

Opinnäytetyö (AMK)

Konetekniikka

2023

Elias Khabbal

VIDEOLATENSSI MATERIALUKSEN ETÄOHJAUSASEMASSA

– Mittausjärjestelmän suunnittelu ja toteuttaminen



Opinnäytetyö (AMK) | Tiivistelmä

Turun ammattikorkeakoulu

Konetekniikka

2023 | 32 sivua

Elias Khabbal

Videolatenssi merialuksen etäohjauksessa

Tässä opinnäytetyössä käydään läpi verkon läpi kautta kulkevan videolähetyksen perusrakennetta, sen eri toteutustapoja ja tekniikoita.

Esimerkkinä tällaisten aiheiden sovelluksista olisi laivojen, hinaajien ja lauttojen etäohjauksen ja -valvonnan analysointi.

Työssä myös selostetaan, kuinka yksi mahdollinen toteutustapa videolatenssin mittaussasetelmasta voidaan koota. Työn tavoitteina on kerätä tietoa sekä toimia ponnahduslautana tuleville latenssitesteille tarjoten hyvät raamit jatkokehitykselle. Näin tuotekehitys sekä sen vaatima tietotaito kulkee luotettavasti eteenpäin ja saadaan dokumentoitua alusta pitäen järjestelmän toimintatapa.

Opinnäytetyössä latenssinmittaukseen käytettiin ohjelmistopohjaista, kevyttä videopaketteja muuttamatonta mittausratkaisua. Mittausohjelmistoa varten täytyi hieman muokata videolähetyksen kulkemaa reittiä, jotta videokuvan pakettien data ei muuttuisi matkan varrella.

Testimittauksien lopputuloksena löydettiin, että suurin latenssin tekijä on videosignaalin siltaamiseen verkkojen välillä käytettävä laite. SRT-protokollan lisäämä latenssi oli odotettua, joten myös mittauksen tulokset olivat odotettuja. Videolähetyksen datankulun priorisointi ja muut verkon palvelulaatua optimoivat toiminnot voisivat olla tapoja madaltaa signaalin latenssia ja parantaa etäohjaus-aseman käytettävyyttä.

Asiasanat:

latenssi, kauko-ohjaus, etäseuranta, itseohjautuvat laivat, suoratoisto

Bachelor's Thesis | Abstract

Turku University of Applied Sciences

Mechanical Engineering

2023 | 32 pages

Elias Khabbal

Video latency in remote operation of a marine vessel

This thesis examines the basic structure, different implementation methods, and techniques of video transmission through a network. As an example of the applications of such topics would be analyzing the remote operation and monitoring of ships, tugboats and ferries.

The thesis also explains how one possible implementation of a video latency measurement setup can be assembled. The objectives of the thesis are to gather information and serve as a launching pad for future latency tests, providing a good framework for further development. This ensures that product development and the required expertise move forward reliably, and the system's operating method is well documented from the beginning.

A software-based, lightweight, packet-invariant measurement solution was used for latency measurement in the thesis. The route taken by the video transmission had to be slightly modified for the measurement software to prevent the packet data of the video image from changing along the way.

As a result of the test measurements, it was found that the biggest latency factor is the device used to bridge the video signal between networks. The latency added by the SRT protocol was as expected, hence the resulting outcomes were as expected. Prioritizing the data flow of the video transmission and other functions to optimize the quality of service of the network could be ways to reduce the signal latency and improve the usability of the remote control station.

Keywords:

Latency, remote controlling, remote monitoring, autonomous ships, streaming

Sisältö

Käytetyt lyhenteet tai sanasto	6
1 Johdanto	7
2 Käyttökohteet	9
2.1 Etäoperointikeskus	9
2.2 Autoship	11
2.2.1 Lyhyitä merimatkoja kulkeva rehualus	12
2.2.2 Sisämaissa toimiva katamaraani	13
3 Taustatieto	15
3.1 Kuljetuskerroksen protokollat	15
3.1.1 TCP	16
3.1.2 UDP	17
3.2 Sovelluskerroksen protokollat	18
3.2.1 SRT	18
3.2.2 MPEG-TS	19
3.2.3 RTP	19
3.2.4 RTSP	20
4 Testilaitteiston kasaus ja soveltuvuus selvityksen suorittaminen	22
4.1 Qosium-mittaussovellusten asennus	23
4.2 Ensimmäinen testi	25
5 Viimeistely mittausjärjestelmä	27
6 Vaikuttajien analysointi	29
7 Lopuksi	30
Lähteet	31

