KYMENLAAKSON AMMATTIKORKEAKOULU
Tietotekniikan koulutusohjelma / tietoverkkotekniikka

Jan Lampikari

DHCPV6-PALVELIN OPERAATTORIKÄYTÖSSÄ

Opinnäytetyö 2014

TIIVISTELMÄ

KYMENLAAKSON AMMATTIKORKEAKOULU
Tietotekniikan koulutusohjelma

LAMPIKARI, JAN	DHCPv6-palvelin operaattorikäytössä
Opinnäytetyö	50 sivua + 6 liitesivua
Työn ohjaaja	Yliopettaja Martti Kettunen
Toimeksiantaja	Haminan Energia Oy
Maaliskuu 2014	
Avainsanat	DHCPv6, ICMPv6, IPv6, Linux, Virtualisointi

Pidemmän aikaa on jo käyty keskustelua siitä, miten pahenevaan IPv4-osoitepulaan tulisi reagoida. Pidennettäisiinkö IPv4:n elinikää keinotekoisin menetelmin, esimerkiksi osoitteenmuunnostekniikoiden avulla, jotta uuteen IPv6:een siirtymistä voitaisiin lykätä? Yritettäisiinkö molempia protokollia ajaa päällekkäin, ja sen jälkeen vähän kerrallaan siirtyä IPv6:n käyttöön? Siirtymän helpottamiseksi DHCP mahdollistettiin myös IPv6:ssa. DHCPv6 on edeltäjäänsä paljon hienostuneempi, ja tukee IPv6:tta.

Tämän työn tarkoituksena oli toteuttaa Haminan Energia Oy:lle Linux-pohjainen, virtualisoitu DHCPv6-palvelin, ja tutkia sen jälkeen asiakasreitittimien IPv6-ominaisuuksia. Päätavoitteena oli rakentaa käyttökelpoinen palvelin, joka olisi kykenevä jakamaan IPv6-osoitteita prefix delegation -menetelmällä, sekä konfiguroida käyttövalmiiksi vähintään yksi asiakasreititin, jossa olisi IPv6-tuki.

Palvelin toteutettiin CentOS-käyttöjärjestelmällä, jonka päälle asennettiin DHCPv6-ohjelmistoksi ISC DHCP. Testattujen asiakasreitittimien ohjelmistona käytettiin Linux-järjestelmään perustuvaa OpenWrt -levityspakettia. Vertailun vuoksi mukaan otettiin myös yksi täysin kaupallinen tuote. Palvelimen sekä asiakasreitittimien testausta varten rakennettiin oma, muista verkoista eristetty testiverkko, jossa testejä voitiin ajaa häiritsemättä tuotantoverkon toimintaa.

DSLAM-laitteiden IPv6-toiminnallisuutta ei ehditty testaamaan, sillä laitetoimittajalta ei saapunut ajoissa tarkoitukseen sopivaa ohjelmistoa. Muut työlle asetetut tavoitteet kuitenkin täyttyivät, ja DHCPv6-palvelin saatiin toteutettua varatun ajan puitteissa.

ABSTRACT

KYMENLAAKSON AMMATTIKORKEAKOULU
University of Applied Sciences
Information Technology

LAMPIKARI, JAN	DHCPv6 Server in a Service Provider Environment
Bachelor's Thesis	50 pages + 6 pages of appendices
Supervisor	Martti Kettunen, Principal Lecturer
Commissioned by	Haminan Energia Oy
March 2014	
Keywords	DHCPv6, ICMPv6, IPv6, Linux, Virtualization

For quite a while there have been discussions about how to react to the worsening IPv4 address shortage. Should the life span of IPv4 be extended with artificial actions in order to postpone the transition to IPv6, such as address translation techniques, or should both protocols be run simultaneously in a dual-stack configuration removing the necessity to rush IPv6 migration? To help the transition, DHCP was made available in IPv6 as well. DHCPv6 is a much more sophisticated DHCP with IPv6 support.

The purpose of this study was to build a virtualized, Linux-based DHCPv6 server for Haminan Energia Oy, and then explore the IPv6 features of different customer routers. The main objectives were to build a fully working server with prefix delegation support, and configure at least one IPv6 capable customer router ready for market.

The server was implemented using CentOS operating system and ISC DHCP software. The tested customer routers had a Linux-based OpenWrt distribution package as their operating system. For comparison, one fully commercial product was also tested. A dedicated and fully isolated test network was built in order to run different scenarios without interfering with the production network.

The IPv6 functionality of DSLAM devices could not be evaluated as the device vendor was unable to deliver a suitable software for that purpose on time. All the other objectives were achieved, and the DHCPv6 server implementation was successful within the allotted time frame.

SISÄLLYS

LYHENNELUETTELO

ARP	Address Resolution Protocol, IPv4:n käyttämä protokolla siirtokerroksen MAC-osoitteiden selvittämiseen.
CIDR	Classless Inter-Domain Routing, reitityksessä käytettävä tapa kirjoittaa osoitemuoto, esim. 2001:DB8/16 tai 192.168.0.1/24.
DAD	Duplicate Address Detection, Neighbor Discovery -protokollan toiminnallisuus, joka estää kahden saman IPv6-osoitteen käytön jonkin linkin alueella.
DHCP	Dynamic Host Configuration Protocol, automaattinen IP-osoitteiden konfigurointiprotokolla verkkoon kytkeytyville laitteille.
DHCPv6	Dynamic Host Configuration Protocol version 6, automaattinen IPv6-osoitteiden konfigurointiprotokolla verkkoon kytkeytyville laitteille.
DNS	Domain Name System, nimipalvelujärjestelmä, joka muuntaa automaattisesti web-osoitteita IP-osoitteiksi, ja myös toisinpäin.
DNSMASQ	Kevyt ohjelmisto DNS ja DHCP palveluiden tarjoamiseen pieniin verkkoihin.
DSLAM	Digital Subscriber Line Access Multiplexer, on laite, jolla tilaajaliittymät yhdistetään operaattorin runkoverkkoon.
ESXI	Virtuaalikoneiden hallintaan käytetty järjestelmä.
IANA	The Internet Assigned Numbers Authority, globaalilla tasolla Internet-protokollaan liittyvien resurssien kuten IP-osoitealueiden jakoa hallitseva järjestö.

ICMP	Internet Control Message Protocol, verkon hallintaan käytettävä protokolla, jonka avulla esimerkiksi virheviestit välitetään eteenpäin.
ICMPv6	IPv6:n käyttämä kehittyneempi versio ICMP:stä.
IGMP	Internet Group Management Protocol, IPv4:ssä multicast-ryhmäjäsenyyksien hallinointiin käytetty protokolla.
IETF	Internet Engineering Task Force, organisaatio, joka vastaa Internet-protokollien kehityksestä ja standardoinnista.
IPng	Internet Protocol - Next Generation, hanke, jossa alettiin tutkia ratkaisuja IPv4:n korvaamiselle.
IPv4	Internet Protocol version 4; Internet-protokollan versio 4.
IPv6	Internet Protocol version 6; Internet-protokollan versio 6.
ISATAP	Intra-Site Automatic Tunnel Addressing Protocol, protokolla, jonka tarkoituksena on helpoittaa IPv6:een siirtymistä.
ISC	Internet Systems Consortium, voittoa tavoittelematon järjestö, joka vastaa muun muassa BIND sekä ISC DHCP -ohjelmistojen kehityksestä.
IT	Information Technology; Tietotekniikka.
LAN	Local Area Network, lähiverkko, jossa verkon laitteet on rajattu maantieteellisesti pienelle alueelle, esimerkiksi kotiin tai toimistoon.
LLMNR	Link-Local Multicast Name Resolution, protokolla, jonka avulla sekä IPv4 että IPv6 -isäntäkoneet voivat suorittaa nimikyselyjä.

MLD	Multicast Listener Discovery, IGMP:n korvaaja IPv6:lle.
MTU	Maximum Transmission Unit, suurin datayksikkö, joka voidaan välittää eteenpäin.
NAT	Network Address Translation on osoitteenmuunnostekniikka, jolla voidaan muuntaa IP-osoitteita toisiksi IP-osoitteiksi.
ND	Neighbor Discovery on protokolla, joka muodostaa laitteiden välisiä naapuruussuhteita.
NS	Neighbor Solicitation on tapa laitteelle pyytää toisen laitteen siirtokerroksen osoitetta, jonka IPv6-osoite tiedetään.
NTP	Network Time Protocol, verkon laitteiden kellojen synkronointiin käytettävä protokolla.
NUD	Neighbor Unreachability Detection, ND-protokollan ominaisuus, joka pitää kirjaa aktiivisista ja saatavilla olevista yhteyksistä
Ping	Packet Internet Groper on TCP/IP:n työkalu, jolla voidaan selvittää, onko jokin tietty isäntä saatavilla verkon alueella.
RA	Router Advertisement on reitittimien tapa kertoa samassa verkossa sijaitseville isäntäkoneille, että niiden kautta on osoitteiden konfigurointiin liittyvää tietoa saatavilla.
RADVD	Router Advertisement Daemon on RA-viestejä lähettävä palvelu.
RAM	Random Access Memory on keskusmuisti, jota käytetään esimerkiksi tietokoneissa.

RIPE NCC	RIPE Network Coordination Centre on yksi viidestä alueellisesta RIR:stä, joka vastaa Internet-resurssien allokoinnista.
RIR	Regional Internet Registry on järjestö, joka vastaa eri Internet-resurssien, kuten IP-osoitteiden allokoinnista.
RS	Router Solicitation on isäntäkoneiden tapa pyytää reitittimiltä RA-viestiä, jotta ne pystyisivät konfiguroimaan itselleen osoitetietoja.
SSH	Secure Shell on protokolla, jonka avulla esimerkiksi eri laitteisiin muodostettavien etäyhteyksien tietoturvaa voidaan parantaa.
TCP	Transmission Control Protocol mahdollistaa yhdessä IP:n kanssa tietokoneiden välisen kommunikoinnin.
TTL	Time To Live on IPv4-paketin otsikkokenttä, joka kertoo paketin elinajan.
UDP	User Datagram Protocol on tietokoneiden väliseen viestintään käytetty protokolla, jossa viestejä kutsutaan datagrammeiksi.
ULA	Unique-Local Address, IPv6:n vastine IPv4:ssä käytetyille privaateille osoitteille.
VLAN	Virtual Local Area Network on virtuaalinen lähiverkko, jossa layer 2 -tasolla verkon laitteet voidaan eristää toisistaan.
VM	Virtual Machine; Virtuaalikone.

WAN	Wide Area Network on laajaverkko, jossa verkon laitteet ovat maantieteellisesti kaukana toisistaan, kuten esimerkiksi operaattoriverkossa.

1 JOHDANTO

IPv4-osoitteet loppuvat kesken ja tilalle tuodaan kovaa vauhtia uusia 128-bittisiä IPv6-osoitteita. Uusien osoitteiden käyttöönotto ei kuitenkaan ole ollut täysin ongelmatonta, ja näin siirtymäaika uusien osoitteiden käyttöönottoon on koko ajan venynyt pidemmäksi. IPv4:lle kehitetään edelleen jatkuvasti uusia ratkaisuja, joilla sen elinaikaa voitaisiin pidentää ja IPv6:n käytön aloittamista voitaisiin lykätä. NAT:n (Network Address Translation) käyttö on yleistynyt kaikissa verkoissa, ja oikeiden globaalien osoitteiden saaminen päätelaitteille vaikeutuu kokoajan. IPv4-maailmassa DHCP (Dynamic Host Configuration Protocol) -palvelun olemassaolo on lähes elintärkeää, mutta IPv6:ssa DHCP:lle on kehitetty korvaavia toimintoja. Saman linkin alueella sijaitsevat päätelaitteet voivat muun muassa automaattisesti konfiguroida itselleen IPv6-osoitteet, jos vain samalla linkillä sijaitsee Router Advertisement -viestejä lähettävä reititin. DHCP-palvelua ei siis välttämättä tarvita. IPv4:n DHCP-palveluun tottuneena vastaavaa hallittavuutta kuitenkin on alettu kaipaamaan myös IPv6:n kanssa. Internet-palveluntarjoajille DHCP-palvelu on kuitenkin pakollinen, sillä muuten asiakkaiden hallinasta tulisi lähes mahdotonta. Osoitteita on jaettava kontrolloidusti, jotta aina tiedetään, kenelle mikäkin osoite kuuluu ja kuinka paljon osoitteita on jaettuna. Tähän tarkoitukseen kehitettiin seuraaja DHCPv6, seuraaja DHCP:lle.

Haminan Energia Oy:ssa ei ollut vielä aloitettu IPv6:een valmistautumista, mutta tulevaisuuden kannalta ensimmäinen kehitysaskel kohti IPv6:tta haluttiin ottaa. Tätä kautta löytyi myös sopiva aihe, Linux-pohjainen DHCPv6-palvelin, jonka toteuttaminen olisi mahdollista niin resurssien kuin käytettävissä olevan ajankin puitteissa. Työ toteutettiin Haminan Energia Oy:n tiloissa tiiviissä yhteistyössä tietoliikenneosaston kanssa. Työn tavoitteena oli toteuttaa DHCPv6-palvelin, joka kykenisi jakamaan IPv6-osoitteita prefix delegation -menetelmää käyttäen erikseen testatuille IPv6-asiakasreitittimille. Prefix delegation -menetelmällä DHCPv6-palvelin jakaa IPv6-osoitealueen asiakasreitittimen LAN-portille, josta reititin alkaa jakaa sitä eteenpäin siihen kytkeytyneille laitteille. Keskeisimmäksi tutkimusongelmaksi muodostui

IPv4:n tarjoaman optio 82 -kentän toiminnallisuuden säilyttäminen IPv6:ssa, jotta myös jokaisen jaetun IPv6-osoitteen takana oleva asiakas pystyttäisiin tunnistamaan.

Työmäärän pitämiseksi kohtuullisena, rajattiin työ ainoastaan DHCPv6-palvelimen rakentamiseen sekä asiakaslaitteiden testaamiseen. IPv6:n käyttöönottoon liittyvien toimenpiteiden, kuten runkoverkkoon tehtävien reititysten ja nimipalvelimen rakentaminen olisi kasvattanut työmäärää valtavasti, joten ne jätettiin työn ulkopuolelle kokonaan. Käytännön toteutus suoritettiin kolmessa vaiheessa. Ensin selvitettiin, miten palvelin tulisi toteuttaa. Sen jälkeen palvelin kasattiin valmiiksi, ja lopuksi palvelimen toimintaa testattiin erilaisin menetelmin.

2 IPV6-PROTOKOLLA

IPv6 eli Internet Protokollan versio 6 on kehitetty versio edeltäjästään IPv4:stä. IPv6 kehitettiin korjaamaan IPv4:ssä esiintyneet puutteet ja rajoitukset sekä vastaamaan tuleviin tietoverkkojen haasteisiin ja vaatimuksiin. IP yhdessä TCP:n (Transmission Control Protocol) kanssa muodostaa TCP/IP:n, joka pitää sisällään suuren määrän eri protokollia mahdollistaen tietokoneiden välisen kommunikoinnin. Jokainen laite, joka käyttää TCP/IP:tä, tarvitsee itselleen uniikin osoitteen, jotta se voidaan tunnistaa ja erottaa muista tietoverkon laitteista. IP määrittää nämä osoitteet, ja IP:n avulla myös TCP/IP pystyy lähettämään IP-paketteja laitteesta toiseen. Tämän seurauksena mikä tahansa laite, jolla on IP-osoite, voi liittyä TCP/IP -verkkoon ja alkaa lähettää sekä vastaanottaa IP-paketteja. (Hagen 2006, 4; Odom 2012, 23 ja 31-32; Teare 2010, 691.)

2.1 IPv6:n historia

1990-luvun alussa IETF (The Internet Engineering Task Force) alkoi kehittää seuraajaa IPv4:lle. Jo silloin huomattiin, että IPv4-osoitteita on vain suhteellisen rajallinen määrä, ja tämä ongelma piti ratkaista. Vuonna 1993 IETF aloitti IPng (Internet Protocol - Next Generation) hankkeen tutkiakseen erilaisia ratkaisuja IPv4:n korvaamiselle. (Hagen 2006, 3.)

