jamk.fi

IPTV-PALVELUN SUUNNITTELU SPIDERNETIIN

Tommi Pitkäaho

Opinnäytetyö
Lokakuu 2015

Tietotekniikan koulutusohjelma
Tekniikan ja liikenteen ala

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

jamk.fi

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Pitkäaho, Tommi	Opinnäytetyö	26.10.2015
	Sivumäärä 68 + 16	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: X

Työn nimi
IPTV-palvelun suunnittelu SpiderNetiin

Koulutusohjelma
Tietotekniikan koulutusohjelma

Työn ohjaaja(t)
Mika Rantonen
Antti Häkkinen

Toimeksiantaja(t)
Jyväskylän ammattikorkeakoulu
Saharinen, Karo

Tiivistelmä

Toimeksiannon tavoitteena oli suunnitella ja selvittää toteutusmahdollisuuksia palveluntarjoajan tason IPTV-palvelulle Jyväskylän ammattikorkeakoulun SpiderNet-laboratorioympäristössä. Opinnäytetyössä tuli selvittää IPTV-palvelun toteuttamiseen tarvittavia laitteita sekä ohjelmistoja, joiden pohjalta voitaisiin tehdä tulevaisuudessa mahdollisia hankintaehdotuksia.

Opinnäytetyössä perehdyttiin IPTV-liikenteen teoriaan varsinkin multicast-liikenteen, IGMP-protokollan sekä erilaisten digitaalisiin televisiolähetyksiin liittyvien standardien osalta.

Opinnäytetyön lopputuloksena saatiin kattava katsaus digitaalisiin televisiolähetyksiin ja IPTV:n toimintaan. Laite- ja ohjelmistoehdotuksia laadittiin testitulosten sekä aiemmin dokumentoitujen IPTV-implementaatioiden pohjalta. Tekijänoikeuslakien vuoksi DVB-tekniikkaa käyttäviä televisiolähetyksiä jakavia laitteita ei käytetty. Palvelun toimintaa simuloitiin lähettämällä yhdeltä palvelimelta eri televisiokanavia esittäviä videoklippejä sekä IP-kameran videokuvaa multicast-tekniikoita käyttäen.

Avainsanat (asiasanat)
IPTV, Multicast, IGMP, DVB

Muut tiedot

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

jamk.fi

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Description

Author(s)	Type of publication	Date
Pitkäaho, Tommi	Bachelor's thesis	26.10.2015
		Language of publication: Finnish
	Number of pages 68 + 16	Permission for web publication: x

Title of publication
Designing an IPTV service for SpiderNet

Degree programme
Information Technology

Tutor(s)
Rantonen, Mika
Häkkinen, Antti

Assigned by
JAMK University of Applied Sciences
Saharinen, Karo

Abstract

The objective of the thesis was to design a service provider grade IPTV service for the SpiderNet networking laboratory environment in JAMK University of Applied Sciences. The research done could then be used as a reference for possible future software and hardware acquisitions for the laboratory environment.

The theory of the thesis discusses the methods and standards of IPTV regarding multicast traffic, IGMP protocol and various technologies regarding digital television broadcasting.

The end result was a review of the technologies involved in digital television and IPTV broadcasting, and suggestions on how to apply these in building an IPTV service in a campus networking laboratory. Hardware and software recommendations were given based on performed tests and studied IPTV implementation cases. Due to copyright issues, an actual DVB-compliant IPTV-environment could not be implemented in SpiderNet. Instead, the IPTV service was simulated using multicasting sample video and live IP-camera footage from a single server through the laboratory network to a client device.

Keywords/tags (subjects)

IPTV, Multicast, IGMP, DVB

Miscellaneous

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Sisältö

Kuviot

Taulukot

Lyhenteet

3DES	Triple Data Encryption Algorithm
AES	Advanced Encryption Algorithm
ARP	Address Resolution Protocol
ATSC	Advanced Television Systems Committee
BGP	Border Gateway Protocol
BPSK	Binary Phase Shift Keying
CAM	Conditional Access Module
CAS	Conditional Access System
CBR	Constant Bit Rate
CI	Conditional Interface
CP	Copy Protection
CSA	Common Scrambling Algorithm
DMB	Digital Multimedia Broadcasting
DRM	Digital Right Management
DTMB	Digital Terrestrial Multimedia Broadcast
DVB	Digital Video Broadcasting
ECM	Entitlement Control Message
EIGRP	Enhanced Interior Gateway Routing Protocol
EMM	Entitlement Management Message
FEC	Forward Error Correction

IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IP	Internet Protocol
IPTV	Internet Protocol Television
IPv4	Internet Protocol Version 4
IRD	Integrated Receiver / Decoder
ISDB	Integrated Services Digital Broadcasting
LAN	Local Area Network
MPEG	Moving Picture Experts Group
NAT	Network Address Translation
OFDM	Orthogonal Frequency-division multiplexing
PIM	Protocol Independent Multicast
PPI	Pixels per Inch
QoS	Quality of Service
RP	Rendezvous Point
RTP	Real Time Transport Protocol
RTSP	Real Time Streaming Protocol
TCP	Transmission Control Protocol
TDM	Time-division Multiplexing
TS	Transport Stream
TTL	Time to Live

UDP	User Datagram Protocol
VBR	Variable Bit Rate
VLAN	Virtual LAN
VoD	Video on Demand
VTP	VLAN Trunking Protocol
XOR	Exclusive Or

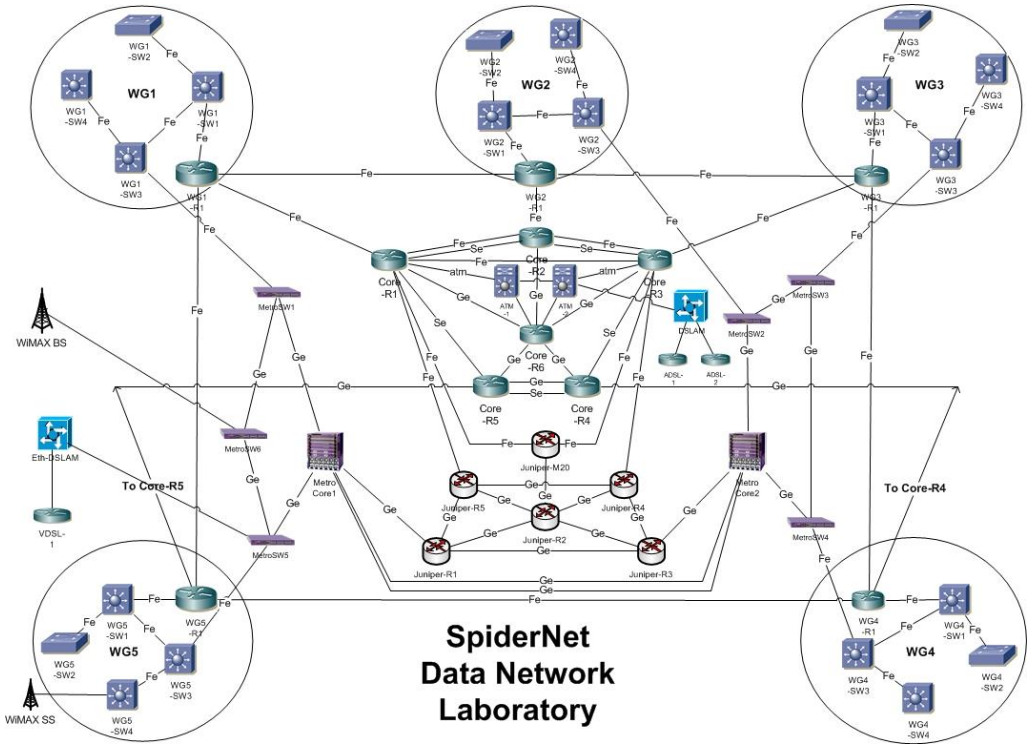
1 Lähtökohdat

1.1 SpiderNet-laboratorioympäristö

SpiderNet on Jyväskylän ammattikorkeakoulun Lutakon kampuksella sijaitseva laboratorioympäristö, joka keskittyy tietoverkkoihin. Laboratorioverkossa voi työskennellä operaattoritason tietoliikennelaitteistolla ja -ohjelmistoilla, joiden tarkoituksena on tukea tietoverkkoalan opiskelijoiden opintoja ja edistää käytännön harjoittelua. SpiderNet-verkossa on valmistajista Airspan Networksin, Cisco Systemsin, Juniper Networksin, Extreme Networksin sekä Zhonen verkkolaitteita. Laboratorioympäristö tukee laajaa valikoimaa tiedonsiirtotekniikoita ja -protokollia Access-, Local Area Network (LAN)- sekä Core-tasoilla. SpiderNetissä päätelaitteiden emulointi on toteutettu VMWare ESXi -virtualisointipalvelimen kautta, jossa jokainen laboratorioverkon viidestä työryhmästä sisältää yhden Linux-palvelimen ja kaksi Windows-käyttöjärjestelmän työasemaa. (SpiderNet 2009.)

1.2 Toimeksianto ja opinnäytetyön tavoitteet

Toimeksiannon tavoitteena oli selvittää mahdollisuudet toteuttaa Jyväskylän ammattikorkeakoulun Spidernet-laboratorioympäristössä Internet Protocol Television (IPTV) -jakeluratkaisu. Työlle ilmeni tarve, kun keväällä 2015 sen aikainen IPTV-implementaatio ei ollut toimiva. Työssä päätettiin selvittää uusi ratkaisu headend-palvelimelle, joka toimisi IPTV-lähetysten jakajana laboratorioverkossa päätelaitteille. Tarkoituksena oli etsiä open source -ratkaisuja, joten palvelimen käyttöjärjestelmäksi päätettiin alusta asti sopivimmaksi Linux-pohjainen käyttöjärjestelmä. IPTV-liikenteen siirtoon käyttettäisiin SpiderNet-verkon infrastruktuuria itse konfiguroiduilla reititysprotokollilla (ks. Kuvio 1).



Kuvio 1. SpiderNet-laboratorioverkon topologia (SpiderNet 2009)

Opinnäytetyön tutkimusmenetelmiksi päätettiin pääasiassa perehtyä aiemmin toteutettuihin ja raportoituihin IPTV-ratkaisuihin. Erityisesti avoimen lähdekoodin käyttöjärjestelmiä ja headend-ohjelmistoja käyttäviä implementaatioita tutkittiin. Toteutuksen osalta kirjallinen aineisto tarjosi rajatusti hyötyä, joten suurin osa implementaatiomenetelmistä perustui Internetin kautta käyttäjien jakamiin ratkaisuihin sekä työn tekijän laatimiin päätelmiin.

2 IPTV-liikenteen teoriaa

2.1 Mikä on IPTV?

IPTV on nykyaikainen tapa lähettää ja vastaanottaa multimediaa IP-protokollaa käyttäen perinteisemmän antenni-, satelliitti- tai kaapeliverkon sijaan. Puhekielessä IPTV:stä voidaan myös käyttää nimitystä laajakaistatelevisio. Tietoverkkoon liittyvä televisiopalvelu tarjoaa käyttäjille uudenlaiset Video on Demand- (VoD) ja verkkotallennuspalvelut. Muista televisiopalveluista tutut maksullisten kanavien tai kanavapakettien tarjoaminen on mahdollista myös IPTV-implementaatioissa käyttäen salattuja tiedonsiirtokanavia. Kanavien salaus tulee yleensä palveluntarjoajalta, minkä takia niiden luvaton murtaminen omatekoisissa IPTV-palveluissa on vaikeaa tai mahdotonta salauksesta riippuen (Mäkinen 2013, 44–45).

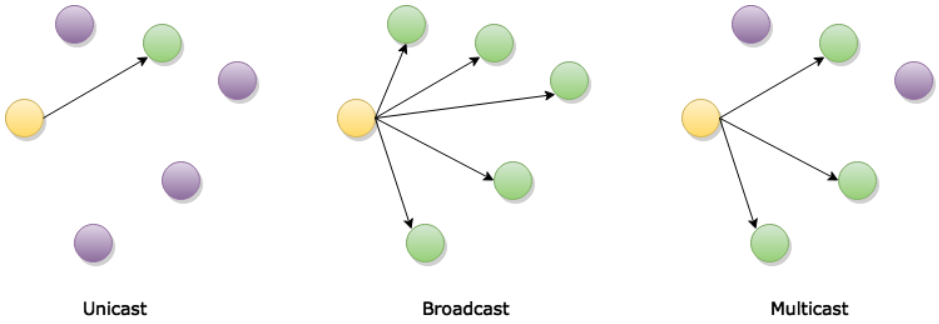
Käyttäjän näkökulmasta IPTV ei eroa merkittävästi muista televisiopalveluista, jos se on operaattorin hallitsema. Laitteistovaatimuksena operaattorin tarjoamissa IPTV-implementaatioissa käyttäjälle on IPTV-liikennettä käsittelevä reititin. Reititinlaite mainostaa tarjolla olevia televisiopalveluita, joita käyttäjä voi katsoa television kautta lähiverkkoon kytketyn IPTV-sovittimen avulla. IPTV:tä voi katsoa myös muilla laitteilla kuten pöytätietokoneella tai älypuhelimella, mikäli laite on kytketty lähiverkkoon ja sisältää IPTV-lähetysten katseluun soveltuvan ohjelmiston. IPTV-sovitin voi olla myös suoraan kytkettynä palvelimeen, joka hoitaa myös lähiverkon reitityksen. Käyttäjien omissa IPTV-ratkaisuissa käytetään DVB-standardia noudattavia lisälaitteita, joilla voidaan vastaanottaa ja jakaa TV-lähetyksiä monista eri lähteistä. (O'Driscoll 2008, 2-3.)

IPTV:n suorituskykyyn vaikuttavat verkon tiedonsiirtokapasiteetti ja IPTV-lähetyksiä vastaanottavien päätteiden sekä verkon muiden laitteiden määrä ja kuormitus. IPTV-tiedonsiirron nopeusvaatimus riippuu myös siirrettävän datan laadusta audion ja videon suhteen. Jotta IPTV:n laatu pysyy riittävänä, palveluntarjoaja saattaa jakaa asiakkaalle allokoidun kaistannopeuden IPTV- ja muun dataliikenteen välille. Tietyn latausnopeuden tarjoava verkkoyhteys voi olla palveluntarjoajan puolesta asetettu Quality of Service (QoS) -sääntöjen mukaisesti varaamaan jokin vähimmäisraja IPTV-liikenteelle. (Qiu 2010.)

2.2 Multicast

2.2.1 IPv4-tiedonsiirron menetelmät

IP-liikenteen eli pakettikytkentäistä Internet Protocol (IP) -protokollaa käyttävän dataliikenteen siirto IPv4-verkoissa jaetaan yleisesti kolmeen eri toimintatapaan (ks. Kuvio 2).



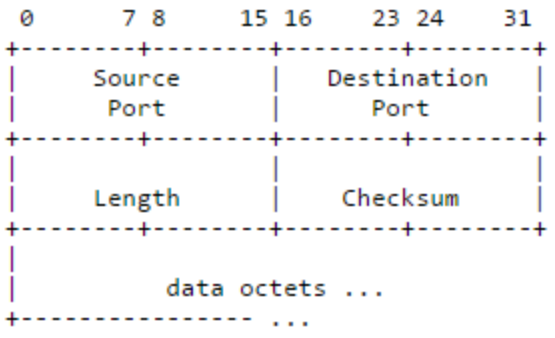
Kuvio 2. IPv4-tiedonsiirtotekniikat

Unicast on lähettäjältä yhdelle vastaanottajalle suunnattu siirtotapa, jossa tiedonsiirrolle varataan n.k. pisteestä pisteeseen eli point-to-point -yhteys. Yksittäisten yhteyksien mahdollinen korkea lukumäärä varsinkin isommissa verkoissa kuormittaa verkon sietokykyä eikä siten ole optimaalinen toimintatapa siirtää sama viesti usealle vastaanottajalle. Unicast-liikenne on yksinkertaisuudestaan huolimatta laajalti käytetty yleisissä Transmission Control Protocol (TCP) -protokollaa käyttävissä yhteyksissä, joissa pyritään varmistamaan viestin eheyden säilyvyys sekä tietoturvallisuus lähettäjältä vastaanottajalle. (Fairhurst 2009.)

Broadcast-liikenteessä lähettäjän viesti siirretään jokaiselle mahdolliselle vastaanottajalle lähiverkossa. Broadcast-viestit ovat hyödyllisiä esimerkiksi verkkonaapuruuksia määritettäessä laitetasolla Address Resolution Protocol (ARP) -protokollalla, jolloin lähettävä laite pystyy rakentamaan verkkotopologian kysymällä jokaiselta verkon laitteelta niiden sijaintia. Broadcast-liikenne voi kuitenkin aiheuttaa ylimääräistä liikennettä verkossa, kun jokaiselle vastaanottajalle lähetetään viesti, jolla osa vastaanottajista ei välttämättä tee mitään (RFC 966). Broadcast-liikenteelle on standardin mukainen oma IP-osoite jokaisessa verkossa, esim. 192.128.128.0/24-verkossa 192.128.128.255/24 on broadcast-osoite (RFC 919).

IPv4 multicastissa vastaanottajat voivat liittyä tiettyihin Internet Group Membership Protocol (IGMP) -ryhmiin, joissa lähettäjä eli multicastia tukeva reititin tai ryhmän muu laite siirtävät dataa. Usean vastaanottajan ryhmiä käyttämällä voidaan lähettää keskitetysti haluttu data usealle laitteelle ilman että käytetään yhtä kanavaa jokaiselle vastaanottajalle. IGMP-ryhmiin liittyminen ja niistä poistuminen toimii dynaamisesti, ja sama käyttäjä voi kuulua useaan ryhmään samanaikaisesti. Myöskään IGMP-ryhmän jäsenten maksimimäärä ei ole rajoitettu, ellei sitä tarkoituksella haluta määrittää. Multicast-liikenne tukee myös reititystä toisin kuin broadcast, joten se soveltuu myös useamman verkon yli ulottuvaan liikenteeseen. Multicastille on broadcastin tapaan varattu oma osuus IPv4-osoiteavaruudesta. (Semeria & Maufer n.d., 1.)

Broadcast- ja multicast-liikenteessä tiedonsiirtoon käytetään yleensä yhteydellisen TCP:n sijaan yhteydetöntä User Datagram Protocol:ia (UDP). UDP-liikenteessä käytetään minimaalista varmistusta pakettien siirtymisestä vastaanottajalle, minkä takia se on erittäin kevyt ja nopea tapa siirtää dataa verkossa laajalle käyttäjämäärälle esimerkiksi IPTV-palvelussa. UDP-kehyksen rakenteeseen kuuluvat hyötykuorman lisäksi ainoastaan protokollan käyttämät lähde- ja vastaanottoportit, paketin pituus bitteinä sekä tarkistussumma (ks. Kuvio 3). Tarkistussumma on vapaaehtoinen, eivätkä siinä esiintyneet virheet aiheuta pakettien uudelleenlähetystä kuten TCP:ssä. (RFC 768.)



Kuvio 3. UDP-kehys (RFC 768)

2.2.2 Protocol Independent Multicast

Protocol Independent Multicast (PIM) eli protokollavapaa multicast ei käytä omaa reititystopologiaansa, vaan saa sen alla kulkevalta reititysprotokollalta. PIM voidaan asettaa toimimaan neljässä eri moodissa.

Dense Modella (PIM-DM) kaikki multicast-liikenne lähetetään kaikille rajapinnoille reitittimeltä. PIM Prune -viestillä voidaan kieltäytyä multicast-liikenteestä. Jos reitittimen kaikille porteille tulee PIM Prune -viesti, menee reititinkin Prune -tilaan. Dense Mode -tilaa käytetään silloin kun verkossa on useita vastaanottajia multicast-liikenteelle. Palvelun laadun ylläpitämiseksi verkon floodausta tulee rajoittaa mahdollisimman hyvin. PIM-DM-implementaatioissa käytetään yhtä tai useampaa Rendezvous Point (RP) -verkkolaitetta, joka toimii multicast-lähetysten mainostajana. (RFC 3973.)

Sparse Mode (PIM-SM) -toimintoa käytetään silloin, kun verkossa ei ole paljon vastaanottajia multicast-paketeille. Sparse Mode -moodissa multicast-liikennettä lähetetään vain niihin portteihin, joista reititinlaite on saanut PIM Join -viestin. Kohdereititin eli Designated Router lähettää tietyn väliajoin Join ja Prune -viestejä multicast-ryhmän Rendezvous Point -reitittimen jokaiselle ryhmälle, joissa on aktiivisia jäseniä. Jokaisen reitittimen tulee olla liittynyt RP-reitittimeen, jotta ne voivat lähettää multicast-paketteja. (RFC 4601.)

Source Specific (PIM-SSM) -moodissa multicast-viestejä lähetetään vain niille vastaanottajille, jotka ovat pyytäneet multicast-liikennettä sen lähettäjän osoitteesta. Tällä menetelmällä voidaan vähentää verkon kuormaa huomattavasti. Source Specific vaatii toimiakseen IGMPv3-viestejä tukevan lähiverkon (ks. luku 2.3.5). Source-Specific Multicast ei vaadi toimiakseen Rendezvous Point -reititintä. (RFC 4608.).

Bidirectional eli kaksisuuntainen PIM (BIDIR-PIM) on suunniteltu käytettäväksi verkoissa, joissa on useita yhtaikaisia vastaanottajia ja lähettäjiä jotka keskustelevat keskenään. BIDIR-PIM voi aiheuttaa verkossa muihin multicast-moodeihin verrattuna enemmän resursseja laajemman multicast-reititystaulun takia. BIDIR-PIM vaatii toimiakseen Dense- ja Sparse Moden tapaan Rendezvous Point -reitittimen, johon lähiverkkojen Designated Router -reitittimet ilmoittavat multicast-liikenteestä. (RFC 5015.)

2.2.3 IPv4-osoitteistus

Internet-protokollien standardointivastaava Internet Engineering Task Force (IETF) -organisaatio on määrittänyt nykyaikaisen luokattoman IPv4-osoiteavaruuden, joka voidaan jakaa kolmeen ryhmään:

- julkiset osoitteet
- yksityiset osoitteet
- erityisosoitteet, t.s. varatut osoitteet

Julkiset osoitteet on tarkoitettu Internetin kautta reititettäville kohteille, minkä vuoksi osoitteiden tulee olla uniikkeja toisistaan. Julkiset osoitteet ovat yleensä sidottuja nimettyyn toimialuenimeen, mikä helpottaa käyttäjien pääsyä palveluun esim. verkkoselaimen kautta. Julkisten osoitteiden saaminen omaan käyttöön on useimmiten tavallisille käyttäjille maksullista, ja niiden jakamisen hoitaa käyttäjän palveluntarjoaja. Osoitteistukseltaan julkisia osoitteita ovat kaikki muut IP-osoitteet paitsi määritetyt yksityiset sekä erityisesti varatut osoitteet. (RFC 1918.)

Yksityiset osoitteet on tarkoitettu lähiverkkojen sisäiseen liikenteeseen eikä niitä voida reitittää julkisten verkkojen läpi ilman osoitteen käännöstä esimerkiksi Network address translation (NAT) -metodia käyttämällä. NAT:ia käyttämällä yksityinen osoite voidaan piilottaa palveluntarjoajan määrittämän julkisen osoitteen alaiseksi ennen reitittämistä Internetin kautta. Yksityisissä suljetuissa verkoissa voidaan myös vapaasti käyttää julkisia osoitteita, mikäli verkon ei ole tarkoitus yhdistyä Internetiin. (RFC 1631.)

