [110/3] via 200.100.100.1, 00:01:02, Fa0/0
O IA   10.0.3.0/24 [110/3] via 200.100.100.1, 00:01:02, Fa0/0
200.100.100.0/30 is subnetted, 2 subnets
C      200.100.100.0 is directly connected, Fa0/0

```

```
C      200.100.100.4 is directly connected, Fa0/1
O*E2 0.0.0.0/0 [110/1] via 200.100.100.5, 00:29:14, Fa0/1
```

```
Backbone2#show ip route
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
        B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
        OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
        NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
        external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
        L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
        default, U - per-user static route, o - ODR, P - periodic
        downloaded static route
```

```
Gateway of last resort is 0.0.0.0 to network 0.0.0.0
```

```
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
O      10.0.0.0/29 [110/3] via 200.100.100.6, 00:02:57, Fa0/1
O IA   10.0.1.0/24 [110/4] via 200.100.100.6, 00:02:57, Fa0/1
O IA   10.0.2.0/24 [110/4] via 200.100.100.6, 00:02:57, Fa0/1
O IA   10.0.3.0/24 [110/4] via 200.100.100.6, 00:02:57, Fa0/1
      200.100.100.0/30 is subnetted, 2 subnets
O      200.100.100.0 [110/2] via 200.100.100.6, 00:31:03, Fa0/1
C      200.100.100.4 is directly connected, Fa0/1
S*    0.0.0.0/0 is directly connected, Fa0/1
```

Kun yhteys palautetaan liittämällä verkkokaapeli takaisin reitittimeen, alkaa Backbone2 paketteja uudelleen auennutta yhteyttä pitkin. Muut verkon reitittimet saavat tiedon reitistä ja laskevat uudelleen reititystaulunsa. Kaikkien verkon reitittimien reititustaulut palaavat katkosta edeltävään muotoonsa, mutta verkon multiaccess-osan reitittimien arvojärjestys on muuttunut. Backbone2 ei ole enää DR, vaan DROther. Edellinen BDR, eli Backbone1 on nyt DR ja Area3Router, jolla on seuraavaksi suurin IP-osoite (10.0.3.1), on nyt BDR. Alkuperäinen arvojärjestys palautuu, jos kaikki verkon reitittimet käynnistetään uudelleen.

Testien perusteella OSPF huomaa nopeasti katkenneet reitit ja laskee vaihtoehdoisen reitin katkenneen tilalle, jos sellainen löytyy. Katkenneen reitin kytkeytyessä uudelleen, reitti huomataan nopeasti ja se otetaan käyttöön, jos se on parempi tai yhtä hyvä kuin sillä hetkellä käytössä oleva. Esimerkkiverkon tapauksessa vain yhdessä testissä

oli mahdollisuus käyttää vaihtoehtoisia reittejä. Tämä osoittaa, että verkon topologialla on myös merkitystä sen toimintaan.

7 YHTEENVETO

Tutustuin työn alussa nykyisiin tietoverkkoihin ja tietoliikenteeseen. Kerroin aluksi yleisesti tietoverkoista ja niiden kaapelointitavoista. Kerroin myös lyhyesti Internetin historiasta. Lähiverkkotyyppien osalta kerroin Ethernetin kehityksestä. Seuraavaksi tutustuin OSI- ja TCP/IP-viitemalleihin sekä niiden mukaisesti toimivaan tiedonsiirtoon. Tietoverkkojen yhteydessä tutustuin myös niissä käytettävien IP-osoitteiden jakamiseen. Selvitin ARP:n, DHCP:n, NAT:n ja DNS-palvelimien toiminnan ja tarkoituksen. IP-osoitteisiin tutustuin tarkemmin reitityksen yhteydessä.