Vuonna 1994 IETF suositteli uuden protokollan eli IPv6:n luomista. Tätä varten perustettiin erillinen ALE (Address Lifetime Expectation) työryhmä, jonka tehtävänä oli tutkia, voitaisiinko IPv4:n elinaikana ehtiä kehittää uusilla ominaisuuksilla toimiva protokolla, vai riittäisikö aika ainoastaan osoitepulan ratkaisemiseen. Siihen aikaan

tehtiin myös ennuste, että IPv4-osoitteet loppuisivat vuosien 2005 ja 2011 välisenä aikana. (Hagen 2006, 3.)

2.2 IPv6:n tarve

IANA (Internet Assigned Numbers Authority) jakoi viimeiset vapaana olevat IPv4-osoitealueensa 3. helmikuuta 2011 RIR:ien (Regional Internet Registry), eli eri maaosien IP -osoitteiden jakelusta vastaavien järjestöjen, käyttöön. Näitä järjestöjä on ympäri maapalloa viisi kappaletta, ja Euroopan alueella näistä viidestä toimii RIPE NCC -niminen järjestö. 14. helmikuuta 2012 RIPE NCC alkoi jakaa IPv4-osoitteita sen viimeisestä osoiteavaruudesta, joka sillä on enää jäljellä. (ARIN 2011; RIPE NCC 2012.)

Verkkoon liitettävien laitteiden määrän kasvaessa koko ajan uudet IPv6-osoitteet tulevat tarpeeseen, sillä IPv4-osoitteiden teoreettinen maksimi on ainoastaan 4,3 miljardia osoitetta. (Hagen 2006, 5.)

2.3 IPv6-osoitealue

IPv6-osoitteet ovat 128-bittisiä, eli niillä voidaan muodostaa yhteensä 2^{128} erilaista osoitetta. IPv6 käyttää desimaalijärjestelmän sijaan heksadesimaalijärjestelmää, sillä heksadesimaalilukuja on helpompi muuntaa binääriluvuiksi. IPv6-osoite on jaettu kahdeksaan 16- bitin heksadesimaaliryhmään, ja nämä ryhmät on eroteltu toisistaan kaksoispisteellä. Heksadesimaaliluvut välillä A - F voidaan kirjoittaa joko isolla tai pienellä kirjaimella. IPv6-osoite näyttää kirjoitettuna kokonaisuudessaan seuraavanlaiselta:

2001:DB8:0000:0000:1234:5678:ABBA:BEEF

Osoitteita voidaan myös kirjoittaa lyhennetyssä muodossa. Niitä voidaan lyhentää yhdistämällä yhdestä 16-bitin ryhmästä peräkkäiset nollat yhdeksi nollaksi, ja jos näitä 16-bitin ryhmiä, jossa kaikki numerot ovat nollia on peräkkäin, voidaan ne jättää kirjoittamatta kokonaan ja korvata kahdella kaksoispisteellä. Kaksi kaksoispistettä peräkkäin ei voi kuitenkaan esiintyä IPv6-osoitteessa kuin kerran, sillä tietokoneet käsittelevät osoitteita aina täyden 128 bitin osoitteina, vaikka osoite olisikin lyhennetyssä muodossa. Lyhennettynä aiemmin esitetty IPv6-osoite näyttäisi tältä:

2001:DB8::1234:5678:ABBA:BEEF

(Davies 2012, 59; Hagen 2006, 36-38; Teare 2010, 700.)

2.4 IPv6-osoitteiden prefixit

IPv6:ssa prefixejä käytetään ilmoittamaan aliverkon koko sekä antamaan verkkotieto reititystä varten. Prefixin pituus kertoo aliverkon peitteen. Prefixejä käytetään liityntäporttien tunnistamiseen aliverkoissa ja reitittimet käyttävät niitä reitittämiseen. Prefixeissä käytetään CIDR (Classless Inter-Domain Routing) merkintätapaa, joka on useasti käytössä myös aliverkotetuilla IPv4-osoitteilla. IPv6 prefix kirjoitetaan muodossa IPv6-osoite/prefixin pituus. Esimerkiksi 2001:DB8:1234:5678::/64 on aliverkon prefix ja 2001:DB8:12::/48 on summattu reitin prefix. (Davies 2012, 60; Hagen 2006, 38; Teare 2010, 701.)

2.5 IPv6-osoitteiden eri tyypit

IPv4:ssä käytettiin unicast-, broadcast- sekä multicast-osoitteita. IPv6:ssa broadcast-osoitteista on luovuttu ja niiden tilalla käytetään multicast-osoitteita. Uutena osoitetyyppinä IPv6:ssa on otettu käyttöön RFC 1546:ssa määritelty anycast-osoite. IPv6-osoitteet voidaankin luokitella kolmeen päätyyppiin, unicast-, multicast- sekä anycast-osoitteisiin. (Hagen 2006, 36; Teare 2010, 704.)

2.5.1 Unicast-osoitteet

IPv6:ssa unicast-osoitteita on montaa eri tyyppiä, joita ovat muun muassa global unicast, link-local, unique-local, site-local sekä embedded IPv4. Site-local-osoitteet ovat kuitenkin jo vanhentuneita, eikä niitä ei enää tueta. Unicast-osoitteen avulla voidaan yksilöidä jokaisen IPv6-laitteen (node) liityntäportti (interface). Nämä osoitteet toimivat samalla tavalla kuin IPv4:n unicast-osoitteet. Paketti, joka on lähetetty unicast-osoitteeseen kulkeutuu juuri siihen liityntäporttiin, jolle kyseinen unicast-osoite kuuluu. Unicast-osoitteet käsittävät koko IPv6-osoitealueen lukuunottamatta multicast-osoitteille varattua aluetta. (Hagen 2006, 36; Teare 2010, 704 ja 708.)

Interface Identifier

IPv6:ssa linkillä tarkoitetaan siirtotietä, jonka välityksellä verkon laitteet keskustelevat keskenään käyttämällä siirtokerrosta (link layer). Interface ID:tä käytetään yksittäisten liityntäporttien tunnistamiseen eri linkkien välillä. Tätä voidaan myös verrata IPv4-osoitteista tuttuun host-osaan. Nämä ID:t ovat aina 64-bittisiä ja ne voidaan luoda dynaamisesti siirtokerroksen (data link layer) osoitteesta. (Teare 2010, 701 ja 704.)

Tyypillisillä liityntäporteilla Interface Identifier muodostetaan IEEE EUI-64 proseduurilla asettamalla kirjainjono fffe keskelle 48-bittistä MAC (Media Access Control) -osoitetta sekä antamalla MAC-osoitteen ensimmäisen tavun seitsemänneksi vasemmanpuoleiselle bitille arvo 1. Tämä uusi 64-bittinen osoite on Interface Identifier ja sitä käytetään IPv6-osoitteen host-osana. Vasemmanpuoleiset 64 bittiä muodostavat verkon prefixin. (Blanchet 2006, 80-21; RFC 2373.)

Global Unicast -osoitteet

IPv6 global unicast -osoitteet vastaavat IPv4:n julkisia osoitteita. Ne ovat globaalisti reititettävissä ja tavoitettavissa IPv6-verkossa. RFC 4291 -standardi määrittelee globaalien osoitteiden kattavan kaiken muun paitsi määrittämättömät osoitteet, loopback-osoitteet, link-local -osoitteet sekä multicast-osoitteet. Global unicast -osoite koostuu kolmesta kentästä, 45-bittisestä global routing prefixistä, 16-bittisestä subnet ID:stä ja 64-bittisestä Interface ID:stä. Global routing prefix kertoo kenelle kyseinen osoitealue on varattu. Palveluntarjoajat ja muut ylemmät tahot määräävät tämän osan osoitteesta, ja sillä on hierarkkinen rakenne. Subnet ID:n avulla voidaan tunnistaa aliverkot, ja interface ID kertoo jonkin tietyn aliverkon tietyn liityntäportin. (Davies 2012, 62-63; Hagen 2006, 40.)

Link-Local Unicast -osoitteet

Link-local-osoitteilla on rajoitettu alue ja ne luodaan dynaamisesti kaikille IPv6-liityntäporteille käyttämällä erityistä link-local prefixiä FE80::/10 ja 64-bittistä interface ID:tä. Niitä käytetään osoitteiden automaattiseen konfigurointiin ja sen lisäksi niitä käyttävät neighbor discovery, router discovery sekä monet reititysprotokollat. Link-local unicast -osoitetta voidaan käyttää myös yhdistämään laitteita samassa paikallisessa verkossa ilman globaaleja osoitteita. (Teare 2010, 707.)

Unique-Local -osoitteet

Unique-local-osoitteet ovat privaatteja osoitteita eli niitä ei reititetä julkisessa IPv6 internetissä. Näitä osoitteita ei myöskään mainosteta verkon ulkopuolelle. (Davies 2012, 66-67.)

2.5.2 Multicast-osoitteet

Multicast-osoitteille on varattu alue FF00::/8. Ensimmäiset kahdeksan bittiä osoitteessa kertovat osoitteen olevan multicast, seuraavat neljä bittiä on varattu lipuille ja sitä seuraavat neljä rajaavat osoitealueen. Loput 112 bittiä osoitteesta kertovat multicast-ryhmän ID:n eli tunnisteen. Multicast-osoite yksilöi joukon IPv6-liityntäportteja. Kuten IPv4:ssäkin, multicast-osoitteeseen lähetetty paketti kulkeutuu kaikille multicast-ryhmään kuuluville jäsenille. Mikä tahansa liityntäportti voi kuulua mihin tahansa määrään multicast-ryhmiä, eli toisinsanoen yksi IPv6-laite voi kuunnella monia multicast-osoitteita samaan aikaan. Multicast-liikenne on erittäin tärkeää IPv6:ssa sillä se sisältää monia IPv6:n ydintoimintoja ja sen lisäksi korvaa aikaisemmin IPv4:ssä käytetyn broadcast-liikenteen. Multicast-osoitteet tarjoavat myös paljon tehokkaamman tavan välittää tietoa. Haluttu tieto lähetetään ainoastaan niille vastaanottajille jotka ovat erikseen ilmoittaneet haluavansa vastaanottaa sitä liittymällä kyseiseen multicast-ryhmään. RFC 2375:ssä on määritelty tietyt multicast-osoitealueet, jotka on varattu pysyvästi eri tarkoituksiin. (Davies 2012, 68; Hagen 2006, 36 ja 52-53; Teare 2010, 704-705 ja 708.)

Solicited-Node Multicast-osoite

Solicited-node multicast-osoite on multicast-osoite, johon jokaisen IPv6-laitteen täytyy kuulua. Sitä käytetään selvittämään tunnetun IPv6-osoitteen link-layer-osoite yhdessä NS (Neighbor Solicitation)-viestin kanssa. Solicited-node multicast-osoite muodostetaan prefixistä FF02::1:FF00:0/104 ja loput 24 bittiä otetaan unicast-osoitteesta. (Davies 2012, 70; Hagen 2006, 50.)

2.5.3 Anycast-osoitteet

Anycast-osoite on yksittäinen globaali unicast-osoite, joka on asetettu useammalle kuin yhdelle liityntäportille. Toisin sanoen, kun yksittäinen unicast-osoite annetaan useammalle liityntäportille, tulee siitä anycast-osoite. Osoitteen puolesta niitä ei voi erottaa unicast-osoitteista, ja näin ollen myös laitteet, joita ei ole konfiguroitu

anycastia varten tunnistavat osoitteet tavallisina unicast-osoitteina. Jos anycast-osoitteita käytetään, täytyy jokainen laite, jolla anycast-osoite on, konfiguroida tietoiseksi siitä. Anycast-osoitteet on suunniteltu tarjoamaan redundanttisuutta sekä kuormanjakoa sellaisissa tilanteissa, joissa useampi reititin tarjoaa samaa palvelua. Anycast-osoitteita käyttämällä voidaan paketti lähettää aina lähimmälle anycast-ryhmän jäsenelle. (Hagen 2006, 49; Teare 2010, 711.)

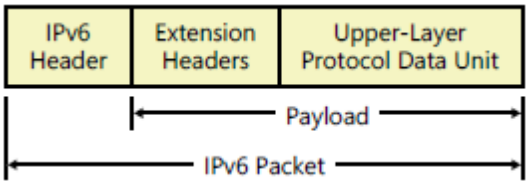
2.5.4 Muut osoitteet

IPv6-protokolla käsittää myös muutamia erikoisosoitteita. Näitä ovat muun muassa määrittämättömät osoitteet sekä loopback-osoitteet. Määrittämätön osoite on muotoa 0:0:0:0:0:0:0:0 tai :: ja vastaa IPv4:n osoitetta 0.0.0.0. Tätä osoitetta käytetään yleensä lähdeosoitteena ennen kuin uniikki osoite saadaan määritettyä. Sitä ei koskaan määritetä liityntäporttiin eikä kohdeosoitteeksi. Loopback-osoite on puolestaan muotoa 0:0:0:0:0:0:0:1 tai ::1 ja sitä käytetään loopback-liityntäporteissa sallien IPv6-laitteen lähettää paketteja itselleen. Se vastaa IPv4:n osoitetta 127.0.0.1. Loopback-osoitteeseen kohdennettuja paketteja ei saa reitittää eteenpäin IPv6-reitittimellä. (Davies 2012, 67.)

Tämän lisäksi on vielä joukko osoitteita, joiden tehtävänä on helpottaa siirtymistä IPv4:stä IPv6:een. Niihin kuuluvat IPv4-compatible-osoitteet, IPv4-mapped-osoitteet, 6to4-osoitteet, ISATAP-osoitteet sekä Teredo-osoitteet. (Davies 2012, 67.)

2.6 IPv6-paketin rakenne

IPv6-paketin rakenne on määritelty RFC 2460 -standardissa. IPv6-paketti koostuu IPv6 otsikkokentästä sekä datakuormasta. IPv6 otsikkokentän (IPv6 Header) pituus on 40 tavua ja otsikkokenttä on aina mukana IPv6-paketissa. Lisäotsikoita (Extension Headers) voi olla mukana datakuormassa useampia ja niiden pituus vaihtelee, mutta ne eivät kuitenkaan ole välttämättömiä. Ylempien kerrosten protokollien kuten esimerkiksi TCP:n (Transmission Control Protocol) ja UDP:n (User Datagram Protocol) datayksikkö (Upper-Layer Protocol Data Unit) koostuu yleensä niiden omista otsikoista ja datakuormista. IPv6:n otsikkokenttä ja lisäotsikot korvaavat IPv4:ssä esiintyvän otsikkokentän ja optiot. (Davies 2012, 91-92.)



Kuva 1. IPv6-paketin rakenne. (Davies 2012, 91.)

2.7 IPv6-otsikon rakenne

IPv6:n otsikko on kevennetty versio IPv4:n otsikosta ja se on määritelty RFC 2460 -standardissa. Otsikon pituus on asetettu kiinteästi 40:n tavun mittaiseksi. Lähdeosoitteelle ja kohdeosoitteelle varatut kentät vievät molemmat 16 tavua otsikolle varatusta tilasta, joten jäljelle jää ainoastaan 8 tavua yleisille otsikkotiedoille. Sellaiset kentät, joita käytetään hyvin harvoin tai ei ollenkaan, on poistettu, ja tilalle on tehty kenttä, joka tarjoaa paremman tuen reaaliaikaiselle liikenteelle. (Davies 2012, 93; Hagen 2006, 17.)

IPv6:n otsikossa on viisi kenttää vähemmän kuin IPv4:n otsikossa. Pois jääneet kentät ovat otsikon pituus (Header Length), fragmenttitunnus (Identification), liput (Flags), fragmentin paikka (Fragment Offset) ja otsikon tarkistussumma (Header Checksum). Otsikon pituus -kenttä poistettiin, koska sitä ei tarvittu enää otsikon kanssa, jossa se on määritelty kiinteäksi. IPv4:ssä puolestaan otsikon pituus saattaa vaihdella 20:sta tavusta 60:een tavuun eri optioiden kanssa, joten tieto otsikon pituudesta on tärkeää. (Hagen 2006, 17.)