Yksityiset osoitteet on määritetty käyttämään kolmea osoiteblokkia (RFC 1918):

- 10.0.0.0 – 10.255.255.255
- 172.16.0.0 – 172.31.255.255
- 192.168.0.0 – 192.168.255.255

Varattuihin tai erityisosoitteisiin kuuluu monia osoitteita, joita käyttämällä voidaan määrittää pakettien vastaanottajaksi jokin muu kuin yksittäinen IP-osoitteen omaava laite. Erityisosoitteet sisältävät myös monia IETF:n määrittämiä kokeellisia osoitteita, joilla ei ole varsinaista julkista käyttötarkoitusta. Tärkeimpiä varattuja osoitteita ovat loopback-, multicast- ja broadcast-osoitteet. Varattuja osoitteita ovat seuraavat:

- 0.0.0.0/8, lähdeosoitteena käytetty osoite, jolla viitataan kyseiseen verkkoon (RFC 1700)
- 100.64.0.0/10, palveluntarjoajien sisäisille verkoille
- 127.0.0.0/8, loopback-osoite
- 169.254.0.0/16, IP-osoitteeton liikenne
- 192.0.0.0/24, IETF:n varaama verkko
- 192.0.2.0/24, kokeellinen verkko
- 192.88.99.0/24, 6to4 anycast-liikenne
- 192.18.0.0/15, kokeellinen verkko
- 198.51.100.0/24, kokeellinen verkko
- 203.0.113.0/24, kokeellinen verkko
- 224.0.0.0/4, multicast-liikenne
- 240.0.0.0/4, varattu tulevaisuutta varten

- 255.255.255.255/32, limited broadcast -osoite, joka on varattu tulevaisuutta varten. (RFC 1918; RFC 5735.)

2.2.4 Virtuaalilähiverkot

Virtual Local Area Network (VLAN) eli virtuaalilähiverkoilla voidaan jakaa fyysinen tietoliikenneverkko useisiin loogisiin verkkoihin. Käyttäjien jakaminen pienempiin verkkoihin helpottaa usein verkkojen hallintaa sekä luo selkeämmän kuvan eri käyttäjäryhmistä ja niiden tarpeista. Virtuaalilähiverkoiksi voidaan jakaa mm. toimiston eri osastojen lähiverkot, jotka eivät tule olemaan tietoisia muista rakennuksen virtuaalilähiverkoista ja joilla ei ole oikeuksia toimia niissä mutta ne jakavat yhteisen fyysisen polun ulos asiakkaan omasta verkosta. (Virtual LAN: Applications and Technology 2004, 3.)

VLANien hallintaa varten luodaan usein oma määrätty virtuaalilähiverkko, jolle on määrätty laajemmat oikeudet muiden virtuaalilähiverkkojen hallintaa varten. Virtuaalilähiverkkojen käytön hyötyjä ovat pääsyn rajoittaminen sekä kulujen minimoiminen. Ilman virtuaalilähiverkkojen käyttöä esim. kolmen toimiston osaston yhdistäminen yhteen ulkoverkkoon liittyvään reitittimeen tarvittaisiin jokaiselle osastolle omat kytkimet sekä kaapelit. Virtuaalilähiverkot toimivat OSI-mallin siirtokerroksella, laitteina on reitittimiä ja/tai kytkimiä. Kytkimillä voidaan siirtää virtuaalilähiverkkojen sisäistä liikennettä, mutta virtuaalilähiverkosta toiseen dataa siirrettäessä tarvitaan reititystä joko reitittimellä tai OSI-mallin verkkokerroksella eli Layer 3 -tasolla toimivalla kytkimellä. Datan siirtoon verkkojen sisällä reitittimien ja/tai kytkimien välillä käytetään trunk-yhteyttä, jossa yksi tai useampi looginen yhteys siirtyy yhdessä fyysisessä linkissä. (Virtual LAN: Applications and Technology 2004, 3-4.)

2.3 IGMP-protokolla

2.3.1 Yleistä

Internet Group Membership Protocol (IGMP) -protokolla mahdollistaa multicast-liikenteen verkossa ja on toimintatapa, jolla mm. IPTV-liikenne toimii. Protokolla toimii IGMP-ryhmien avulla, joita esimerkiksi reititinlaite voi tarjota siihen liittyviin verkkoihin. Verkon käyttäjälaitteet voivat liittyä haluttuihin ryhmiin. IPTV:n tapauksessa jokin tietty televisiokanava voi vastata jotakin IGMP-ryhmää, johon liittymällä käyttäjälaiteella voidaan katsoa haluttua kanavaa. IGMP-liikenteen viestejä analysoimalla voidaan selvittää lähiverkon IPTV-liikenteen mahdollisia ongelmia. (RFC 966.)

Protokolla toimii TCP/IP-stäkin osana, joten se vaatii IP-osoitteistuksen jokaiselle multicastiin osallistuvalle laitteelle sekä reititysmahdollisuuden lähetyslaitteelta. IGMP-protokolla on laajalti tuettu useimmissa kuluttajatasonkin käyttöjärjestelmissä. IGMP:n yleisiä toimintavaatimuksia reititinlaitteelta ovat

- routing table eli reititystaulu, johon kirjataan reitittimeen liittyvien verkkojen jäsenten IP-osoitteet ja sijainti verkkotopologiassa
- network membership table eli verkon jäsenyystaulu, johon multicast-ryhmiin kuuluvien jäsenten verkon tallennetaan
- local host membership table eli paikallinen jäsenyystaulu, johon kirjataan reititinlaitteeseen suoraan kiinnittyvien verkkojen multicast-ryhmät (RFC 966.)

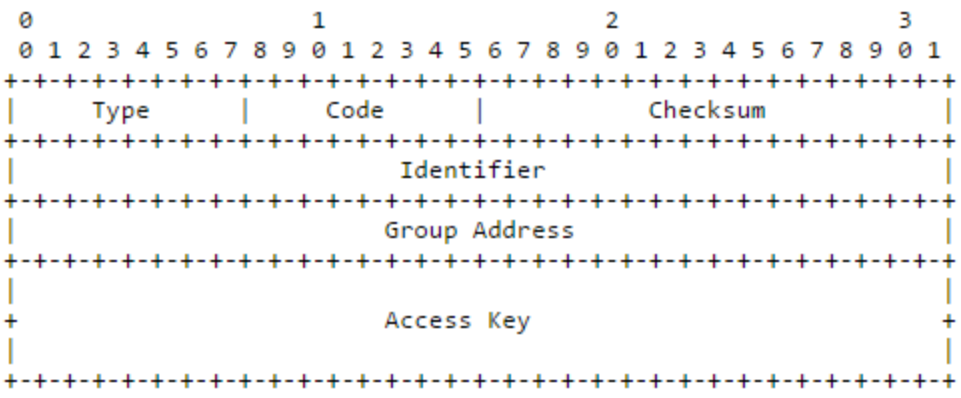
IGMP-liikenteen toiminta perustuu IGMP-viesteihin. Protokollasta on laadittu neljä eri versiota, joiden mukana on tuotu mahdollisuuksia yhä modernimpia multicast-ratkaisuja varten. IGMP-viestien rakenne on muuttunut jokaisessa versiossa, mutta perustoiminnallisuuden osalta uudemmat versiot ovat myös yhteensopivia aiempien versioiden kanssa (RFC 3376).

2.3.2 IGMP versio 0

Ensimmäinen versio IGMP-protokollan toiminnasta laadittiin ehdotettavaksi vuonna 1985. Luonnoksessa määritettiin multicast-liikenteen periaate siten, että sen tarkoituksena on siirtää IP-paketteja käyttäjäryhmälle, jolla on yksi tietty IP-osoite. Ryhmälle määrätyn paketin vastaanottaa jokainen IGMP-ryhmän jäsen. Luonnos määritti myös IGMP-ryhmien rakenteen siten, että ryhmien IP-osoitteet voivat olla joko kiinteitä tai väliaikaisia. IGMP:n tuelle määritettiin kolme eri tasoa:

- Taso 0: ei tukea multicast-liikenteelle
- Taso 1: laite voi toimia multicast-liikenteen raportoijana tai resurssien sijaintina, mutta ei kykene liittymään IGMP-ryhmään.
- Taso 2: täysi tuki IGMP-protokollalle (RFC 988.)

IGMP:n ensimmäinen versio määritti IGMP-viestin rakenteen (ks. Kuvio 4).



Kuvio 4. IGMPv0-viestin rakenne (RFC 988)

Kaikki IGMP-viestit kapseloidaan IP-pakettiin IP-protokollatunnuksella kaksi. Type-kentällä määritetään viestille tarkoitus, joita on neljä sekä joko Request tai Reply -muuttuja:

- 1 = Create Group Request, 2 = Create Group Reply
- 3 = Join Group Request, 4 = Join Group Reply
- 5 = Leave Group Request, 6 = Leave Group Reply

- 7 = Confirm Group Request, 8 = Confirm Group Reply

Code-kenttä määrittää viestin olemaan joko julkinen (bitti 0) tai yksityinen (bitti 1), jota käytetään vain jos viestityyppinä on Create Group. Reply-viesteissä Code-kenttä antaa vastauksen lähetetylle pyynnölle:

- 0 = pyyntö myönnetty
- 1 – 4 = pyyntö hylätty
    - 1 = ei resursseja
    - 2 = väärä code
    - 3 = väärä ryhmän osoite
    - 4 = väärä pääsyavain
- 5 – 255 = pyyntö keskeytynyt, yritetään uudelleen *x* sekunnin kuluttua.

Checksum on viestin eheyden tarkistuskenttä. Identifier-kentän arvoja voivat olla:

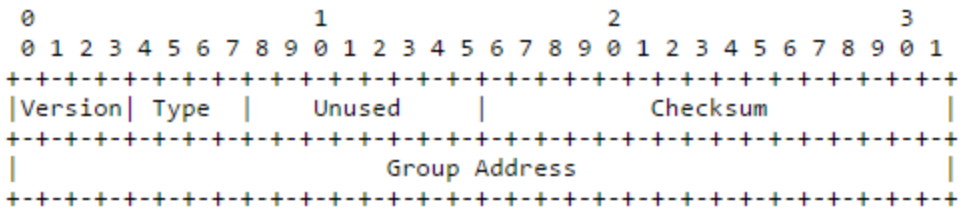
- Confirm Group Request -viestissä arvo 0, muissa Request-tyypin viesteissä saman lähettäjän viestien järjestysluku
- Reply-viesteissä tulee vastata lähetetyn Request-viestin arvoa

Group Address -kentällä tunnistetaan IGMP-ryhmän IP-osoite. Create Group Reply -viesteissä kenttä sisältää luodulle ryhmälle määritetyn IP-osoitteen. Reply-tyypin viesteissä kentän tulee vastata Request-tyypin samaa arvoa.

Access Key -kentässä määritetään 64-bittinen tunnistusavain, mikäli ryhmä on Create Group Request -tyypin viestissä määritetty yksityiseksi. Muissa tapauksissa kentän arvo on nolla. (RFC 988.)

2.3.3 IGMP versio 1

IGMP versio 1 laajenti protokollan mekanismeja ja määritti IGMP-ryhmien sisäisiä rooleja siten, että ryhmän ylläpito siirtyi multicast-reitittimiltä myös käyttäjille. IGMPv1 toi mukanaan myös uuden, yksinkertaisemman pakettirakenteen viesteille (ks. Kuvio 5). (RFC 1054.)

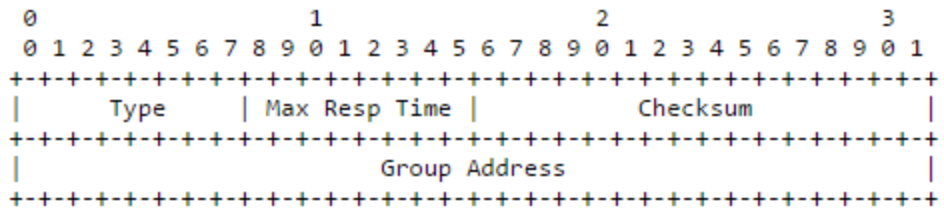


Kuvio 5. IGMPv1-viestin rakenne (RFC 1054)

- Version-kenttä asetetaan ykköseksi
- Type-kentälle on kolme mahdollisuutta
    - 1 = Host Membership Query
    - 2 = Host Membership Report
    - 3 = DVMRP
- Unused-kenttä ei ole käytössä tässä versiossa protokollasta
- IGMP Checksum-kenttä on viestien tarkistussumma
- Group Address-kenttä sisältää Host Membership Query -tyypin viestissä nollan. Host Membership Report -viestissä kenttä sisältää raportoitavan ryhmän IP-osoitteen. (RFC 1054.)

2.3.4 IGMP versio 2

IGMP versio 2 salli käyttäjän IGMP-ryhmästä poistumisen ripeän raportoinnin reititinlaitteelle. Nopea ilmoitus ryhmän jäsenyyksien muutoksista optimoi verkon suorituskykyä. Paketin rakenne sai uudessa versiossa muokatun rakenteen (ks. Kuvio 6).



Kuvio 6. IGMPv2-viestin rakenne (RFC 2236)

- Type-kenttä voi sisältää neljä eri arvoa
    - 0x11 = Membership Query
    - 0x16 = Version 2 Membership Report
    - 0x17 = Leave Group
    - 0x12 = Version 1 Membership Report
- Max Resp Time määrittää ajan, kuinka kauan reititinlaite odottaa vastausta käyttäjältä

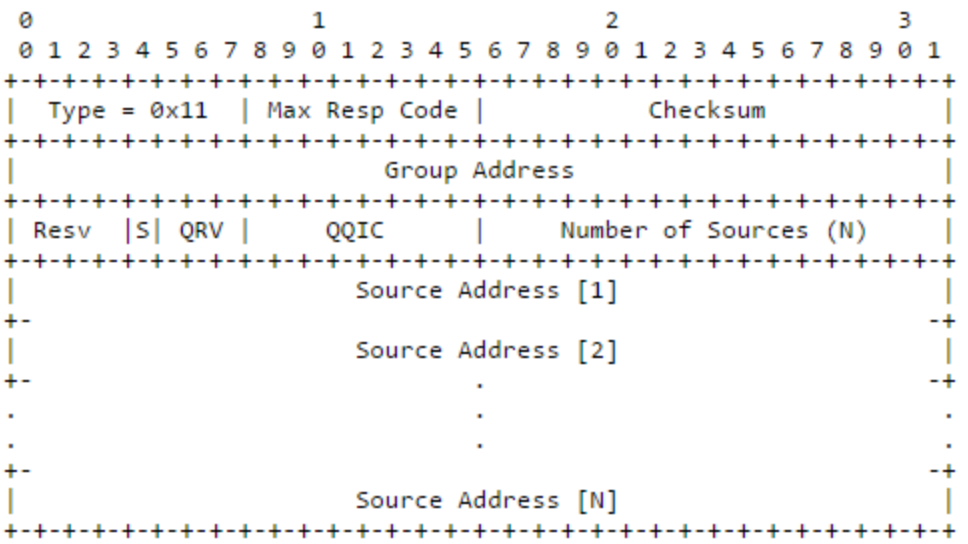
Muut kentät ovat käyttötarkoitukseltaan aiempien versioiden mukaisia. (RFC 2236.)

2.3.5 IGMP versio 3

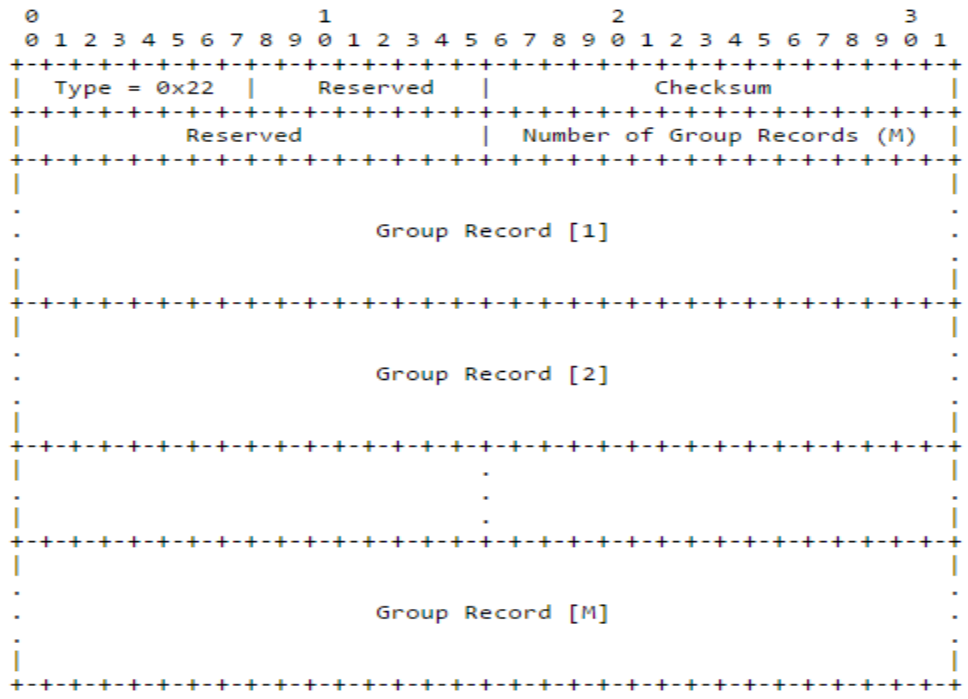
Versio 3 on IGMP-protokollan uusin iteraatio, joka määriteltiin vuonna 2002. Protokolla toi tässä versiossa tuen Source Specific Multicast -tekniikalle, jossa multicast-vastaanottaja pyytää tietyltä multicast-lähetyksen tarjoajalta luvan liittyä ryhmään. Ryhmiä ei varsinaisesti mainosteta, joka parantaa verkon suorituskykyä. (RFC 3569; RFC 4607.)

Uusi versio kasvatti IGMP-viestin kokoa. Reititinlaitteet lähettävät verkon käyttäjille IGMP Membership Query -viestin (ks. Kuvio 7), johon käyttäjälaitteet vastaavat

Membership Report -viestillä. Käyttäjän Membership Record -viesti sisältää tyyppikohtaisena Group Record -kenttinä tiedot jokaisesta IGMP-ryhmästä, johon käyttäjä on liittynyt (ks. Kuvio 8).



Kuvio 7. IGMPv3 Membership Query –viestin rakenne (RFC 3376)



Kuvio 8. IGMPv3 Membership Report –viestin rakenne (RFC 3376)

2.4 Digital Video Broadcasting

2.4.1 Yleistä

Digital Video Broadcasting (DVB) on joukko avoimia standardeja, joilla määritetään digitaalisten televisiolähetysten siirto niiden kehysrakenteen, kanavakoodauksen sekä moduloinnin osalta. DVB:n lisäksi muita digitaalisia televisiolähetyksiä koskevia standardeja ovat Advanced Television Systems Committee (ATSC), Integrated Services Digital Broadcasting (ISDB), Digital Terrestrial Multimedia Broadcast (DTMB) sekä Digital Multimedia Broadcasting (DMB). DVB on laajalti käytössä Euroopan maissa, mukaan lukien Suomen digitaalitelevisioverkossa. IPTV-liikenteessä toimitilojen televisiolähetyksiin käytetty siirtomedia (antenni-, kaapeli, satelliitti- tai langaton verkko) määrittää käytetyn DVB-standardin, joka on oleellinen osa palvelun headend-palvelimen laitteistoa suunniteltaessa. DVB-standardeihin mukautuvia palvelimille ja työasemille suunniteltuja vastaanottimia on markkinoilla tarjolla reilusti. (Poole n.d.)

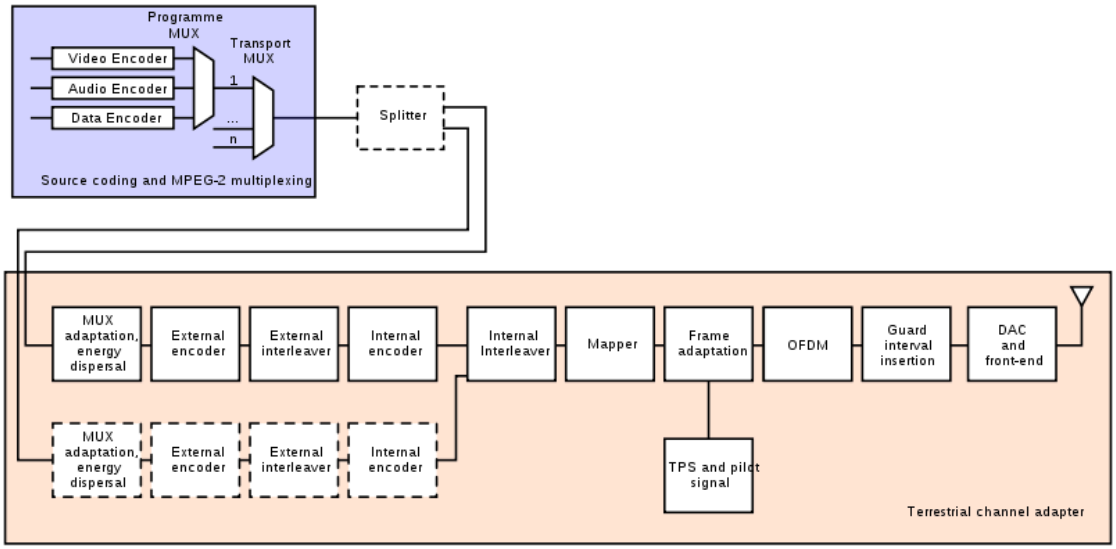
2.4.2 DVB-T

DVB-standardit jaetaan käyttötarkoituksen ja siirtomedian mukaan. DVB-T (Terrestrial) -standardi määrittää televisioliikenteen maanpäällisessä antenniverkossa. DVB-T käyttää moduloinnissa Orthogonal frequency-division multiplexing (OFDM) -tekniikkaa, jota käytetään myös mm. nykyaikaisissa ADSL-verkkokytkentätekniikoissa. DVB-T:n maksimaalinen tiedonsiirtonopeus on 27 Mbit/s ja jokainen taajuuskanava voi lähettää neljästä viiteen ohjelmaa yhtaikaisesti (Papinniemi 2010, 4). OFDM-modulaatiota käytettäessä tietovuot eivät aiheuta häirintää toistensa kanssa verrattuna esimerkiksi analogisiin antennilähetyksiin. DVB-T:n muita ominaisuuksia ovat

- kolme modulaatiovalintaa (QPSK, 16QAM, 64QAM)
- viisi Forward Error Correction (FEC) –virheenkorjaustasoa
- kantoaaltojen määrä 8k (6817 kpl aaltoja) tai 2k (1705 kpl aaltoja)
- videon lähetystaajuus 50 tai 60 hertsiä

- kaistanleveys 6, 7 tai 8 megahertsiä.

Kuviossa 9 on kuvattu DVB-T -lähettimen rakenne ja kuinka palveluntarjoajan enkooderista saapuva data muutetaan standardinmukaiseksi digitaaliseksi tietovuoksi. Monipuoliset ominaisuudet mahdollistavat palveluntarjoajille kattavan mahdollisuuden tehdä tarpeisiin sopivia ja mukautuneita ratkaisuja. (Poole n.d.)