Työn IP-osoitteita ja reititystä käsittelevässä osassa selvitin aluksi IP-osoitteiden rakenteen. Reitityksen osalta kerroin, että reitti voidaan määrittää joko staattisesti tai dynaamisesti. Staattisessa reitityksessä kahden IP-osoitteen välille määritetään yhteys. Dynaamisessa reitityksessä käytetään reititysprotokollia, joihin määritetään IP-osoitteet tai IP-osoitealueet, joiden välillä tiedonsiirtoa tapahtuu.

Protokollien osalta selvitin, että reititettävät protokollat muokkaavat dataa siirtoa varten ja reititysprotokollat laskevat reitittimille reittejä, mitä pitkin dataa siirretään. Reititettävistä protokollista mainitsin jo viitemalleista kertoessani TCP- ja UDP-protokollat, joita käytetään yleisesti tiedonsiirrossa. Selvitin myös, että reititysprotokolla on reitittimessä toimiva tietokoneohjelma, joka perustuu matemaattiseen algoritmiin. Algoritmeja on erilaisia, mutta niillä on samanlainen tarkoitus. Ne laskevat lyhimmän reitin kahden pisteen välillä. Eri algoritmit suorittavat tehtävänsä eri tavoin ja päätyvät verkon rakenteesta riippuen usein eri ratkaisuihin.

Reititysprotokolliin tutustuessani selvitin, että ne jaetaan itsenäisten järjestelmien (AS) sisällä käytettäviin (IGP) ja niiden yhdistämiseen käytettäviin (EGP). Toimintaperiaatteen osalta reititysprotokollat jakautuvat etäisyysvektori- ja linkkitilaprotokolliin. Etäisyysvektori-protokollat sopivat pieniin verkkoihin, ne käyttävät vain vähän reitittimen muistia, mutta niiden ajoitetut reitityspäivitykset kuluttavat paljon verkon

kaistanleveyttä. Linkkitilaprotokollat soveltuvat myös suuriin verkkoihin, ne käyttävät enemmän reitittimen muistia, mutta vievät vähemmän verkon kaistanleveyttä.

Reititysprotokollista keskityin tarkemmin OSPF-protokollaan. Selvitin, että OSPF on IETF:n kokoaman ryhmän 1990-luvun alussa kehittämä reititysprotokolla, joka on suunniteltu TCP/IP-malliin perustuviin verkkoihin. Tutustuin OSPF:n ominaisuuksiin, kuten linkkitilatietokantaan ja protokollapaketteihin. Selvitin, että linkkitilatietokanta rakentuu LSA-nimisistä reittitiedoista. Reittitietoja OSPF jakaa naapureidensa kanssa tiettyjen protokollapakettien välityksellä. Selvitin myös miten OSPF toimii multiaccess-verkossa ja miten lähiverkon voi jakaa alueisiin OSPF:n avulla. Käytännön osuudessa tein esimerkiverkon, jossa yhdistin reitittäjiä suoraan toisiinsa Point-to-Point-yhteyksillä sekä kytkimen kautta. Aiheutin verkkoon vikatilanteita ja tutkin miten OSPF reagoi niihin. Testit osoittivat OSPF:n toimivan teoriassa kerrotulla tavalla. OSPF on yksi parhaista lähiverkkojen sisällä käytettävistä reititysprotokollista. Se on nopea, skaalautuva ja toimii kaikkien reititinvalmistajien laitteissa. OSPF on huomioon otettava vaihtoehto valittaessa verkon reititysprotokollaa.

LÄHTEET

Albrightson, Bob, Boyle, Joanne ja Garcia-Luna-Aceves J. J. EIGRP - a fast routing protocol based on distance vectors. PDF-dokumentti. <http://vir.liu.se/~TDTS02/papers/eigrp.pdf>. Ei päivitystietoa. Luettu 19.10.2012.

Ballew, Scott M. 1998. IP -verkkojen hallinta Cison reitittimillä. Tehokäyttäjän opas. Espoo: Suomen Atk-kustannus Oy. ISBN 951-762-644-4.