2.8 IPv6-otsikkokenttien sisältö

Version (Versio)

4 bittiä pitkä versio-kenttä kertoo käytetyn IP-version. IPv6:lla tälle kentälle on asetettu arvo 6. (Davies 2012, 95; Hagen 2006, 19.)

Traffic Class (Luokkakenttä)

Luokkakenttä määrittelee IPv6-paketin luokan tai prioriteetin ja on 8 bittiä pitkä. Tämä kenttä tarjoaa vastaavan toiminnallisuuden kuin IPv4:n palveluluokkakenttä (Type Of Service). Se helpottaa reaaliaikaisen sekä kaiken muun erikoiskohtelua vaativan

datan käsittelyä tarjoten reitittimille tietoa IPv6-pakettien luokista ja prioriteeteista. (Davies 2012, 95; Hagen 2006, 19.)

Flow Label (Vuon tunniste)

Vuon tunniste kertoo, että jokin paketti kuuluu johonkin tiettyyn pakettisarjaan. Tämän kentän pituus on 20 bittiä ja sitä käytetään priorisoimaan liikennettä, esimerkiksi reaaliaikaisen äänen ja videon siirtoa. (Davies 2012, 95; Hagen 2006, 19.)

Payload Length (Kuorman pituus)

Tämä kenttä on 16 bittiä pitkä ja sillä kerrotaan kuorman pituus. IPv6:ssa tämä lasketaan eri tavalla kuin IPv4:ssä. (Davies 2012, 95; Hagen 2006, 20.)

Next Header (Seuraava otsikko)

Seuraava otsikko -kenttä kertoo joko ensimmäisen lisäotsikon tyypin tai ylemmän kerroksen PDU:n protokollan, ja on 8 bittiä pitkä. (Davies 2012, 95.)

Hop Limit (Elinaika)

Elinaika -kenttä määrittää linkkien maksimimäärän joita IPv6-paketti voi kulkea, ennen kuin se hylätään. Kentän pituus on 8 bittiä. IPv4:ssä vastaava kenttä on TTL (Time To Live). (Davies 2012, 95.)

Source Address (Lähdeosoite)

Tämän kentän pituus on 128 bittiä ja se ilmoittaa paketin alkuperäisen lähteen IPv6-osoitteen. (Davies 2012, 95; Hagen 2006, 22.)

Destination Address (Kohdeosoite)

Kohdeosoite -kentän pituus on myös 128 bittiä, ja sillä ilmaistaan paketin määränpään IPv6-osoite. (Davies 2012, 96; Hagen 2006, 22.)

Kuva 2. IPv6-otsikon rakenne. (Davies 2012, 94.)

2.9 IPv6:n lisäotsikot

IPv4:ssä otsikkokenttä sisältää kaikki optiot ja näin jokaisen reitittimen täytyy tarkistaa niiden olemassaolo ja tarvittaessa prosessoida ne. IPv6:ssa reititysoptiot on siirretty lisäotsikoiden alle (Extension Headers). Ainoa lisäotsikko, jonka IPv6-reitittimet joutuvat prosessoimaan on Hop-by-Hop -optio. Tämä nopeuttaa otsikon prosessointia ja näin olleen parantaa myös IPv6-pakettien reititysnopeutta. RFC 2460 määrittelee kaikki lisäotsikot, joita jokaisen IPv6-laitteen täytyy tukea. Näitä lisäotsikkoja ovat hyppyoptio-otsikko (Hop-by-Hop Options header), kohdeoptio-otsikko (Destination Options header), reititysotsikko (Routing header), lohkomisotsikko (Fragment header), todennusotsikko (Authentication header) ja salausotsikko (Encapsulating Security Payload header). (Davies 2012, 99.)

3 ICMPV6

IPv6 käyttää päivitettyä versiota ICMP:stä (Internet Control Message Protocol), ICMPv6:tta, ja se on määritelty RFC 4443 -standardissa. ICMPv6 on paljon tehokkaampi kuin edeltäjänsä ja se sisältää uusia toiminnallisuuksia. Päätarkoitus on kuitenkin sama, ja se tarjoaa eri toimintoja ilmoittamaan toimitus- sekä reititysvirheistä. Samalla siitä löytyy myös yksinkertainen echo-toiminto vianhakua varten. (Davies 2012, 117; Hagen 2006, 60.)

ICMPv6-viestit voidaan luokitella kahteen eri luokkaan, virheviesteihin ja informaatioviesteihin. Virheviestit ilmoittavat reititys- sekä toimitusvirheistä ja ne voidaan luokitella useampaan eri alaluokkaan. Informaatioviestit tarjoavat puolestaan yksinkertaisen diagnostiikkaominaisuuden vianhakua varten ja niitä on kahta eri tyyppiä; Echo Request ja Echo Reply. Näitä kahta viestiä käytetään yhdessä yleisimmistä TCP/IP:n

työkaluista, pingissä (Packet Internet Groper). Pingin avulla voidaan selvittää onko jokin tietty isäntä tavoitettavissa verkon alueella. Myös ND (Neighbor Discovery) sekä MLD (Multicast Listener Discovery) käyttävät ICMPv6:n informaatioviestejä hyväkseen. (Davies 2012, 119 ja 124; Hagen 2006, 60 ja 69.)

3.1.1 Neighbor Discovery

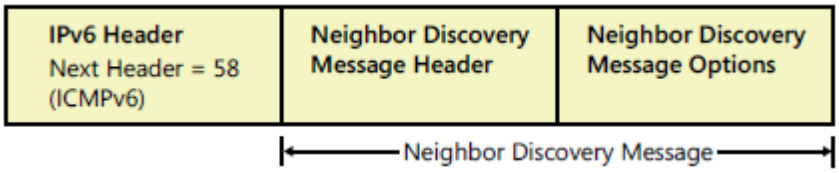
Neighbor Discovery -protokolla (NDP) muodostaa laitteiden välisiä naapuruussuhteita. NDP korvaa IPv4:ssä käytetyt ARP:n (Address Resolution Protocol), ICMP:n (Internet Control Message Protocol) router discovery:n sekä ICMP:n uudelleenohjausviestin (redirect message). NDP tarjoaa myös muita lisätoiminnallisuuksia, kuten NUD:n (Neighbor Unreachability Detection) sekä DAD:n (Duplicate Address Detection). NUD pitää kirjaa aktiivisista ja saatavilla olevista yhteyksistä. DAD puolestaan havaitsee, mikäli samalla linkillä on käytetty samaa IPv6-osoitetta kahteen kertaan. (Davies 2012, 131 ja 167; Hagen 2006, 73 ja 85.)

3.1.2 Neighbor Discovery -prosessit

Router discovery -prosessin aikana isäntä (host) selvittää oman linkin reitittimet, ja on vastaava kuin ICMPv4:n router discovery. Prefix discovery -prosessilla isännät selvittävät paikallisen linkin alueen verkkojen prefixit. Parameter discovery sallii isäntien selvittää muita lisäparametreja, muun muassa linkin MTU-arvon sekä sallittujen hyppyjen määrän. Address autoconfiguration -prosessin aikana IPv6-osoitteet konfiguroidaan liityntäporteille joko jonkin osoitepalvelimen kuten DHCPv6-palvelimen (Dynamic Host Configuration Protocol for IPv6) kanssa, tai ilman. Address resolution vastaa IPv4:n ARP:aa (Address Resolution Protocol), ja sen avulla selvitetään, mikä linkkikerroksen osoite vastaa mitäkin IPv6-osoitetta. Next-hop determination selvittää sen naapurin IPv6-osoitteen, jolle pakettia ollaan reitittämässä. Neighbor unreachability detection -prosessilla laite pystyy kertomaan että naapurin IPv6-kerros ei enää vastaanota paketteja. Duplicate address detection selvittää, onko jokin osoite käytössä jo naapurilaitteella. Redirect function ilmoittaa isännälle aina paremman reitin IPv6-osoitteesta saavuttaakseen kohteen, ja se on vastaava kuin ICMPv4:n redirect message. (Davies 2012, 132-133.)

3.1.3 Neighbor Discovery -viestit

ND-viestit käyttävät ICMPv6-viestirakennetta, ja ne koostuvat viidestä ICMP-viestistä. Näitä viestejä ovat Router Solicitation / Router Advertisment -viestipari, Neighbor Solicitation / Neighbor Advertisement -viestipari, sekä ICMP Redirect -viesti. (Davies 2012, 133; Hagen 2006, 75.)



Kuva 3. ND-viestin rakenne. (Davies 2012, 133.)

Neighbor Solicitation

Neighbor Solicitation on laitteelle tapa pyytää toisen laitteen siirtokerroksen osoitetta, jonka IPv6-osoite tiedetään. Näin mahdollistetaan näiden kahden laitteen välinen kommunikointi. IPv4:ssä vastaava protokolla on ARP (Address Resolution Protocol). (Blanchet 2006, 115.)

Kun laite haluaa lähettää datagrammin naapurilleen saman linkin alueella, se lähettää Neighbor Solicitation -viestin Solicited-Node Multicast -osoitteeseen. Kaikki tätä osoitetta kuuntelevat laitteet huomaavat viestin, mutta ainoastaan todellinen kohde vastaa tähän, koska viesti pitää sisällään tämän kohteen IPv6-osoitteen. (Blanchet 2006, 116.)

Neighbor Advertisement

Neighbor Advertisement -viesti lähetetään vastauksena Neighbor Solicitation -viestiin. Vastaus lähetetään suoraan Neighbor Solicitation -viestin lähettäneelle laitteelle, ja viestiin sisällytetään Neighbor Solicitation -viestillä pyydetty linkkikerroksen osoite. (Blanchet 2006, 116.)

Router Advertisement

Reitittimet lähettävät Router Advertisement -viestejä, tai lyhyemmin RA-viestejä, isäntäkoneille, jotka pitävät sisällään tietoa linkin prefixistä sekä oletusreitittimestä.

Näin isäntäkoneet voivat automaattisesti konfiguroida itsensä. RA-viestejä lähetetään satunnaisin väliajoin, jotta vältyttäisiin samanaikaisilta viestien lähetyksiltä, jos samalla linkillä on useampi reititin. Oletuksena tämä väli vaihtelee 200:n ja 600:n sekunnin välillä. (Blanchet 2006, 81.)

RA-viestin lähdeosoitteeksi merkitään lähettävän reitittimen liityntäportin link-local-osoite, ja kohdeosoitteeksi merkitään multicast-osoite FF02::1. Jollei toisin määritellä, reitittimet lähettävät RA-viestejä kaikista liityntäporteistaan. Kaikki laitteet, jotka on määritelty automaattisesti konfiguroimaan itsensä, kuuntelevat RA-viestejä, ja prosessoivat ne välittömästi sellaisen saadessaan. (Blanchet 2006, 82.)

Router Solicitation

RA-viesti saattaa olla lähetetty juuri ennen kuin jokin laite on ehtinyt käynnistyä, ja näin ollen se joutuisi odottamaan seuraavaa RA-viestiä, jotta se voisi konfiguroida itsensä. Jotta laite saisi itselleen välittömästi käynnistymisen aikana RA-viestin, se lähettää käynnistyessään Router Solicitation -viestin (RS) kaikille reitittimille sen linkin alueella. Tällä viestillä se pyytää reitittimiä lähettämään heti uuden RA-viestin, jotta tämä laite voisi konfiguroida itsensä odottamatta. (Blanchet 2006, 83-84.)

Laite asettaa RS-viestin lähdeosoitteeksi oletuksena sen oman link-local-osoitteen. Jos link-local-osoitetta ei voida käyttää, käytetään sen sijaan määrittelemätöntä osoitetta ::. (Blanchet 2006, 84.)

3.2 Address Auto-Configuration

Yksi IPv6:n hyödyllisimmistä ominaisuuksista on sen kyky konfiguroida itsensä automaattisesti ilman osoitteen konfigurointi protokollaa, kuten DHCPv6:tta. IPv6-isäntäkone voi automaattisesti konfiguroida itselleen link-local-osoitteen jokaiselle liityntäportilleen. Router discovery -ominaisuuden avulla isäntäkone voi selvittää naapurireitittimien osoitteet, linkin prefixin sekä muita konfigurointi parametreja. (Davies 2012, 205.)

3.2.1 Auto-Configuration -tyypit

Auto-Configuration -tyyppejä on kolme erilaista, ja jokaisessa niissä link-local-osoite konfiguroidaan automaattisesti. (Davies 2012, 205-206.)

Stateless (Tilaton)

Tässä osoitteiden sekä muiden asetusten konfigurointi perustuu Router Advertisement -viestejen vastaanottamiseen. Näissä viesteissä on Managed Address Configuration -lippu sekä Other Stateful Configuration -lippu asetettu arvoon 0, ja ne sisältävät yhden tai useamman Prefix Information -option 64-bittisillä prefixeillä. (Davies 2012, 205.)

Stateful (Tilallinen)

Tässä konfigurointi pohjautuu osoitteen konfigurointi protokollan, esimerkiksi DHCPv6:n, käyttöön. Sen avulla voidaan saada osoitteita sekä muita konfigurointi asetuksia. Isäntäkone käyttää tilatonta auto-configuration -tyyppiä, jos se saa Router Advertisement -viestin jossa ei ole Prefix Information -optioita, tai jos Managed Address Configuration -tai Other Stateful Configuration -lippu on asetettu arvoon 1. Isäntäkone voi käyttää tilallista auto-configuration -menetelmää myös, jos paikallisella linkillä ei ole yhtään reititintä. (Davies 2012, 206.)

Stateless ja Stateful

Molempien yhdistelmässä konfigurointi perustuu myös Router Advertisement -viestien vastaanottamiseen, jotka sisältävät Prefix Information -optioita, ja joissa joko Managed Address Configuration -lippu tai Other Stateful Configuration -lippu on asetettu arvoon 1. (Davies 2012, 206.)

4 DHCPV6

DHCPv6 on määritelty RFC 3315 -standardissa, ja se mahdollistaa tilalliset (stateful) sekä tilattomat (stateless) osoitekonfiguraatiot IPv6-isäntäkoneille. IPv6-isäntäkone käyttää DHCPv6:tta lippujen perusteella, jotka naapurireititin on sisällyttänyt Router Advertisement -viestiin. (Davies 2012, 210.)

4.1.1 Managed Address Configuration -lippu

Managed Address Configuration -lippu, tai lyhennettynä M-lippu, asetettuna arvoon 1, kertoo isäntäkonetta käyttämään konfigurointi protokollaa saadakseen tilallisia osoitteita. (Davies 2012, 210.)

4.1.2 Other Stateful Configuration -lippu

Other Stateful Configuration -lippu, tai lyhennettynä O-lippu, asetettuna arvoon 1, ohjeistaa isäntäkonetta käyttämään konfigurointi protokollaa saadakseen muita konfigurointi asetuksia. (Davies 2012, 211.)

4.1.3 Managed Address Configuration -lipun sekä Other Stateful Configuration -lipun yhdistelmät

Kun M-lippu ja O-lippu on molemmat asetettu arvoon 0, verkossa ei ole DHCPv6-palvelinta ollenkaan. Isäntäkoneet käyttävät Router Advertisement -viestejä muiden kuin link-local-osoitteiden konfigurointiin, ja muita keinoja kuten manuaalista konfigurointia muiden asetusten konfigurointiin. (Davies 2012, 211.)

Jos M-lippu sekä O-lippu on molemmat asetettu arvoon 1, käytetään DHCPv6-palvelinta sekä osoitteiden konfigurointiin että muiden asetusten konfigurointiin. Tätä kutsutaan myös DHPv6 stateful -tilaksi, jossa DHCPv6-palvelin jakaa tilallisia (stateful) osoitteita IPv6-isäntäkoneille. (Davies 2012, 211.)