Kuvio 9. DVB-T -lähettimen rakenne

2.4.3 DVB-T2

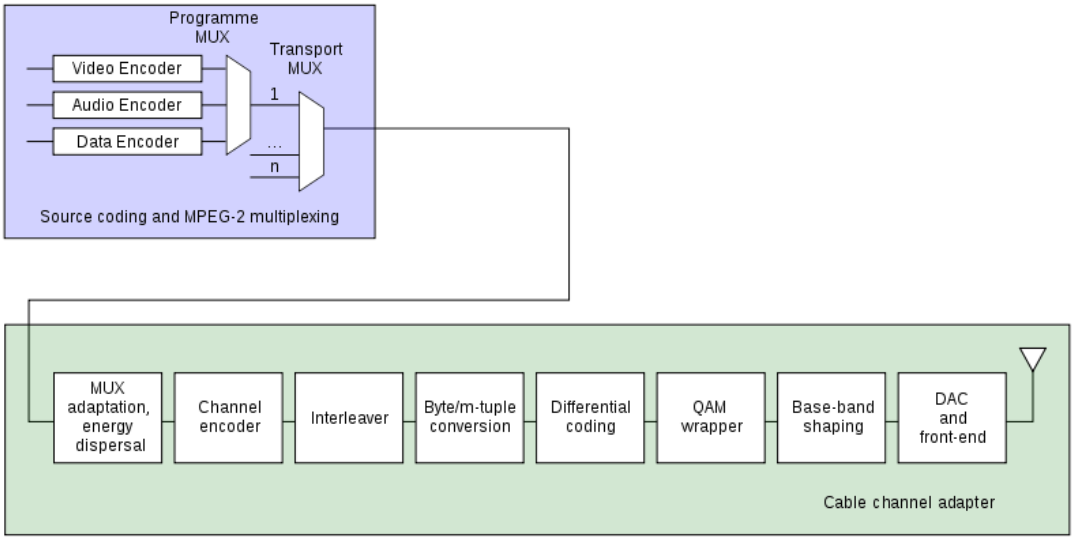
DVB-T2 on edistyneempi versio DVB-T:sta. Uudelle standardille selvisi tarve varsinkin teräväpiirtolähetysten ja monikanavaisen audion yleistyessä. Modulaatiovalinnoilta T2 laajentui 256QAM:aan, kantoaaltojen määrän valinta kasvoi viiteen (1k, 2k, 8k, 16k ja 32k) ja virheenkorjauksen metodeja päivitettiin kehittyneempiin vaihtoehtoihin (Poole n.d.). Tiedonsiirtokapasiteetti toisessa versiossa kasvoi jopa 35 Mbit/s nopeuteen (Papinniemi 2010, 7-8).

2.4.4 DVB-C

DVB-C (Cable) on digitaalisten kaapeliverkon kautta siirrettävien televisiolähetysten määrittävä standardi. DVB-T:n tapaan kaapeliversio käyttää OFDM:ää 16- ja 256-QAM moodeissa. Muuten toimintatavaltaan molemmat standardit ovat hyvin saman-

tapaisia (ks. Kuvio 10), mutta DVB-C tukee suurempaa määrää taajuusalueita. Taajuusalueiden määrä kasvattaa yhtaikaisten ohjelmalähetysten maksimimäärän kuuteen nopeudella 38 Mbit/s (Papinniemi 2010, 4).

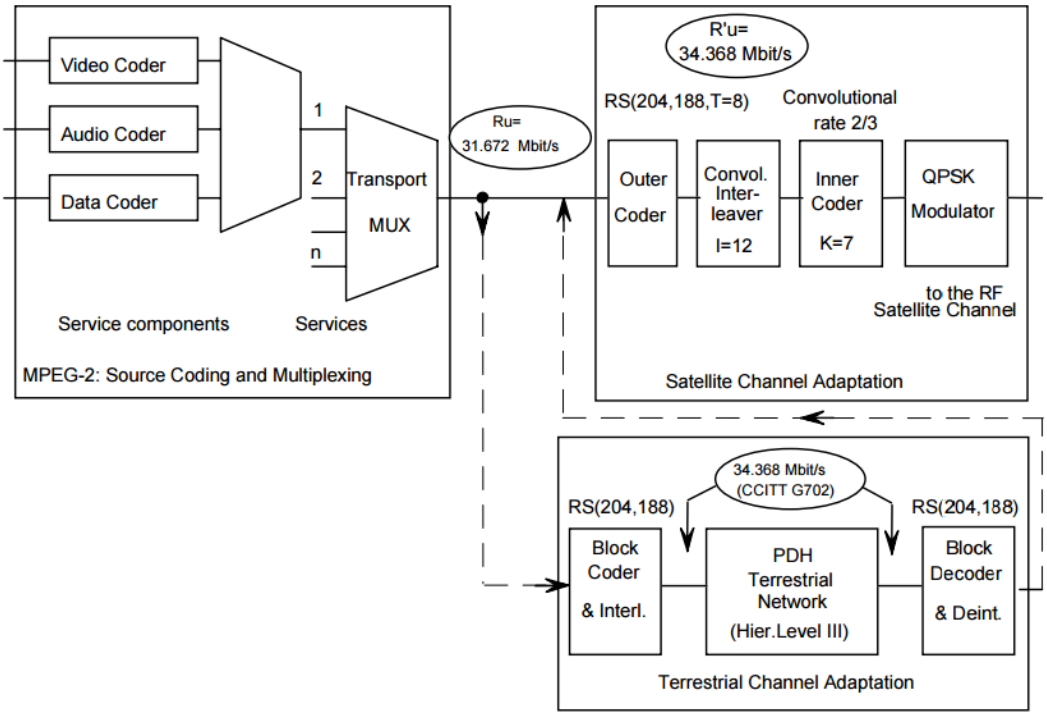
DVB-C soveltuu IPTV-järjestelmään täysin samalla lailla kuin DVB-T. Laitehankinnoissa tulee huomioida käytössä oleva televisioverkon tyyppi sekä yhteensopiva sovitin.



Kuvio 10. DVB-C -lähettimen rakenne

2.4.5 DVB-S

DVB-S (Satellite) on kolmas vaihtoehto digitaalisten televisiolähetysten siirtoon. DVB-S -standardi määrittää televisiolähetysten lähettämisen ja vastaanottamisen satelliittiverkon kautta, jonka lähettimen rakenne on kuviossa 11. Modulaatio suoritetaan kaksivaihe-eroisella vaiheavainnuksella eli Binary Phase Shift Keying:llä (ETSI TR 101 198 1997). DVB-S:n tiedonsiirtonopeus on maksimissaan 63 Mbit/s ja se tukee viidestä kymmeneen yhtaikaista kanavalähetystä (Papinniemi 2010, 5).



Kuvio 11. DVB-S -lähettimen rakenne (ETSI TR 101 198 1997)

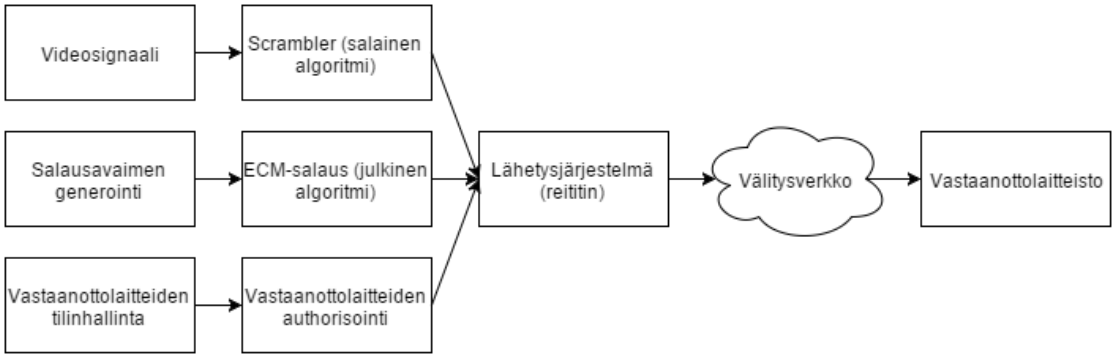
2.4.6 DVB-H & DVB-SH

DVB-H (Handheld) -standardi on tarkoitettu mobiililaitteilla vastaanotettaviin televisiolähetyksiin. Teknologialle esiintyi tarve, kun pienikokoiset kannettavat laitteet alkoivat yleistyä. Toiminnaltaan DVB-H vastaa maanpäälliseen verkkoon suunniteltua DVB-T:a. DVB-H ja DVB-T käyttävät samaa antenniverkkoa, jossa ne toimivat toisistaan erillään multipleksoinnin avulla. Handheld-standardi kuitenkin kaatui markkinoilla olevien laitteiden vähäisyyden takia. DVB-SH (Satellite services to Handhelds) sekä IPTV-streaming -palvelut syrjäyttivät DVB-H:n mobiiliratkaisuissa. Satelliittiverkkoon perustuva mobiilitelevisio käyttää moduloinnissa kahta mahdollista ratkaisua, jossa Time-division Multiplexing (TDM) -teknologia esiintyi ensimmäisen kerran digitaalisessa televisiolähetyksissä:

- SH-A -moodissa satelliittilinkki sekä vastaanottolaite käyttävät OFDM:ää
- SH-B -moodissa satelliittilinkki käyttää TDM:ää ja vastaanottolaite OFDM:ää (ETSI TR 102 377 2009.)

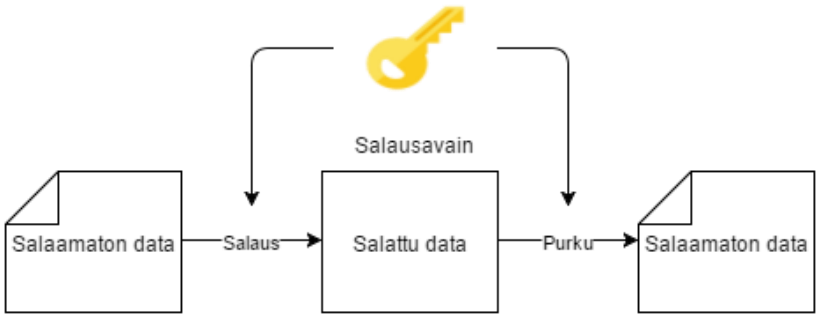
2.5 DigiTV-lähetysten salausarkkitehtuuri

Digitaalisten televisiolähetysten sisällönvälitysjärjestelmässä on operaattorin hallitsemana kolme lähetysten salaukseen ja authorisointiin vaikuttavaa elementtiä (ks. Kuvio 12). Conditional Access System (CAS) -järjestelmä vastaa lähetysten lähettämisestä niitä tilanneille asiakaslaitteille. CAS:n toimintaa voidaan IPTV-ratkaisuissa verrata IGMP-protokollan viestien vaihtoon, joilla asiakaslaite voi esimerkiksi ilmoittaa halutusta kanavan vaihdosta. CAS-järjestelmään kuuluu myös lähetysten scrambling-prosessi ja siihen liittyvä signaalin kryptaus eli salaus. Salaukseen kuuluvat viestit lähetetään usein lähetysdatan kanssa samassa tietovuossa, mutta ne voidaan halutessa määrittää siirrettäväksi eri verkkojen kautta kuin itse lähetys. Lähetysten scrambling-algoritmi on yleensä salainen ja operaattorin yksinoikeudella määrittämä. Salausavaimia voidaan siirtää myös vastaanottolaitteeseen syötettävällä TV-kortilla, joka on digiboksien/-TV:iden käyttäjille todennäköisesti tuttu ominaisuus. (Minoli 2008, 251- 252.)



Kuvio 12. DVB-lähetyksen CAS-järjestelmä

CAS-järjestelmän viestit voidaan jakaa kahteen luokkaan. Entitlement Control Message (ECM) -viestejä lähetetään tietyin aikavälein CAS-järjestelmästä tilaajille, joilla vaihdetaan lähetyksen salausavaimet. Salausavaimista käytetään digitaalisissa televisiolähetyksissä nimitystä Control Word (CW). Lähetyksissä käytetään yleensä symmetrisiä salausalgoritmeja kuten Triple Data Encryption Algorithm (3DES) ja Advanced Encryption Standard (AES). Symmetrisissä salausalgoritmeissa lähetettävä data salataan samalla avaimella. Prosessissa lähetyspäässä salaamaton data suojataan verkossa siirtoa varten. Salattu data voidaan avata vastaanottolaitteella aiemmin vastaanotetulla salausavaimella (ks. Kuvio 13). (Minoli 2008, 251-252.)

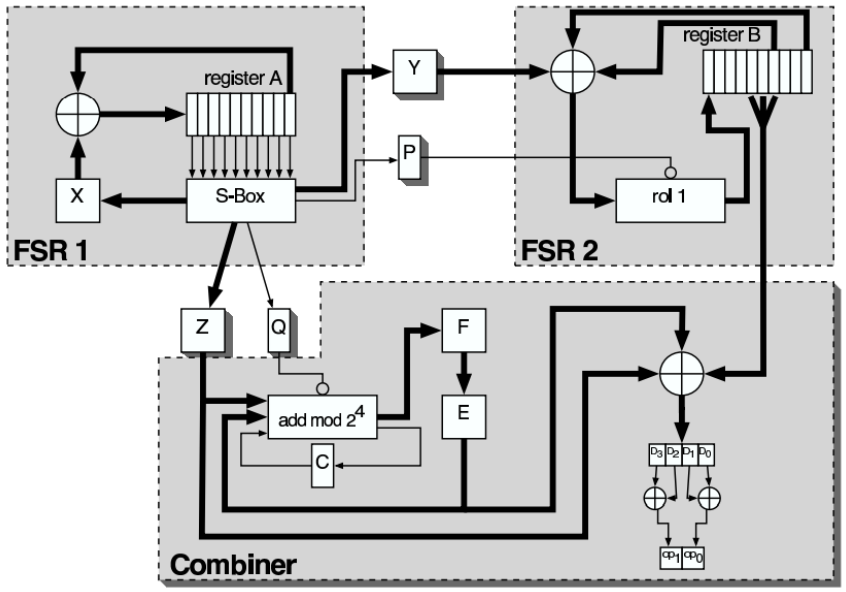


Kuvio 13. Symmetrinen salaus

Toinen CAS-järjestelmän käyttämä viestityyppi on Entitlement Management Message (EMM). EMM-viesteillä lähetysjärjestelmä antaa vastaanottolaitteelle hyväksynnän katsoa lähetystä. (Minoli 2008, 252.)

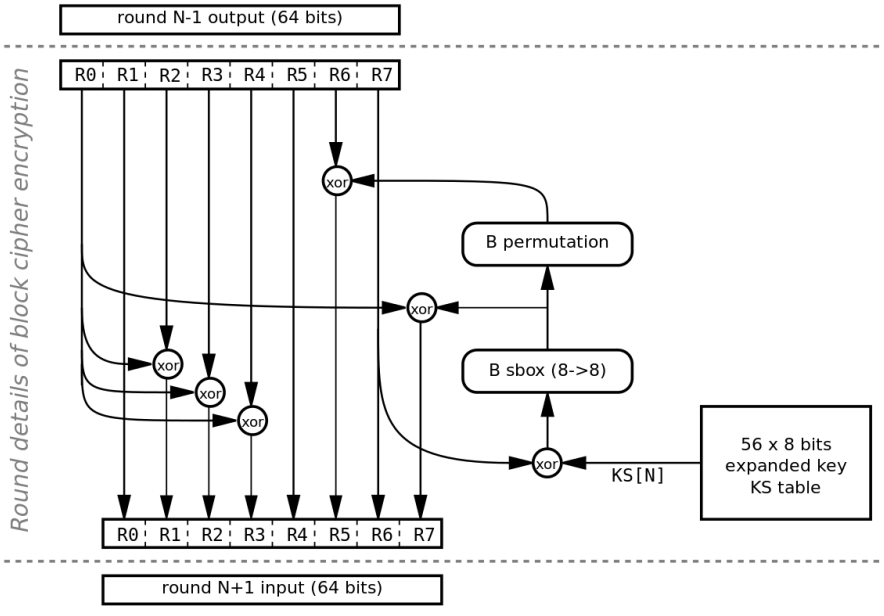
DVB-lähetysten kryptauksessa on ollut käytössä vuodesta 1994 lähtien Common Scrambling Algorithm (CSA) -algoritmi. CSA-algoritmi oli vuoteen 2002 asti palvelun-tarjoajille saatavilla vain erityisluvalla, jossa määritettiin lisäksi Non-Disclosure Agree-ment -sopimus lupia jakavan ETSI-järjestön kanssa. Sopimuksen mukaan sopimuksen hyväksyneet osapuolet eivät saaneet implementoida algoritmia sovellustasolla, joka olisi mahdollistanut algoritmin purkamisen ja siten esittänyt ilmeisen tietoturvariskin. Vuonna 2002 tilanne muuttui ja ohjelmisto FreeDec julkaisi ohjelmistopohjaisen CSA-implementaation, jonka julkaisun jälkeen CSA-algoritmin purkaminen alkoi. (Wein-mann & Wirt 2004, 1.)

CSA-algoritmin rakenne jaetaan kahteen osaan. TV-lähetyksen tietovuon paketin lop-puun liitetään stream-koodi ja paketin alkuun 64-bittinen block-koodi, joista salaus-avain muodostuu. Lähetyksen alussa stream-koodin 32 ensimmäistä toistoa ovat alustusta varten. Alustuksen jälkeen stream-koodi tuottaa prosessointikellon jokai-sella syklillä kaksi bittiä pseudosatunnaisia arvoja, joille suoritetaan XOR (Exclusive Or) -looginen operaatio. Stream-koodin loogiseen rakenteeseen kuuluu kaksi feed-back-shift-register (FSR) -komponenttia ja yhdistäjä muistilla (ks. Kuvio 14). Kompo-nenttien yhdistäjä tuottaa aiemmin mainitut kaksi pseudosatunnaista bittiä. Toisessa FSR-komponentissa on huomioitavana osana S-Box-osa, jonka toiminta oli yksi suu-rimmista mysteereistä algoritmin murtamiselle ennen CAS:n ohjelmistopohjaista im-plementaatiota. (Weinmann & Wirt 2004, 2-4.)



Kuvio 14. Stream-koodin looginen rakenne (Weinmann & Wirt 2008, 4)

Block-koodin tehtävä on tuottaa O-permutaatiolla 64-bittinen arvojono, joka suoritetaan 56 kertaa (ks. Kuvio 15). Operaatioon vaikuttaa CW-koodin arvo, josta muodostetaan laajennettu avainarvo. Avainarvosta otetaan yksi bitti jokaisella suorituskerralla. Tekijänä on myös stream-koodin tapaan S-Box-komponentin tuottama arvo. (Weinmann & Wirt 2004, 6-7.)



Kuvio 15. Block-koodin looginen rakenne

CSA-algoritmin purkamismahdollisuus esittää realistisen uhan DVB-tekniikkaa käyttävissä digitaalisissa televisiolähetyksissä, varsinkin koska käytännössä kaikki eurooppalaiset maksulliset televisiopalvelut on myös salattu CSA:lla (Weinmann & Wirt 2004, 1). CSA-algoritmiin kohdistettuja tutkimusryhmien suorittamia kokeellisia hyökkäyksiä on raportoitu ainakin kahteen haavoittuvuuteen. Yhdessä hyökkäyksessä syötettiin block-koodiin satunnainen virhe, jonka vaikutus algoritmissa sai 64-bittisen tulosarvon kahdeksan viimeistä bittiä tuottamaan virheellisen arvon. Näistä arvoista voitiin siten laskea käänteisoperaatiolla varsinaisen avaimen arvo (Wirt 2004). Toinen hyökkäys kohdistettiin MPEG-2 -liikennettä koskevaan salaukseen, jossa liikenteen kapsuloinnista johtuen tietovuo koostui suuresta määrästä nollia. Avain saatiin haettua käyttämällä n.k. sateenkaaritaulua (Tews, Weiner & Wälde 2011, 41-44).

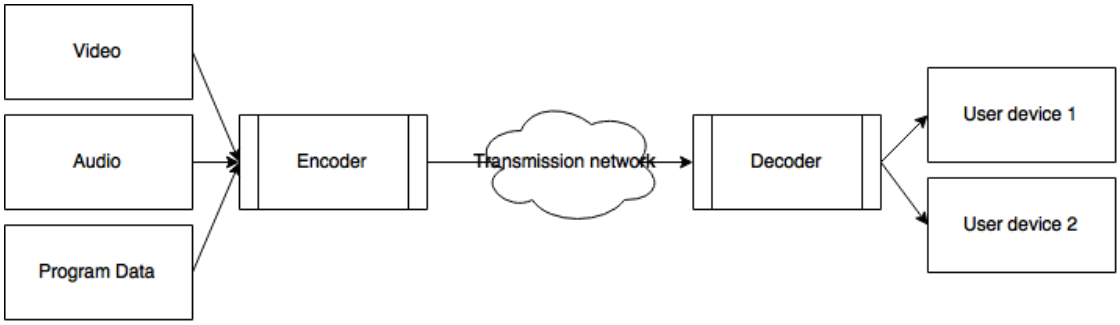
2.6 Televisiolähetysten tekniikkaa

2.6.1 Video- ja audiodatan pakkauksenhallinta

Pakkauksenhallinta eli koodekki on algoritmi tai ohjelmisto, joka yhdistää video- ja audiodatassa kooderin ja dekooderin toiminnallisuudet sekä hallitsee mahdollisia tiedonsiirtoon liittyviä muuttujia. Koodekit erotellaan käyttötarkoituksensa mukaisesti joko puhe-, audio- tai videokoodekkeihin. Televisioliikenteessä vastaanottolaitteen tulee pystyä vähintään purkamaan koodattu data soveltuvalla koodekilla (dekoodaus), monipuolisemmat laitteet voivat lisäksi muuttaa datan eri muotoon lähetettäväksi (koodaus). Koodekeilla voidaan myös manipuloida videon ja audion laatua. Häviöttömissä koodekeissa datan laatu pysyy koskemattomana ja häviöllisissä koodekeissa datasta poistetaan jokin osa, jolloin sen toistolaatu heikkenee. Häviöllisten koodekkien käyttötarkoitus on pienentää tiedostokokoa, jolloin sen vaikutus siirtoon tarvittavaan kaistanleveyteen on kevyempi. (Good, Bazzano & Lombardi 2010.)

Kuviossa 16 on kuvattu yksinkertaisesti koodekin toiminta. Vasemmalla lähetyspäässä video-, audio- ja ohjelmadata ovat omina tietovuoinaan. Kooderi muuttaa raa'at tietovuot soveltuvien koodekkien mukaisiin muotoihin. Koodatut tietovuot pa-

kataan koodekin mukaiseen kuljetus eli siirtotason formaattiin, jolloin ne voidaan siirtää esim. antenni- tai IP-verkossa. Vastaanottopäässä paketit dekoodataan joko keskitetyllä palvelimella tai suoraan vastaanottolaitteella.



Kuvio 16. Pakkauksenhallinta televisioliikenteessä

2.6.2 Kuvanlaatu

IPTV-palveluissa voidaan törmätä nopeasti kaistanleveyden asettamiin ääriarvoihin. IPTV kulkee samassa verkossa muun datan kanssa, joten sen vaikutus tulee huomioida koko verkon suorituskyvyn ylläpidon varmistamiseksi. Käytetyn tiedonsiirtometodin (unicast, broadcast, multicast) lisäksi IPTV-liikenteen videon ja äänen laatu vaikuttaa suoraan tarvittavaan kaistanleveyteen.