Kuvat

Kuva 1. Autoshopin erilaiset digitaaliset ympäristöt. (Technologies, 2021)	11
Kuva 2. Aluksen suunniteltu kulkureitti. (Demo & Use Case, 2021)	13
Kuva 3. Katamaraanin esimerkkireitti (Demo & Use Case, 2021)	14
Kuva 4. OSI-mallin mukaisesti eritelty datan kulku kahden pisteen välillä. (Emmet, D. 2017)	15
Kuva 5. TCP-yhteyden SYN/ACK-käyttely. (Conrad ym. 2012.)	16
Kuva 6. Testilaitteiston kokonaisuutta havainnollistava kuva	22

Käytetyt lyhenteet tai sanasto

Lyhenne	Lyhenteen selitys
Bastion	Erityisesti suojattu palvelin joka suojaa verkkoa hyökkäyksiltä ja mahdollistaa turvallisen pääsyn tiettyihin resursseihin
NTP	Network Time Protocol, protokolla aikatiedon synkronoimiseen laitteiden välillä
OSI	Open Systems Interconnection, malli joka kuvaa tiedonsiirtoprotokollien yhdistelmää
PoC	Proof of Concept, todistus idean osoittaminen toteuttamiskelpoiseksi
Proxy	Välityspalvelin, välittää verkkopyynnön toiseen verkkoon toimien välittäjänä käyttäjän ja palvelimen välillä
PTZ	Pan, Tilt, Zoom, kameran kolme eri ohjaustapaa
VLAN	Virtual Local Area Network, tietoverkon jakaminen eri loogisiin osiin

1 Johdanto

Merenkulun etäohjattujen ja automatisoitujen toimintamuotojen hidas, mutta varma kasvu johtaa moniin tärkeisiin ja relevantteihin kysymyksiin käytön luotettavuudesta. Olemassaolevia kehitysprojekteja löytyy yksityiseltä puolelta suurilta tekijöiltä, sillä tällaisella uudella tekniikalla mahdollistettaisiin matalimmat toimintakulut ja pienemmät miehityskustannukset. Myös julkinen sektori on herännyt tähän; EU rahoittaa laajaa yhteistyötä alan johtajien kesken. Tässä opinnäytetyössä käsitellään yhtä tällaista projektia ja avataan sen tavoitteita samalla reflektoiden etäohjauksen vaatimiin kriteereihin.

Opinnäytetyöni toimeksiantajana toimii Norjan valtion pääomistuksessa olevan konsernin Kongsberg Gruppenin tytäryhtiö Kongsberg Maritime, joka kehittää erilaisia tuotteita ja järjestelmiä merenkulkuun. Näihin kuuluu muun muassa paikannusta, navigointia ja automaatiota aluksiin ja offshore-laitteistoihin sekä järjestelmiä ja tuotteita merenpohjan kartoitukseen ja valvontaan. Turun toimipiste kuuluu Remote Operation Solutions osioon Kongsberg Maritimea. Turussa tehdään pääosin tutkimus- ja kehitystyötä, mutta myös asiakasprojekteja valmistellaan lähetystä ja asennusta vaille valmiiksi pakkauksiksi. Kehitystyö painottuu pääosin myytävän etäohjauspisteen kehittämiseen. Suuri osa ajasta kuluu myös erillisten projektien eteenpäin viemisessä.

Työni tavoitteina on selkeyttää etäohjaukseen vaikuttavan latenssin lähteet ja rakentaa runkoa ja ohjeistusta tulevien samankaltaisten mittausten suorittamiseksi. Aihe on tärkeä, koska latenssi esimerkiksi videokuvassa vaikuttaa suoraan operoijan suorittamien liikkeiden ajoitukseen. Alkuperäisenä hypoteesinä on, että mediasisällön pakkaukset ja muut protokollamuuttajat olisivat suurimpana latenssin kasvun lähteenä.

Työni alussa kappaleessa 2 selitän etäoperoinnin periaatteita merenkulussa sekä esittelen myös muutaman olemassaolevan projektin. Kappaleessa 3 kerron hieman taustatietoa teknisistä tekijöistä. Taustatiedon ja käyttökohteiden jälkeen kappaleessa 4 siirrytään itse mittauslaitteiston kasaamiseen ja itse

testien suorittamiseen. Kappaleessa 5 viimeistellyn mittausjärjestelmän esittelyn ohessa käsitellään myös esiintyneitä ongelmia ja rajoituksia. Näistä keskustellaan lisää myös pohdinnassa kappaleessa 7, jossa työn aikana opitut asiat ovat tiivistettyinä kehityskohteiden ja huomioitavien rajoitusten ohella.