Sellaisessa yhdistelmässä jossa M-lippu on asetettu arvoon 0, ja O-lippu asetettu arvoon 1, DHCPv6-palvelinta käytetään ainostaan muiden asetusten konfigurointiin. Naapurireitittimet on konfiguroitu mainostamaan muiden kuin link-local-osoitteiden prefixejä, joista IPv6-isäntäkoneet voivat muodostaa itselleen tilattoman (stateless) osoitteen. Tätä kutsutaan DHCPv6 stateless -tilaksi. (Davies 2012, 211.)

Jos M-lippu on asetettu arvoon 1, ja O-lippu arvoon 0, DHCPv6-palvelinta käytetään osoitteiden konfigurointiin, mutta ei muiden asetusten konfigurointiin. IPv6-isäntäkoneet kuitenkin tyypillisesti tarvitsevat myös muita asetuksia, kuten IPv6 DNS-palvelimien osoitteita, joten tämä on epätodennäköinen yhdistelmä. (Davies 2012, 211.)

4.2 DHCPv6-viestit

DHCPv6-viestit ovat UDP (User Datagram Protocol)-viestejä, kuten DHCPv4:ssä. DHCPv6-asiakkaat kuuntelevat DHCPv6-viestejä UDP portista 546. DHCPv6-palvelimet sekä DHCPv6 relay agentit kuuntelevat DHCPv6-viestejä puolestaan UDP portista 547. DHCPv6-viestien rakenne on paljon yksinkertaisempi kuin DHCPv4:n. (Davies 2012, 212.)

Kuva 4. DHCPv6-viestin rakenne. (Davies 2012, 212.)

Msg-Type on yhden tavun mittainen kenttä, ja se kertoo DHCPv6-viestin tyypin. Kolmen tavun mittainen Transaction-ID-kenttä on asiakkaan määrittämä ja sitä käytetään ryhmittelemään DHCPv6-viestien vaihtoa. DHCPv6-palvelimet kopioivat Transaction-ID -kentässä esiintyvän arvon pyyntöviesteistä ja käyttävät samaa arvoa vastausviesteissä. Options-kenttä ei ole kiinteää pituutta, ja sisältää yhden tai useampia optioita, jotka pitävät sisällään asiakkaan ja palvelimen tunnistetietoja, tilallisia IPv6-osoitteita sekä muita konfigurointiasetuksia. (Davies 2012, 212.)

4.3 DHCPv6-viestityypit

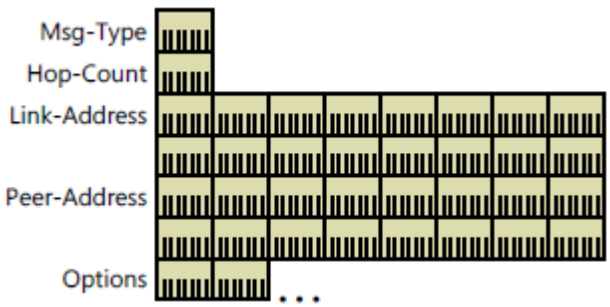
Alla olevassa taulukossa on listattuna RFC 3315:ssä määritellyt DHCPv6-viestien eri tyypit.

Msg-Type	Message	Kuvaus
1	Solicit	DHCPv6-asiakkaan lähettämä viesti DHCPv6-palvelinten löytämiseksi.
2	Advertise	DHCPv6-palvelimen vastaus Solicit-viestiin, jolla se ilmoittaa olevansa saatavilla.
3	Request	DHCPv6-asiakas pyytää osoitteita tai muita konfigurointiasetuksia joltain tietyltä DHCPv6-palvelimelta.
4	Confirm	DHCPv6-asiakkaan kaikille palvelimille lähettämä viesti, jonka avulla varmistetaan, että DHCPv6-asiakkaan konfiguraatio on validi sillä linkillä, johon se on yhdistynyt.
5	Renew	DHCPv6-asiakkaan lähettämä viesti jollekin tietylle DHCPv6-palvelimelle, jolla se pyytää pidennystä sen saamien osoitteiden elinaikoihin, ja samalla päivitystä konfigurointiasetuksiin.
6	Rebind	DHCPv6-asiakkaan lähettämä viesti mille tahansa DHCPv6-palvelimelle silloin, jos Renew-viestiin ei ole saatu vastausta.
7	Reply	DHCPv6-palvelimen lähettämä viesti jollekin tietylle DHCPv6-asiakkaalle vastauksena Solicit-, Request-, Renew-, Rebind-, Information-Request-, Confirm-, Release- tai Decline-viestiin.
8	Release	DHCPv6-asiakkaan lähettämä viesti, jolla se kertoo, ettei enää käytä jotain tiettyä määrättyä osoitetta.
9	Decline	DHCPv6-asiakkaan lähettämä viesti jollekin tietylle DHCPv6-palvelimelle, jolla se kertoo, että DHCPv6-palvelimen tarjoama osoite on jo käytössä.
10	Reconfigure	DHCPv6-palvelimen lähettämä viesti DHCPV6-asiakkaalle, jolla se kertoo, että sillä on uutta tai päivitettyä tietoa konfigurointiasetuksiin. Tämän viestin jälkeen DHCPV6-asiakas lähettää DHCPv6-palvelimelle takaisin joko Renew- tai Information-Request-viestin.
11	Information-Request	DHCPv6-asiakkaan lähettämä viesti pyytääkseen konfigurointiasetuksia (Ei kuitenkaan osoitteita).
12	Relay-Forward	DHCPv6 relay agentin lähettämä viesti jolla se välittää DHCPv6-viestin eteenpäin DHCPv6-palvelimelle. Relay-Forward-viesti pitää sisällään DHCPv6-asiakkaan viestin pakattuna DHCPv6 Relay-Message -option sisään.
13	Relay-Reply	DHCPv6-palvelimen DHCPv6-asiakkaalle lähettämä viesti, joka on lähetetty DHCPv6 relay agentin läpi. Reply-Reply-viesti pitää sisällään DHCPv6-palvelimen viestin pakattuna DHCPv6 Relay-Message -option sisään.

Kuva 5. DHCPv6-viestityypit. (Davies 2012, 213.)

4.4 DHCPv6-viestit relay agentin ja palvelimen välillä

Palvelinten ja relay agenttien väliseen viestinvaihtoon on olemassa erillinen viestirakenne. Tällä mahdollistetaan lisätietojen tallennus, jotta tiedetään esimerkiksi mistä aliverkosta alkuperäinen DHCPv6-viesti tuli, ja näin DHCPv6-palvelin pystyy päättämään minkä aliverkon prefixin se jakaa DHCPv6-asiakkaalle. (Davies 2012, 214.)



Kuva 6. DHCPv6-viestin rakenne relay agentin ja palvelimen välillä. (Davies 2012, 214.)

Hop-Count on yhden tavun mittainen kenttä, ja se kertoo kuinka moni relay agentti on saanut jonkin kyseisen viestin. 16:sta tavun mittainen Link-Address kenttä pitää sisällään osoitteen (ei link-local), joka on määritelty relay agentin sille liityntäportille, joka on samassa verkossa DHCPv6-asiakkaan kanssa. Link-Address-kentästä DHCPv6-palvelin pystyy päättelemään oikean IPv6-osoitealueen josta jakaa osoite. Peer-Address-kenttä on 16 tavua pitkä, ja pitää sisällään sen DHCPv6-asiakkaan IPv6-osoitteen, josta viesti alunperin lähetettiin, tai sen relay agentin osoitteen, joka viimeisenä viestin välitti eteenpäin. (Davies 2012, 214.)

Peer-Address-kentän alapuolella on DHCPv6-optiot. Sen sisällä on Relay Message-optio, joka pitää sisällään viestin jota välitetään eteenpäin, sekä muut relay-optiot. Relay Message-optio paketoi DHCPv6-viestit sisäänsä joita vaihdetaan DHCPv6-asiakkaan ja DHCPv6-palvelimen välillä. (Davies 2012, 214.)

4.5 IPV6:n nimipalvelujärjestelmä

IPv6:ssa nimien muistaminen osoitteiden sijaan on paljon tärkeämpää kuin ennen viitattaessa verkon eri resursseihin, sillä IPv6-osoitteessa saattaa olla yhteensä 32 heksadesimaalilukua. IPv6:ssa nimipalveluille on kaksi protokollaa, DNS ja LLMNR. DNS tarjoaa helpon tavan muuntaa nimiä IPv6-osoitteiksi ja toisinpäin, ja se on suora käännös IPv4:n nimipalvelujärjestelmästä. LLMNR on puolestaan uusi protokolla samaan tarkoitukseen verkoille joissa ei ole DNS-palvelinta. LLMNR sallii sekä IPv6 että IPv4 -isäntien suorittaa nimikyselyitä yksinkertaisten request ja reply -viestien vaihdolla. (Davies 2012, 227-228.)

4.6 Yhteenveto IPv4:n ja IPv6:n eroista

IPv4:ssä osoitteet ovat 32-bittisiä, kun taas IPv6:ssa ne ovat 128-bittisiä. IPsec -otsikkotuki on IPv4:ssä valinnainen. IPv6 vaatii IPsec-otsikkotuen. IPv4 ei käytä otsikkokentässä minkäänlaista tunnistusmekanismia pakettivirroille, jotka pyytävät reitittimiltä erikoiskohtelua. IPv6 käyttää tähän Flow Label -kenttää, johon tunnistusmekanismi sisällytetään. Pakettien paloittelu tehdään versiossa 4 sekä lähettävällä isäntäkoneella että reitittimillä, huonontaen samalla reitittimen suorituskykyä. Versio 6 rajoittaa paloittelun ainoastaan lähettävälle isäntäkoneelle. IPv4:llä ei ole minkäänlaisia vaatimuksia siirtokerroksen paketin koolle. IPv6:n siirtokerroksen täytyy tukea vähintään 1280:n tavun mittaisia paketteja. IPv4:n otsikkokentässä on aina tarkistussumma

ja optiot, IPv6:n otsikkokentässä taas ei ole kumpiakaan. Siirtokerroksen osoitteiden selvittämiseksi IPv4 käyttää broadcast -viesteihin perustuvaa ARP:ia, IPv6 hoitaa saman multicast -lähetyksenä Neighbor Solicitation -viesteillä. Internet Group Management Protocol:aa (IGMP) käytetään IPv4:ssä hallinnoimaan multicast-ryhmäjäsenyyksiä. IPv6:ssa IGMP on korvattu Multicast Listener Discovery:lla (MLD). (Davies 2012, 8; RFC 2236.)

5 VIRTUALISOINNISTA YLEISESTI

Virtualisoinnilla on teknologiana pitkä historia. Lähes kaikki IT-alan yritykset käyttävät virtualisointia jossain muodossa, ja virtualisoinnista on tulossa yksi peruspilareista koko IT-infrastruktuurissa. Monille yrityksille palvelinvirtualisointi on arkipäivää. Kaikki uudet sovellukset luodaan virtuaalikoneille ellei sovelluksen kehittäjä pysty perustelemaan tarvetta fyysisille resursseille. (Ruest 2009, XV.)

Virtualisointi poistaa palvelin käyttöjärjestelmissä riippuvuuden fyysiseen laitteistoon, ja antaa mahdollisuuden liikutella niitä ja suorittaa järjestelmäpalautuksia helpommin. Palvelinten pääkäyttäjät voivat siirtää lennosta live migration -ominaisuuden avulla virtuaalikoneen toiseen fyysiseen resurssiin ja suorittaa palvelimen laitteistohuollon keskellä työpäivää. Virtualisointi muuttaa lähes kaiken sen, miten järjestelmiä, tallennustiloja, tietoverkkoja, tietoturvaa, käyttöjärjestelmiä sekä sovelluksia hallitaan. (Ruest 2009, XVI.)

Virtualisoinnin ytimenä toimii virtuaalikone (virtual machine tai VM), joka on tiukasti eristetty ohjelmistosäilö, jolla on oma käyttöjärjestelmä ja sovellukset. Jokaisen virtuaalikoneen ollessa täysin itsenäinen ja erillään muista, voidaan niitä ajaa useampia samanaikaisesti yhdellä tietokoneella. Ohut ohjelmistokerros, hypervisor, erottaa virtuaalikoneet isäntäkoneesta ja samalla dynaamisesti allokoi resursseja jokaiselle virtuaalikoneelle niiden tarpeiden mukaan. Virtualisoinnin avulla voidaan maksimoida palvelinten käyttöaste ja laskea palvelinten määrä minimiin. Ohjelmistojen ja resurssien provisiointi on nopeampaa ja helpompaa, ja kaikki kriittiset sovellukset voidaan virtualisoida ja näin parantaa suorituskykyä, luotettavuutta, skaalautuvuutta sekä laskea käyttökuluja. (VMware 2014)

6 POHJUSTUS TYÖHÖN

Työn tavoitteena oli toteuttaa Haminan Energian asiakasverkkoon IPv6-standardin mukainen DHCP-palvelin. Palvelinta testattaisiin erilaisten asiakaspäätelaitteiden sekä asiakaskeskittimien kanssa, ja tarvittaessa niihin tehtäisiin muutoksia IPv6:n käyttöönottoa varten. Palvelin oli tarkoitus toteuttaa avoimen lähdekoodin ohjelmistolla ja lopputulokseksi toivottiin suoraan tuotantokäyttöön soveltuvaa toteutusta. Koko projekti suoritettiin yhteistyössä Haminan Energia Oy:n tietoliikenneosaston kanssa.

6.1 Palvelimelle asetetut vaatimukset

Palvelimelle asetettiin tiettyjä vaatimuksia joita sen tulisi täyttää, jotta se voitaisiin myöhemmin ottaa tuotantokäyttöön. Palvelin tulisi toteuttaa virtualisoituna ratkaisuna, ja alustana toimisi VMware vSphere 5. Käyttöjärjestelmäksi toivottiin jotain Debian jakeluun perustuvaa, mutta myös CentOS -pohjainen ratkaisu kävisi. DHCP-ohjelmistoa ei välittömästi lyöty lukkoon, vaan sen pohjalta tulisi tehdä selvitystyötä jotta ohjelmistoksi valikoituisi varmasti sellainen, jossa IPv6-tuki olisi mahdollisimman pitkällä. Osoitteiden jako tultaisiin toteuttamaan prefix delegation -menetelmää käyttäen, jossa asiakasreitittimen LAN-portille jaetaan halutun kokoisia prefixejä, ja josta reititin sen jälkeen jakaa osoitteita eteenpäin siihen kytkeytyneille päätekoneille. Myös erittäin tärkeänä toiminnallisuutena olisi asiakkaiden yksilöiminen tavalla tai toisella, mielellään kuitenkin vastaavanlaisella ratkaisulla, jonka optio 82 tarjoaa IPv4-maailmassa. Toisin sanoen DHCP-kyselyn mukana pitäisi optio-kenttien avulla kuljettaa tietoa palvelimelle, josta asiakas voidaan yksilöidä esimerkiksi keskittimen portin tai asiakasnumeron perusteella. Lopuksi kaikki DHCP-keskustelut sekä jaetut osoitteet tulisi tallentaa erillisiin lokeihin, joista muun muassa edelläkin mainitut tiedot tulisi löytyä.

6.2 Selvitystyö ennen palvelimen asennusta

DHCP-ohjelmiston tärkein valintakriteeri oli sen tuki IPv6:lle. Windows-pohjaiset ratkaisut jätettiin kokonaan pois listalta, koska itse palvelinkin toteutettaisiin Linuxilla. Loppujenlopuksi lupaavimmilta ohjelmistoilta vaikuttivat ISC DHCP ja Dnsmasq. Molemmista löytyi IPv6-tuki, ja molemmat tarjosivat monia mahdollisuuksia osoitteiden jakoon. ISC DHCP vaatisi ohjelmiston lisäksi myös radvd:n (Router Advertisement Daemon) käyttöä, kun taas Dnsmasq selviäisi tehtävästä omillaan. Tarkempi

selvitys kuitenkin osoitti, että ISC DHCP -ohjelmistolla oli huomattavasti paremmin dokumentoidut ohjesivut, sillä pystyi käsittelemään suurempaa määrää asiakkaita, sen IPv6-tuki oli pidemmällä sekä se osasi käsitellä optio-kenttiä paremmin. Vaikkakin ISC DHCP:n konfigurointi näytti esimerkkikonfiguraatioita tarkasteltaessa huomattavasti monimutkaisemmalta, valittiin se kuitenkin käytettäväksi edellä mainittujen ominaisuuksiensa vuoksi.