Cisco Press 1998. OSPF network design solutions. Sähköinen kirja. Macmillan Computer Publishing. ISBN 1578700469.

Cisco Systems 1996. Cisco – IOS Tutorial. HTML-dokumentti. <http://www.cisco.com/warp/cpropub/45/tutorial.htm>. Päivitetty 29.8.1997. Luettu 11.11.2012.

Cisco Systems 2002. Cisco verkkoakatemia. Ensimmäinen vuosi. Helsinki: Edita Prima Oy. ISBN 951-826-559-3.

Cisco Systems 2012a. Clock Rate Command Enhancements Feature Module. HTML-dokumentti. http://www.cisco.com/en/US/docs/ios/11_2/feature/guide/clkrate.html. Ei päivitystietoa. Luettu 11.11.2012.

Cisco Systems 2012b. OSPF commands. HTML-dokumentti. http://www.cisco.com/en/US/docs/ios/12_2/iproute/command/reference/1rfospf.html#wp1019916. Ei päivitystietoa. Luettu 14.10.2012.

Doyle, Jeff 2005. OSPF and IS-IS: Choosing an IGP for Large-Scale Networks. Sähköinen kirja. Addison Wesley Professional. ISBN 0-321-16879-8.

Halabi, Sam 1996. OSPF design guide. Cisco Systems. USA: Cisco Systems.

Hendrick, C. 1988. Routing Information Protocol. Standardi. HTML-dokumentti. <http://rsync.tools.ietf.org/html/rfc1058>. Päivitetty 10.6.1988. Luettu 12.1.2012.

InformIT 2012. Creating and Configuring Loopback Interfaces. HTML-dokumentti. http://www.informit.com/library/content.aspx?b=CCNA_Practical_Studies&seqNum=45. Ei päivitystietoa. Luettu 11.11.2012.

Jaakohuhta, Hannu 2002. Lähiverkot – Ethernet. IT Press. Helsinki: Edita Prima Oy.

Khabardar, Percy ja Finn, Norman William 1998. Layer 3 switch unicast protocol. Patentti. HTML-dokumentti. <http://www.google.fi/patents?hl=fi&lr=&vid=USPAT6049834&id=ghYEAAAAEBAJ&oi=fnd&dq=layer+3+switch&printsec=abstract#v=onepage&q=layer%20%20switch&f=false>. Päivitetty 4.5.1998. Luettu 16.4.2012.

Malkin, G. 1998. RIP version 2. Standardi. HTML-dokumentti. <http://wiki.tools.ietf.org/html/rfc2453>. Päivitetty 8.11.1998. Luettu 15.1.2012.

Malkin, G. ja Minnear, R. 1997. RIPng for IPv6. Standardi. HTML-dokumentti. <https://tools.ietf.org/html/rfc2080>. Päivitetty 16.1.1997. Luettu 12.1.2012.

Mockapetris, P. 1987. Domain names – concepts and facilities. Standardi. HTML-dokumentti. <http://www.zytrax.com/books/dns/apd/rfc1034.txt>. Päivitetty 15.11.1987. Luettu 25.2.2012.

Moy, John T 1997. OSPF version 2. Standardi. HTML-dokumentti. <http://ebook.tools.ietf.org/html/rfc2178>. Päivitetty 12.7.1997. Luettu 18.12.2011.

Moy, John T. 2000. OSPF: Anatomy of an Internet Routing Protocol. PDF-dokumentti. New Jersey: Addison Wesley. ISBN 0-201-63472-4.

Oran, David R. 1990. OSI IS-IS Intra-domain Routing Protocol. Standardi. <http://tools.ietf.org/html/rfc1142>. Luotu 20.2.1990. Luettu 12.5.2012.

Parkhurst, William R. 1998. Cisco router OSPF. Desing & implementation guide. PDF-dokumentti. USA: McGraw-Hill.

Parziale, Lydia ja Britt, T. David 2006. TCP/IP tutorial and technical overview. PDF-dokumentti. USA: IBM Redbooks.

Tanenbaum, Andrew S. 1996. Computer Networks. Third edition. New Jersey: Prentice-Hall International. ISBN 0-13-394248-1.

Thomas, Thomas M. 2003. OSPF network design solutions. PDF-dokumentti. Indianapolis: Cisco Press.

Dijkstran algoritmi (Tanenbaum 1996, 352)