2 Käyttökohteet

2.1 Etäoperointikeskus

Kansainvälisen merenkulkujärjestön, IMO:n määritelmien mukaan etäohjattu alus määritellään tason kaksi tai kolme autonomiseksi alukseksi. Tason 2 laivaa ohjataan etänä toisesta sijainnista, mutta komentokannella on paikalla miehistöä, jotka voivat ottaa aluksen hallintaan milloin tahansa. Tason 3 laiva taas on täysin etäohjattu ilman paikallista miehistöä. (IMO, 2021) Sillä nämä teknologiat ovat alati kehittyviä, etenkin etäohjauspisteen vaatimukset lainsäädäntötasolla ovat osittain puutteellisia. Monet termit, kuten "miehistö" ja "vastuuhenkilö" koskien tason 2 ja 3 autonomisia laivoja vaativat kansainvälisesti hyväksytyjä määritelmiä. Näistä puutteista huolimatta monet yritykset kehittävät omia ratkaisujaan autonomisista aluksista pystyäkseen toimittamaan markkinoille valmiita tuotteita lainsäädännön ja sosiopoliittisen ympäristön ollessa valmiita tälle muutokselle. (Saarni Jouni, Nordberg-Davies Sini and Saurama Antti, 2018)

Kongsbergin kehittämä toteutus etäohjatusista työasemista kulkee nimellä Remote Operation Center (ROC). Joustava etäohjaus ja -hallintajärjestelmä, jota pystytään ottamaan käyttöön isoilla konttialuksilla sekä pienemmillä aluksilla kuten hinaajilla tai sisävesikäyttöön tarkoitetuilla lautoilla. Tarkoituksena on poistaa tarve kalliille ja yhteen alukseen lukitulle ja yhteen funktioon omistetulle kuljettajahenkilöstölle. Tällaiset paljon aikaa ja suurta henkilöinvestointia vievä miehistykset olisi mahdollista korvata esimerkiksi rannikolla sijaitsevalla ohjauskeskuksessa työskentelevällä ohjaajamiehistöllä. Näin saataisiin parannettua työntekijöiden työolosuhteita, -aikoja sekä vähentämään ns. "luppoaikaa" ohjauksessa. Kun pääasiallinen ohjaava henkilö ei enää sijaitse itse aluksella, tekijöiden ei tarvitse jäädä kiinni aluksille mahdollisesti huonoissa olosuhteissa matkojen ajaksi. Usein satamista ja laitureilta irtautumisessa on paljon odottelua ja tyhjää aikaa, joka on sekä henkilöstölle, että yritykselle tehokkuutta kuluttavaa toimintaa pidemmän päälle.

Tämän odotusajan aikana etänä aluksia ohjaava henkilöstö pystyisi ottamaan ohjaukseen tai tarkasteluun toisia aluksia, jotka vaativat aktiivista kontrollia. Myös henkilöstön vaihto kesken aluksen kulun olisi mahdollista, helpottaen työntekijöiden työn ja vapaa-ajan tasapainotusta. Haasteelliset ja vaaralliset työympäristöt voidaan vaihtaa turvalliseen ja staattiseen operointipisteeseen, jossa ergonomiasta ja työntekijöiden hyvinvoinnista voidaan huolehtia paremmin. (*Autonomous Shipping*, 2022)

Älykkäällä anturoinnilla ja valvontakameroilla voidaan replikoida ohjaussillalla saatavissa oleva informaatio ohjaukseen riittävällä tasolla. Aluksen ympäristöä tarkkaillaan ja mitataan eri keinoin, kuten tutkilla, kameroilla sekä navigointilaitteilla. Kaiken sisällytetyn teknologian tarkoituksena mahdollistaa aluksen tarvittavan sisäisen ja ulkoisen tietoisuuden välittäminen ohjaajalle. Näin saadaan tietoa aluksen omista hälytyksistä ja järjestelmistä sekä informaatiota toisten alusten kulusta ja sääolosuhteista. Myös itse aluksen ohjausjärjestelmä pystyy käyttämään kaikkea tätä dataa omissa tekoälyyn perustuvissa ominaisuuksissaan. Esimerkiksi älykäs automaattinen esteiden väistöjärjestelmä vapauttaa ohjaajan resursseja ja mahdollistaa suuremman automaation tason aluksissa. Yksittäisen aluksen vähentynyt manuaalisen ohjauksen tarve tarkoittaa, että samanaikaisesti yksi henkilö kykenee ohjaamaan useampaa määrää aluksia kerralla.