Optio 82 -kentän korvaajiksi löytyivät optiot 18 ja 37. Optio 18 pitää sisällään Interface-ID -kentän ja optio 37 puolestaan Remote-ID -kentän. Näistä jompaa kumpaa tai molempia yhdessä käyttämällä saataisiin talteen tietoa siitä, kenelle prefixejä on jaettu. (IPAM 2014)

Käyttöjärjestelmäksi valikoitui lopulta CentOS 64-bittisenä versiona, sillä Debianilla suoritetut kokeilut ISC DHCP:n käyttöönotosta sekä muusta konfiguroinnista osoittautuivat paljon hankalammiksi. Siinä missä CentOS otti vastaan ohjelmistot ja konfiguraatiot ilman ongelmia, tuotti Debian ainoastaan valtavan määrän erilaisia virheilmoituksia. Palvelimen tulisi olla helposti käytettävä ja ennen kaikkea vakaa, eikä opinnäytetyölle varatun ajan puitteissa olisi edes järkevää käyttää kaikkea aikaa pelkästään käyttöjärjestelmään liittyvien virheiden selvittämiseen, joten valinta oli selvä.

7 KÄYTÄNNÖN TOTEUTUS

7.1 Virtualisointialustan luominen

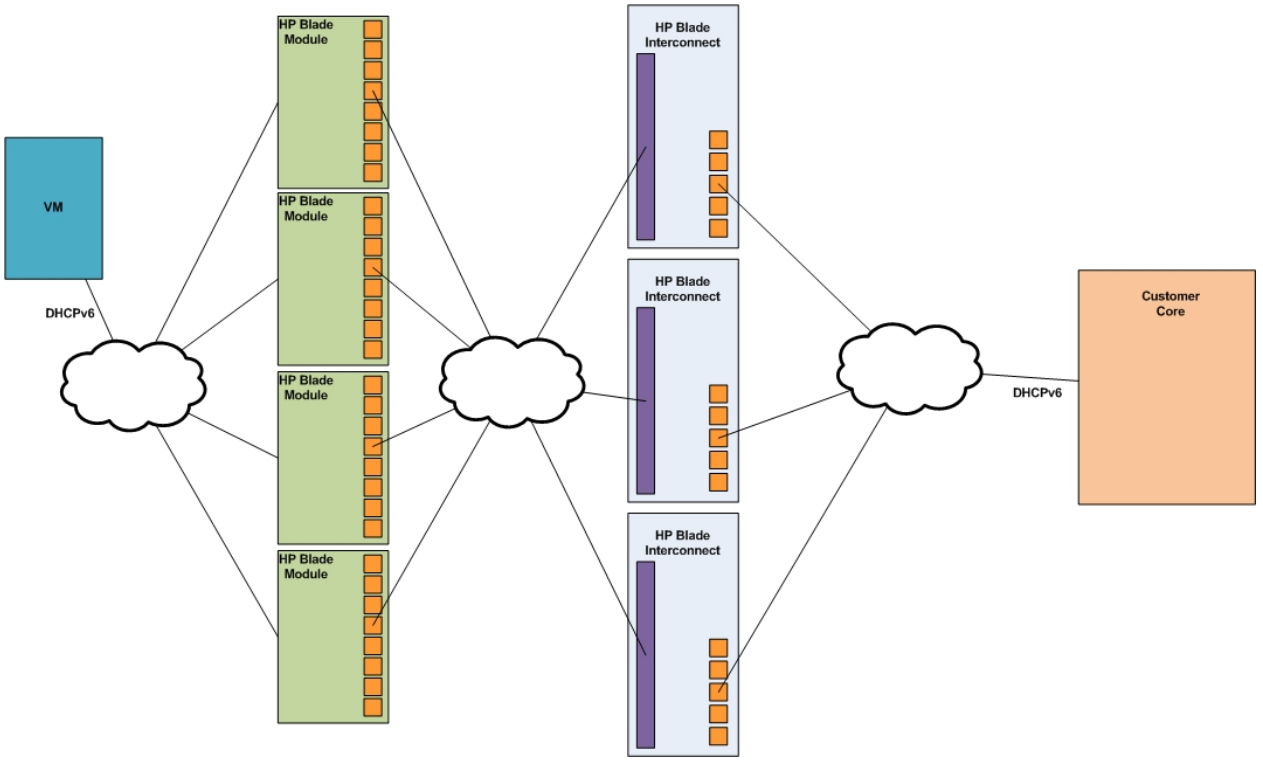
DHCPv6-palvelimelle luotiin aluksi VMware vSphere -ympäristöön datakeskukseen (datacenter) oma tila (datastore), jonne palvelin sijoitettiin. Datacenter pitää sisällään kaiken virtuaalisen palvelinympäristön ajamiseen tarvittavat komponentit, ja datastore puolestaan säilöö kaikki virtuaalikoneisiin liittyvät tiedostot. (VMware vSphere Documentation.)

Sen jälkeen palvelin liitettiin klusterissa sopivalle ESXi host:lle. Klusteri koostuu monesta isäntäkoneesta (host), ja jokaisen isäntäkoneen resurssit ovat osa sen klusterin resursseja, johon kukin isäntäkone kuuluu. Klusterin alla ajettavat isäntäkoneet voivat pitää sisällään monia yksittäisiä virtuaalikoneita. (VMware vSphere Documentation.)

Seuraavaksi ESXi host:lle määriteltiin DHCPv6-palvelinta varten omat port group -määritykset, eli toisin sanoen verkot joita pitkin DHCP -liikennöinti sekä hallintayhteydet kulkisivat. Nämä uudet port group:t liitettiin sitten ESXi:n oikeisiin virtuaaliverkkokortteihin, jotta liikennöinti näihin uusiin verkkoihin onnistuisi.

Myös Blade-järjestelmään piti tehdä tarvittavat muutokset DHCPv6-palvelinta varten. Blade Interconnect Bay:n molempien reunojen portit liitettiin oikeisiin VLAN:hin, jotta liikennöinti DHCPv6-palvelimen sekä haluttujen VLAN:ien välillä onnistuisi.

Lopuksi tehtiin vielä tarvittavat fyysiset kytkennät Blade-kehikon sekä runkokytkimen välillä. Kuvassa x kokonaisuudessaan DHCPv6-palvelimen sekä Blade-järjestelmän väliset kytkennät.



Kuva 7. Blade-järjestelmän sekä DHCPv6-palvelimen väliset kytkennät.

7.2 Palvelimen komponenttien konfigurointi ja käyttöjärjestelmän asennus

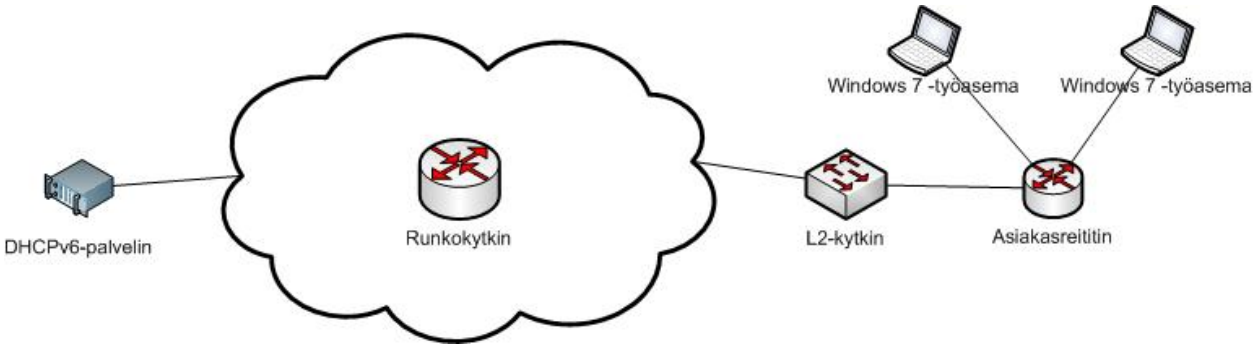
Kun virtualisointialusta oli valmis, voitiin DHCPv6-palvelimelle asentaa sopivat komponentit sekä asentaa käyttöjärjestelmä. Palvelimelle annettiin 1 gigatavu RAM-muistia, yksi yhdellä ytimellä varustettu prosessori, 20:n gigatavun kiintolevy sekä kaksi verkkokorttia. Kiintolevy osoittautui kuitenkin myöhemmin liian pieneksi lokitiedostojen vaatiman tilan takia, ja sen tilaa jouduttiin kasvattamaan 80:een gigata-

vuun. Verkkokorteista toinen varattiin DHCP-liikennettä varten, ja toinen hallintayhteyksien käyttöön. Molemmat verkkokortit liitettiin sen jälkeen vielä oikeisiin port groupeihin jotka oli tehty virtualisointialustan luonnin yhteydessä.

Käyttöjärjestelmäksi valittiin jo aiemminkin mainittu CentOS:n 64-bittinen versio. Asennus suoritettiin käyttämällä CentOS:n tarjoamaa graafista käyttöliittymää. Asennuksen päätyttyä annettiin palvelimen molemmille verkkokorteille IP-osoitteet. Ensimmäinen verkkokortti konfiguroitiin palvelemaan DHCPv6-asiakkaita, ja sille annettiin ainoastaan IPv6-osoite valitusta osoiteavaruudesta. Jälkimmäinen verkkokortti puolestaan liitettiin IPv4-hallintaverkkoon, ja sille asetettiin sopiva IPv4-osoite mahdollistaen sen etähallinnan.

7.3 Testiympäristön rakentaminen

Testiympäristöksi pyrittiin rakentamaan mahdollisimman yksinkertainen verkko, jossa erilaisten konfiguraatioiden kokeileminen olisi mahdollisimman helppoa. Myös laitteiden lisääminen ja poistaminen tuli onnistua vaivattomasti, jotta palvelimen toimintaa voitaisiin testata monella eri asiakaslaitteella ja päätekoneella. Samalla myös pystyttäisiin kartoittamaan parhaiten toimiva asiakasreititin jota voitaisiin myöhemmin alkaa jakaa asiakkaille. Testiympäristöstä jouduttiin kuitenkin jättämään pois asiakaskeskittimet eli DSLAM-laitteet, sillä opinnäytetyölle varatun ajan puitteissa laitetoimittaja ei saanut toimitettua IPv6:een kykenevää ohjelmistoversiota.



Kuva 8. Palvelimen sekä asiakaslaitteiden testausta varten rakennettu testiympäristö.

Runkokytkimeen luotiin kaksi VLAN:ia, joilla simuloitiin kahta erillistä asiakasverkkoa. Molemmat VLAN:t luotiin seuraavilla komennoilla:

*vlan ****

name DHCPv6_Zone_1

*vlan ****
name DHCPv6_Zone_2

vlan -rivi määrittää VLAN:n numeron ja *name* -rivi taas VLAN:n nimen.

VLAN:ja varten piti myös luoda niille omat virtuaaliset liityntäportit, joiden kautta molempien kuvitteellisten asiakasverkkojen liikenne voitiin reitittää sekä DHCPv6-palvelimelle että kuvassakin esiintyvälle L2-kytkimelle. Tämä tehtiin konfiguroimailla molemmat liityntäportit seuraavasti:

interface Vlan Testi1
ipv6 address 2A00:B9A0:350::1/48
ipv6 enable
ipv6 nd ra suppress all
ipv6 dhcp relay destination 2A00:B9A0:302::2 Vlan Palvelin

interface Vlan Testi2
ipv6 address 2A00:B9A0:360::1/48
ipv6 enable
ipv6 nd ra suppress all
ipv6 dhcp relay destination 2A00:B9A0:302::2 Vlan Palvelin

interface Vlan -käskyllä kerrotaan kytkimelle että kyseessä on virtuaalinen liityntäportti. *ipv6 enable* ja *ipv6 address* -rivit sallivat IPv6:n sekä määrittävät liityntäportille IPv6-osoitteen. *ipv6 nd ra suppress all* -käskyllä kerrotaan kytkimelle olla ottamatta kantaa sille tuleviin RA-viestihin, jotta se ei muodostaisi omin päin L3-tason naapuruutta sen viereisen laitteen kanssa ja samalla estäisi DHCPv6:n oikeaa toimintaa. *ipv6 dhcp relay destination* -rivi määrittelee lopuksi sen osoitteen, johon kytkimelle tulevat DHCP-kyselyt tulee ohjata.

L2-kytkimelle verkot tuotiin trunk-linjaa pitkin ja sen jälkeen ne ohjattiin edelleen erillisiin portteihin. Kytkemällä asiakasreitittimen L2-kytkimen perään saatiin se liitet-

tyä kumpaan tahansa luotuun asiakasverkkoon. L2-kytkimen konfiguraatio oli seuraavanlainen:

```
vlan Testi1
  name "DHCPv6_Zone_1"
  untagged 5
  tagged 10
  no ip address
  exit
vlan Testi2
  name "DHCPv6_Zone_2"
  untagged 6
  tagged 10
  no ip address
  exit
```

Molempien VLAN:ien alle määriteltiin name -komennolla niiden nimet. Untagged -käskyllä määriteltiin kytkimen access-portti kummallekin VLAN:lle ja tagged -käskyllä puolestaan kytkimen trunk-portti. No ip address kertoo että IP-osoitetta ei ole annettu.

7.4 ICS DHCP -ohjelmiston asennus ja konfigurointi

Palvelimen konfigurointi aloitettiin sallimalla IPv6-pakettien reititys. Tämä tehtiin lisäämällä /etc/sysctl.conf -tiedostoon rivi net.ipv6.conf.all.forwarding=1. Sen jälkeen palvelimelle asennettiin yum -paketinhallinan kautta uusin versio ISC DHCP -ohjelmistosta. Asennuksen jälkeen kävi kuitenkin ilmi, ettei tämä versio kyennyt käsittelemään Relay-Forward -viestien sisällä kuljetettuja optio -kenttiä. Näiden kenttien käsittely ja niiden sisältämän tiedon talteenotto olisi kuitenkin hyvin tärkeä toiminnallisuus, joten jo asennettu versio päädyttiin korvaamaan uusimmalla mahdollisella 4.3.0 release candidate -versiolla. Tähän versioon oli lisätty v6relay -niminen toiminto, joka osaisi lukea ja prosessoida DHCPv6 relay agentin asettamia optio-kenttiä. Uutta versiota varten jouduttiin rakentamaan myös uusi käynnistys skripti, sillä vanhalla skriptillä DHCPv6-palvelua ei saanut enää käynnistymään. Käytetty skripti löytyy liitteestä; Liite 2. dhcpd6-palvelun käynnistys skripti.

ISC DHCP:n konfiguraatio rakennettiin /etc/dhcp/ polun takaa löytyvään dhcpd6.conf -tiedostoon. Ensimmäisenä konfiguraatiotiedostoon piti määrittää lease-tiedostolle uusi sijainti, sillä SELinux esti sitä toimimasta sen alkuperäisessä sijainnissaan. Uusi sijainti määriteltiin heti tiedoston alkuun lisäämällä sinne seuraavanlainen rivi:

```
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";
```

Tällä kerrottiin ohjelmalle että sen tulisi kirjoittaa kaikki jaetut IPv6-osoitteet polun /var/lib/dhcpd/ takaa löytyvään tiedostoon dhcpd6.leases.

Seuraavaksi ohjelmalle kerrottiin missä verkossa DHCPv6-palvelin sijaitsee, ja se lisättiin tiedostoon subnet6 -määrityksellä.

```
subnet6 2a00:b9a0:302::/48
{
}
```

Hakasulkeiden sisään voitaisiin kyseiselle verkolle tehdä lisämäärityksiä, mutta koska tämä verkko oli varattu palvelimelle ja sieltä ei jaettaisi osoitteita, kommentoitiin siitä kaikki muut rivit pois.