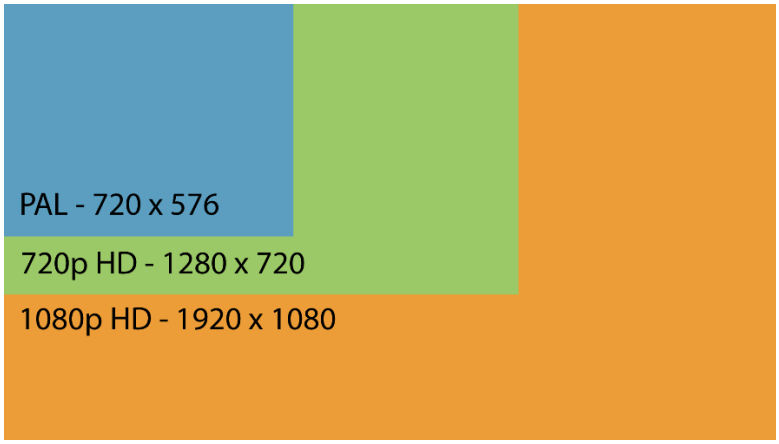
Kuvanlaadusta puhuttaessa viitataan yleensä sen kuvasuhteen mukaiseen pikselitiheyteen eli resoluutioon. Resoluutio ilmoitetaan yleensä muodossa **leveys kertaa korkeus** käyttäen numeerisia arvoja pikseleiden määrästä tuumaa kohden (pixels per inch, PPI). Puhekielessä saatetaan kuvan resoluutiolla tarkoittaa myös (virheellisesti) kuvan silmin havaittavaa laatua. Pikselitiheyteen voidaan vaikuttaa useassa eri vaiheessa videon siirtoprosessia mm. koodekkien avulla. (Randall 1998, 217-218.)

Suomessa käytettyjä resoluutioita ovat:

- PAL (576p) – 720x576 pikseliä
- 720p - 1280x720 pikseliä
- 1080i - 1440x1080 tai 1920x1080 pikseliä, analogisen television kaltainen lomitus

- 1080p – 1920x1080 pikseliä

Suomessa analogisissa televisiolähetyksissä käytettiin PAL-standardin mukaista resoluutiota. PAL-resoluutiota käyttävät myös Standard-Definition TV (SDTV) –digitaalilähetykset. Korkealaatuisemmissa High-Definition TV (HDTV) –lähetyksissä eli teräväpiirtolähetyksissä kuvan resoluutio on joko 720 tai 1080 pikseliä korkeussuunnassa. 1080p –resoluutio on näytettävien pikselien määrältä jo viisinkertainen analogiseen PAL-lähetykseen verrattuna (ks. Kuvio 17).



Kuvio 17. Videolähetysten resoluutioita.

Kuvan nähtävään laatuun vaikuttaa lähetettävän videon pikselitiheys, siirrossa tapahtuva mahdollinen hävikki sekä vastaanottavan näyttölaitteen fyysinen kyky näyttää resoluutioita. Näyttölaitteiden natiiviresoluutio määrittää korkeimman mahdollisen pikselitiheyden, jossa ei tapahdu minkäänlaista pikselien sovittamisesta johtuvaa venyttämistä tai litistämistä. Natiiviresoluutiolla jokaista pikseliä vastaa yksi näyttölaitteen kuvapiste.

2.6.3 Virkistystaajuus ja kuvan lomitus

Videokuvan virkistystaajuudella tarkoitetaan yksittäisten kuvien eli kehysten näyttökertoja sekunnissa (frames per second, FPS). Videotekniikassa liikkuva kuva muodostuu näyttölaitteseen syötettävistä yksittäisistä kuvista, joiden vaihtuminen saa aikaan liikkuvan kuvan. Korkeampi virkistystaajuus saa kuvan näyttämään sulavammalta. Videon virkistystaajuuteen voidaan vaikuttaa videon lähtökohteessa ja näyttölaitteessa. Näyttölaitteissa on resoluution tapaan maksimaalinen arvo myös virkistystaajuudelle.

Virkistystaajuudella on merkitystä videon lähetysvaiheessa sekä vastaanottolaitteessa. Ilman videon koodausta koodekin avulla virkistystaajuuden kasvattaminen nostaa tarvittavaa kaistanleveyttä lineaarisesti. Koodekkia käyttämällä pikselien toistettava väri pystytään muistamaan esimerkiksi videossa kuvattavan tilanteen taustan tai muun objektin osalta, joka saattaa olla täysin liikkumaton. Pistetiedon optimoinnilla videon koko voidaan saada jopa täsmälleen samaksi, vaikka videon virkistystaajuus olisi kaksinkertainen. Virkistystaajuudet voidaan jakaa joko lomitettuihin tai progressiivisiin päivitystapoihin. Progressiivisella virkistystaajuudella jokainen kehys esitetään peräkkäin. Lomitetulla virkistystaajuudella näyttökuvaan syötetään lomittain kaksinkertainen määrä kehyksiä, jolla saadaan kaksinkertainen virkistystaajuus alkuperäiseen videon virkistystaajuuteen verrattuna. Taulukossa 1 on listattu yleisiä televisiolähetysten virkistystaajuuksia.

Taulukko 1. Yleisiä televisiolähetysten virkistystaajuuksia

Virkistystaajuus	**Progressiivinen / lomitettu**	**Käyttökohde**
24p	progressiivinen	elokuvat NTSC-alueella
25p	progressiivinen	24p-virkistystaajuus PAL-alueella
48p	progressiivinen	elokuvat, tarkoitus vähentää liikkeen aiheuttamaa laadun heikkenemistä
50i/60i	lomitettu	24p/25p-virkistystaajuus lomitettuna
50p/60p	progressiivinen	uusimpien HDTV:iden tukema virkistystaajuus

2.6.4 Bittinopeus

Yhdistämällä pikselitiheys, virkistystaajuus ja äänenlaatu saadaan bitrate eli bittinopeus, joka määrittää video- ja äänidatan tarvitseman kaistanleveyden lähetyksen keskeytymättömään toistoon. Bittinopeus ilmoitetaan muodossa bittiä sekunnissa, joka on yleinen tapa ilmoittaa nopeuksia monissa dataliikenteen siirtotavoissa. Huomattavaa on, että bittinopeudella viitataan usein yhteyden vaatimaan maksimaaliseen nopeuteen, jonka tulisi varmistaa yhteyden vakaus. Televisiolähetysten koodauksessa voidaan käyttää joko jatkuvaa (constant bit rate, CBR) tai vaihtelevaa bittinopeutta (variable bit rate, VBR). Jatkuvassa bittinopeudessa kooderin tulee täyttää pistetiedon säilyttämisestä saatu optimoitu pakettitila tyhjällä tilalla. Vaihtelevaa bit-

tinopeutta käytettäessä bittinopeudeksi saadaan keskimääräinen nopeus. Intensiiviset vaihtelut kuvassa eivät välttämättä nosta bittinopeutta, vaan nämä kohdat voidaan sijoittaa tyhjiin paketteihin. Silmämääräisesti suurimmat bittinopeuden vaihtelut havaitaan datan vaihdekohdassa, kuten videon kohtauksen vaihtumisessa. (Hwang 2009, 122.)

Nostamalla videon tai audion laatua tarvittava bittinopeus kasvaa, joka on yleinen rajoittava tekijä korkealaatuisempien televisiolähetysten lähettämisessä, 720p-tason media vaatii n. 4-5 Mbit/s nopeuden ja 1080p-tason media n. 8-12 Mbit/s (Patterson 2012). IPTV-teknologiassa multicast osoittautuukin merkittäväksi tekijäksi kaistanleveyden optimoinnissa.

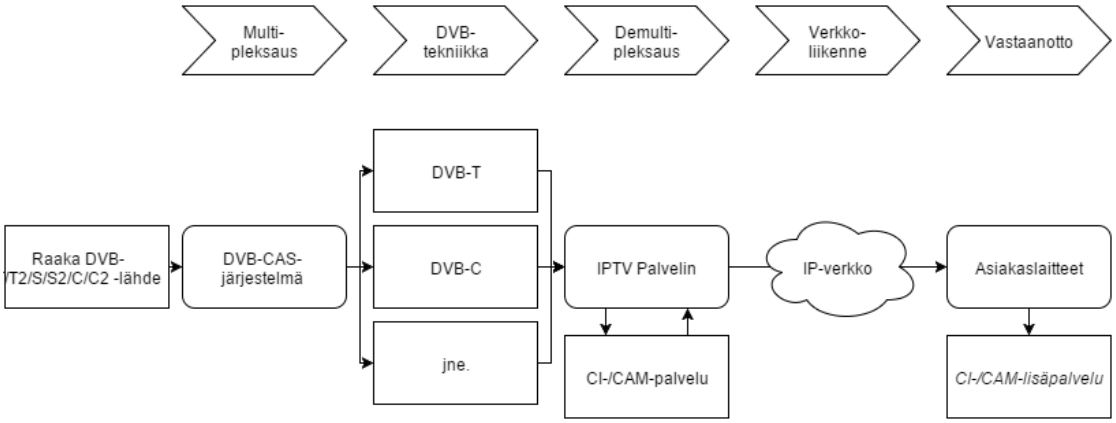
3 Toteutus

3.1 Televisiolähetysten julkinen esittäminen

Työn toteutusmahdollisuutta pohdittaessa törmättiin tekijänoikeuslain asettamiin rajoituksiin. Tekijänoikeuslain määrityksen mukaan televisiolähetysten jakaminen julkisilla esityspaikoilla vaatii maksullisen lisenssin hankkimisen lisenssin omaavalta järjestöltä. Laki koskee koulujen sisäisiä tietoverkkoja ja mainitsee työn kaltaisen tutkimustyön (Koulun sisäverkko n.d.). Asia varmistettiin myös yhteydenotolla tekijänoikeusjärjestö Kopiostoon, joka hallitsee oppilaitoksille suunnattuja tekijänoikeuslisenssejä. Tekijänoikeusseikkojen takia työssä ei voitu toteuttaa DVB-lähetyksiä jakavaa implementaatiota.

3.2 Operaattoreiden IPTV-implementaatioita

Työnannossa oli tarkoituksena toteuttaa operaattorin IPTV-palvelu, jonka periaatteena on tarjota DVB-lähetyksiä muutettuna IP-verkossa siirrettävään formaattiin. Operaattorin tuotantotasoisessa IPTV-palvelussa DVB-lähetykset vastaanotetaan Integrated Receiver / Decoder (IRD) -laitteilla. Palvelimella tai palvelinklusterilla demultipleksataan (demux) DVB-lähetykset ja avataan CAS-järjestelmässä määritetty salaus. CI-/CAM-lisäkorteilla puretaan kanavien salaus. Maksullisille lisäkanaville tarvitaan omat CI-/CAM-kortit. Kanavista rakennetaan luettelo, joka saadaan analysoimalla DVB-kuljetusstreamin sisältöä. Lähetykset voidaan purkamisen jälkeen enkoodata haluttuun formaattiin ja lähettää ne asiakaslaitteille IP-verkon kautta käyttäen multicastia (ks. Kuvio 18). IP-verkon osiin voin myös kuulua IPTV-middleware -laitteita, jotka toimivat sovellustasolla toimien esimerkiksi palvelun käyttöliittymänä tai ilmoittajana. (Wielert 2011.)



Kuvio 18. Esimerkki operaattorin IPTV-palveluarkkitehtuurista

Tällaista toteutusta käyttää Suomessa palveluntarjoajista ainakin TeliaSonera AB, jonka IPTV-palvelussa asiakkaille jaetaan IPTV-lähetyksiä mainostava reititin ja set-top box -laite, joka yhdistetään televisioon. Tutkinnan perusteella IPTV-liikenne siirretään Soneran palvelussa UDP:lla käyttäen IGMPv2:ta. Palvelun toimintaa tutkittiin lähettämällä Kreatel Communications:n valmistamasta set-top box -laitteen IGMPv2 membership report -viesti, jolla laite pyytää lupaa liittyä ryhmään katsomaan kanavaa seitsemän (ks. Kuvio 19). Kanavaa vaihtamalla multicast-viestin kohdeosoite vaihtui, joka on IPTV-implementaatioissa yksi suosituimpia toimintatapoja. Soneran tarjoama set-top box ei ollut käyttäjän konfiguroitavissa. Lähetyksiä ei myöskään pystytty vastaanottamaan millään muulla laitteella kuin operaattorin itse tarjoamalla laitteella, joten käytössä on jokin operaattorin asettama authorisaatio IGMP-viestien välityksessä.

```
89874 812.058235 192.168.1.65      239.16.116.7       IGMPv2     60 Membership Report group 239.16.116.7
⊞ Frame 89874: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
⊞ Ethernet II, Src: KreatelC_66:b5:e4 (00:02:9b:66:b5:e4), Dst: IPv4mcast_10:74:07 (01:00:5e:10:74:07)
⊟ Internet Protocol Version 4, Src: 192.168.1.65 (192.168.1.65), Dst: 239.16.116.7 (239.16.116.7)
    Version: 4
    Header Length: 24 bytes
  ⊞ Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector 6; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 32
    Identification: 0x0000 (0)
  ⊞ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 1
    Protocol: IGMP (2)
  ⊞ Header checksum: 0xbf16 [validation disabled]
    Source: 192.168.1.65 (192.168.1.65)
    Destination: 239.16.116.7 (239.16.116.7)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
  ⊞ Options: (4 bytes), Router Alert
⊞ Internet Group Management Protocol
```

Kuvio 19. TeliaSonera IPTV:n IGMP-jäsenyyspyyntö.

Muiden palveluntarjoajien IPTV-ratkaisuja tutkiessa löydettiin tietoa käytetyistä IPTV-palvelun laitteista. Mikkelin Puhelinyhtiössä oli dokumentoinnin mukaan vuonna 2013 IPTV-palvelimena käytössä rack-mountattu Sun Microsystemsin valmistama Sun

Fire X4150 -palvelin ja siihen liitetty Sun Fire X4500 -tallennuspalvelin. Televisiolähetykset vastaanotetaan palvelimella kuituyhteyden kautta Appear-TV:n valmistamasta DVB-lähetyksiä vastaanottavasta vahvistinlaitteesta. Toteutuksesta ei selvinnyt asiakasmäärää, jonka pohjalta IPTV-palvelu oli suunniteltu. Palvelussa ei tarjottu set-top -laitteistoa asiakkaille, vaan IPTV-lähetykset olivat katsottavissa mediasoitinohjelmistoilla. (Lampinen 2013.)

Päijät-Hämeen Puhelin Oyj:n IPTV-palvelun toteutusdokumentissa vuonna 2006 IPTV-palvelimena oli käytetty IPTV-lähetysten jakamiseen tarkoitettua Skystream iPlex -palvelinta. Käytössä oli DVB-S -lähetykset, mutta IRD-laitteesta ei ollut tarkempia spesifikaatioita lueteltuna. Asiakkaiden set-top box -laitteet lähettivät Soneran tapaan IGMPv2 -viestejä ja multicast-reititykseen käytettiin PIM-SM -moodia (Savolainen 2006, 40-41). Toteutuksessa todettiin ongelmia kaistanleveyden riittämättömyydessä IPTV-lähetyksille kuparikaapeleiden laadun heikkouden takia sekä Core-verkon ohjelmistopuolen ongelmista (Savolainen 2006, 45). Huomioitavaa dokumentoinnista kuitenkin on, että tästä toteutuksesta on tämän opinnäytetyön tekohetkellä jo lähes kymmenen vuotta.

Tutkitut operaattorien IPTV-ratkaisut olivat tuotantokäyttöön tarkoitettuja, joten niitä ei lähdettäisi käyttämään perustana laitevalinnoille SpiderNet-toteutuksen laboratorioverkossa. SpiderNet-toteutusta suunnitellessa löydettiin laboratoriokäyttöön suunnitellun IPTV-palvelun dokumentointi, jossa laadittiin vaatimuksia sekä toteutusohjeita useasta DVB-lähteestä toistettavalle IPTV-palvelulle. Kernenin (2012) dokumentoinnin mukaan suunniteltu vaatimuslista palvelulle laadittiin seuraavanlaisesti:

- IPTV-palvelin, jonka emolevyssä on tarpeeksi rajapintoja vastaanottolaitteille eli DigiTV-virittimille (receiver). **Jokainen viritinrajapinta tulee palvelimessa toistamaan vain yhtä kanavaa**, joten jokainen haluttu lähetettävä kanava tarvitsee oman rajapintansa (Kernen 2012, 12). Markkinoilla on myös ainakin neljää DVB-rajapintaa tukevia virittimiä sekä hybrid-kortteja, jotka tulevat yhtaikaisesti useita DVB-tekniikoita.

- Huomioidaan ympäristön DVB-lähde tai -lähteet virittimiä hankittaessa (ks. luku 2.4.)

- Palvelimelle Conditional Interface (CI) -tytärkortit, joihin liitetään Conditional Access Module (CAM) -kortit. CAM-korteilla vastaanotetaan CW-koodit DVB-lähetysten lähteestä palvelimella. CAM-korttien kyky hoitaa descrambling riippuu mallista, joten huomioitavaa on hankkia tarpeeksi laadukkaita descrambling-kortteja, jotta riittävä määrä kanavia voidaan purkaa yhtaikaisesti.

- Haluttaessa maksullisten kanavien descrambling-toimintoon tarvittavat CAM-pääsykortit.

Palvelua suunniteltaessa tulisi mapata lista kaikista kanavista, joita halutaan jakaa. Jakamismahdollisuus riippuu DVB-lähetysten tarjoajasta ja käytetystä DVB-tekniikasta. Palveluntarjoaja ei välttämättä salli minkäänlaista jakamista ilman maksullista lisenssiä. Jakolupia voi tiedustella palveluntarjoajalta. (Kernen 2012.)

Ohjelmiston puolesta palvelimella on vapauksia. Se voidaan selvitysten perusteella toteuttaa ainakin Windows-, Linux-, Mac OSX- ja FreeBSD -käyttöjärjestelmillä. Hankitut digivirittimet määrittävät yleensä käyttöjärjestelmämahdollisuudet, mutta suurin osa virittimistä tukee vähintään muutamaa Linux-distroa. (Kernen 2012.)

Palvelimen vaatimusten lisäksi vastaanottopäässä tulee olla riittävä laitteisto lähetysten katsomista varten. Koska kyseessä on IPTV-toteutus, tarvitaan operaattorin lähetyspäähän reititin, joka lähettää televisiokanavat IP-verkon yli. Vastaanottavilla laitteilla tulee olla sopiva ohjelmisto sekä laitteisto, jolla lähetyksiä voidaan katsoa. IPTV-palveluita voi katsoa televisiosta IPTV set-top box -laitteella eli IPTV-boksilla tai laajalla valikoimalla PC- sekä kannettavia työasemia ja mobiililaitteita.

3.3 Lähtökohdat SpiderNet-toteutukselle

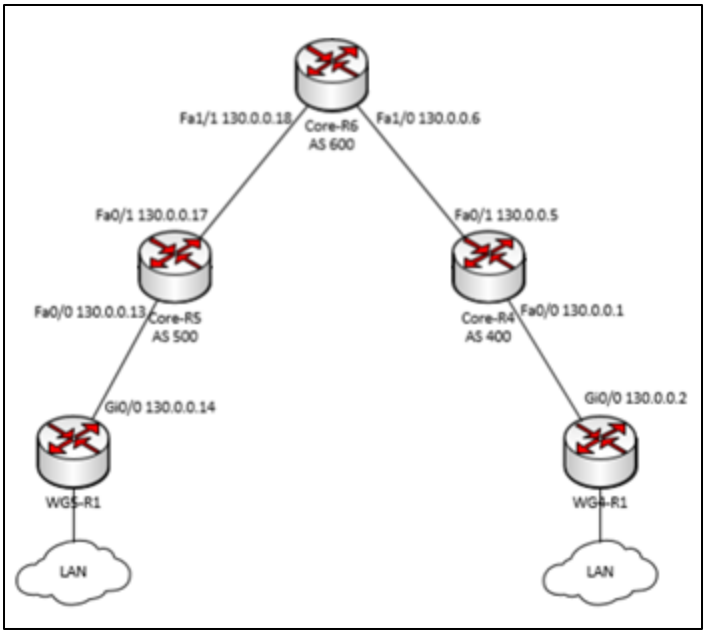
Spidernet-laboratorioympäristössä oli ollut IPTV-ratkaisuna kokeilussa työaseman USB-rajapintaan liitetty langaton DVB-viritin, jonka toiminnasta ja aiemmasta käytöstä ei ollut kirjallista tietoa. Työasema ei ollut dedikoitu palvelin, ja opiskelijat käyttivät sitä satunnaisesti laboratoriotöissä.

Luvussa 3.1 käsiteltyjen tekijänoikeusseikkojen takia työssä päätettiin siirtyä varsinaisten televisiolähetysten jakamisesta simuloituun ratkaisuun. Toteutuksessa käytettiin keskitettyä palvelinta, joka lähetti eri videoklippejä toistolla eli streamaisi niitä varsinaisten televisiokanavien sijaan. Ratkaisu perustui Kymenlaakson ammattikorkeakoulun laboratorioverkko SimuNetissä toteutettuun IPTV-ratkaisuun, jossa ei myöskään käytetty DVB-laitteita (Suleva 2011). Tämän työn toteutuksessa Linux-pohjainen palvelin sijaitsi yhdessä SpiderNet-verkon työryhmässä, josta liikenne siirrettiin SpiderNetin intraverkon läpi toiseen työryhmään vastaanotettavaksi. Tässä toteutuksessa ei voitu selvittää DVB-signaalin muutosta IP-verkon kautta jaettavaksi mediaksi. Toteutuksen tarkoituksena oli toimia perustana palvelimen ja verkon rakenteelle, mikäli laboratorioverkossa toteutettaisiin DVB-lähetyksiä käyttävää ratkaisua. Multicast-liikenteen reitittämiseen tultaisiin käyttämään reititysprotokollatonta PIM-reititystä. Multicast-liikenteen lähetys ja vastaanotto päätettiin toteuttaa Kernenin (2012) ohjeistuksen mukaan Source Specific Multicast -tekniikalla ja siten käyttäen IGMPv3:a. Tarkoituksena oli dokumentoida konfiguraatiot siten, että niiden pohjalta voitaisiin toistaa tehty palvelu sekä mahdollisesti lisämuutoksilla lähettää DVB-lähetyksiä.

Liikenteen reitittämisen konfigurointi perustui Jyväskylän ammattikorkeakoulun QoS-verkkojen suunnittelu ja toteutus -kurssilla opinnäytetyön tekijän dokumentoimiin laitekonfiguraatioihin. Näissä laitekonfiguraatioissa käytettiin vain yhtä multicast-kanavaa, joten muutoksia tuli tehdä. IPTV-palvelimen konfigurointiin ei käytetty aiempia ohjeita.

3.4 Verkkolaitteiden konfigurointi

Konfiguroitavia laitteita olivat työryhmien Ciscon valmistamat WG4 ja WG5 sekä Core -verkkolaitteet (ks. Kuvio 20). Laitteet konfiguroitiin konsoliyhteydellä LabraNetin verkossa. Taulukossa 2 on listattu laitteiden mallit ja niille annetut verkon loogiset nimet topologiassa.



Kuvio 20. Verkon reitittimet

Taulukko 2. Verkkolaitteiden loogiset nimet ja laitemallit

Looginen nimi	**Laitteen valmistaja ja malli**
Core-R4	Cisco Systems 7204 -reititin
Core-R5	Cisco Systems 7204 -reititin
Core-R6	Cisco Systems 7206 -reititin
WG4-R1	Cisco Systems 2821 -reititin
WG4-SW1	Cisco Systems 3550 -reititin
WG4-SW2	Cisco Systems 3550 -reititin
WG5-R1	Cisco Systems 2821 -reititin
WG5-SW1	Cisco Systems 2950 -reititin

IPTV-palvelua simuloiva palvelinympäristö oli työryhmässä WG5 ja asiakaslaitteista vastasi työryhmä WG4. Palvelinympäristössä WG5 oli käytössä vain yksi kytkin WG5-SW1, johon palvelin oli kytketty. Työasemaryhmässä WG4 käytettiin kahta kytkintä WG4-SW1 sekä WG4-SW2, joiden läpi oli kytketty yhteys työasemilta WG4-R1 -reitittimille.