Tämän työn tarkoituksena on kehittää mittaustyökaluja ja dataintegraatiota etäohjaajalle avustavalle tietoa. Valvontakameroiden videokuvan viiveen tietäminen on tärkeää ohjaajalle tai valvojalle, jotta hän kykenee ottamaan huomioon tämän ajallisen poikkeavuuden ohjauksissa. Jos videolähetyksen viive vaihtelee esimerkiksi 0,5 ja 1,5 sekunnin välillä, tarvittavat ohjausliikkeet voivat olla suurestikin erilaiset.

Tuotetut tulokset ovat myös tuotteen käyttäjille, että ostajille lisäinformaatiota toiminnasta ja luotettavuudesta. Turvallisuuden ja luotettavuuden varmistaminen on ehdotonta myös lainsäädännön kannalta. Meri- ja rahtiteollisuudessa on tarkat säädökset lupien saamisen suhteen ja etenkin

uusien teknologioiden ympärillä pyörii paljon tarkastuslautakuntia ja toiminnan verifiointia.

2.2 Autoship

Autoship on suurin EU tasolla rahoitettu yhteistyöprojekti autonomisten laivojen kehitystyötä varten. Sen tarkoituksena on kehittää EU:n autonomisten laivojen teknologian kilpailukykyisyyttä ja nopeuttaa siirtymistä laajempaa laivojen autonomiaa kohti. Projektin aikana kehitetyt toimitukset toimivat tulevaisuudessa samankaltaisten projektien ja investointien optimoinnissa yhteisten standardien kehityksen ohella. Arviona on mahdollista kaupalliset toteutukset teknologioista vuoden 2023 loppuun mennessä.



Kuva 1. Autoshipin erilaiset digitaaliset ympäristöt. (Technologies, 2021)

Autoshipin lainsäädöllisen ja teollisen yhteistyön kehityksen lisäksi painotetaan simulaation sekä muiden digitaalisten teknologioiden edistämistä. Näin saadaan kerättyä arvokasta käyttäjäkokemusta ilman reaali maailman testauksia. Projektissa myös rakennetaan ja operoidaan kahta erilaista autonomista alusta, toinen lyhyen matkan merenkulkuun ja toinen sisämaan vesiväyliä kulkemaan. Näiden alusten fokuksena on tavaroiden kuljetus, keventäen riippuvuutta

maantiekuljetuksille ja parantaen rahdin kuljetusmahdollisuuksia jatkossa. Tämänkaltaiset uudet laivat auttavat teollisuutta optimoimaan investointejaan kehittämällä kilpailukykyisyyttään ja uudistamalla laivastoaan. Suuremmalla määrällä monitorointia, datan käyttöä ja kommunikointia laajemmalla verkko edetään kohti tehokkaampaa toimintaa joka saralla merilogistiikka-alalla. Älykkäiden laivojen yhteentoimivuus ja IoT teknologia parantaa jokaisen operaation turvallisuutta ja nopeutta.

Kaiken tämän saavuttamiseksi kehitykseen tuodaan mukaan merenkulkuviranomaiset, logistiikan arvoketju ja päättäjät, jotka yhdessä päättävät projektiin sopivat valmistuskriteerit ja suunnitteluparametrit. Koska tämä projekti on EU:n rahoittama, myös projektin viestintä tulee arvioida, jotta saadaan "laaja yhteiskunnallinen hyväksyntä". Työskentelyn aikana tehdään analyysejä lainsäädöllisestä, yhteiskunnallisesta, taloudellisesta ja (kyber)turvallisuustekijöistä projektiin liittyen ja näiden perusteella projektin valmistuttua kerätään kokoon tiivistelmä kehitysehdotuksista, joka toimitetaan Kansainväliselle merenkulkujärjestölle alan kehitystä varten. (*Autonomous Shipping*, 2022)

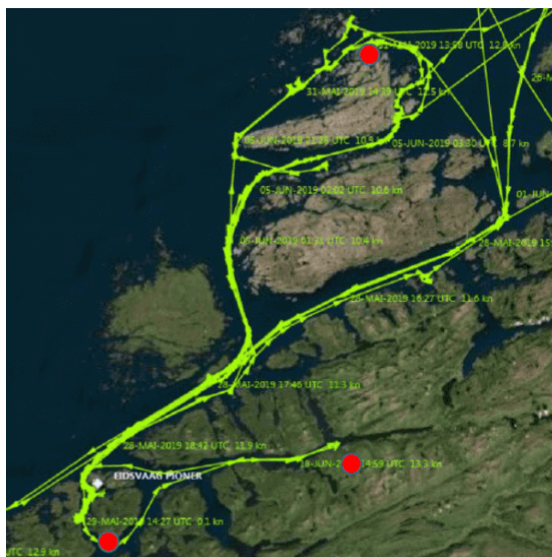
2.2.1 Lyhyitä merimatkoja kulkeva rehualus