Seuraavaksi luotiin kahdelle testiverkolle omat osoitealueet, joista jaettaisiin /64 kokoisia prefixejä asiakasreitittimen LAN-portille prefix delegation -menetelmällä. Samalla määritettiin kuvitteellisen nimipalvelimen osoite.

```
subnet6 2a00:b9a0:350::/48
{
    option dhcp6.name-servers fec0:0:0:1::1;
    option dhcp6.domain-search "domain.example";
    prefix6 2a00:b9a0:350:: 2a00:b9a0:350:fffc:: /64;
}

subnet6 2a00:b9a0:360::/48
{
    option dhcp6.name-servers fec0:0:0:1::1;
    option dhcp6.domain-search "domain.example";
```

prefix6 2a00:b9a0:360:: 2a00:b9a0:360:fffc:: /64;

}

Subnet6 -määrityksellä kerrottiin verkko, josta DHCPv6-pyyntöjä tulisi. Prefix6 -rivillä määritettiin puolestaan verkko, josta kyseiselle verkolle jaettaisiin prefixejä. Jaettavien prefixien alueeksi muodostui prefix6 -määrityksessä esiintyvien kahden osoitteen välinen alue. option dhcp6.name-servers sekä option dhcp6.domain-search riveillä kerrottiin kuvitteellisen nimipalvelimen osoite.

Tämän jälkeen DHCP:lle konfiguroitiin IPv6-osoitteiden vuokra-ajat. Ajaksi määritettiin yksi päivä eli 86400 sekuntia. Nämä määritykset laitettiin subnet6 -määrityksien ulkopuolelle.

default-lease-time 86400;
max-lease-time 86400;
preferred-lifetime 86400;

default-lease-time kertoo kuinka kauaksi aikaa osoite vuokrataan kun asiakas ei pyydä jotain tiettyä vuokra-aikaa. max-lease-time määrittää puolestaan ylärajan vuokra-ajalle. preferred-lifetime antaa taas ajan sille kuinka kauan vuokrattu IPv6-osoite pysyy validina. Kaikki ajat määritettiin samaksi sillä eri valmistajien päätelaitteet voivat pyytää osoitteita eri tavalla. Tämä voisi aiheuttaa taas sen, että joillekin laitteille osoitteita myönnetään pidemmiksi aikaa kuin toisille.

Seuraavaksi tiedostoon kirjoitettiin rivi, jolla lokitietojen keräys voitiin käynnistää. Tämä rivi lisättiin myös subnet6 -määritysten ulkopuolelle:

log-facility local7;

log-facility kertoo sen palvelun mihin lokia kirjoitetaan. local7 oli sopivasti vapaana, ja se todettiin testauksen kautta hyvin toimivaksi, joten se jätettiin ISC DHCP:n määrittämään oletusarvoon, eli local7:ään.

Lokitietojen keräämisen lisäksi täytyi ohjelmalle kertoa vielä mitä kaikkea lokiin haluttaisiin kerätä. Samoilla määrityksillä muokattiin myös IPv6-osoitteiden lease-tiedostoon kirjautuvia tietoja. Erityisesti DHCPv6-keskustelun Relay-Forward viestis-

sä asetettujen optioiden 18 ja 37 tiedot tuli saada kirjoitettua lease-tiedostoon niin, että jokaisen vuokratun IPv6-osoitteen alla lukisi joko portti tai asiakasnumero. Näin voitaisiin myöhemmin yhdistää kukin jaettu IPv6-osoite johonkin tiettyyn asiakkaaseen. Seuraavat rivit lisättiin jälleen subnet6 -määritysten ulkopuolelle:

```
on commit {
    if option dhcp6.ia-na = option dhcp6.ia-na{
    set iana = binary-to-ascii(16,16,":",substring(suffix(option dhcp6.ia-na,24),0,16));
    }
    if option dhcp6.ia-pd = option dhcp6.ia-pd{
    set iapd = binary-to-ascii(16,16,":", suffix(option dhcp6.ia-pd,16));
    set pdsize = binary-to-ascii(10,8,":",substring(suffix(option dhcp6.ia-pd,17),0,1));
    }
    set ifname = v6relay(1, option dhcp6.interface-id);
    set lla = (binary-to-ascii(16, 8, ":", suffix(option dhcp6.client-id, 6)));
    if option dhcp6.ia-na = option dhcp6.ia-na{
    log(info, concat("ONCOMMIT IA_NA: ", iana, " To: ", lla, " INT: ", ifname));
    }
    if option dhcp6.ia-pd = option dhcp6.ia-pd{
    log(info, concat("ONCOMMIT IA_PD: ", iapd, "/", pdsize, " To: ", lla, " INT: ",
    ifname));
    }
}
```

Kaikki on commit -käskyn alle kirjoitetut komennot perustuvat ehtojen täyttymiseen sekä muuttujien asettamiseen. If -alkuiset rivit vaativat aina jonkun ehdon täyttymistä, ja set -alkuiset rivit puolestaan asettavat aina jonkin muuttujan, jos sitä edellä asetettu ehto on täyttynyt.

Ensimmäisessä ehtolauseessa pyydetään palvelinta lukemaan, löytyykö DHCPv6 optio 3 IA-NA DHCPv6-viestin sisältä. Jos se löydetään, asetetaan muuttuja iana, jonka arvoksi kirjoitetaan option 3 arvo ascii-muodossa.

Järjestyksessä alaspäin seuraava ehtolause suorittaa täysin saman toiminnon kuin edellinenkin, mutta se etsii DHCPv6 optio 25 IA-PD:tä, ja sen löytäessään kirjoittaa sen muuttujaan iapd myöskin ascii-muodossa. IA-PD:llä varataan jokin tietty IPv6 prefix

jaetuksi. Sen lisäksi asetetaan myös muuttuja pdsize, joka lukee samaa optiota käyttäen jaetun prefixin koon, ja kirjoittaa sen ascii-muodossa talteen.

Toista ehtolauserakennetta seuraa kaksi set-riviä, joilla asetetaan lisää muuttujia. Ensin asetetaan muuttuja ifname, jolla kutsutaan v6relay -toimintoa. V6relay osaa lukea Relay-Forward -viestien sisältä asetetut optio-kentät 18 ja 37, ja tässä tapauksessa sitä pyydettiin lukemaan optio-kentän 18 sisältö sekä asettamaan sen kentän sisältämä arvo muuttujaan ifname. Jälkimmäiseen muuttujaan (lla) luettiin DHCPv6-option 1 sisältämä client-ID ascii-muotoisena.

Alimmaisena olevat ehtolauseet tulostavat kaiken aiemmin muuttujiin luetun datan ulos luettavassa muodossa. Ensin tarkistetaan vielä onko optiot IA-NA sekä IA-PD olemassa, jonka jälkeen kaikki muuttujien sisältö tulostetaan järjestyksessä tiedostoon.

Kokonaisuudessaan tämä esitelty lokitietojen keräystoiminto tuottaa seuraavanlaisen tulosteen lease-tiedostoon:

```
ia-pd "\001\000\000\000\000\003\000\001\220\366RD\200i" {
  cltt 1 2014/02/17 06:23:34;
  iaprefix 2a00:b9a0:350:fffc::/64 {
    binding state active;
    preferred-life 86400;
    max-life 86400;
    ends 2 2014/02/18 06:23:34;
    set lla = "90:f6:52:44:80:69";
    set ifname = "VlTesti1";
    set pdsize = "64";
    set iapd = "2a00:b9a0:350:fffc:0:0:0:0";
```

Tästä tulosteesta voidaan helposti nähdä milloin prefix on jaettu, kuinka pitkä elinaika sillä on, milloin se vanhenee, kenelle se on jaettu, kuinka suuri jaettu prefix on sekä mikä on jaetun prefixin tarkka osoite. Asiakkaan yksilöiminen onnistuu helposti muuttujien lla sekä ifname sisältöjä tarkastelemalla. Tästä kyseisestä tulosteesta voidaan esimerkiksi päätellä, että prefix on jaettu jonkin tietyn virtuaaliportin taakse, ja prefixin vastaanottaneen laitteen MAC-osoite on 90:f6:52:44:80:69.

7.5 RADVD:n konfigurointi

RADVD:tä eli Router Advertisement Daemon:ia tarvittiin palvelimella lähettämään RA-viestejä IPv6-asiakkaille jotta DHCPv6-keskustelu voitaisiin käydä asiakkaan ja palvelimen välillä. RADVD:lla on muitakin toimintoja, mutta tässä tapauksessa sitä tarvittiin ainoastaan RA-viestien lähetykseen. Tästä syystä myös sen konfigurointi oli hyvin yksinkertaista. RADVD:n konfiguraatiotiedosto radvd.conf löytyy polun /etc/ takaa. ISC DHCP:n hoitaessa kaiken osoitteiden jakoon liittyvän työn, konfiguroitiin RADVD ainoastaan asettamaan RA-viestissä olevat M ja O liput arvoon 1. Samalla määriteltiin RA-viestien lähetysintervallit sekä portti johon RA-viestejä lähetettäisiin. Konfiguraatiotiedosto näytti seuraavanlaiselta:

```
interface eth1
{
        AdvSendAdvert on;
        AdvManagedFlag on;
        AdvOtherConfigFlag on;
        MinRtrAdvInterval 30;
        MaxRtrAdvInterval 100;
};
```

interface -rivillä määritettiin portti johon RA-viestejä lähetettäisiin. AdvSendAdvert -rivillä kerrottiin RADVD:ta lähettämään RA-viestejä ja vastaamaan RS-viesteihin. AdvManagedFlag ja AdvOtherConfigFlag -riveillä määritettiin molempien M sekä O lippujen arvoksi 1. MinRtrAdvInterval sekä MaxRtrAdvInterval -riveillä määritettiin puolestaan se aika, jonka sisällä RA-viesti lähetetään.

7.6 NTP:n konfigurointi

Jotta palvelin saatiin pysymään ajassa, tuli se määrittää käyttämään NTP:tä (Network Time Protocol). Samalla myös varmistuttaisiin siitä, että lease-tiedoston aikaleimat olisivat luotettavia. NTP:n konfigurointitiedosto löytyi polun /etc/ takaa nimellä ntp.conf. Tästä tiedostosta kommentoitiin pois muut NTP-palvelimet ja sinne lisättiin ainoastaan Haminan Energia Oy:n verkon oman NTP-palvelimen osoite. Lisätty rivi oli muotoa server xxx.xxx.xxx.xxx.

7.7 Lokiskriptien teko ja ajastus

Kaikki DHCP-keskustelut sekä lease-tiedot haluttiin saada talteen erillisiin lokitiedostoihin. Jotta tämä onnistuisi, niitä varten tarvittiin skriptit jotka sen jälkeen ajastettiin ajettavaksi tietyin aikavälein. Haminan Energia Oy:lla oli olemassa jo valmiiksi toimiva DHCPv4-palvelin, joten saman toiminnallisuuden säilyttämiseksi skriptit kopioitiin suoraan DHCPv4-palvelimelta myös DHCPv6-palvelimen käyttöön. Näin kaikki lokitiedostot löytyisivät samasta paikasta kuin ennenkin.

Ensimmäisenä kuitenkin DHCP-keskusteluja varten tarvittiin oma loki jota voitaisiin sen jälkeen aina tietyin aikavälein kopioida talteen. ISC DHCP-ohjelmiston konfigurointivaiheessa itse ohjelma määritettiin jo kirjoittamaan tietyt tiedot lokiin, mutta erillistä lokitiedostoa ei vielä määritelty. Uusi lokitiedosto määriteltiin /etc/ -polun takaa löytyvään rsyslog.conf -tiedostoon. Tähän tiedostoon lisättiin seuraava rivi:

local7. /var/log/dhcpd6.log*

Tällä rivillä kerrottiin rsyslog -ohjelmalle, että kirjoita local7 -alkuiseen palveluun tulevat viestit polun /var/log/ takaa löytyvään dhcpd6.log -tiedostoon.

Seuraavaksi varmuuskopiointiskriptit ajastettiin Linuxin cron -ajastustoiminnon avulla. /etc/ -polun takaa löytyvä crontab -tiedosto muokattiin sisältämään ajastukset kolmelle eri skriptille. Tähän tiedostoon lisättiin seuraavat rivit:

* * * * * *root /root/scriptit/lease.sh*
* * * * * *root /root/scriptit/dhcpd6loki.sh*
*59 23 * * * root /root/scriptit/paiva.sh*

Näissä * -merkillä kerrotaan ensin kellonaika jolloin skripti ajetaan. Viisi kappaletta * -merkkejä tarkoittaa, että skripti ajetaan joka minuutti. Alimpana listassa oleva ajastus puolestaan tarkoittaa, että skripti ajetaan jokaisena päivänä kello 23:59. Kellonajan jälkeen crontab:lle kerrotaan käyttäjänimi jolla skripti ajetaan, ja joka tässä tapauksessa on root. Käyttäjänimen jälkeen täytyy ohjelmalle kertoa vielä polku jonka takaa skripti löytyy.

Lopuksi kaikki skriptit kopioitiin omaan kansioonsa /root/scriptit ja ne nimettiin vastaamaan crontab -ohjelmassa kutsuttuja skriptejä.

7.8 Palomuurin konfigurointi

Linuxin palomuuri on konfiguroitu oletuksena estämään DHCPv6 -liikenne. Palomuuriin täytyi tehdä säännöt, joilla sallittaisiin DHCPv6-kyselyt, RA-viestit, NS- ja NA-viestit sekä DHCPV6:een liittyvät virheilmoitukset. Palomuurin konfiguraatio tehtiin /etc/sysconfig -polun takaa löytyvään ip6tables -tiedostoon. Ensin sallittiin DHCPv6:lle UDP-portti 547 jota se käyttää palvelimella toimiessaan:

-A INPUT -p udp -m udp --dport 547 -j ACCEPT

Sen jälkeen sallittiin ICMP 134 -tyyppiä olevat RA-viestit link-local-osoitealueelta:

-A INPUT -p icmpv6 -s fe80::/10 --icmpv6-type 134 -j ACCEPT

Myös ICMP:n tyyppiä 135 ja 136 olevat NS- ja NA-viestit tuli sallia sekä link-local-osoitealueelta kuin myös verkosta johon palvelin sijoitettiin:

-A INPUT -p icmpv6 -s fe80::/10 --icmpv6-type 135 -j ACCEPT
-A INPUT -p icmpv6 -s palvelimen_verkko --icmpv6-type 135 -j ACCEPT
-A INPUT -p icmpv6 -s fe80::/10 --icmpv6-type 136 -j ACCEPT
-A INPUT -p icmpv6 -s palvelimen_verkko --icmpv6-type 136 -j ACCEPT

ICMP:n tyypit 1-4 sallittiin vielä virheilmoituksia varten:

-A INPUT -p icmpv6 --icmpv6-type 1 -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type 2 -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type 3 -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type 4 -j ACCEPT

Loppuun jätettiin vielä kaiken lopun ICMPv6-liikenteen oletuksena kieltävät säännöt, jotta mikään ylimääräinen portti ei jäisi vahingossa auki:

-A INPUT -j REJECT --reject-with icmp6-adm-prohibited

-A FORWARD -j REJECT --reject-with icmp6-adm-prohibited

7.9 Tarvittavien palveluiden automaattinen käynnistys

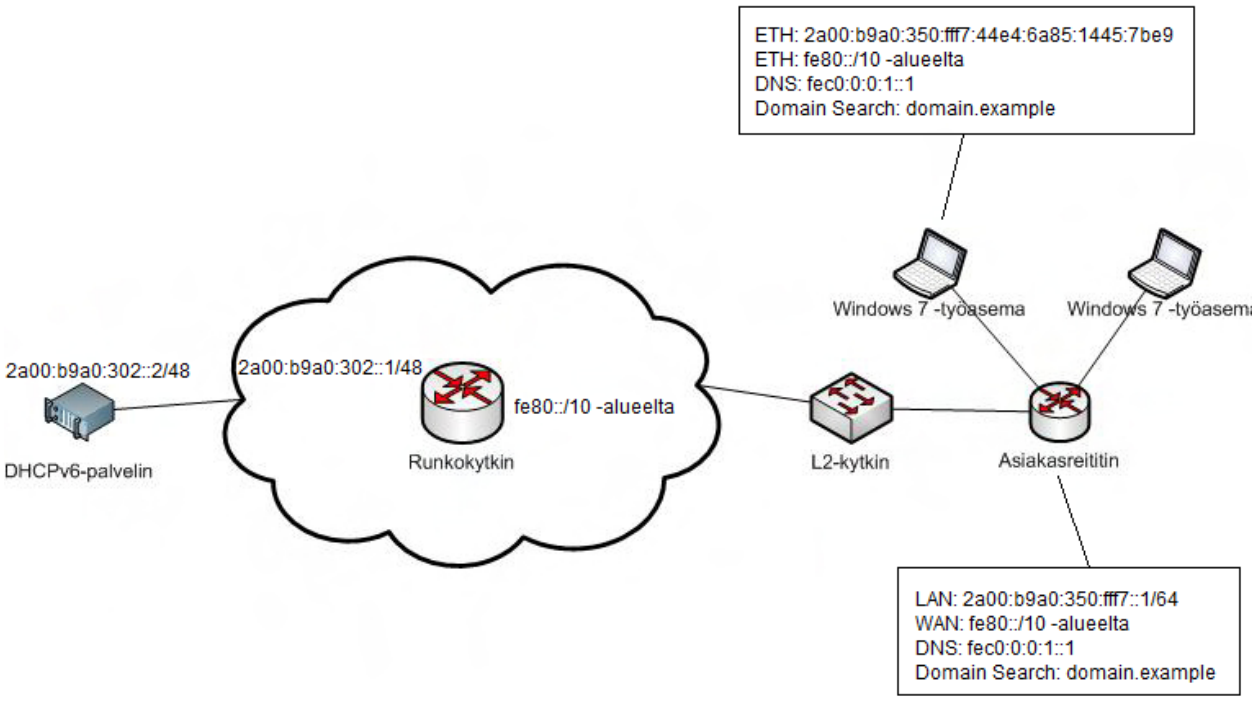
Mikäli palvelinta jouduttaisiin käynnistämään uudelleen, tai mikäli se jonkin vian seurauksena käynnistyisi uudelleen itsekseen, asetettiin palvelimen toiminnalle tärkeät palvelut käynnistymään järjestelmän käynnistymisen yhteydessä. Tätä varten muokattiin /etc/rc.d/ -polun takaa löytyvää rc.local -tiedostoa. Rc.local on skripti, joka ajetaan vasta kaikkien muiden skriptien ajon jälkeen järjestelmän käynnistymisen yhteydessä. Asettamalla palveluiden käynnistyskäskyt tähän tiedostoon voitiin varmistua siltä, että esimerkiksi verkkokorttien ajurit on varmasti ajettu ennen DHCPv6-palvelun käynnistämistä. Rc.local -tiedostoon ajastettiin käynnistyväksi itse DHCPv6-palvelu, RADVD sekä NTP. Näitä varten kyseiseen tiedostoon lisättiin seuraavat rivit:

/etc/init.d/dhcpd6 start
/etc/init.d/radvd start
/etc/init.d/ntpd start

Rc.local -tiedoston käyttö on hyvin yksinkertaista. Palvelun käynnistyskäskyksi riittää, että sille määritetään polku josta kyseinen palvelu löytyy ja samalla käsketään start -komennolla käynnistämään se.

7.10 Palvelimen testaus

Kun kaikki tarpeellinen ohjelmisto sekä konfigurointi oli saatu tehtyä, voitiin palvelimen testaus aloittaa. Testauksessa käytettiin jo aiemmin esiteltyä testiverkkoa. Testauksen päätavoitteina oli varmistaa, että palvelin osaa jakaa prefixejä eri verkkoihin sekä tehdä niistä tarvittavat lokimerkinnät. Samalla voitaisiin kokeilla erilaisia asiakasreitittimiä sekä selvittää niiden IPv6-valmiutta. Halutussa lopputuloksessa testiverkko näyttäisi seuraavanlaiselta:



Kuva 9. Testiverkko IPv6-osoitteilla.

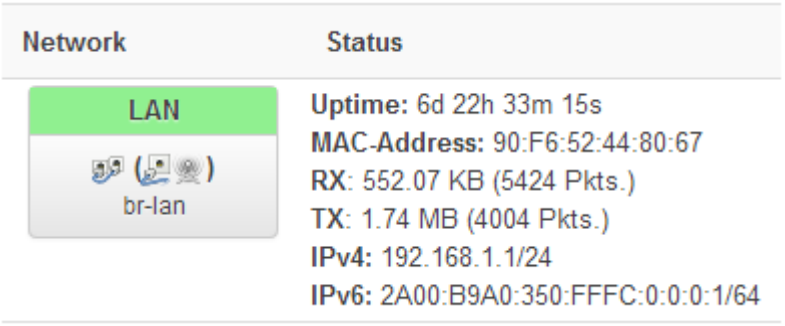
7.10.1 Testaus TP-Link TL-WR740N -reitittimen kanssa

Ensimmäinen testi suoritettiin käyttäen TP-Link:n reititintä. Tämä kyseinen malli valikoitui sopivaksi testilaitteeksi, sillä sille oli saatavilla OpenWrt:n uusin kokeellinen ohjelmisto. Ohjelmistoversio oli Trunk -linjan koodinimeä Barrier Breaker kantava revisio 39319.

Uusi ohjelmisto asennettiin laitteelle ongelmitta suoraan graafista käyttöliittymää käyttäen. Tämän jälkeen laitteelle asennettiin SSH-yhteyden kautta web-käyttöliittymä LuCI sekä lisäpaketti sille IPv6:n graafista tukea varten. TP-Link konfiguroitiin DHCPv6-asiakkaaksi muuttamalla LAN-portin DHCPv6-asetuksia. Router Advertisement Service asetettiin Server-tilaan, ja NDP-Proxy asetettiin hybriditilaan. DHCPv6-palvelu otettiin myös pois päältä, sillä päätekoneiden haluttiin konfiguroivan itselleen osoitteen stateless autoconfiguration -toiminnolla. Samoin myös ULA-prefixin jakelu otettiin pois käytöstä, sillä tarkoituksena olisi jakaa työasemille IPv6-osoitteita globaalilta alueelta, eikä käyttää privaatteja osoitteita.

Sen jälkeen TP-Link kytkettiin kiinni testiverkkoon, ja TP-Link:n perään kytkettiin yksi Windows 7 -työasema. TP-Link kätteli itselleen välittömästi link-local-naapuruuden relay-kytkimen kanssa ja sen jälkeen pyysi IPv6-prefixiä DHCPv6-

palvelimelta. Prefix ilmestyi reitittimen LAN-portille josta se onnistuneesti jakoi sitä eteenpäin työasemalle. Lopuksi työasema neuvotteli itselleen mainostetusta prefixistä stateless autoconfiguration -toiminnolla uniikin osoitteen itselleen.



Kuva 10. TP-Link on pyytänyt itselleen onnistuneesti IPv6-prefixin.

7.10.2 Testaus Asus RT-N56U -reitittimen kanssa

Toisena päätettiin kokeilla täysin kaupallisen tuotteen IPv6-toiminnallisuutta DHCPv6-palvelimen kanssa. Tämäkin testi oli tärkeä, sillä asiakkaiden ei voida olettaa käyttävän aina palveluntarjoajan hyväksymiä laitteita. Laitteeksi valikoitui yleisesti suosittu reititin jossa on IPv6-tuki.

Laite kytkettiin testiverkkoon kiinni ja asetettiin DHCPv6-client -tilaan. Kyseinen laite ei kuitenkaan suostunut aloittamaan minkäänlaista viestinvaihtoa DHCPv6-palvelimen kanssa, vaikka siitä otettiin palomuuri kokonaan pois käytöstä. Tämän jälkeen laitteen ohjelmisto päivitettiin uusimpaan mahdolliseen ja se kytkettiin takaisin testiverkkoon. Sama tilanne kuitenkin toistui eikä laite suostunut lähettämään palvelimelle RS-viestiä jolla se pyytäisi itselleen konfiguraatiotietoja. Laitteen lokitietoja tarkasteltaessa selvisi, että laitteen mielestä DHCPv6-palvelimessa olisi vikaa. Pienen selvitystyön jälkeen kävi kuitenkin ilmi, että tämä olisi laitteen ohjelmiston toimintaan liittyvä virhe, ja koska uudempaa ohjelmistoversiota ei ollut saatavilla, jouduttiin laitteen testaaminen lopettamaan.

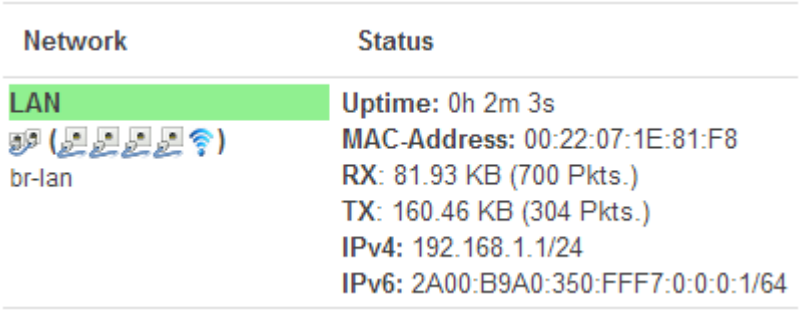
7.10.3 Testaus Inteno VG50A -modeemin kanssa

Tärkein testi suoritettiin Intenon VG50A -modeemilla. Kyseinen laite on aktiivisessa käytössä useissa asiakasliittymissä, joten sen IPv6-toiminnallisuuden varmistaminen

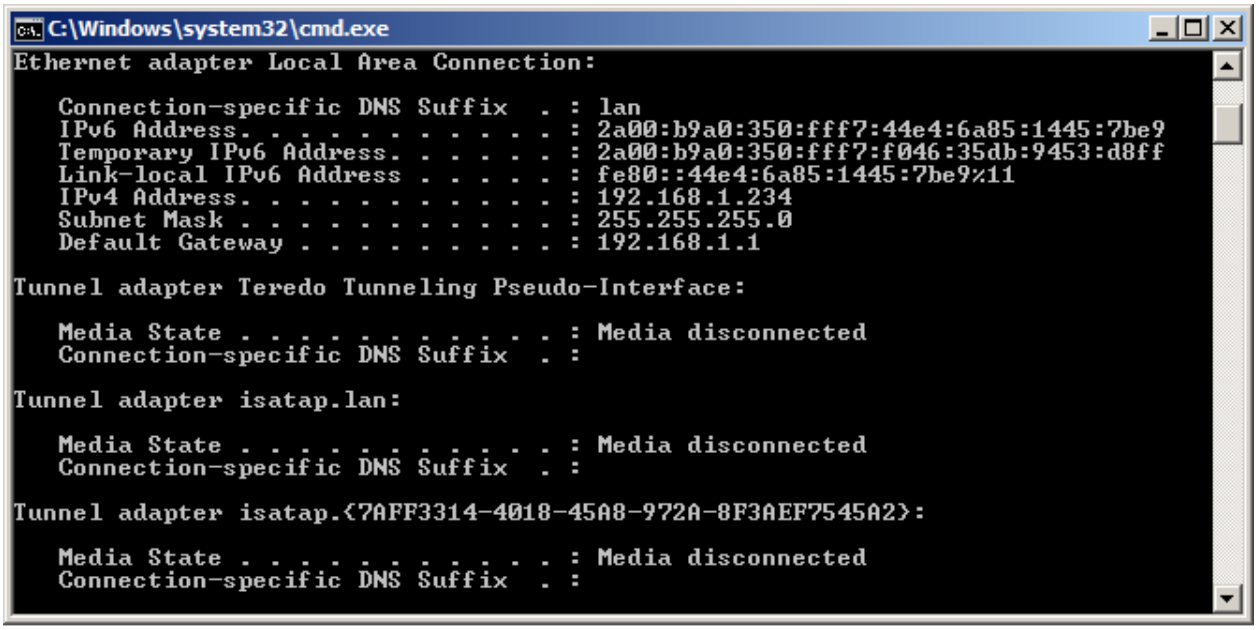
olisi hyvin tärkeää. Laitteessa oli ohjelmistona Intenon muokkaama versio OpenWrt:n Attitude Adjustment 12.09.1:stä.

Ensimmäisenä Intenon WAN6 Interface täytyi sillata laitteen fyysisiin WAN-liityntäportteihin, jotta IPv6-toiminnallisuus saatiin käyttöön. Laite konfiguroitiin jälleen DHCPv6-asiakkaaksi. Ohjelmistosta löytyneen virheen vuoksi se jouduttiin kuitenkin asettamaan tilaan, jossa se ei pyytäisi itselleen minkäänlaista IPv6-osoitetta, sillä muuten se ei kelpuuttanut itselleen minkäänlaista prefixiä. ULA-prefixin jakelu otettiin pois käytöstä ja sen jälkeen määriteltiin vielä DHCPv6:n asetukset TP-Link:iä vastaaviksi. Router Advertisement Service asetettiin Server Mode -tilaan, ja DHCPv6-palvelu sekä NDP-Proxy otettiin pois päältä. Lopuksi määriteltiin vielä mitä liityntäporttia DHCPv6 palvelisi, ja miltä liityntäportilta se ottaisi vastaan käskyjä. Palveltavaksi portiksi asetettiin LAN-portti, ja palvelevaksi portiksi WAN6.

Konfiguroinnin jälkeen Inteno kytkettiin testiverkkoon kiinni, ja sen perään kytkettiin jälleen Windows 7 -työasema. Kuten uudemmalla versiolla varustettu TP-Link:n reititin, myös tämä laite kätteli välittömästi itselleen link-local-naapuruuden relay-kytkimen kanssa, ja pyysi sen jälkeen DHCPv6-palvelimelta prefixin itselleen. Laite asetti prefixin LAN-liityntäporttiin ja alkoi sen jälkeen jakaa sitä eteenpäin työasemille. Tässäkin tapauksessa työasema konfiguroi onnistuneesti stateless autoconfiguration -toiminnolla itselleen uniikin IPv6-osoitteen annetusta prefixistä.



Kuva 11. Inteno on pyytänyt itselleen IPv6-prefixin ja asettanut sen LAN-portille.



Kuva 12. Windows 7 -työasema on muodostanut Intenon tarjoamasta prefixistä itselleen uniikin IPv6-osoitteen.

7.10.4 Palvelimen oikean toiminnan varmistaminen

Täyden varmuuden saamiseksi siitä, että palvelin jakaa prefixit oikein, tarkasteltiin DHCPv6-viestienvaihtoa Wireshark -ohjelman avulla. Wireshark käynnistettiin tutkimaan paketteja palvelimen DHCPv6-palveluja tarjoavasta portista samalla kun asiakasreititin kytkettiin kiinni testiverkkoon. Tuloksena oli seuraavanlainen viestinvaihto:

```
DHCPv6    231 Relay-forw L: 2a00:b9a0:360::1 Solicit XID: 0xb8208 CID: 0003000190f652448069
DHCPv6    298 Relay-reply L: 2a00:b9a0:360::1 Advertise XID: 0xb8208 CID: 0003000190f652448069
DHCPv6    215 Relay-forw L: 2a00:b9a0:360::1 Solicit XID: 0x46b61a CID: 0003000190f652448069
DHCPv6    234 Relay-reply L: 2a00:b9a0:360::1 Advertise XID: 0x46b61a CID: 0003000190f652448069
DHCPv6    258 Relay-forw L: 2a00:b9a0:360::1 Request XID: 0x20a0e0 CID: 0003000190f652448069
DHCPv6    230 Relay-reply L: 2a00:b9a0:360::1 Reply XID: 0x20a0e0 CID: 0003000190f652448069
```

Kuva 13. DHCPv6-asiakkaan ja DHCPv6-palvelimen välinen keskustelu.