Core-verkossa EIGRP- ja BGP-protokollat asetettiin mainostamaan toistensa reittejä sekä staattisia reittejä komennoilla:

router eigrp 40
redistribute bgp 400
network <mainostettava verkko>
router bgp 400
redistribute static

redistribute eigrp 40
neighbor <liittyvän laitteen IP> <aliverkon peite>

Staattiset reitit asetettiin Core-R4 ja Core-R5-laitteilla kohti työryhmiä komennolla *ip route <kohdeverkon IP> <kohdeverkon alipeite> <kohdelinkki>*. Jokaisessa Core-laitteissa enabloitiin SSM-multicast komennoilla *ip multicast-routing* ja *ip pim ssm default*. Käytössä olevissa rajapinnoissa määritettiin IP-osoitteiden lisäksi komennot *ip pim sparse-mode* sekä *ip igmp version 3*.

Työryhmien reitityksen hoiti WG*x*-R1 -reitittimet. Core-verkkoon päin määritettiin staattiset reitit kaikelle liikenteelle, jotka eivät olleet sisäverkoissa komennolla *ip route 0.0.0.0 0.0.0.0 <Core-verkon linkin IP-osoite>*. Operaattoriratkaisua noudattaen reitittimillä konfiguroitiin myös NAT. Kaksi NAT-poolia määritettiin, yksi staattisesti palvelimelle komennolla *ip nat pool X <haluttu NATattu osoite> <haluttu NATattu osoite> prefix 24*, jossa X on määritetty nimellä *server* tai *other*. Jotta palvelin saatiin liittymään aina *server*-pooliin, täytyi luoda pääsylista komennoilla:

access-list 4 permit <palvelimen IP-osoite> <palvelimen aliverkon peite>
access-list 4 deny any
access-list 5 permit <lähiverkon IP-osoite> <lähiverkon aliverkon peite>
access-list 5 deny any

NAT-poolit liitettiin pääsylistoihin komennoilla:

ip nat inside source list 4 pool server
ip nat inside source list 5 pool other overload

Overload-määritteellä mahdollistetaan useampien työasemien käännetty osoite olemaan sama ulkoverkossa, säästäen käytettyä IP-osoiteavaruutta. Reitittimille annettiin NATattu osoite komennolla *ip nat inside source static <reitittimen sisäverkon rajapinnan IP-osoite> <NAT-osoite>*, jolloin voitiin testata icmp-liikenteen kulku lähiverkkojen reitittimien välillä (ks. Kuvio 21).

```
WG5-R1#traceroute 172.16.1.1

Type escape sequence to abort.
Tracing the route to 172.16.1.1

  1 130.0.0.13 0 msec 0 msec 0 msec
  2 130.0.0.18 0 msec 0 msec 0 msec
  3 130.0.0.5 0 msec 0 msec 0 msec
  4 138.108.55.1 0 msec
```

Kuvio 21. Traceroute työryhmien välillä

Molemmassa työryhmässä Core-verkkoon liittyi WGx-R1 -reititin, josta oli yhteys työasemille ja palvelimelle yhden kytkimen kautta. Työryhmissä sisäinen liikenne konfiguroitiin käyttämään virtuaalisia lähiverkkoja (VLAN) sekä niitä tukevaa VLAN Trunking Protocol:ia (VTP). VTP asetettiin reitittimissä toimimaan VTP Server -moodissa komennoilla:

vtp mode server
vtp domain <VTP-toimialueen nimi>
vtp password <VTP-toimialueen salasana>

Kytkimissä VTP konfiguroitiin samoilla parametreillä mutta VTP Server -moodin sijaan VTP Client -moodiin.

Palvelimen työryhmässä WG5 luotiin VLAN 10 IPTV-liikenteelle, vastaanottopään työryhmässä WG4 luotiin VLANit 10, 20, 30 ja 40 eri liikennetyyppejä ja työasemia varten.

VLANit yhdistettiin WGx-R1-reitittimillä virtuaalisilla rajapinnoilla kytkimeen liittyvässä fyysisessä rajapinnassa. Virtuaalirajapita luotiin ja kommentoitiin komennoilla:

interface gigabitethernet x/x.<VLAN ID>.
encapsulation dot1Q <VLAN ID>
ip address x.x.x.x x.x.x.x
description VLAN <VLAN ID> <VLAN:in annettu nimi>

Kytkimissä VLANeja kuljettavat rajapinnat asetettiin trunk-moodiin komennolla *switchport trunk encapsulation dot1Q* ja *switchport mode trunk.* Palvelin ja työasemat olivat kytkettyinä kytkimiin, joissa VLANit liitettiin rajapintoihin komennolla *switchport mode access* ja *switchport access vlan <VLAN ID>*. Liittämällä VLAN 10 palvelimeen liittyvään rajapintaan saatiin IPTV-liikenne kulkemaan halutussa VLANissa.

VLAN:eihin kuuluville laitteille konfiguroitiin IP-osoitteistus reitittimen DHCP-palvelulta komennolla:

service dhcp
ip dhcp pool <DHCP-poolin nimi>
network <VLAN-verkon osoite> <aliverkon peite>
default-router <reitittimen virtuaalirajapinnan VLAN-osoite>

Palvelimelle saatiin aina tietty staattinen osoite käyttämällä *network*-määritteen sijasta komentoja:

host <IP-osoite> /24
hardware-address <palvelimen rajapinnan MAC-osoite>

Kytkimissä asetettiin myös komento *ip igmp snooping*, joka mahdollistaa IGMP-ryhmien kartoituksen ilman multicast-lähetysten floodaamista jokaiseen avoimeen porttiin. IGMP Snooping myös mappaa automaattisesti rajapinnan, josta on lähin yhteys multicast-liikennettä reitittävään laitteeseen.

Multicast konfiguroitiin työryhmien reitittimissä samaan tapaan kuin Core-verkon laitteissa, enabloiden *ip multicast-routing* sekä *ip pim ssm default* globaalissa konfiguraatiomoodissa ja liittäen jokaiseen käytössä olleseen rajapintaan *ip pim sparse mode* sekä *ip igmp version 3*.

Multicast-liikenteen toiminta voitiin palvelimen asennuksen ja toimintaanpanon jälkeen todentaa katsomalla reitittimien multicast-reititystauluja komennolla *show ip mroute* (ks. Kuvio 22, Kuvio 23 & Kuvio 24). Tauluista nähtiin, että lähetetty multicast-stream ryhmään 232.1.1.1 kulki IPTV-palvelimen VLAN-rajapinnasta vastaanottoverkon VLAN-rajapintaan. IGMP-protokollan toiminta todennettiin *show ip igmp membership* -komennolla, josta nähtiin työaseman pyyntö liittyä multicast ryhmään lähteestä 109.163.249.2, joka on palvelimen NATattu IP-osoite (ks. Kuvio 25).

```
WG5-R1#sh ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.2.127.254), 00:00:18/00:02:41, RP 0.0.0.0, flags: SP
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null

(192.168.1.100, 232.1.1.1), 00:33:14/00:03:27, flags: sT
  Incoming interface: GigabitEthernet0/1.10, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:11:45/00:02:32

(*, 224.0.1.40), 00:39:05/00:02:19, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:39:05/00:02:19
```

Kuvio 22. WG5-R1 multicast-reititystaulu

```
WG4-R1#sh ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(192.168.1.100, 232.1.1.1), 00:23:54/00:02:55, flags: sTI
  Incoming interface: GigabitEthernet0/0, RPF nbr 130.0.0.1
  Outgoing interface list:
    GigabitEthernet0/1.10, Forward/Sparse, 00:03:52/00:02:55

(*, 224.0.1.40), 00:32:42/00:02:37, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:32:43/00:02:36
```

Kuvio 23. WG4-R1 multicast-reititystaulu

```
Core-R6#sh ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(192.168.1.100, 232.1.1.1), 00:12:33/00:02:43, flags: sT
  Incoming interface: FastEthernet1/1, RPF nbr 130.0.0.17
  Outgoing interface list:
    FastEthernet1/0, Forward/Sparse, 00:12:33/00:02:43

(*, 224.0.1.40), 2d02h/00:02:02, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    FastEthernet1/0, Forward/Sparse, 2d02h/00:02:02
```

Kuvio 24. Core-R6:n multicast-reititystaulu

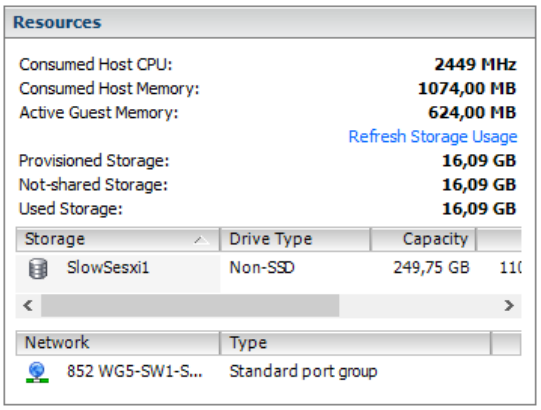
```
WG4-R1#sh ip igmp mem
Flags: A  - aggregate, T - tracked
       L  - Local, S - static, V - virtual, R - Reported through v3
       I - v3lite, U - Urd, M - SSM (S,G) channel
       1,2,3 - The version of IGMP the group is in
Channel/Group-Flags:
       / - Filtering entry (Exclude mode (S,G), Include mode (*,G))
Reporter:
       <mac-or-ip-address> - last reporter if group is not explicitly tracked
       <n>/<m>      - <n> reporter in include mode, <m> reporter in exclude

 Channel/Group                Reporter        Uptime   Exp.  Flags  Interface
/*,232.1.1.1                  172.16.1.5      00:02:30 stop  3MA    Gi0/1.10
 109.163.249.2,232.1.1.1                     00:00:12 02:47 RA     Gi0/1.10
 *,224.0.1.40                 130.0.0.2       07:17:39 02:47 3LA    Gi0/0
```

Kuvio 25. WG4-R1 IGMP-ryhmien jäsenyydet

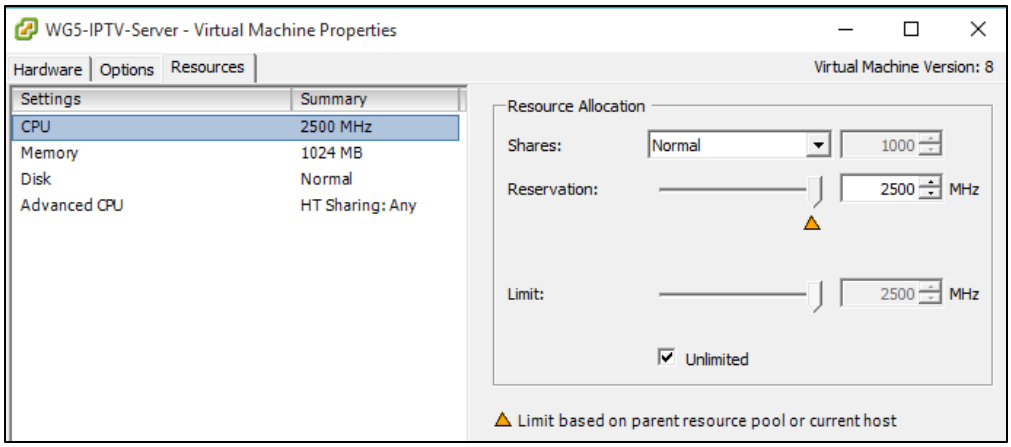
3.5 Palvelimen asennus ja konfigurointi

Palvelimeksi valittiin Linux-distribuutio Ubuntu 14.04.3 LTS, joka oli työn tekovaiheessa uusin versio. Käyttöjärjestelmä soveltuu mainiosti mediapainoitteiseen käyttöön aiempien kokemusten perusteella. Samaa käyttöjärjestelmää käytettiin myös työasemassa. Videoklippien streamaukseen ja katsomiseen käytettäisiin avoimen lähdekoodin ilmaista VideoLAN Media Player (VLC) -ohjelmistoa, jota voi palvelinkäytössä ohjata graafisen käyttöliittymän lisäksi esimerkiksi telnetillä tai HTTP-protokollalla verkkoselaimen kautta. Videotiedostojen lisäksi VLC:llä voidaan streamata esimerkiksi DVB-lähetyksiä tai IP-kameran ulostuloa, joka oli tärkeä ominaisuus ohjelmistoa valittaessa ottaen huomioon toteutuksen alkuperäisen suunnitelman. VLC on yhteensopiva mediasoittimena sekä palvelimena lähes minkä tahansa käyttöjärjestelmän kanssa, mutta DVB-lähetysten uudelleenjakaminen on tuettu vain Linux-pohjaisissa käyttöjärjestelmissä.

Palvelin päätettiin toteuttaa LabraNetin virtualisoidussa VMWare ESXi -ympäristössä. Ympäristössä olevat virtuaalityöasemat ja -palvelimet voitiin liittää SpiderNetin työryhmäverkkoihin. IPTV-palvelulle perustettiin verkkoympäristön ylläpitäjien puolesta työn tekijän hallittavaksi resurssiosio, johon palvelin sekä työasemat asennettin. Resurssiosioon oli käyttöoikeudet ympäristön ylläpitäjillä sekä työn tekijällä. Palvelimen asennuksessa ei esiintynyt mitään poikkeavaa. Virtuaalityöasemien asennuksen lopputuloksena pystytettiin yksi IPTV-palvelua hallitseva palvelin ja yksi työasema, jolla vastaanotettaisiin lähetykset. Molemmille virtuaalilaitteille allokoitiin alustavasti yksiytiminen 2500 MHz prosessori ja 1024 MB muistia. 2500 MHz oli yhden prosessoriytimen osalta toteutusta varten asetettu maksimiarvo (ks. Kuvio 26 & Kuvio 27). Muistin ja prosessoriytimien määrää voitiin kasvattaa tarpeen mukaan.



Kuvio 26. IPTV-palvelimen käyttämät asetetut resurssit



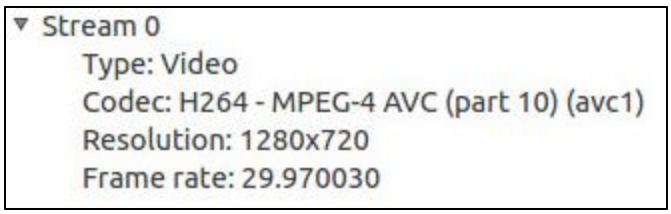
Kuvio 27. IPTV-palvelimen prosessorin resurssiraja

Ympäristössä oli valmiina jokaiselle SpiderNetin työryhmälle allokoituja Linux-palvelimia ja Windows-työasemia, joita ei tässä työssä käytetty. IPTV-palvelin asetettiin

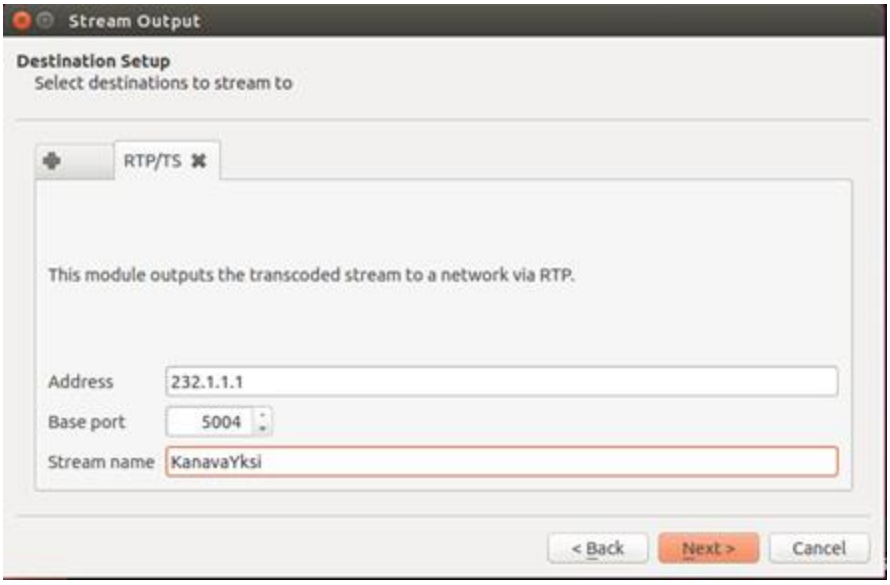
WG5-SW1-SRV-palvelimelle tarkoitettuun verkkoon, jolloin se saatiin yhdistettyä SpiderNetin fyysiseen WG5-SW1 -kytkimeen. Myös vastaanottava työasema liitettiin samaan tapaan haluttuihin verkkoihin.

VLC asennettiin palvelimelle komentoriviltä komennolla *sudo apt-get install vlc* ja oletuksena enabloitu graafinen käyttöliittymä saatiin näkymään komennolla *vlc*. Soittimen asetuksista tuli sallia useampi yhtaikainen instanssi valikosta Tools -> Preferences ja poistamalla valintaruutu kohdasta *Allow only one instance*. Tämän jälkeen voitiin käynnistää toinen soittimen instanssi graafisessa käyttöliittymässä toista mediastreamia varten.

Lähetysten laaduksi päätettiin asettaa 720p-tarkkuus nopeudella noin 30 videokehystä sekunnissa, jota noudatettiin lähetettäviä videoklippejä etsiessä. Maksuttomat ja tekijänoikeusvapaat videoklipit ladattiin YouTube-verkkovideopalvelusta YouTubeDLG-ohjelmistolla. Ensimmäisen kanavan streamaus aloitettiin palvelimella oikeaklikkaamalla käyttöliittymän pääikkunaa ja valitsemalla Open Media -> Open File. Streamattavia tiedostoja oli ladattu palvelimelle kolme (loop01.mp4, loop02.mp4 ja loop03.mp4), jotka olivat identtisiä laadultaan, hieman vaihdellen kestoltaan ja siten tiedostokooltaan (ks. Kuvio 28). Mediaa streamatessa tiedostokoolla tai kestolla ei ollut merkitystä. Streamattavaksi videoklipiksi valittiin ensin loop01.mp4. Valinnan jälkeen valittiin Play -> Stream. Ohjattu streamaus -näkymä näyttää valitun tiedoston, jonka hyväksynnän jälkeen tuli valita haluttu siirtoprotokolla medialle. Siirtometodina käytettiin Real Time Transport Protocol (RTP) / MPEG Transport Stream (TS) –valinta. Koska multicast-tekniikkana oli käytössä PIM-SSM, valittiin lähdeosoitteeksi SSM-spesifinen osoite 232.1.1.1 oletusportilla 5004 (ks. Kuvio 29).



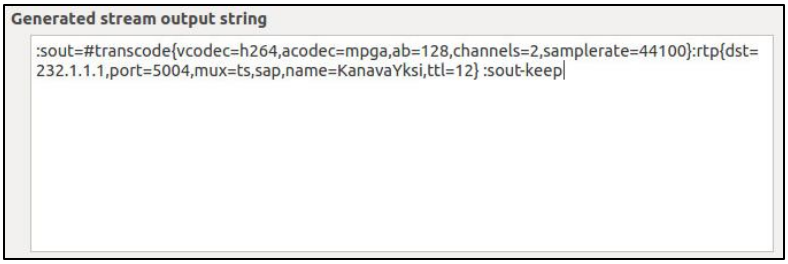
Kuvio 28. Videoklippien laatu.



Kuvio 29. Kanavan multicast-osoitteen valinta.

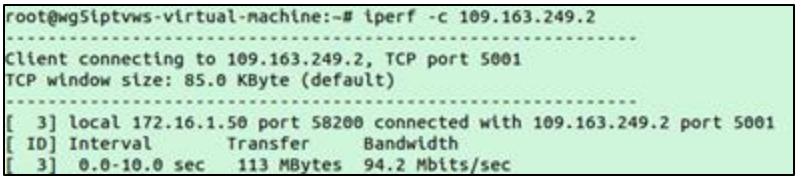
Stream name -kenttään voidaan sijoittaa lähetyksen haluttu nimi, jota käytettäisiin RTP-tiedonsiirrossa Session Announcement Protocol (SAP) -protokollan kanssa. SAP-protokollaa käytetään yleensä lähiverkon sisäiseen multicast-streamien mainostukseen, jota ei tässä työssä käytetty, sillä lähetykset vastaanotettiin eri työryhmässä. SAP-protokolla on myös IETF:n mukaan vasta kokeellisessa vaiheessa (RFC 2974).

Seuraavaksi määritettiin median transkoodaus, jossa voidaan haluttaessa muuttaa videon tyyppiä ja laatua lähetyspäässä. Koska lähetettävät tiedostot olivat mp4-formaatissa, transkoodauksen valintana pidettiin myös MPEG-4 eli H.264 (MPEG-2) + MP3. Transkoodauksen valinnan jälkeen ohjelma näytti laaditut asetukset generoituna koodinpätkänä, johon lisättiin muuttuja *ttl=12* (ks. Kuvio 30). Time to Live -arvo määrittää verkossa ylitettyjen reititettyjen hoppien määrän, joka voidaan asettaa topologiaan sopivaksi. Ilman TTL-arvoa multicast-lähetyksen reitti mainostui työryhmästä toiseen mutta itse video- ja audiodata ei liikkunut lähetyspään reitittimestä verkkoon, sillä **VLC:n oletusarvo TTL:lle on yksi**.



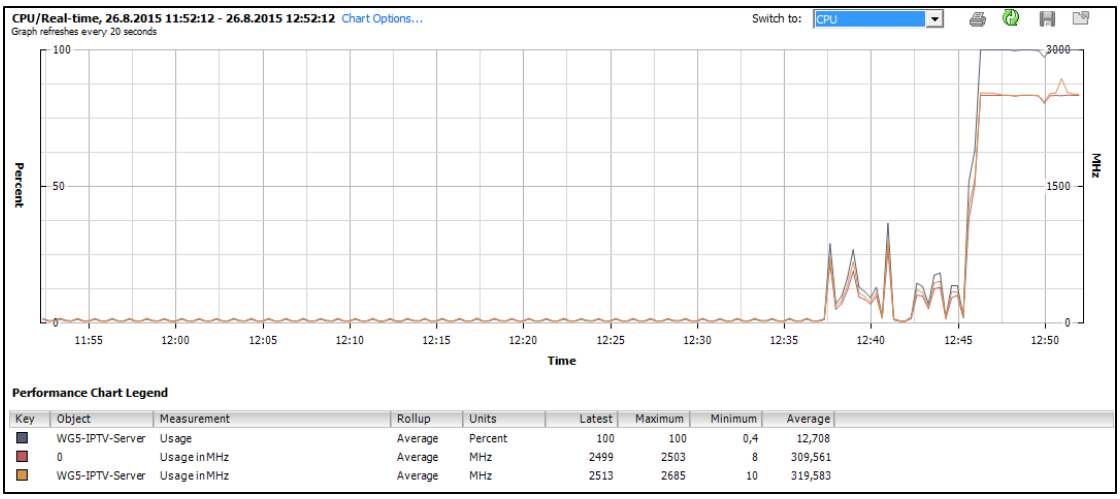
Kuvio 30. Stream-asetusten generoitu koodi, MPEG4-transkoodaus

MP4-transkoodattu lähetys osoittautui välittömästi liian raskaaksi palvelimelle, jonka takia työasemalta katseltun mediastreamin laatu oli erittäin heikkoa. Kaistan riittävyys mitattiin iperf-ohjelmistolla, joka mittasi palvelimen ja työaseman välisen yhteyden nopeudeksi n. 90 Mbit/s (ks. Kuvio 31), joten syynä oli selkeästi palvelimen riittämätön prosessointiteho.