Kuva 4. Autoship- projektin kalarehua kuljettava alus

Tämä lyhyen matkan merenkulkuun tarkoitettu alus kuljettaa kalojen ravintoa Norjan Kristiansandista Skrettingin ja Cargillin tehtailta Tanskaan Hirtshaliin. Kokonaismatka on noin 140 kilometriä, josta suurin osa tapahtuu avoimella merellä. Kalanrehun toimittajana aluksen täytyy toimia kuljetuksessa, telakoitua

ja irtautua satamista, siirtää erilaista varastoitavaa tavaraa, toimia yhdessä kalakasvattamojen kanssa sekä toimia tiettyssä sääikkunassa. Tämä ei kuitenkaan ole täysin autonominen, mutta pyrkii toimimaan korkeatasoisesti autonomisten ohjausten kannalta. Aluksen kansi on myös ajoittain täysin miehittämätön. Tämän vähäisen huomion vaativuuden myötä navigoijalla on kyky seurata ja monitoroida useita aluksia kerralla.



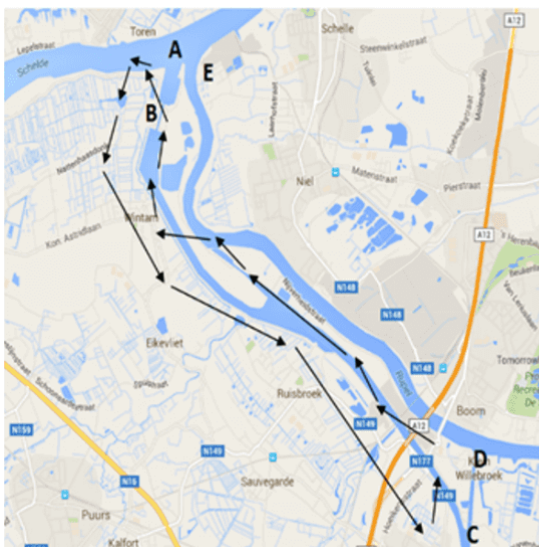
Kuva 2. Aluksen suunniteltu kulkureitti. (Demo & Use Case, 2021) Kuva 5. Suunniteltu aluksen kulkureitti

Aluksen kommunikaatio ja ohjausyhteydet toteutetaan lyhyen matkan kommunikoinnilla lähellä rannikkoa. Muulloin kaikki yhteydet kulkevat satelliittiyhteyksien ylitse. Huomioitavaa on satelliittiyhteyksien pienempi kaistanleveys ja korkeampi latenssi verrattuna esimerkiksi LTE, CDMA ja WCDMA tekniikoiden tarjoamiin palveluihin. Näiden yhteyksien kautta rannikolta käsin suoritetaan reitinsuunnittelua, etäohjattavia toimintoja, monitorointia ja tukea yllättävissä tilanteissa. (Demo & Use Case, 2021)

2.2.2 Sisämaissa toimiva katamaraani

Tämän projektin alus keskittyy yhden katamaraanin operoimiseen Antwerpenissä Belgiassa. Sen tehtävänä on kulkea autonomisesti ja

täysaikaisesti miehittämättömänä kuljettaen kauppatavaraa lavoilla, irtotavarana tai konteissa. Katamaraani toimii jatkuvana operaationa liikkuen, suorittamalla satamatoimet ja navigoimalla tietyllä reitillä autonomisesti. Toisin kuin Short Sea Shipping-aluksella, kanavia ja jokia pitkin kulkiessa voidaan käyttää puhelinyhteyksiä sekä vielä lyhyempien matkojen kommunikointivälineitä. Sillä tämä katamaraani on autonominen, yhteyksiä käytetään lähinnä logistiikan suunnitteluun, monitorointiin ja poikkeustilanteiden hoitamiseen. Suurin osa navigoinnin prosessoinnista suoritetaan paikan päällä aluksella, joten yhteydet eivät rasitu läheskään yhtä paljon, vaikka ne olisivatkin käytössä, vaikkapa jonkin läheisen tapahtuman ansiosta.