```

#define MAX_NODES 1024          /*päätteiden maksimimäärä*/
#define INFINITY 1000000000    /*saavuttamattoman reitin arvo*/
int n, dist[MAX_NODES][MAX_NODES]; /*dist[i][j] etäisyys välillä i ja j */

void shortest_path(int s, int t, int path[])
{
struct state                    /*laskettava reitti*/
{
int predecessor;              /*edeltäjä*/
int lenght;                   /*etäisyys päätteeltä lähteeseen*/
enum {permanent, tentative} label; /*leiman tila*/
}
state [MAX_NODES];

int i, k, min;
struct state * p;

for (p = &state[0]; p < &state[n]; p++) /*asetta tila*/
{
p->predecessor = -1;
p->lenght = INFINITY;
p->label = tentative;
}
state[t].lenght = 0;          state[t].label = permanent;
k = t;                        /*k on kohde*/
do                             /*onko kohteeseen parempaa reittiä?*/
{
for (i = 0; i < n; i++)      /*tässä verkossa on n päätettä*/
if (dist[k][i] != 0 && state[i].label == tentative)
{
if (state[k].lenght + dist[k][i] < state[i].lenght)
{
state[i].predecessor = k;
state[i].lenght = state[k].lenght + dist[k][i];
} }
k = 0;          min = INFINITY;
for (i = 0; i < n; i++)
if (state[i].label == tentative && state[i].lenght < min)
{
min = state[i].lenght;
k = i;
}
state[k].label = permanent;
}
while (k != s);              /*kopioi tämä reitti reititystauluun*/
i = 0;          k = s;
do { path[i++] = k; k = state[k].predecessor; }
while (k >= 0); }

```

Esimerkkiverkon reitittimien asetukset ja reititystaulut**ISP-reititin**

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname ISP
ISP(config)#interface fa0/0
ISP(config-if)#ip address 200.100.100.2 255.255.255.252
ISP(config-if)#no shutdown
ISP(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
ISP(config-if)#interface fa0/1
ISP(config-if)#ip address 200.100.100.6 255.255.255.252
ISP(config-if)#no shutdown
ISP(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
ISP(config-if)#exit
ISP(config)#ip dhcp excluded-address 10.0.1.1
ISP(config)#ip dhcp excluded-address 10.0.2.1
ISP(config)#ip dhcp excluded-address 10.0.3.1
ISP(config)#ip dhcp pool ArealPool
ISP(dhcp-config)#default-router 200.100.100.2
ISP(dhcp-config)#network 10.0.1.0 255.255.255.0
ISP(dhcp-config)#exit
ISP(config)#ip dhcp pool Area2Pool
ISP(dhcp-config)#default-router 200.100.100.2
ISP(dhcp-config)#network 10.0.2.0 255.255.255.0
ISP(dhcp-config)#exit
ISP(config)#ip dhcp pool Area3Pool
ISP(dhcp-config)#default-router 200.100.100.2
ISP(dhcp-config)#network 10.0.3.0 255.255.255.0
ISP(dhcp-config)#exit
ISP(config)#
ISP(config)#router ospf 1
ISP(config-router)#network 200.100.100.0 0.0.0.3 area 0
ISP(config-router)#network 200.100.100.4 0.0.0.3 area 0
ISP(config-router)#exit
ISP(config)#exit
ISP#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
        B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate default, U - per-user static route, o - ODR, P - periodic downloaded static route

Gateway of last resort is 200.100.100.5 to network 0.0.0.0