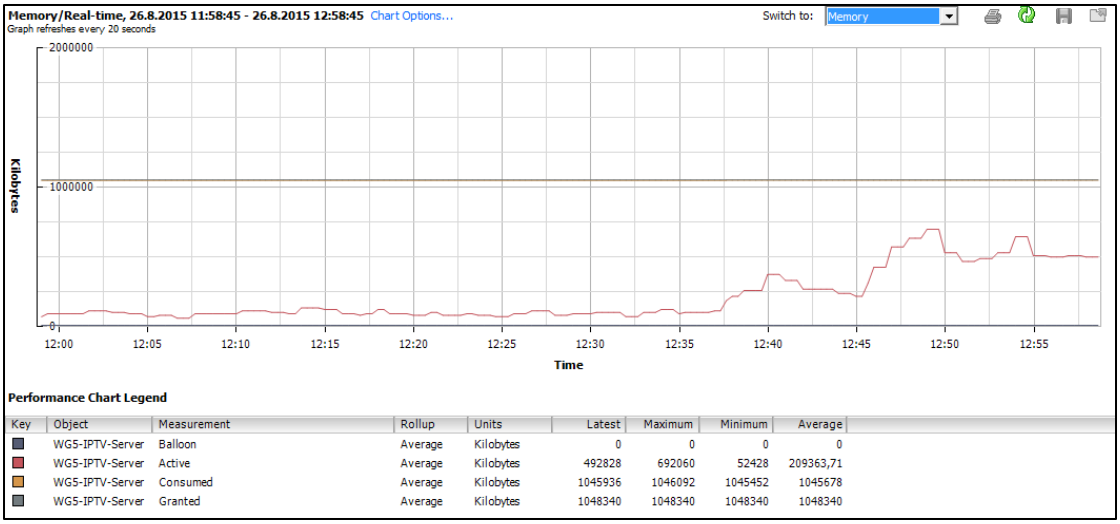


Kuvio 31. Työryhmien välisen kaistan mittaus

Suorituskykyä voitiin seurata ESXi-ympäristön resurssienvalvontatyökaluilla. Prosessorin osalta yhden MPEG4-streamin lähetys nosti prosessorinkäytön välittömästi sataan prosenttiin (ks. Kuvio 32). Muistin käyttö riitti yhteen streamiin, joka kulutti keskimäärin 500 MB (ks. Kuvio 33). Kuvioista nähdään myös kuinka vähän resursseja palvelin vei idle-tilassa ennen lähetystä.

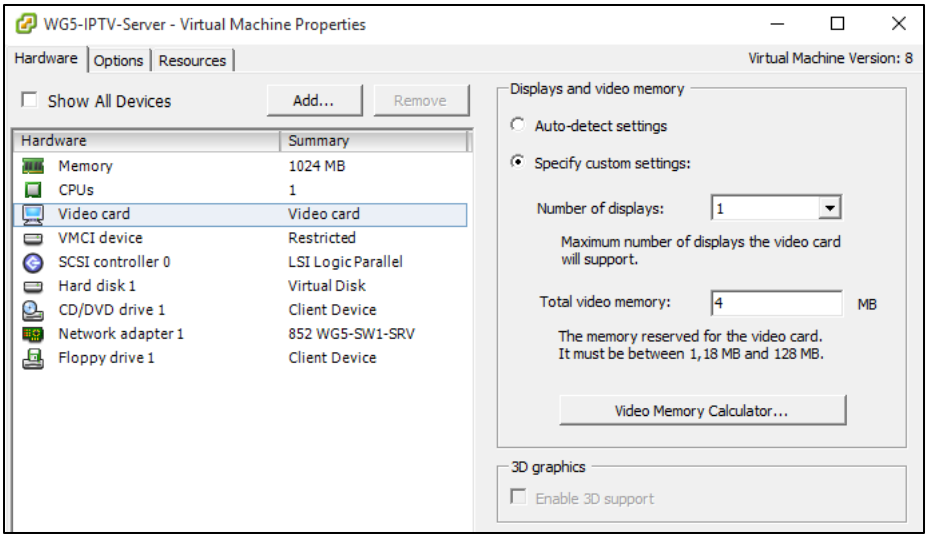


Kuvio 32. IPTV-palvelimen prosessorinkäyttö, MPEG-4 -transkoodaus



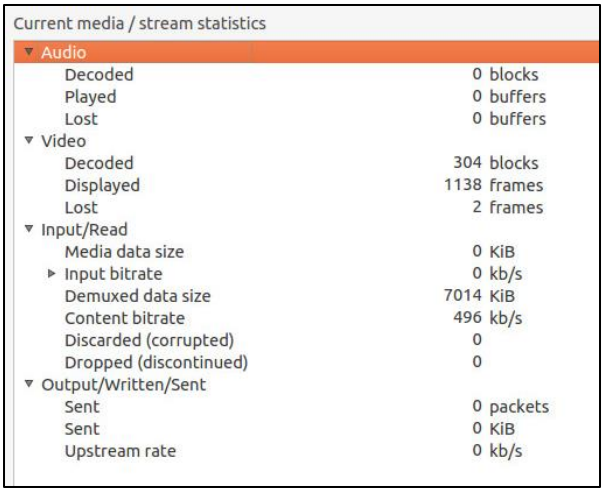
Kuvio 33. IPTV-palvelimen muistinkäyttö, MPEG-4 -transkoodaus

Suorituskykyyn liittyviä ongelmia tutkiessa huomattiin, että ESXi-ympäristössä ei ollut tarjolla palvelimelle minkäänlaista fyysistä grafiikkaprosessoria, joka on videodatan ja varsinkin enkoodauksen käsittelyssä yleensä pääkomponentti. Virtuaalipalvelimen asetuksista pystyttiin asettamaan videoadapteriksi ainoastaan virtualisoitu ”Video card”, eli valittavana ei ollut dedikoitua grafiikkaprosessoria (ks. Kuvio 34).



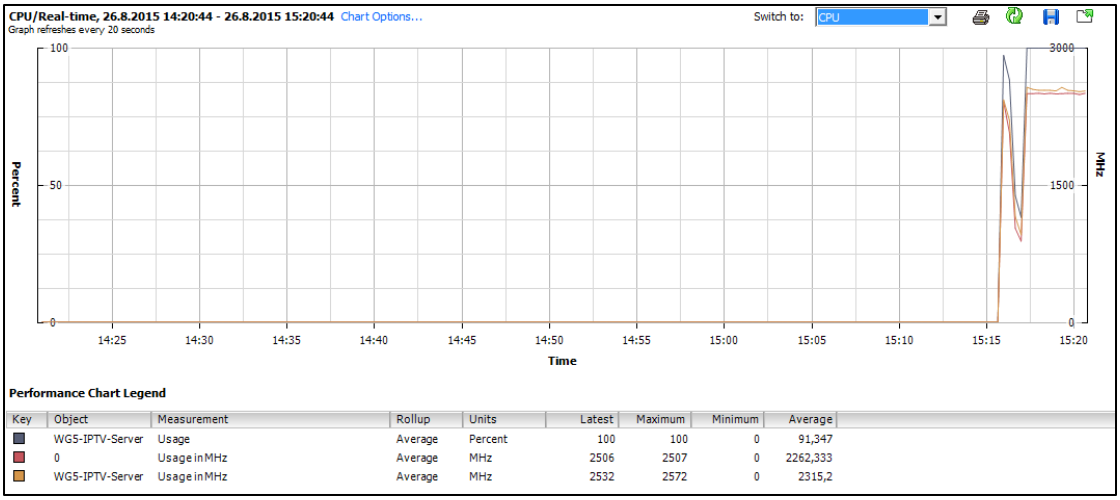
Kuvio 34. IPTV-palvelimen grafiikkaprosessori

Työasemalla paikallisesti toistettuna videot näkyivät täydellisesti, joten allokoitu vastaanottopään graafinen prosessointiteho riitti toistaa lähetystä. Streamin lähettämiä videokehyksiä voitiin analysoida vastaanottopäässä käyttämällä VLC:n statistiikkatyökalua. Kuvion 28 mukaista lähetystä vastaanotettiin työasemalla minuutin ajan. Statistiikkojen mukaan kehyksiä saapui noin 1100, vaikka minuutin ajalla n. 30 FPS x 60 s olisi pitänyt vastaanottaa noin 1800 kehystä (ks. Kuvio 35).

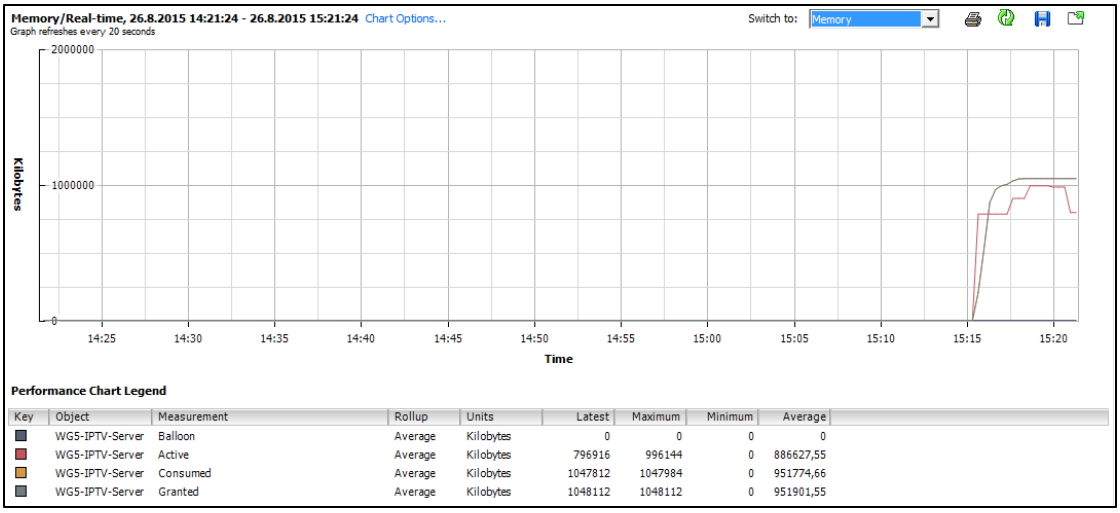


Kuvio 35. H.264 + MP3 statistiikat minuutin ajalta

Kevyemmillä transkoodausoptioilla kuten iPod Standard Definition -enkoodaus näkyi ilman suurempia katkoksia mutta tietysti huomattavasti pienemmällä pikselitiheydellä. Palvelimen suorituskykyä seurattiin samaan tapaan kuin MPEG-4 -lähetyksessä, jossa havaittiin samanlainen kapasiteetin kulutus (ks. Kuvio 36 & Kuvio 37).

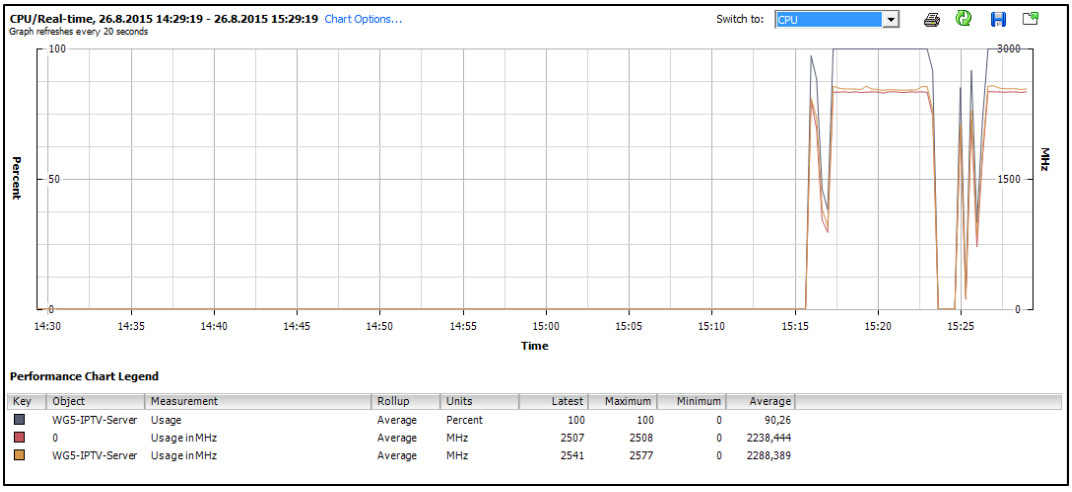


Kuvio 36. IPTV-palvelimen prosessorikäyttö, iPod SD –transkoodaus

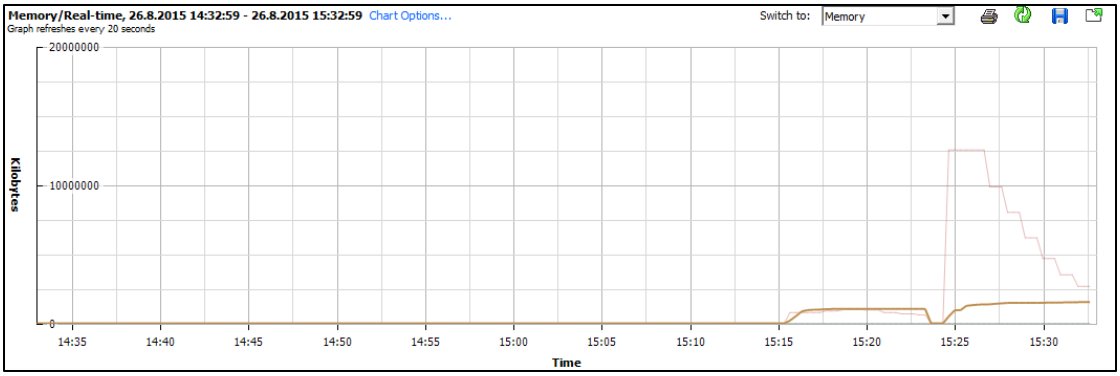


Kuvio 37. IPTV-palvelimen muistinkäyttö, iPod SD -transkoodaus

ESXi-ympäristössä oli mahdollisuus kasvattaa grafiikkaprosessorin muistia oletusarvosta 4 MB arvoon 128 MB, joka tehtiin. Tämän lisäksi muistin kapasiteettia kasvatettiin 1 GB:stä 16 GB:iin. Prosessorin pullonkaula esiintyi jälleen, kun MPEG-4 -transkoodauksellinen stream laitettiin päälle (ks. Kuvio 38). Muistia käytettiin nyt reilu 1 GB, joten lisäallokaatiosta oli hyötyä (ks. Kuvio 39). Vastaanottolaitteella kuva oli edelleen katkonaista, jonka syy voitiin siten rajata suurella todennäköisyydellä prosessoriin tai grafiikkaprosessoriin. Grafiikkaprosessorin analysointiin ei ollut ympäristössä työkaluja. Transkoodauksen vaikutus prosessoriin kokeiliin myös, mutta sataprosenttinen prosessoritehon käyttö esiintyi myös ilman minkäänlaista transkoodausta lähetyksessä. Muistin noston jälkeen kokeiltiin myös prosessoriydinten määrän kasvattamista yhdestä neljään, jolla ei ollut merkitystä havaitun lähetyksen kuvanlaatuun.

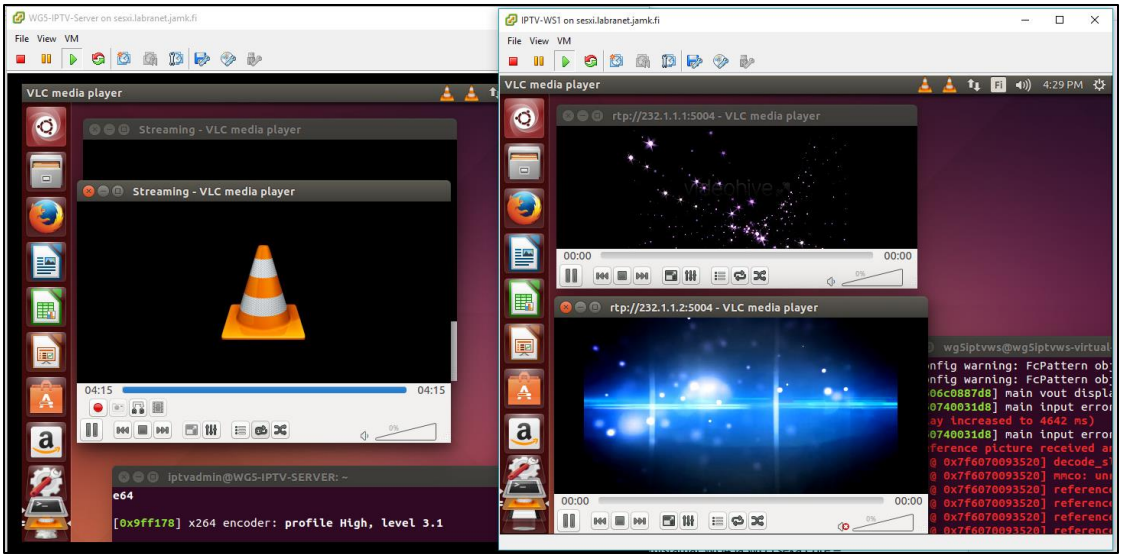


Kuvio 38. IPTV-palvelimen prosessorinkäyttö #2, MPEG-4 transkoodaus



Kuvio 39. IPTV-palvelimen muistinkäyttö #2, MPEG-4 transkoodaus

Havaituista suorituskyvyn puutteellisuuksista huolimatta streamausta testattiin. Asetukset yhdelle lähetykselle olivat nyt valmiit, ja ne voitiin toistaa muita kanavia varten. Palvelimella käynnistettiin toinen VLC-instanssi, jossa jaettiin tiedostoa *loop02.mp4* osoitteeseen 232.1.1.2. Osoitteistus perustui luvussa 3.2 esitettyyn TeliaSonera AB:n IPTV-toteutuksen multicast-osoitteistukseen. Molempia käynnissä olevia kanavia voitiin nyt tarkastella työasemalla yhtaikaisesti (ks. Kuvio 40). Palvelimen tehokkuus riitti streamata kahta kanavaa vain muutaman sekunnin ajan ja kolmella kanavalla työasema ei vastaanottanut lähetyksiä ollenkaan kun käytössä oli kuvion 28 mukainen transkoodaus.



Kuvio 40. Kahden kanavan toisto, iPod Standard Definition –transkoodaus

Streamien toiminta sujui ongelmitta pois lukien palvelimen asettamat rajoittuvuudet ja siitä johtunut kehysten puute. Työasemalla avattiin VLC Media Playerilla verkkosijainti molempiin kanaviin osoitteessa *rtp://<palvelimen NATattu osoite>@<multicast-ryhmän osoite>:<portti>*.

Graafisen käyttöliittymän käytön sijaan tutkittiin myös streamien ajamista komentoriviltä ja toiston hallintaa HTTP-käyttöliittymän kautta. VLC:tä ei voida ajaa komentorivillä root-käyttäjänä tai sudo-moodissa. Komentoon *vlc* lisättiin muutamia argumentteja ja ajettiin se:

IPTVAdmin@WG5-IPTVServer$ vlc file:///var/vlma/loop01.mp4 -I http –daemon --sout "#rtp{dst=232.1.1.1,port=5004,ttl=12,mux=ts}" –sout-keep –http-port 9090 –http-password <http-käyttöliittymän salasana>

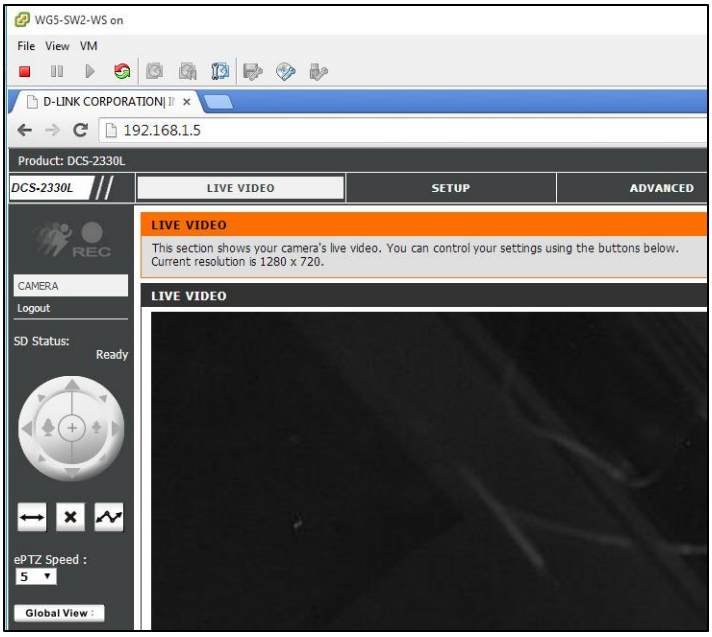
Komennolla lähetettiin *loop01.mp4* -tiedostoa tausta- eli daemonprosessina, jota voitiin kontrolloida verkkoselaimen kautta osoitteessa *http://127.0.0.1:9090*. Taustaprosessoinnin avulla voitiin käynnistää komentoriviltä jokainen kanava muuttamalla *–sout* -argumentin ominaisuuksia jokaiselle multicast-osoitteelle. Komentoon voidaan tarvittaessa asettaa lähetyksen käyttämä transkoodaus *–sout:*n jälkeen argumentilla *#transcode{vcodec=<videokoodekki>,acodec=<audiokoodekki>,vb=<videon bittinopeus>,ab=<audion bittinopeus>,samplerate=<audion näytteenottotaajuus>"*. Komentoriviä voidaan käyttää myös streamin katsomiseen, mutta työasemilla graafisen käyttöliittymän käyttö oli riittävä.

Lähetykset haluttiin ajettaviksi skripteinä palvelimella. */var/vlma/stream.sh* –skriptiin määritettiin jokaisen kanavan lähettävä daemon-komento yllä olevan esimerkin mukaisesti. *stream.sh* –skripti asetettiin ajettavaksi tiedostoksi komennolla *chmod +x /var/vlma/stream.sh*. Skriptillä voitiin siten käynnistää yhtaikaisesti jokainen haluttu IPTV-lähetys yhdellä komennolla. Luotuun skriptiin voitaisiin tarvittaessa lisätä myös muita VLC-komentoja. VLC:n instansseihin ei löydetty dokumentaatioista eikä käyttäjien ratkaisuista valmista prosessien tappokomentoa, jonka takia lähetysten keskeyttäminen onnistui ainoastaan komennolla *killall vlc*.

3.6 IP-kameran käyttö

Työssä päätettiin lisätä palvelun kolmanneksi "televisiokanavaksi" IP-kameran videosyöte. D-Link DCS-2330-L –mallinen kamera liitettiin WG5-SW2 –kytkimen porttiin FastEthernet 0/17. Samaan kytkimeen liitetyllä SpiderNetin Windows-työasemalla suoritettiin kameran asennus. Asennusohjelmisto ladattiin Internetistä D-Link:in tukisivustolta. Asennusvelhon ajamisessa noudatettiin annettuja asennusohjeita.