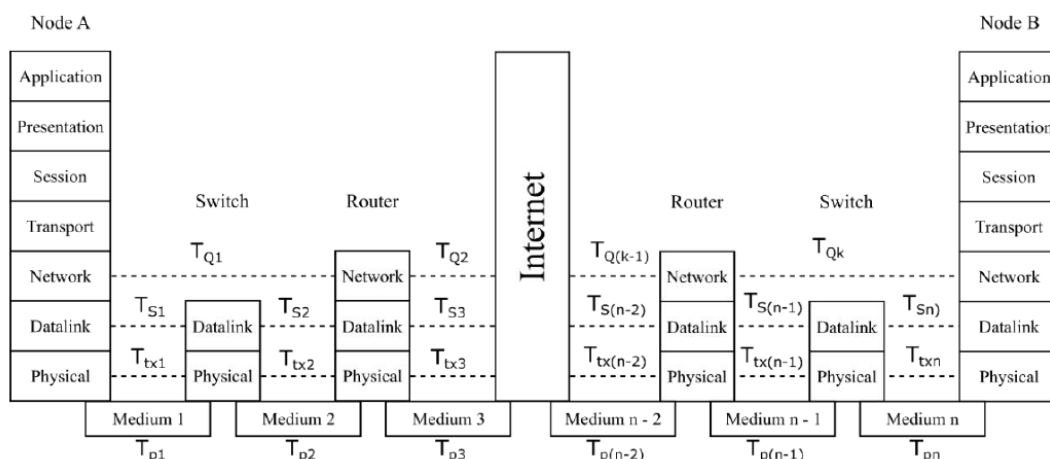
Kuva 3. Katamaraanin esimerkkireitti (Demo & Use Case, 2021)

Kuten yllä olevasta reittikuvasta näkee, katamaraanin kulkema reitti on aivan asutuksen ja suurien teiden lähellä. Tämä toimii kahtena positiivisena vaikuttajana. Katamaraanin kautta kuljetettava rahti on suoraan pois teiden kautta suoritetusta kuljetuksesta, vähentäen liikennettä ja ruuhkaa maanteillä. Toisekseen tämä urbaani sijainti mahdollistaa hyvät langattomat tietoliikenneyhteydet verrattuna edellisessä luvussa mainittuun avomerellä kulkevaan alukseen. (Demo & Use Case, 2021)

3 Taustatieto

Tässä kappaleessa käsitellään vaadittavaa perustietoa verkkotopologiasta ja erinäisistä verkon osissa käytetyistä protokollista. Eri tarkoitukseen kulkevaa dataa paketoidaan mahdollisimman onnistuneen kuljetuksen varmistamiseksi sekä mediamuodon käyttötapojen mukaan erilaisten protokollien muotoihin. Työn mittauksia varten rakennettua datareittiä luodessa näiden yksityiskohtien ja erojen tunnistaminen on tärkeässä osassa.

OSI mallin mukaisesti verkot ovat jaettu 7 osaan, ylemmästä kerroksesta alempaan lueteltuna; sovellus-, esitystapa-, istunto-, kuljetus-, verkko-, siirtoyhteys-, ja fyysinen kerros. Kuva 4 visualisoi verkon eri tasot pisteestä A pisteeseen B internetin välityksellä. Näistä kerroksista tässä työssä keskitytään kuljetus- ja sovelluskerrokseen.



Kuva 4. OSI-mallin mukaisesti eritelty datan kulku kahden pisteen välillä. (Emmet, D. (2017))

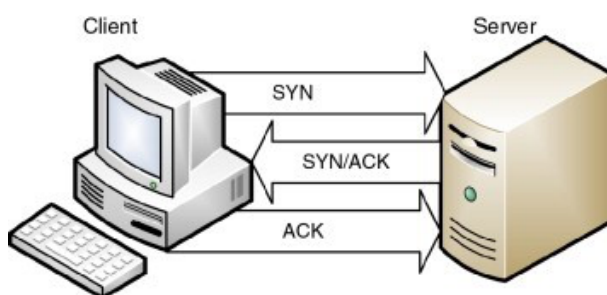
3.1 Kuljetuskerroksen protokollat

Verkossa kuljetuskerroksen päärooli on toimia viestinvälittäjänä suoraan eri sovelluksille saman verkon sisäisesti, suorittaa virheiden tarkistusta sekä segmentoida datablokit. Kuljetuskerros luo sovelluskerrokselle loogisen

kommunikointiväylän yleensä käyttäen joko TCP- tai UDP-protokollaa. Näin sovelluksilla on pisteestä pisteeseen toimiva väylä, jossa kuljetuskerros hoitaa, luo ja sulkee yhteyksiä. Verkkokerrokselta tullessa paketeista poistetaan osoite- ja muita tunnisteita ja lähetetään eteenpäin kohti sovelluskerrosta. Ja toiseen suuntaan kulkevaan liikenteeseen lisätään tarvittavat tunnisteet verkkokerroksen liikennettä varten. (Noergaard, 2010)