```

10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
O      10.0.0.0/29 [110/2] via 200.100.100.1, 00:02:34,
FastEthernet0/0
          [110/2] via 200.100.100.5, 00:02:34,
FastEthernet0/1
O IA   10.0.1.0/24 [110/3] via 200.100.100.1, 00:02:34,
FastEthernet0/0
          [110/3] via 200.100.100.5, 00:02:34,
FastEthernet0/1
O IA   10.0.2.0/24 [110/3] via 200.100.100.1, 00:02:34,
FastEthernet0/0
          [110/3] via 200.100.100.5, 00:02:34,
FastEthernet0/1
O IA   10.0.3.0/24 [110/3] via 200.100.100.1, 00:02:34,
FastEthernet0/0
          [110/3] via 200.100.100.5, 00:02:34,
FastEthernet0/1
200.100.100.0/30 is subnetted, 2 subnets
C      200.100.100.0 is directly connected, FastEthernet0/0
C      200.100.100.4 is directly connected, FastEthernet0/1
O*E2 0.0.0.0/0 [110/1] via 200.100.100.5, 00:02:34, FastEthernet0/1

```

Backbone1-reititin

```

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Backbone1
Backbone1(config)#interface fa0/0
Backbone1(config-if)#ip address 10.0.0.1 255.255.255.248
Backbone1(config-if)#no shutdown
Backbone1(config-if)#

```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

```
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
Backbone1(config-if)#interface fa0/1
Backbone1(config-if)#ip address 200.100.100.1 255.255.255.252
Backbone1(config-if)#no shutdown
Backbone1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
Backbone1(config-if)#exit
Backbone1(config)#ip route 200.100.100.2 255.255.255.252 fa0/1
Backbone1(config)#ip route 0.0.0.0 0.0.0.0 fa0/1
Backbone1(config)#ip access-list extended NAT
Backbone1(config-ext-nacl)#permit ip 10.0.0.0 0.0.0.7 any
Backbone1(config-ext-nacl)#permit ip 10.0.1.0 0.0.0.255 any
Backbone1(config-ext-nacl)#permit ip 10.0.2.0 0.0.0.255 any
Backbone1(config-ext-nacl)#permit ip 10.0.3.0 0.0.0.255 any
Backbone1(config-ext-nacl)#exit
Backbone1(config)#ip nat inside source list NAT interface fa0/1
overload
Backbone1(config)#router ospf 1
Backbone1(config-router)#network 10.0.0.0 0.0.0.7 area 0
Backbone1(config-router)#network 200.100.100.0 0.0.0.3 area 0
Backbone1(config-router)#default-information originate
Backbone1(config-router)#exit
Backbone1(config)#exit
Backbone1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
        B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
        OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
        NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
        external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
        L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
        default, U - per-user static route, o - ODR, P - periodic
        downloaded static route
```

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

```
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C      10.0.0.0/29 is directly connected, FastEthernet0/0
O IA   10.0.1.0/24 [110/2] via 10.0.0.3, 00:03:56, FastEthernet0/0
O IA   10.0.2.0/24 [110/2] via 10.0.0.4, 00:03:56, FastEthernet0/0
O IA   10.0.3.0/24 [110/2] via 10.0.0.5, 00:03:56, FastEthernet0/0
200.100.100.0/30 is subnetted, 2 subnets
C      200.100.100.0 is directly connected, FastEthernet0/1
```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