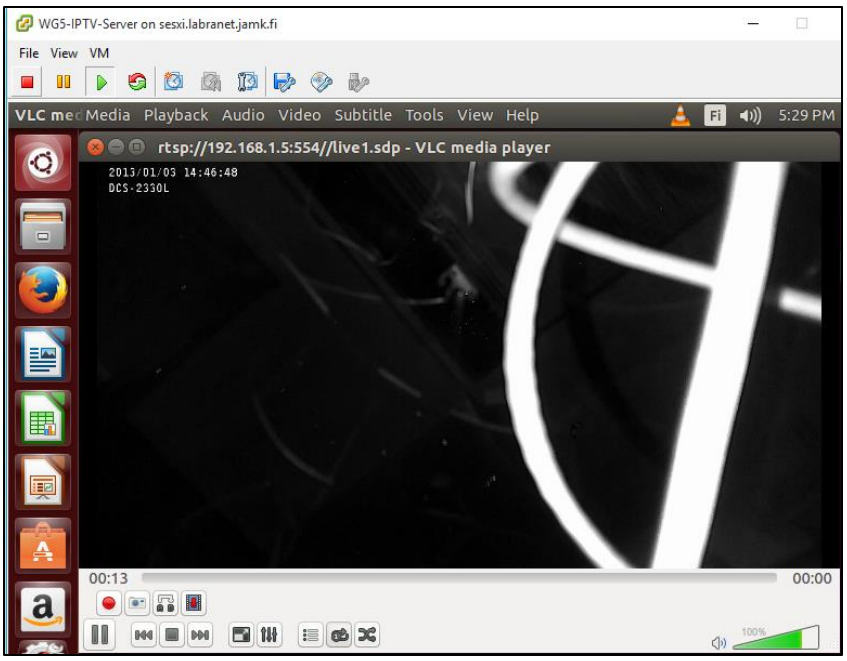
DCS-2330-L –kamera tukee Internetin kautta seurattavaa lähetystä, mutta asennuksessa valittiin verkkoyhteydeksi **lähiverkon kautta käyttö**. Tätä valintaa käyttämällä ei vaadittu asennuksessa Internetyhteyttä kameralle. Asennuksen yhteydessä ilmeni ongelmia DHCP:n kanssa. Kameran IP-osoite oli oletuksena 192.168.0.20, joka ei toiminut 192.168.0.0/24- IP-osoitteistusta käyttävässä VLAN-pohjaisessa verkossa. Kameralle saatiin sopiva IP-osoite ajamalla asennusohjelma uudelleen käyttäen verkkoasetuksena staattista IP-osoitetta. Staattisella osoitteella kameran asennusohjelmisto ei voinut Internetyhteyden puutteen takia ajaa loppuun, mutta jälleen ajettaessa valitsemalla lähiverkon kautta käyttö **kamera sai reitittimen DHCP-palvelimelta soveltuvan IP-osoitteen 192.168.1.5**. Kameraa voitiin asennusprosessin jälkeen hallita verkkoselaimelta lähiverkossa (ks. Kuvio 41).



Kuvio 41. IP-kameran käyttöliittymä

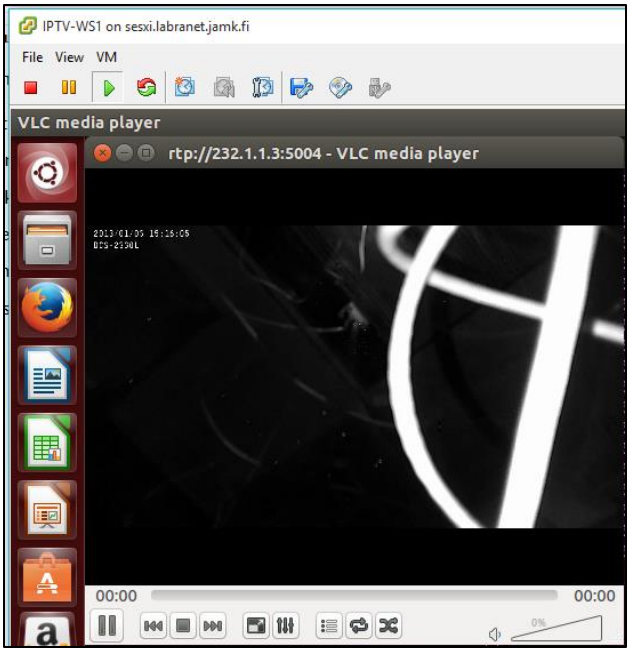
Kameran kaappaama kuva tuli saada IPTV-palvelimelle, josta se tultaisiin lähettämään SSM-multicastina osoitteeseen 232.1.1.3. Kamera ei tukenut SSM-multicastia lainkaan, eikä kameran IP-osoitteen jakaminen palvelinverkon ulkopuolelle ollut muutenkaan suotavaa. Kameralähetyksen seuraaminen IPTV-palvelimelta testattiin ensin käyttämällä RTSP (Real Time Streaming Protocol)-lähetystä. Palvelimella avattiin VLC verkko-osoitteeseen *rtsp://192.168.1.5:554//live1.sdp*. *Live1.sdp* on RTSP-protokollan kommunikointiväylä, joka voitiin profiilin tapaan konfiguroida kameran hallintapaneelista. Osoitteen syötettyä VLC vaati myös kameran hallintatilin

käyttäjätunnuksen ja salasanan. Tämän jälkeen kameran kuvaa voitiin seurata IPTV-palvelimelta (ks. Kuvio 42).



Kuvio 42. IP-kameran lähetys vastaanotettuna IPTV-palvelimella

IPTV-palvelimelta konfiguroitiin seuraavana VLC:n graafisen käyttöliittymän streamausvelhon kautta kameran lähetyksen uudelleenstreamaus WG4-työryhmään. Streamin lähteeksi asetettiin edellisen mukainen RTSP-yhteys (käyttäjätili ja salasana tarvittiin uudelleen) ja lähetys streamattiin SSM-osoitteeseen 232.1.1.3 käyttäen RTP:tä. **VLC:tä käytettäessä tuli muistaa asettaa TTL-arvo riittävän suureksi**. Kameran lähetystä voitiin nyt seurata WG4-työryhmän työasemalta VLC:llä verkko-osoitteesta *rtp://<palvelimen NAT-osoite>@<SSM-osoite>:<määritetty portti>* (ks. Kuvio 43). Huomattavaa oli, että uudelleenstreamatun kameralähetyksen katsomiseen ei tarvittu kameran käyttäjätiliä eikä salasanaa. Multicast-liikenteen toiminta varmistettiin myös WG5-R1-reitittimeltä siten, että liikenne kulki halutusti palvelimen ja työaseman eikä kameran ja työaseman välillä (ks. Kuvio 44). *show ip multicast route* –komennolla nähtiin, että 232.1.1.3-osoitteeseen lähetetty multicast-liikenne tuli palvelimelta IP-osoitteesta 192.168.1.100.



Kuvio 43. IP-kameran lähetys vastaanotettuna WG4-työasemalla

```
WG5-R1#sh ip mrou
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       V - RD & Vector, v - Vector
Outgoing interface flags: H - Hardware switched, A - Assert winner
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 239.255.255.250), 2d14h/00:02:34, RP 0.0.0.0, flags: SJC
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/1.10, Forward/Sparse, 2d14h/00:02:34

(192.168.1.100, 232.1.1.3), 00:12:52/00:03:25, flags: sT
  Incoming interface: GigabitEthernet0/1.10, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 00:05:10/00:02:46

(*, 224.0.1.40), 5w1d/00:02:35, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet0/0, Forward/Sparse, 5w1d/00:02:35
```

Kuvio 44. IP-kameran multicastin todennus

Seuraavana laadittiin daemon-skripti IPTV-palvelimelle aiempien multicast-lähetysten tapaan. Komentoon tuli huomioida RTSP-käyttäjänimen ja salasanan lisäys:

vlc rtsp://192.168.1.5/live1.sdp –rtsp-user=admin –rtsp-pwd=<kameran salasana> -I http –daemon –sout"#rtp{dst=232.1.1.3,port=5004,ttl=12,mux=ts}" –sout-keep –http-port 9090 –http-password <http-salasana>

Oheinen komento lisättiin aiemmin konfiguroitujen tiedostolähetysten tapaan määrättyyn */var/vlma/stream.sh* –skriptitiedostoon.

3.7 Tietoturva

Palvelimen kovennus konfiguroitiin siten, että sisääntulevat yhteydet sallittiin vain verkon 192.168.1.0/24 sekä multicast-liikenteen osalta. Kaikki Iptables-komennot suoritettiin sudo-ohjelman kautta. Palvelimella ajettiin ensin komento, jolla asetetaan policy pudottamaan kaikki paketit, joita ei myöhemmillä säännöillä aseteta sallituksi:

IPTVAdmin@WG5-IPTVSERVER$ iptables –P INPUT DROP
iptables –P FORWARD DROP
iptables –P OUTPUT DROP

Lähiverkon liikenne sallittiin komennolla *iptables –A INPUT –s 192.168.1.0/24 –j ACCEPT* sekä *iptables –A INPUT –s 192.168.1.0/24 –j ACCEPT*. Seuraavaksi sallittiin ICMP-liikenne komennoilla *iptables –A INPUT –p icmp –j ACCEPT* sekä *iptables –A OUTPUT –p icmp –j ACCEPT*. Multicast-liikenteen toimivuudelle ei tarvittu muita konfiguraatioita palvelimella. Iptables-asetukset olivat nyt valmiit (ks. Kuvio 45).

```
iptvadmin@WG5-IPTV-SERVER:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  192.168.1.0/24       anywhere
ACCEPT     icmp --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  192.168.1.0/24       anywhere
ACCEPT     icmp --  anywhere             anywhere
```

Kuvio 45. Iptables-säännöt palvelimella

Seuraavaksi voitiin asettaa luodut Iptables-säännöt pysyviksi, muuten jokainen sääntö tulee konfiguroida uudelleen palvelimen käynnistyksen jälkeen. *sudo sh –c ”iptables-save > /etc/iptables.rules”* –komennolla tallennettiin luodut säännöt tekstitiedostoksi. Iptables ja Ubuntussa oletuksena päällä oleva Network-manager -palvelu eivät ole yhteensopivia, joten palvelimelle tuli konfiguroida staattinen IP-osoite. Network-manager suljettiin komennolla *sudo service network-manager stop*. */etc/network/interfaces* -tiedosto avattiin tekstieditorilla ja muokattiin seuraavaksi:

auto lo
uface lo inet loopback

```
iface eth0 inet static
 address 192.168.1.100
 netmask 255.255.255.0
 gateway 192.168.1.1
 pre-up iptables-restore < /etc/iptables.rules
```

Komennon *sudo ifdown eth0 && ifup eth0* ajamisen jälkeen palvelimella oli staattinen IP-osoite. WG5-R1 -reitittimestä voitiin nyt poistaa palvelimelle tarkoitettu MAC-osoite:

```
WG5-R1#service dhcp
 no ip dhcp pool Server
```

Verkkolaitteiden tietoturvassa noudatettiin Cisco Systemsin laatimia Best Practices -ohjeita, joista otettiin osa tarpeen mukaan käyttöön. Konfigurointi aloitettiin sulkemalla kaikki ei-käytössä olevat fyysiset rajapinnat jokaisessa reitittimessä ja kytkimessä komennolla *interface x/x -> shutdown*. Paikalliset käyttäjätunnukset, joilla on konfigurointioikeus verkkolaitteissa, tuli suojata. Komennolla *enable secret <salasana>* asetettiin salasana privileged EXEC -moodille. Salasanoja ei kovennettu tämän enempää, sillä kryptattuja salasanoja ei voida resetoida ilman fyysistä pääsyä verkkolaitteille. Kryptauksen tarpeellisuus tulee kuitenkin huomioida ympäristöä asennettaessa, muuten salasanat ovat plain text -formaatissa verkkolaitteiden konfiguraatiotiedostossa. Salasanojen kryptaus verkkolaitteilla voidaan enabloida komennolla *service password-encryption*. Korkeatasoisempi käyttäjän autentikointi voidaan myös suorittaa esim. keskitetyllä RADIUS-palvelimella, jota ei käsitelty tässä toteutuksessa.

Core-verkon reititysprotokollat kovennettiin siten, että verkon reittitauluja ei jaeta jos laitteella ei ole oikeuttavaa kryptografista avainta. EIGRP-reititysprotokollalle jokaisella Core-verkon reitittimellä luotiin avain, joka sidottiin reititykseen liittyvään rajapintaan:

```
Core-Rx(config) key chain eigrpkey
key 1
 key-string <salasana>
interface <Core-verkon linkki>
 ip authentication mode eigrp 10 md5
 ip authentication key-chain eigrp 10 eigrpkey
```

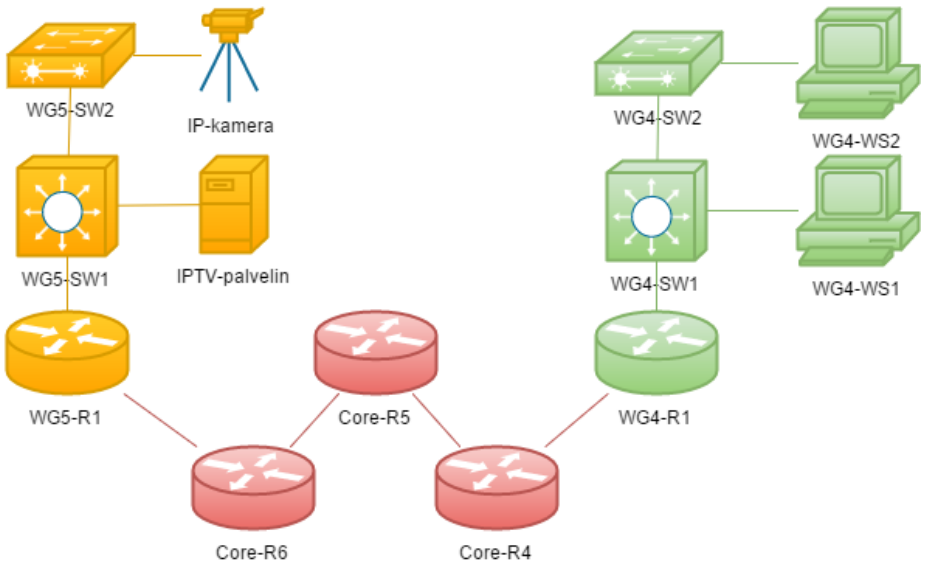
BGP-reititys kovennettiin EIGRP:n tapaan MD5-autentikoinnilla ja lisäksi TTL-varmistuksella, jolla määritetään reititystietojen mainostuksen raja.

Core-Rx(config) router bgp <bgp instanssi>
neighbor <peer-reitittimen IP-osoite> password <salasana>
neighbor <peer-reitittimen IP-osoite> ttl-security hops <hoppien maksimimäärä>

4 Tulokset

Opinnäytetyön toteutuksen tuloksena saatiin SpiderNetin Core-verkon ylittävä usean yhtaikaisen lähteen audio- ja videolähetys (ks. Kuvio 46). WG5-työryhmän palvelimen mediatiedostot sekä verkkoon liitetyn IP-kameran videokuva lähetettiin VLC-ohjelmiston avulla multicast-liikenteenä verkkoon. Lähetyksiä voitiin tarkastella toisesta työryhmästä WG4 Core-verkon läpi. Asennetun palvelimen ja verkon konfiguraatioiden mukaan voitaisiin myös rakentaa varsinaisia DigiTV-DVB-lähetyksiä jakava palvelin, mikäli SpiderNet-laboratorioverkossa niitä päätettäisiin jakaa.

Työn tavoitteisiin operaattorin IPTV-palvelun toteutuksesta ei päästy tekijänoikeuslakien asettamien esteiden takia. DVB-lähetysten arkkitehtuuria ei voitu tutkia ilman varsinaista laitteistoa tai DVB-lähetysten simulointiin tarkoitettua testauslaitteistoa, jota SpiderNet-ympäristössä ei ollut. DVB-lähetysten puutteen takia toteutus ei vastannut varsinaista operaattorin IPTV-järjestelmää.



Kuvio 46. Toteutuksen topologia

5 Laite- ja ohjelmistoehdotuksia

Asennettu IPTV-palvelin ei riittänyt teknisiltä ominaisuuksiltaan toimeksiannon asettamiin vaatimuksiin IPTV-palvelulta. Palvelimen prosessointitehokkuus ei riittänyt videon käsittelyyn käytetyllä kuvanlaadulla. Ongelma voidaan ratkaista palvelimen prosessorin päivityksellä sekä grafiikkaprosessoreilla. Prosessorin hankintaehdotuksia on vaikea antaa, sillä se riippuu laajalti tarjotun palvelun kanavien määrästä ja laadusta. Grafiikkaprosessoreita ovat järeät tuotantokäyttöön suunnatut prosessorit (esim. NVIDIA Tesla-sarja) tai edullisemmat kuluttajatason PC-laitteille suunnatut PCI/PCIe -rajapintoihin liitettävät prosessorit. Suosituksena voidaan antaa hankittavaksi toteutusympäristöstä (virtuaalinen / fyysinen palvelin) riippuen hankittavaksi grafiikkaprosessori, joka hoitaa dedikoidusti televisiolähetysten enkoodauksen prosessointia lähetysformaattiin. Grafiikkaprosessorin hankinnalla lievitetään palvelimen prosessorin kuormaa, jolloin toteutuksessa törmättyyn sataprosenttinen prosessorikäyttö ja siitä johtunut kehysten putoaminen voidaan välttää.

DVB-pohjaisessa toteutuksessa tulisi viritinratkaisua suunnitellessa aloittaa ympäristössä käytetystä DVB-standardista ja halutusta jaettavien kanavien määrästä ja DVB-standardin asettaman multipleksoinnin mahdollistamasta kanavamäärästä, jolla saadaan määritettyä tarvittavien virittimien määrä. Toimeksiannossa ei määritetty haluttua kanavien määrää, joten suoranaisia ehdotuksia virittimien määrälle ei voida antaa. Salattuja kanavia varten tarvittaisiin lisäksi CI/CI+/CMA -ominaisuuksilla varustettuja virittimiä tai lisäkortteja. Koska SpiderNetissä oli jo yksi DVB-viritin, sen toiminta kannattaisi ensin kartoittaa, mikäli toteutus tehdään. IPTV-palvelua voidaan aina rakentaa asteittain ja useilla eri virittimillä, jonka takia hankintoja ei tarvitse tehdä köntässä.

Viritinsuosituksia laadittaessa jätettiin pois varsinaiset operaattorien käyttämät tuhansien asiakkaiden palveluun tarkoitetut DVB-virittimet, joista ei juurikaan ole tarjolla julkisia hinnoitteluja. Todennäköisesti huomattavasti edullisempi ratkaisu on käyttää palvelimen USB/PCI/PCIe -rajapintaan yhdistettävät virittimiä, kuten Kernenin (2012) ratkaisussa oli tehty. Tällainen implementaatio vastaa toimintatavaltaan riittävästi käsiteltyjä operaattorien IPTV-palveluita mutta pienemmässä skaalassa.

Itse rakennettu IPTV-palvelin antaa valmiin ratkaisun sijaan myös paremmat mahdollisuudet DVB-tekniikoiden toiminnan monitorointiin ja analysointiin. Kernen listaa käyttämiään laitteita dokumentissaan, joita voidaan harkita, mikäli toteutuksessa tarvitaan uusia virittimiä tai CI/CAM-moduuleita (ks. Taulukko 3). Osa luetelluista malleista oli työn tekovaiheessa tuotannosta poistuneita, mutta merkkiehdotuksia voidaan pitää referenssinä hankintoja suunniteltaessa.

Taulukko 3. DVB-virittimiä ja CAM-moduuleita (Kernen 2012, 16)

Laitteen valmistaja ja malli	**Rajapintaliitäntä ja käyttöjärjestelmä**	**Tuettu/tuetut DVB-tekniikat ja lisäominaisuudet (esim. CI/CAM -paikat)**	**Saatavuus suomalaisilta jälleenmyyjiltä**
DViCO FusionHDTV DVB-T Dual Digital 4	PCIe Windows, Linux	DVB-T (x2)	Ei
Hauppauge WinTV Nova-T 500	PCI Windows, Linux	DVB-T (x2)	Kyllä
Neotion ACS 3.1	CAM-moduuli		Ei
NetUp Dual DVB-S2-CI	PCIe Windows, Linux	DVB-S2 (x2), CI (x2)	Ei
PowerCAM Pro	CAM-moduuli		Ei
SMiT Viaccess	CAM-moduuli		Kyllä
TBS 6925	PCIe Windows, Linux	DVB-S2	Ei
TBS 6980 Dual tuner	PCIe Linux	DVB-S2 (x2)	Ei
Technotrend S-1500	PCI Windows	DVB-S	Kyllä
Technotrend S2-3200	PCI Windows	DVB-S2, Budget CI	Kyllä
Technotrend T-1500	PCI Windows	DVB-T, Budget CI+Conax	Kyllä

Viritinhankinnoissa tulee huomioida ajurien yhteensopivuus palvelimen ytimen (kernelin) kanssa. LinuxTV -yhteisö (www.linuxtv.org) on erinomainen lähde yhteensopivuuksia selvittäessä ja tarjoaa myös jäsenten itse suunnittelemia laiteajureita ja muita työkaluja Linux-käyttöjärjestelmän IPTV-palvelimille.

Toimeksiannossa tuli löytää avoimen lähdekoodin ratkaisuja palvelimen DVB-lähetysten jakamisohjelmistolle. Toteutuksessa käytetty VLC-ohjelmisto on toimiva ratkaisu myös DVB-virittimiä käytettäessä, mutta lisäksi etsittiin korvaavia ohjelmia. Taulu-

kossa 4 on listattu suosittuja DVB-lähetyksiä jakavia ohjelmistoja, joita voidaan suositella käytettäväksi DVB-implementaatiossa. MuMuDVB on Kernenin (2012) laatimassa laboratoriototeutuksessa käytetty ohjelmisto. Tässä työssä kyseiseen ohjelmaan ei tutustuttu sen tarkemmin, sillä sen käyttö vaatii DVB-virittimen.

Taulukko 4. DVB-lähetyksien kanssa yhteensopivia IPTV-palvelinohjelmistoja

Ohjelmisto	**Tuetut käyttöjärjestelmät**	**Tuetut DVB-tekniikat**	**Tiedonsiirtoprotokollat**
DVBlast	Linux	DVB-S, DVB-S2, DVB-C, DVB-T	RTP, UDP
MuMuDVB	Linux	DVB-S, DVB-T, DVB-C	Multicast (RTP+UDP), HTTP Unicast
MythTV	Linux	DVB-C, DVB-S, DVB-T	HTTP
Tvheadend	Linux, FreeBSD, Android	DVB-S, DVB-S2, DVB-C, DVB-T	HTTP, SAT>IP, HTSP
VDR IPTV-pluginilla	Linux	DVB-S, DVB-T, DVB-C, max. 4 viritintä	RTP, UDP
VLC	DVB-streamaus vain Linuxissa	DVB-S, DVB-T, DVB-C	HTTP, MS-WMSP, RTSP, RTP/MPEG-TS, RTP A/V-profile, UDP, IceCast

6 Yhteenveto

6.1 Palvelun käytettävyyden parannusehdotuksia

Operaattorin tarjoamissa IPTV-palveluissa tutkinnan perusteella halutaan asiakkaan käyttävän palvelun mukana saatua IPTV-vastaanotinta, joka osaa automaattisesti virittyä oikeille multicast-kanaville. Tällaisia palveluntarjoajia olivat ainakin Sonera Oyj, joka ei tarjonnut minkäänlaista virallista listaa IPTV-kanavista. DNA Welho tarjosi tietosivulla jokaisen yksittäisen kanavan multicast-osoitteen. Muilla laitteilla, kuten pelkän verkkoyhteyden omaavilla työasemilla, näiden kanavien hakeminen voi olla manuaalinen prosessi, mikäli käytössä on DNA Welhon ratkaisu. Sonera Oyj:n tapauksessa kanavat olivat asiakkaiden selvittämiä ja raportoimia. Kanavien haun helpottamiseen voitaisiin käyttää VLC-ohjelmiston tukemia soittolistoja, jotka voitaisiin hakea työasemalle esimerkiksi palvelimen verkkosivun kautta. Käyttämällä palvelinta soittolistan hakuun voitaisiin helpottaa käyttäjien pääsyä multicast-lähetyksiin. Soittolistaominaisuutta ei työssä tutkittu aikarajoitteiden takia.

6.2 Pohdinta

Työn suurimpana haasteena oli aiheen laajuus ja lähes rajaton mahdollisuus palvelun ominaisuuksista. IPTV:n toiminnan teoriaan perehtyminen sekä palvelun toteutuksen suunnittelu oli mielekästä ja aiheesta opittiin paljon uutta. Mielenkiintoista oli varsinkin televisiolähetysten kehittymiseen liittyvät asiat ja uskonkin, että IPTV-palvelut tulevat yleistymään entistä enemmän. DVB-lähetysten jakamiseen liittyneet esteet hidastivat hieman projektin etenemistä, mutta vastaavanlainen palvelu kuitenkin saatiin onnistuneesti rakennettua.