3.1.1 TCP

TCP on yksi eniten käytetyimmistä protokollista sen luotettavuuden ansiosta. Sen luotettavuus johtaa siinä käytetystä kolmisuuntaisesta kättelystä, jolla varmistetaan toimiva kommunikaatio laitteiden välillä.



Kuva 5. TCP-yhteyden SYN/ACK-kättely. (Conrad Eric, Misenar Seth and Feldman Joshua, 2012)

Kuva 5 esittää koneen ja palvelimen välistä TCP yhteyden kättelyä. Kättely koostuu SYN ja ACK paketeista. Kättelyn aloittava kone valitsee sekvenssinumeronsa ja lähettää sen SYN paketissa palvelimelle. Tämän saatuaan palvelin valitsee vastaavasti oman sekvenssinumeronsa SYN/ACK pakettiin. Palvelimen ja koneen välinen pakettien saapumisen varmistus tapahtuu kasvattamalla toisen sekvenssinumeroa, luoden kuittausnumeron (ACK-osuus). Tällä sekvenssi- ja kuittausnumeron käytöllä molemmat osapuolet voivat tunnistaa puuttuvia tai väärässä järjestyksessä liikkuvia sekvenssejä.(Conrad Eric, Misenar Seth and Feldman Joshua, 2012)

Jos itse yhteyden viiveellä ei ole merkitystä, vaan pakettien varma saapuminen ja pakettien oikea järjestys on prioriteettina, TCP on oiva valinta. Sillä myös varmistetaan palvelun toimivuus tilanteissa, jossa internetkaistanleveys on rajattu ja halutaan silti varmistaa kriittisen informaation saapuminen. Esimerkiksi NASA käyttää SCPS-TP protokollaa, joka perustuu TCP:n lisäksi ominaisuuksien säätöä kuten pakettien uudelleenlähetystä, asymmetrian toleranssia ja älykästä kaistan ruuhkautumisen välttämistä. Ilmankin näitä erityisolosuhteita varten rakennettuja lisäyksiä TCP toimii luotettavana palveluna, jota käytetään mm. tiedonsiirrossa ftp protokollalla, internetin selaamiseen sekä sähköpostien lähetykseen. (Tran *et al.*, 2005)

Työssäni TCP on käytössä etenkin latenssia mittaavan ohjelmiston omassa mittauspisteiden välisissä kommunikoinneissa. Siinä välittyvä data on pääosin pelkkiä pakettitietoja ja konfiguraatioihin vaadittavaa tekstiä, aikatietoja ja osoitteita, joten liikenteen kuorman koko on erittäin pieni. Tämän takia TCP:n varmuus ja luotettavuus toimii tarkoitukseen hyvin, sillä tällaisten kevyiden datapakettien virheenkorjaaminen ja mahdollinen uudelleenlähettäminen ei vaadi suurta kaistaa verkosta.

3.1.2 UDP

User Datagram Protocol on toinen erittäin suosittu kuljetusprotokolla, erityisesti käytettynä livelähetyksissä ja verkkopelaamisessa pienen viiveen myötä. Toisin kuin TCP, UDP ei suorita vastaanottajalta tulevia pakettien saapumisvarmistuksia. Näiden sijasta UDP:lla jatketaan pakettien lähetystä kuin kaikki olisivat matkanneet perille normaalisti. Tämän takia protokollaa kutsutaan "best effort" palveluksi, eli jos informaatiota tippuu pois matkan varrella, toimintaa jatketaan sillä tiedolla mitä saatavilla on. Tämä tietenkin johtaa laadun heikentymiseen, sillä ei voida varmistaa kokonaisten kehysten saapuminen jokaisessa tilanteessa. Toisaalta tällä saadaan mahdollistettua erittäin paljon pienempi viive livelähetyksiin kuin TCP:tä käyttäen. Aika- tai ajoituskriittiset palvelut kuten musiikkilähetyksen synkronointi festivaalivaloihin tai

verkkopelaamisessa eri pelaajien kesken jaetut interaktiiviset maailmat vaativat UDP:n mahdollistamaa matalaa viivettä. (Fortinet, 2022)