```
O          200.100.100.4 [110/2] via 200.100.100.2, 00:03:56,
FastEthernet0/1
          [110/2] via 10.0.0.2, 00:03:56, FastEthernet0/0
S*   0.0.0.0/0 is directly connected, FastEthernet0/1
```

Backbone2-reititin

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Backbone2
Backbone2(config)#interface fa0/0
Backbone2(config-if)#ip address 10.0.0.2 255.255.255.248
Backbone2(config-if)#no shutdown
Backbone2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
Backbone2(config-if)#interface fa0/1
Backbone2(config-if)#ip address 200.100.100.5 255.255.255.252
Backbone2(config-if)#no shutdown
Backbone2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
Backbone2(config-if)#exit
Backbone2(config)#ip route 200.100.100.6 255.255.255.252 fa0/1
Backbone2(config)#ip route 0.0.0.0 0.0.0.0 fa0/1
Backbone2(config)#ip access-list extended NAT
Backbone2(config-ext-nacl)#permit ip 10.0.0.0 0.0.0.7 any
Backbone2(config-ext-nacl)#permit ip 10.0.1.0 0.0.0.255 any
Backbone2(config-ext-nacl)#permit ip 10.0.2.0 0.0.0.255 any
Backbone2(config-ext-nacl)#permit ip 10.0.3.0 0.0.0.255 any
Backbone2(config-ext-nacl)#exit
Backbone2(config)#ip nat inside source list NAT interface fa0/1
overload
Backbone2(config)#router ospf 1
Backbone2(config-router)#network 10.0.0.0 0.0.0.7 area 0
Backbone2(config-router)#network 200.100.100.4 0.0.0.3 area 0
Backbone2(config-router)#default-information originate
Backbone2(config-router)#exit
Backbone2(config)#exit
Backbone2#show ip route
Codes:    C - connected, S - static, I - IGRP, R - RIP, M - mobile,
          B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate default, U - per-user static route, o - ODR, P - periodic downloaded static route

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

```
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C      10.0.0.0/29 is directly connected, FastEthernet0/0
O IA   10.0.1.0/24 [110/2] via 10.0.0.3, 00:04:14, FastEthernet0/0
O IA   10.0.2.0/24 [110/2] via 10.0.0.4, 00:04:14, FastEthernet0/0
O IA   10.0.3.0/24 [110/2] via 10.0.0.5, 00:04:14, FastEthernet0/0
      200.100.100.0/30 is subnetted, 2 subnets
O      200.100.100.0 [110/2] via 200.100.100.6, 00:04:14,
FastEthernet0/1
      [110/2] via 10.0.0.1, 00:04:14, FastEthernet0/0
C      200.100.100.4 is directly connected, FastEthernet0/1
S*    0.0.0.0/0 is directly connected, FastEthernet0/1
```

Area1Router

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname ArealRouter
ArealRouter(config)#
ArealRouter(config)#interface fa0/0
ArealRouter(config-if)#ip address 10.0.0.3 255.255.255.248
ArealRouter(config-if)#no shutdown
ArealRouter(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
ArealRouter(config-if)#interface fa0/1
ArealRouter(config-if)#ip address 10.0.1.1 255.255.255.0
ArealRouter(config-if)#ip helper-address 200.100.100.2
ArealRouter(config-if)#ip helper-address 200.100.100.6
ArealRouter(config-if)#no shutdown
ArealRouter(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
ArealRouter(config-if)#exit
```


Esimerkkiverkon reitittimien asetukset ja reititystaulut

```
ArealRouter(config)#router ospf 1
ArealRouter(config-router)#network 10.0.0.0 0.0.0.7 area 0
ArealRouter(config-router)#network 10.0.1.0 0.0.0.255 area 1
ArealRouter(config-router)#exit
ArealRouter(config)#exit
ArealRouter#show ip route

Codes:    C - connected, S - static, I - IGRP, R - RIP, M - mobile,
          B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
          OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
          NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
          external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
          L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
          default, U - per-user static route, o - ODR, P - periodic
          downloaded static route
```

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

```
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.0.0.0/29 is directly connected, FastEthernet0/0
C       10.0.1.0/24 is directly connected, FastEthernet0/1
O IA   10.0.2.0/24 [110/2] via 10.0.0.4, 00:04:38, FastEthernet0/0
O IA   10.0.3.0/24 [110/2] via 10.0.0.5, 00:04:38, FastEthernet0/0
200.100.100.0/30 is subnetted, 2 subnets
O      200.100.100.0 [110/2] via 10.0.0.1, 00:04:38, FastEthernet0/0
O      200.100.100.4 [110/2] via 10.0.0.2, 00:04:38, FastEthernet0/0
O*E2 0.0.0.0/0 [110/1] via 10.0.0.1, 00:04:38, FastEthernet0/0
```