Lähteet

ETSI TR 101 198. 1997. Digital Video Broadcasting (DVB); Implementation of Binary Phase Shift Keying (BPSK) modulation in DVB satellite transmission systems.

ETSI TR 102 377. 2009. Digital Video Broadcasting (DVB); DVB-H Implementation Guidelines.

Fairhurst, G. 2009. Internet Communications Engineering - A Tutorial. Viitattu 16.6.2015. http://www.erg.abdn.ac.uk/users/gorry/course/intro-pages/uni-b-mcast.html

Good, R., Bazzano, D. & Lombardi, E. 2010. The Video Encoding Guide: Codecs, Formats, Containers and Settings Explained. Viitattu 8.8.2015. http://www.masternewmedia.org/the-video-encoding-guide-codecs-formats-containers-and-settings-explained/

Hwang, J. 2009. Multimedia Networking: From Theory to Practice. Cambridge: Cambridge University Press.

Kernen, T. 2012. Home Brew IPTV head-end. Viitattu 11.7.2015. https://tech.ebu.ch/docs/events/opensource12/presentations/RMLL_Kernen_Home_Brew_IPTV_v2_optimised.pdf

Koulun sisäverkko. N.d. Tekijänoikeusjärjestö Kopioston laatima vastaus. Viitattu 25.7.2015. http://www.kopiraitti.fi/ukk/fi_FI/koulun_sisaverkko/

Lampinen, J. 2013. IPTV-palvelualustan valinta. Opinnäytetyö. Jyväskylän ammattikorkeakoulu, teknologiaosaamisen johtamisen koulutusohjelma.

Minoli, D. 2008. IP Multicast With Applications to IPTV and Mobile DVB-H. Hoboken: John Wiley & Sons.

Mäkinen, K. 2013. IPTV-laitteiden korvaaminen omilla laitteilla. Opinnäytetyö. Jyväskylän ammattikorkeakoulu, tietotekniikan koulutusohjelma.

O'Driscoll, G. 2008. Next Generation IPTV Services and Technologies. Hoboken: John Wiley & Sons.

Papinniemi, S. 2010. Uudet standardit DVB-T2, DVB-C2, DVB-S2, DVB-SH. Opinnäytetyö. Mikkelin ammattikorkeakoulu, tietotekniikan koulutusohjelma.

Patterson, J. 2012. Video Encoding Settings for H.264 Excellence. Viitattu 8.8.2015. http://www.lighterra.com/papers/videoencodingh264/

Poole, I. N.d. What is DVB Digital Video Broadcasting. Viitattu 2.8.2015. http://www.radio-electronics.com/info/broadcast/digital-video-broadcasting/what-is-dvb-tutorial.php

Qiu, D. 2010. On the QoS of IPTV and Its Effects on Home Networks. Viitattu 4.7.2015. http://www.hindawi.com/journals/ijdmb/2010/253495/

Randall, N. 1998. What Constitutes a Fine Resolution? PC Magazine, 8, 17, 217-218.

RFC 1054. 1988. Host Extensions for IP Multicasting. Viitattu 25.6.2015. Deering, S.E. & Cheriton, D.R. https://tools.ietf.org/html/rfc1054

RFC 1631. 1994. The IP Network Address Translator (NAT). Viitattu 10.7.2015. Egevang, K. & Francis, P. https://tools.ietf.org/html/rfc1631

RFC 1700. 1994. Assigned Numbers. Viitattu 18.6.2015. Reynolds, J. Postel, J. https://tools.ietf.org/html/rfc1700

RFC 1918. 1996. Address Allocation for Private Internets. Viitattu 21.6.2015. Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J. & Lear, E. https://tools.ietf.org/html/rfc1918

RFC 2236. 1997. Internet Group Management Protocol, Version 2. Viitattu 25.6.2015. Fenner, W. https://tools.ietf.org/html/rfc2236

RFC 2974. 2000. Session Announcement Protocol. Viitattu 19.8.2015. Handley, M., Perkins, C. & Whelan, E. https://tools.ietf.org/html/rfc2974

RFC 3376. 2002. Internet Group Management Protocol, Version 3. Viitattu 27.6.2015. Cain, B. Deering, S. Kouvelas, I. Fenner, B. Thyagarajan, A. http://tools.ietf.org/html/rfc3376

RFC 3569. 2003. An Overview of Source-Specific Multicast (SSM). Viitattu 26.6.2015. Bhattacharyya, E. http://tools.ietf.org/html/rfc3569

RFC 3973. 2005. PIM – Dense Mode. Viitattu 7.7.2015. Adams, A., Nicholas, J. & Siadak, W. https://tools.ietf.org/html/rfc3973

RFC 4601. 2006. Protocol Independent Multicast – Sparse Mode (PIM-SM): Protocol Specification (Revised). Viitattu 7.7.2015. Fenner, B., Handley, M., Holbrook, H. & Kouvelas, I. https://tools.ietf.org/html/rfc4601

RFC 4607. 2006. Source-Specific Multicast for IP. Viitattu 26.6.2015. Holbrook, H. & Cain, B. https://tools.ietf.org/html/rfc4607

RFC 4608. 2006. Source-Specific Protocol Independent Multicast in 232/8. Viitattu 04.07.2015. https://tools.ietf.org/html/rfc4608

RFC 5015. 2007. Bidirectional Protocol Independent Multicast. Viitattu 11.8.2015. Handley, M. Kouvelas, I. Speakman, T. & Vicisano, L.

RFC 5735. 2010. Special Use IPv4 Addresses. Viitattu 1.7.2015. Cotton, M. & Vegoda, L. https://tools.ietf.org/html/rfc5735

RFC 768. 1980. User Datagram Protocol. Viitattu 15.6.2015. Postel, J. https://tools.ietf.org/html/rfc768

RFC 919. 1984. Broadcasting Internet Datagrams. Viitattu 16.6.2015. Mogul, J. https://tools.ietf.org/html/rfc919

RFC 966. 1985. Host Groups: A Multicast Extension to the Internet Protocol. Viitattu 21.6.2015. Deering, S.E. Cheriton, D.R. https://tools.ietf.org/html/rfc966

RFC 988. 1986. Host Extensions for IP Multicasting. Viitattu 21.6.2015. Deering, S.E. Cheriton, D.R. https://tools.ietf.org/html/rfc988

Savolainen, P. 2006. IPTV-toteutus laajakaistaverkossa. Opinnäytetyö. Lahden ammattikorkeakoulu, tietotekniikan koulutusohjelma.

Semeria, C. & Maufer, T. N.d. Introduction to IP Multicast Routing. Viitattu 18.6.2015. http://www4.ncsu.edu/~rhee/clas/csc495j/ip-multicast-part1.pdf

Simpson, W. 2006. Video Over IP: A Practical Guide to Technology and Applications.

SpiderNet. 2009. SpiderNet-laboratorioympäristön esittely. Viitattu 3.7.2015. http://student.labranet.jamk.fi/SpiderNet/

Suleva, L. 2011. IPTV:n asettamat vaatimukset verkolle ja palvelun toteutus SimuNetissä. Opinnäytetyö. Kymenlaakson ammattikorkeakoulu, tietoverkkotekniikan koulutusohjelma.

Tews, E., Weiner, M. & Wälde, J. 2011. WEWoRC 2011 –konferenssidokumentti: Breaking DVB-CSA. Viitattu 15.8.2015. http://hgpu.org/?p=6038

Weinmann, R-P. & Wirt, K. 2004. Analysis of the DVB Common Scrambling Algorithm. Raportti. Viitattu 25.8.2015. https://www-old.cdc.informatik.tu-darmstadt.de/reports/reports/KP/csa_04.pdf

Wielert, M. 2011. IPTV tuotannon näkökulmasta. Opinnäytetyö. Hämeen ammattikorkeakoulu, mediatekniikan koulutusohjelma.

Wirt, K. 2004. Fault attack on the DVB Common Scrambling Algorithm. Viitattu 25.8.2015. http://eprint.iacr.org/2004/289

Virtual LAN: Applications and Technology. 2004. Micrel-yrityksen laatima white paper. Viitattu 28.7.2015. http://www.micrel.com/_PDF/Ethernet/White%20Paper/vlans%20wp.pdf

Liitteet

Liite 1: Verkkolaitteiden konfiguraatiot

WG4-R1

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname WG4-R1
boot-start-marker
boot-end-marker
logging message-counter syslog
no aaa new-model
memory-size iomem 5
dot11 syslog
ip source-route
ip cef
ip dhcp pool VLAN10
  network 172.16.1.0 255.255.255.0
  default-router 172.16.1.1
ip dhcp pool VLAN20
  network 172.16.2.0 255.255.255.0
  default-router 172.16.2.1
ip dhcp pool VLAN30
  network 172.16.3.0 255.255.255.0
  default-router 172.16.3.1
ip dhcp pool VLAN40
  network 172.16.4.0 255.255.255.0
  default-router 172.16.4.1
no ip domain lookup
ip multicast-routing
no ipv6 cef
multilink bundle-name authenticated
voice-card 0
archive
 log config
  hidekeys
interface GigabitEthernet0/0
 description WG4-R1 to Core-R4
 ip address 130.0.0.2 255.255.255.252
 ip pim sparse-mode
 ip nat outside
 ip virtual-reassembly
 ip igmp version 3
 duplex auto
```

```
 speed auto
interface GigabitEthernet0/1
 description WG4-R1 to WG4-SW1
 no ip address
 duplex auto
 speed auto
interface GigabitEthernet0/1.10
 description VLAN 10 Management
 encapsulation dot1Q 10
 ip address 172.16.1.1 255.255.255.0
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
interface GigabitEthernet0/1.20
 description VLAN 20 Data
 encapsulation dot1Q 20
 ip address 172.16.2.1 255.255.255.0
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
interface GigabitEthernet0/1.30
 description VLAN 30 IPTV
 encapsulation dot1Q 30
 ip address 172.16.3.1 255.255.255.0
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
interface GigabitEthernet0/1.40
 description VLAN 40 VoIP
 encapsulation dot1Q 40
 ip address 172.16.4.1 255.255.255.0
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
interface Serial0/0/1
 no ip address
 shutdown
interface FastEthernet0/1/0
 no ip address
 shutdown
```

```
 duplex auto
 speed auto
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 130.0.0.1
no ip http server
no ip http secure-server
ip pim ssm default
ip nat pool server 138.108.55.2 138.108.55.2 prefix-length 24
ip nat pool other 138.108.55.1 138.108.55.1 prefix-length 24
ip nat inside source list 4 pool server
ip nat inside source list 5 pool other overload
ip nat inside source static 172.16.1.1 138.108.55.1
access-list 4 permit 172.16.1.0 0.0.0.255
access-list 4 deny   any
access-list 5 permit 172.16.0.0 0.0.255.255
access-list 5 deny   any
control-plane
line con 0
line aux 0
line vty 0 4
 login
scheduler allocate 20000 1000
```

WG4-SW1

```
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname WG4-SW1
no aaa new-model
ip subnet-zero
no ip domain-lookup
spanning-tree mode pvst
spanning-tree extend system-id
vlan internal allocation policy ascending
interface GigabitEthernet0/1
 description WG4-SW2 to WG4-R1
 switchport trunk encapsulation dot1q
 switchport mode trunk
interface GigabitEthernet0/2
 description WG4-SW2 to WG4-SW1
 switchport trunk encapsulation dot1q
```

```
 switchport mode trunk
 shutdown
interface GigabitEthernet0/3
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/4
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/5
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/6
 switchport mode dynamic desirable
shutdown
interface GigabitEthernet0/7
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/8
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/9
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/10
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/11
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/12
 switchport mode dynamic desirable
 shutdown
interface Vlan1
 no ip address
 shutdown
ip classless
ip http server
ip http secure-server
control-plane
line con 0
line vty 5 15
```

WG4-SW2

```
version 12.1
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
```

```
hostname WG4-SW2
ip subnet-zero
no ip domain-lookup
ip ssh time-out 120
ip ssh authentication-retries 3
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
interface FastEthernet0/1
 description WG4-SW2 to WG4-SW1
 switchport mode trunk
interface FastEthernet0/2
 shutdown
interface FastEthernet0/3
 shutdown
interface FastEthernet0/4
 shutdown
interface FastEthernet0/5
 shutdown
interface FastEthernet0/6
 shutdown
interface FastEthernet0/7
 shutdown
interface FastEthernet0/8
 shutdown
interface FastEthernet0/9
 shutdown
interface FastEthernet0/10
 shutdown
interface FastEthernet0/11
 shutdown
interface FastEthernet0/12
 shutdown
interface FastEthernet0/13
 shutdown
interface FastEthernet0/14
 shutdown
interface FastEthernet0/15
 shutdown
interface FastEthernet0/16
 shutdown
interface FastEthernet0/17
 shutdown
interface FastEthernet0/18
 shutdown
interface FastEthernet0/19
 shutdown
interface FastEthernet0/20
 shutdown
```

```
interface FastEthernet0/21
 description WG4-SW2 to WG4-IPTVWS
 switchport access vlan 30
 switchport mode access
interface FastEthernet0/22
 shutdown
interface FastEthernet0/23
 shutdown
interface FastEthernet0/24
 shutdown
interface Vlan1
 no ip address
 no ip route-cache
 shutdown
ip http server
line con 0
line vty 5 15
```

WG5-R1

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname WG5-R1
boot-start-marker
boot-end-marker
logging message-counter syslog
no aaa new-model
memory-size iomem 5
dot11 syslog
ip source-route
ip cef
ip dhcp pool Server
   host 192.168.1.100 255.255.255.0
   hardware-address 000c.299d.20ec
ip dhcp pool Other
   network 192.168.1.0 255.255.255.0
   default-router 192.168.1.1
no ip domain lookup
ip multicast-routing
no ip igmp ssm-map query dns
no ipv6 cef
multilink bundle-name authenticated
voice-card 0
archive
 log config
  hidekeys
interface GigabitEthernet0/0
```

```
 description WG5-R1 to Core-R5
 ip address 130.0.0.14 255.255.255.252
 ip pim sparse-mode
 ip nat outside
 ip virtual-reassembly
 duplex auto
 speed auto
interface GigabitEthernet0/1
 description WG5-R1 to WG5-SW1
 no ip address
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
 duplex auto
 speed auto
interface GigabitEthernet0/1.10
 description VLAN 10
 encapsulation dot1Q 10
 ip address 192.168.1.1 255.255.255.0
 ip pim sparse-mode
 ip nat inside
 ip virtual-reassembly
 ip igmp version 3
interface Serial0/0/0
 no ip address
 shutdown
 clock rate 2000000
interface Serial0/0/1
 no ip address
 shutdown
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
ip forward-protocol  nd
ip route 0.0.0.0 0.0.0.0 130.0.0.13
no ip http server
no ip http secure-server
ip pim ssm default
ip nat pool server 109.163.249.2 109.163.249.2 prefix-length 24
ip nat pool other 109.163.249.3 109.163.249.50 prefix-length 24
ip nat inside source list 4 pool server
```

```
ip nat inside source list 5 pool other overload
ip nat inside source static 192.168.1.1 109.163.249.1
access-list 4 permit 192.168.1.0 0.0.0.255
access-list 4 deny   any
access-list 5 permit 192.168.0.0 0.0.255.255
access-list 5 deny   any
control-plane
line con 0
line aux 0
line vty 0 4
 login
scheduler allocate 20000 1000
```

WG5-SW1

```
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname WG5-SW1
no aaa new-model
ip subnet-zero
ip routing
no ip domain-lookup
ip multicast-routing
spanning-tree mode pvst
spanning-tree extend system-id
vlan internal allocation policy ascending
interface GigabitEthernet0/1
 description WG5-SW2 to WG5-R1
 switchport trunk encapsulation dot1q
 switchport mode trunk
interface GigabitEthernet0/2
 description WG5-SW2 to WG5-SW2
 switchport trunk encapsulation dot1q
 switchport mode trunk
 shutdown
interface GigabitEthernet0/3
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/4
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/5
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/6
 description WG5-SW1 to WG5-IPTVServer
```

```
 switchport access vlan 10
 switchport mode access
interface GigabitEthernet0/7
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/8
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/9
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/10
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/11
 switchport mode dynamic desirable
 shutdown
interface GigabitEthernet0/12
 switchport mode dynamic desirable
 shutdown
interface Vlan1
 no ip address
 shutdown
ip classless
ip route 0.0.0.0 0.0.0.0 131.0.0.1
ip http server
ip http secure-server
ip pim ssm default
control-plane
line con 0
line vty 5 15
```

WG5-SW2

```
version 12.1
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
hostname WG5-SW2
ip subnet-zero
no ip domain-lookup
ip ssh time-out 120
ip ssh authentication-retries 3
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
interface FastEthernet0/1
 description WG5-SW2 to WG5-SW1
```

```
 switchport mode trunk
interface FastEthernet0/2
 shutdown
interface FastEthernet0/3
 shutdown
interface FastEthernet0/4
 shutdown
interface FastEthernet0/5
 shutdown
interface FastEthernet0/6
 shutdown
interface FastEthernet0/7
 shutdown
interface FastEthernet0/8
 shutdown
interface FastEthernet0/9
 shutdown
interface FastEthernet0/10
 shutdown
interface FastEthernet0/11
 shutdown
interface FastEthernet0/12
 shutdown
interface FastEthernet0/13
 shutdown
interface FastEthernet0/14
 shutdown
interface FastEthernet0/15
 shutdown
interface FastEthernet0/16
 shutdown
interface FastEthernet0/17
 description WG4-SW2 to IP Camera
 switchport access vlan 10
interface FastEthernet0/18
 shutdown
interface FastEthernet0/19
 shutdown
interface FastEthernet0/20
 shutdown
interface FastEthernet0/21
 shutdown
interface FastEthernet0/22
 shutdown
interface FastEthernet0/23
 shutdown
interface FastEthernet0/24
 shutdown
interface Vlan1
```

```
 no ip address
 no ip route-cache
 shutdown
ip http server
line con 0
line vty 5 15
```

Core-R4

```
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname Core-R4
boot-start-marker
boot-end-marker
no aaa new-model
resource policy
ip subnet-zero
ip cef
no ip domain lookup
key chain eigrpkey
 key 1
  key string cisco
ip multicast-routing
no crypto isakmp ccm
interface FastEthernet0/0
 description Core-R4 to WG4-R1
 ip address 130.0.0.1 255.255.255.252
 ip authentication mode eigrp 40 md5
 ip authentication key-chain eigrp 40 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
 speed auto
interface FastEthernet0/1
 description Core-R4 to Core-R6
 ip address 130.0.0.5 255.255.255.252
 ip authentication mode eigrp 40 md5
 ip authentication key-chain eigrp 40 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
 speed auto
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
```

```
 no fair-queue
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
interface GigabitEthernet2/0
 no ip address
 shutdown
 negotiation auto
router eigrp 40
 redistribute bgp 400
 network 130.0.0.0 0.0.0.255
 neighbor 130.0.0.0 password cisco
 neighbor 130.0.0.0 ttl-security hops 2
 auto-summary
router bgp 400
 no synchronization
 bgp log-neighbor-changes
 redistribute static
 redistribute eigrp 40
 neighbor 130.0.0.6 remote-as 600
 no auto-summary
ip classless
ip route 138.108.55.0 255.255.255.0 130.0.0.2
ip route 172.16.1.0 255.255.255.0 130.0.0.2
ip route 172.16.2.0 255.255.255.0 130.0.0.2
ip route 172.16.3.0 255.255.255.0 130.0.0.2
ip route 172.16.4.0 255.255.255.0 130.0.0.2
no ip http server
no ip http secure-server
ip pim ssm default
control-plane
gatekeeper
 shutdown
line con 0
 stopbits 1
line aux 0
line vty 0 4
```

Core-R5

```
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname Core-R5
boot-start-marker
boot-end-marker
no aaa new-model
resource policy
ip subnet-zero
ip cef
no ip domain lookup
no ip dhcp use vrf connected
ip multicast-routing
key chain eigrpkey
 key 1
  key string cisco
no crypto isakmp ccm
interface FastEthernet0/0
 description Core-R5 to WG5-R1
 ip address 130.0.0.13 255.255.255.252
 ip authentication mode eigrp 50 md5
 ip authentication key-chain eigrp 50 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
 speed auto
interface FastEthernet0/1
 description Core-R5 to Core-R6
 ip address 130.0.0.17 255.255.255.252
 ip authentication mode eigrp 50 md5
 ip authentication key-chain eigrp 50 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
 speed auto
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
 no fair-queue
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
```

```
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
 no dce-terminal-timing-enable
interface GigabitEthernet2/0
 no ip address
 shutdown
 negotiation auto
router eigrp 50
 redistribute bgp 500
 network 130.0.0.0 0.0.0.255
 neighbor 130.0.0.0 password cisco
 neighbor 130.0.0.0 ttl-security hops 2
 auto-summary
router bgp 500
 no synchronization
 bgp log-neighbor-changes
 redistribute static
 redistribute eigrp 50
 neighbor 130.0.0.18 remote-as 600
 no auto-summary
ip classless
ip route 109.163.249.0 255.255.255.0 130.0.0.14
ip route 192.168.1.0 255.255.255.0 130.0.0.14
ip route 192.168.2.0 255.255.255.0 130.0.0.14
ip route 192.168.3.0 255.255.255.0 130.0.0.14
ip route 192.168.4.0 255.255.255.0 130.0.0.14
no ip http server
no ip http secure-server
ip pim ssm default
control-plane
gatekeeper
 shutdown
line con 0
 stopbits 1
line aux 0
line vty 0 4
```

Core-R6

```
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
```

```
no service password-encryption
hostname Core-R6
boot-start-marker
boot-end-marker
no aaa new-model
ip cef
no ip domain lookup
ip multicast-routing
key chain eigrpkey
 key 1
  key string cisco
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
 media-type rj45
 no negotiation auto
interface GigabitEthernet0/2
 no ip address
 shutdown
 duplex auto
 speed auto
 media-type rj45
 no negotiation auto
interface GigabitEthernet0/3
 no ip address
 shutdown
 duplex auto
 speed auto
 media-type rj45
 no negotiation auto
interface FastEthernet1/0
 description Core-R6 to Core-R4
 ip address 130.0.0.6 255.255.255.252
 ip authentication mode eigrp 60 md5
 ip authentication key-chain eigrp 60 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
 speed auto
interface FastEthernet1/1
 description Core-R6 to Core-R5
 ip address 130.0.0.18 255.255.255.252
 ip authentication mode eigrp 60 md5
 ip authentication key-chain eigrp 60 eigrpkey
 ip pim sparse-mode
 ip igmp version 3
 duplex auto
```

```
 speed auto
interface ATM2/0
 no ip address
 shutdown
 no atm ilmi-keepalive
interface ATM4/0
 no ip address
 shutdown
 no atm ilmi-keepalive
router eigrp 60
 redistribute bgp 600
 network 130.0.0.0 0.0.0.255
 neighbor 130.0.0.0 password cisco
 neighbor 130.0.0.0 ttl-security hops 1
 auto-summary
router bgp 600
 no synchronization
 bgp log-neighbor-changes
 redistribute eigrp 60
 redistribute static
 neighbor 130.0.0.5 remote-as 400
neighbor 130.0.0.17 remote-as 500
no auto-summary
no ip http server
no ip http secure-server
ip pim ssm default
control-plane
gatekeeper
 shutdown
line con 0
 stopbits 1
line aux 0
line vty 0 4
```