Kaikki viestit olivat joko Relay-Forward tai Relay-Reply -tyyppisiä, sillä palvelimen ja reitittimen välissä oli DHCPv6 relayna toimiva kytkin. Reititin aloitti keskustelun Solicit -viestillä pyytämällä itselleen Non-temporary -osoitetta sekä prefixiä. Sen jälkeen palvelin vastasi Advertise -viestillä, jossa se kertoi ettei Non-temporary osoitetta ole saatavilla, ja tarjosi samalla prefixiä reitittimelle. Seuraavaksi reititin lähetti uuden Solicit -viestin, mutta tällä kertaa pyysi ainoastaan prefixiä itselleen. Palvelin vastasi jälleen Advertise -viestillä ja mainosti samaa prefixiä uudelleen. Reititin hyväksyi tämän ehdotuksen ja lähetti Request -viestin, jolla se pyysi sitä itselleen. Lopuksi palve-

lin hyväksyi pyynnön, ja lähetti takaisin Reply -viestin, jolla se vahvisti luovuttaneensa reitittimelle pyydetyn prefixin.

8 TULOSTEN TARKASTELU JA JATKOKEHITYS

Palvelin saatiin konfiguroitua halutunlaiseksi, vaikka käytetyt ohjelmistoversiot olivat enemmän tai vähemmän kehityksenalaisia. Asetetut tavoitteet täyttyivät kaiken muun paitsi DSLAM:ien testauksen ja testiverkkoon tuonnin osalta hyvin. Valitettavasti laitetoimittajalta ei saapunut opinnäytetyölle varatun ajan puitteissa sellaista ohjelmistoversiota, joka olisi tukenut IPv6:tta. Myöskään yhteyttä IPv6 Internetiin ei voitu muodostaa, sillä runkoverkon puolelta puuttui IPv6-reititys kokonaan, eikä IPv6:een kykenevää nimipalvelinta ollut saatavilla. IPv6-reitityksen rakentaminen sekä sopivan nimipalvelimen järjestäminen olisi vaatinut valtavasti lisää aikaa, ja niiden mukaan ottaminen opinnäytetyön sisältöön olisi kasvattanut työn laajuuden aivan liian suureksi.

IPv6:een siirtyminen on aloitettu jo kauan aikaa sitten, eikä kunnolla toimivia IPv6-ratkaisuja löydy vieläkään. Tietoa IPv6:een liittyen on hyvin vähän saatavilla, ja kaikki IPv6:tta tukevat ohjelmistot ovat edelleen melko puutteellisia. Ennen kuin IPv6:tta voidaan alkaa käyttää laajemmin, vaatisi se huomattavan lisäpanostuksen IPv6-tuen kehittämiseen kaikille TCP/IP:tä käyttäville laitteille. Toistaiseksi näyttäisi siltä, että kaikki avoimen lähdekoodin ohjelmistot ovat pisimmällä IPv6-ominaisuuksiensa kanssa, ja kaikki kaupalliset tuotteet puolestaan ovat hyvinkin rajoittuneita IPv6:n suhteen. Mielenkiintoista olisi ollut esimerkiksi kokeilla, kuinka hyvin Windows -pohjainen palvelin olisi suoriutunut DHCPv6-palvelun tarjoamisessa.

Jatkokehitystä palvelimelle jäi todella paljon. ISC DHCP:n release candidate ohjelmisto pitäisi korvata uuteen heti vakaan version ilmestyessä. IPv6:een kykenevien reitittimien kartoittamista tulisi jatkaa, jotta IPv6:n käyttöönotossa voitaisiin mahdollisesti rajata IPv6-osoitteiden jakoa niin, että osoitteita jaettaisiin ainoastaan jo toimivaksi todetuille laitteille. DSLAM:eille sopivan ohjelmistoversion saapuessa täytyisi niiden testaus aloittaa mahdollisimman pian, jotta mahdolliset ongelmat saataisiin selvitettyä ja samalla tutkittua kumpaa DHCPv6 optiota käyttäen DSLAM:it yksilöivät asiakkaat. Sen jälkeen DHCPv6-palvelin voitaisiin konfiguroida ottamaan talteen tiedot oikeasta optio-kentästä. Palvelimen käytettävyyttä voisi myös kehittää, sillä ainoa tapa toistaiseksi tutkia esimerkiksi lease-tietoja on avata lokitiedostoja johonkin tekstieditoriin. Samoin kaikki konfigurointi täytyy tehdä jonkin tekstieditorin avulla.

Graafisen käyttöliittymän kautta kaiken hoitaminen olisi huomattavasti helpompaa, mutta sellaisen kehittäminen avoimen lähdekoodin ohjelmistolle olisi todennäköisesti turhaa. Ohjelmiston päivittyessä graafinen käyttöliittymä saattaisi lakata toimimasta joiltain osin ellei jopa kokonaan, ja jos DHCP-ohjelmistoa päivitetään vähänkään useammin, tulisi sen ylläpitämisestä todella aikaa vievää. Pelkästään IPv6:een valmistautuminen ja sille sopivien tietoverkkoratkaisujen kehittäminen tulee olemaan todella haastavaa jokaiselle Internet -palveluntarjoajalle, jonka myös tämän työn toteuttaminen osoitti.

LÄHTEET

Blanchet, M. 2006. Migrating to IPv6. England: John Wiley & Sons Ltd.

Davies, J. 2012. Understanding IPv6, Third Edition. California, USA: O'Reilly Media.

Hagen, S. 2006. IPv6 Essentials. Second Edition. California, USA: O'Reilly Media.

THE IANA IPV4 ADDRESS FREE POOL IS NOW DEPLETED. ARIN 2/2011. Saatavissa: https://www.arin.net/announcements/2011/20110203.html [viitattu 19.11.2013].

ISC DHCPv6 Option Configuration. IPAM 2014. Saatavissa: http://www.ipamworldwide.com/dhcp-options/isc-dhcpv6-options.html [viitattu 11.3.2014].

Odom, W. 2012. CCENT/CCNA ICND1 Official Cert Guide, Third Edition. Indianapolis, USA: Cisco Press.

IPv4 Exhaustion. RIPE NCC 9/2012. Saatavissa: http://www.ripe.net/internet-coordination/ipv4-exhaustion [viitattu 19.11.2013].

RFC 2236, Internet Group Management Protocol, Version 2. IETF 11/1997. Saatavissa: http://tools.ietf.org/rfc/rfc2236.txt [viitattu 11.3.2014].

RFC 2373, IP Version 6 Addressing Architecture. IETF 7/1998. Saatavissa: http://www.ietf.org/rfc/rfc2373.txt [viitattu 11.3.2014].

Ruest, D. & N. 2009. Virtualization: A Beginner's Guide. USA: McGraw-Hill Companies.

Teare, D. 2010. Implementing Cisco IP Routing. Indianapolis, USA: Cisco Press.

Virtualization Basics. VMware 2014. Saatavissa: http://www.vmware.com/virtualization/virtualization-basics/how-virtualization-works.html [viitattu 11.3.2014].

VMware vSphere 5.1 Documentation Center. VMware vSphere Documentation. Saatavissa: https://pubs.vmware.com/vsphere-51/index.jsp?topic=%2Fcom.vmware.vsphere.vcenterhost.doc%2FGUID-F9B69C22-247F-44A9-B68D-0EC52E2C06E1.html [viitattu 11.3.2014].

DHCPv6-palvelimen konfiguraatiotiedostojen sijainnit Liite 1

crontab: /etc/crontab
dhcpd6: /etc/dhcp/dhcpd6.conf
dhcpd6.leases: /var/lib/dhcpd/dhcpd6.leases
dhcpd6.log: /var/log/dhcpd6.log
ip6tables: /etc/sysconfig/ip6tables.conf
ntp: /etc/ntp.conf
radvd: /etc/radvd.conf
rc.local: /etc/rc.d/rc.local
rsyslog: /etc/rsyslog.conf
scriptit: /root/scriptit
sysctl: /etc/sysctl.conf

```
#!/bin/bash
# ISC DHCPv6 daemon
# chkconfig: 345 20 80
# description: ISC DHCPv6 daemon
# processname: dhcpd6

DAEMON_PATH="/usr/sbin/"

DAEMON=dhcpd
DAEMONOPTS="-6 -cf /etc/dhcp/dhcpd6.conf"

NAME=dhcpd6
DESC="ISC DHCPv6 daemon"
PIDFILE=/var/run/$NAME.pid
SCRIPTNAME=/etc/init.d/$NAME

case "$1" in
start)
        printf "%-59s" "Starting $NAME:"
        cd $DAEMON_PATH
        PID=`$DAEMON $DAEMONOPTS > /dev/null 2>&1 & echo $!`
        sleep 0.35;
        if [ -f $PIDFILE ]; then
       PID=`cat $PIDFILE`
       if [ -z "`ps axf | grep ${PID} | grep -v grep`" ]; then
                                                                rm -f $PIDFILE
          echo -e "[\e[0;31mFAILED\e[0m]"
       else
          echo -e "[\e[0;32m  OK  \e[0m]"
       fi
    else
       echo -e "[\e[0;31mFAILED\e[0m]"
    fi
;;
status)
    if [ -f $PIDFILE ]; then
       PID=`cat $PIDFILE`
       if [ -z "`ps axf | grep ${PID} | grep -v grep`" ]; then
          printf "%s\n" "Process dead but pidfile exists"
       else
          echo "$NAME (pid $PID) is running..."
       fi
    else
       echo "$NAME is stopped"
```

```
        fi
;;
stop)
        if [ -f $PIDFILE ]; then
                                                cd $DAEMON_PATH
                                                PID=`cat $PIDFILE`
                                                if [ -z $PID ]; then
                                                                kill -HUP $PID
                                                fi
                                                printf "%-59s" "Stopping $NAME:"
           echo -e "[\e[0;32m  OK  \e[0m]"
           rm -f $PIDFILE
        fi
;;

restart)
           $0 stop
           $0 start
;;

*)
        echo "Usage: $0 {status|start|stop|restart}"
        exit 1
esac
```

```
# DHCP for IPv6 Server Configuration file.
#   see /usr/share/doc/dhcp*/dhcpd6.conf.sample
#   see 'man 5 dhcpd.conf'
#   run 'service dhcpd6 start' or 'dhcpd -6 -cf /etc/dhcp/dhcpd6.conf'

# Lease tiedoston sijainti vaihdettu koska SELinux estää sitä muuten toimimasta
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";

# Lease aikojen määritys (24h)
default-lease-time 86400;
max-lease-time 86400;
preferred-lifetime 86400;

# Log facilityksi määritetty local7, loki menee tiedostoon /var/log/dhcpd6.log
log-facility local7;

# Tarvittavat lisätiedot lease tiedostoon sekä lokiin
on commit {
      if option dhcp6.ia-na = option dhcp6.ia-na{
                        set iana = binary-to-ascii(16,16,":",substring(suffix(option dhcp6.ia-na,24),0,16));
          }
          if option dhcp6.ia-pd = option dhcp6.ia-pd{
               set iapd = binary-to-ascii(16,16,":", suffix(option dhcp6.ia-pd,16));
                        set pdsize = binary-to-ascii(10,8,":",substring(suffix(option dhcp6.ia-pd,17),0,1));
          }
          # docsis optio, ei tarvita
          # set cm = v6relay(1, (binary-to-ascii(16, 8, ":", option docsis.cm-mac-address)));
          set ifname = v6relay(1, option dhcp6.interface-id);
          set lla = (binary-to-ascii(16, 8, ":", suffix(option dhcp6.client-id, 6)));

      if option dhcp6.ia-na = option dhcp6.ia-na{
                        log(info, concat("ONCOMMIT IA_NA: ", iana, " To: ", lla, " INT: ", ifname));
          }
          if option dhcp6.ia-pd = option dhcp6.ia-pd{
                        log(info, concat("ONCOMMIT IA_PD: ", iapd, "/", pdsize, " To: ", lla, " INT: ", ifname));
          }
   }

# DHCPv6 palvelimen pitää kuulua johonkin subnet6 alueeseen
# Tältä alueelta ei jaeta osoitteita, tarvitaan palvelinta varten
subnet6 2a00:b9a0:302::/48 {
          # Alue clienteille (esim. reitittimen wan portille)
      # range6 2a00:b9a0:302:fffc:0000:0000:0000:0001 2a00:b9a0:302:fffc:ffff:ffff:ffff:fff0;
}
```

```
# VLANTesti1 osoitealue
subnet6 2a00:b9a0:350::/48 {
	# Alue clienteille (esim. reitittimen wan portille)
	# range6 2a00:b9a0:350:fffc:0000:0000:0000:0001 2a00:b9a0:350:fffc:ffff:ffff:ffff:fff0;

	# Alue clienteille jotka pyytävät väliaikaista osoitetta
	# range6 2001:db8:0:1::/64 temporary;

	# Lisäoptiot
	option dhcp6.name-servers fec0:0:0:1::1;
	option dhcp6.domain-search "domain.example";

	# Reitittimille jaettava prefix (eli reitittimen lan portille)
	prefix6 2a00:b9a0:350:: 2a00:b9a0:350:fffc:: /64;

	# Fixed address hostille
	# host specialclient {
	   #		host-identifier option dhcp6.client-id 00:01:00:01:4a:1f:ba:e3:60:b9:1f:01:23:45;
	#   fixed-address6 2001:db8:0:1::127;
	   # }
}

# VLANTesti2 osoitealue
subnet6 2a00:b9a0:360::/48 {
	# Alue clienteille (esim. reitittimen wan portille)
	# range6 2a00:b9a0:360:fffc:0000:0000:0000:0001 2a00:b9a0:360:fffc:ffff:ffff:ffff:fff0;

	# Alue clienteille jotka pyytävät väliaikaista osoitetta
	# range6 2001:db8:0:1::/64 temporary;

	# Lisäoptiot
	option dhcp6.name-servers fec0:0:0:1::1;
	option dhcp6.domain-search "domain.example";

	# Reitittimille jaettava prefix (eli reitittimen lan portille)
	prefix6 2a00:b9a0:360:: 2a00:b9a0:360:fffc:: /64;

	# Fixed address hostille
	# host specialclient {
	   #		host-identifier option dhcp6.client-id 00:01:00:01:4a:1f:ba:e3:60:b9:1f:01:23:45;
	#   fixed-address6 2001:db8:0:1::127;
	   # }
}
```

```
ia-pd "\001\000\000\000\000\003\000\001\220\366RD\200i" {
  cltt 3 2014/02/26 07:47:51;
  iaprefix 2a00:b9a0:360:fffc::/64 {
    binding state active;
    preferred-life 86400;
    max-life 86400;
    ends 4 2014/02/27 07:47:51;
    set lla = "90:f6:52:44:80:69";
    set ifname = "VlTesti2";
    set pdsize = "64";
    set iapd = "2a00:b9a0:360:fffc:0:0:0:0";
  }
}

ia-pd "\001\000\000\000\000\003\000\001\220\366RD\200i" {
  cltt 3 2014/02/26 07:49:30;
  iaprefix 2a00:b9a0:350:fffb::/64 {
    binding state active;
    preferred-life 86400;
    max-life 86400;
    ends 4 2014/02/27 07:49:30;
    set lla = "90:f6:52:44:80:69";
    set ifname = "VlTesti1";
    set pdsize = "64";
    set iapd = "2a00:b9a0:350:fffb:0:0:0:0";
  }
}

ia-pd "\001\000\000\000\000\003\000\001\000\"\007 GC" {
  cltt 3 2014/02/26 08:35:15;
  iaprefix 2a00:b9a0:350:fffa::/64 {
    binding state active;
    preferred-life 86400;
    max-life 86400;
    ends 4 2014/02/27 08:35:15;
    set lla = "0:22:7:20:47:43";
    set ifname = "VlTesti1";
    set pdsize = "64";
    set iapd = "2a00:b9a0:350:fffa:0:0:0:0";
  }
}
```