Area2Router

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Area2Router
Area2Router(config)#interface fa0/0
Area2Router(config-if)#ip address 10.0.0.4 255.255.255.248
Area2Router(config-if)#no shutdown
Area2Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
Area2Router(config-if)#interface fa0/1
Area2Router(config-if)#ip address 10.0.2.1 255.255.255.0
Area2Router(config-if)#ip helper-address 200.100.100.2
```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

```
Area2Router(config-if)#ip helper-address 200.100.100.6
Area2Router(config-if)#no shutdown
Area2Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
Area2Router(config-if)#exit
Area2Router(config)#router ospf 1
Area2Router(config-router)#network 10.0.0.0 0.0.0.7 area 0
Area2Router(config-router)#network 10.0.2.0 0.0.0.255 area 2
Area2Router(config-router)#exit
Area2Router(config)#exit
Area2Router#show ip route
Codes:    C - connected, S - static, I - IGRP, R - RIP, M - mobile,
          B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
          OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
          NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
          external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
          L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
          default, U - per-user static route, o - ODR, P - periodic
          downloaded static route
```

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

```
    10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.0.0.0/29 is directly connected, FastEthernet0/0
O IA    10.0.1.0/24 [110/2] via 10.0.0.3, 00:04:55, FastEthernet0/0
C       10.0.2.0/24 is directly connected, FastEthernet0/1
O IA    10.0.3.0/24 [110/2] via 10.0.0.5, 00:04:55, FastEthernet0/0
    200.100.100.0/30 is subnetted, 2 subnets
O       200.100.100.0 [110/2] via 10.0.0.1, 00:04:55, FastEthernet0/0
O       200.100.100.4 [110/2] via 10.0.0.2, 00:04:55, FastEthernet0/0
O*E2 0.0.0.0/0 [110/1] via 10.0.0.1, 00:04:55, FastEthernet0/0
```

Area3Router

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Area3Router
Area3Router(config)#interface fa0/0
Area3Router(config-if)#ip address 10.0.0.5 255.255.255.248
Area3Router(config-if)#no shutdown
```

Esimerkkiverkon reitittimien asetukset ja reititystaulut

```

Area3Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
Area3Router(config-if)#interface fa0/1
Area3Router(config-if)#ip address 10.0.3.1 255.255.255.0
Area3Router(config-if)#ip helper-address 200.100.100.2
Area3Router(config-if)#ip helper-address 200.100.100.6
Area3Router(config-if)#no shutdown
Area3Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
Area3Router(config-if)#exit
Area3Router(config)#router ospf 1
Area3Router(config-router)#network 10.0.0.0 0.0.0.7 area 0
Area3Router(config-router)#network 10.0.3.0 0.0.0.255 area 3
Area3Router(config-router)#exit
Area3Router(config)#exit
Area3Router#show ip route
Codes:    C - connected, S - static, I - IGRP, R - RIP, M - mobile,
          B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
          OSPF inter area, N1 - OSPF NSSA external type 1, N2 - OSPF
          NSSA external type 2, E1 - OSPF external type 1, E2 - OSPF
          external type 2, E - EGP, i - IS-IS, L1 - IS-IS level-1,
          L2 - IS-IS level-2, ia - IS-IS inter area, * - candidate
          default, U - per-user static route, o - ODR, P - periodic
          downloaded static route

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

    10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.0.0.0/29 is directly connected, FastEthernet0/0
O IA    10.0.1.0/24 [110/2] via 10.0.0.3, 00:05:07, FastEthernet0/0
O IA    10.0.2.0/24 [110/2] via 10.0.0.4, 00:05:07, FastEthernet0/0
C       10.0.3.0/24 is directly connected, FastEthernet0/1
    200.100.100.0/30 is subnetted, 2 subnets
O       200.100.100.0 [110/2] via 10.0.0.1, 00:05:07, FastEthernet0/0
O       200.100.100.4 [110/2] via 10.0.0.2, 00:05:07, FastEthernet0/0
O*E2   0.0.0.0/0 [110/1] via 10.0.0.1, 00:05:07, FastEthernet0/0

```