UDP on käytössä työni videoputken päätelaitteisiin siirtyessä, sillä oletuksena on, että tämä verkon osuus on melko häiriötön, sillä laitteet sijaitsevat fyysisesti samassa verkossa. Tämä verrattuna verkon langattomaan osuuteen, jossa data muutetaan SRT protokollaan ja siirretään meren yli alukselta ohjauspisteeseen. Tälle matkalle UDP ei olisi sopiva, sillä sen "best effort" toimintamalli johtaisi suuriin pakettien tippumiseen ja huonoon käyttäjäkokemukseen. Ohjauspisteessä aluksen kameroiden videokuvat tiputtaisivat ajoittain yhteyden monitoreille kiinnitettyihin koneisiin, ja ohjaamiseen vaadittava kamerakuvalaatu kärsisi usein verkon heikennyttyä hiemankin.

3.2 Sovelluserroksen protokollat

Sovelluserroksen tehtäväksi jää informaation kuljettaminen lähellä käyttäjiä. Tämä kerros vastaanottaa pyyntöjä ja dataa käyttäjiltä ja siirtää ne muille kerroksille kohti datan vastaanottajaa. Kun käyttäjälle lähetetään dataa, se ohjataan sovelluserrokselle, joka tuo sen käyttäjän sovellukselle esitettäväksi. Itse käyttäjän sovellukset eivät ole esitettynä tässä kerroksessa, vaan kerros toimii vain sovellusten kommunikoinnin mahdollistajana. (Emmet, 2017)

3.2.1 SRT

Secure Reliable Video Transport on Haivisionin kehittämä avoimen lähdekoodin protokolla, jota on kehitetty erityisesti tuomaan UDP:n matalan latenssin suorituskyvyn yhteen verkoille, joilla on suuriakin häviöitä. SRT varmistaa pakettikadenssin olevan sama dekooderin päässä kuin verkkoon sisään tulevassa videosignaaliin, yksinkertaistaen dekooodausprosessin. SRT tarjoaa myös muita ominaisuuksia, kuten sisäänrakennetun AES salauksen linkkitasolla ja mahdollistaa päätelaitteiden toimivan sekä "kuuntelijana" että lähettäjänä, helpottaen verkonohjausta palomuuritasolla. (Nikols, 2022)

Mittauslaitteistossa videolähetyksen kuljetusta varten valittiin SRT protokolla juuri sen luotettavuudesta häviöllisissä verkoissa. Näin voidaan huonoissakin ympäristöissä varmistaa ohjausta varten videoyhteyksien kriittinen toimivuus. Mittauslaitteiston mittaamissa tilanteissa verkko on ajoittain erittäin huonolaatuista ja sisältää suurella todennäköisyydellä pakettien häviämistä. Tätä protokollaa käyttäen saadaan varmistettua videon siirtyminen, vaikka se johtaisikin korkeampaan videolatenssiin. Suurempaan viiveeseen joka on stabiili, on ohjaajan helpompi mukautua kuin ajoittain satunnaisesti katoavaan tai hyppivään videokuvaan.

3.2.2 MPEG-TS

(MPEG-TS on oikeastaan säiliömuoto, eikä itse sovellusprotokolla, mutta se on sisällytetty tässä sillä latenssimittauksissa yksi käytettävistä laitteista ilmoittaa protokollaksi TS-over SRT) TS on standardisoitu mediasäiliö, joka on suunniteltu digitaalista videolähetystä varten. Sillä siinä käytetään pieniä yksittäisiä paketteja, TS minimoi korruption ja häviöiden vaikutuksia ja on kestävämpi häviöllisillä verkoilla. (Trattnig, 2022)Formaatin sisältämä etukäteinen virhekorjaus (Forward error correction) vähentää tarvetta uudelleenlähetykselle. Vastaanottaessa signaalia, FEC mahdollistaa vastaanottajan tarkistavan ja korjaavan virheelliset osiot signaalista ilman datan uudelleenlähetyksiä.(Al-Mualla, Canagarajah and Bull, 2002)

MPEG-TS:ää käytetään SRT:n ohella datakuljetuksessa meren yli kulkevaa häviöllistä yhteyttä simuloivassa verkossa juuri TS:n sisältämien kuljetusvirheitä minimoivien ominaisuuksien vuoksi.

3.2.3 RTP

Real-time Transport Protocol koteloi video- tai audiodataa RTP paketeiksi ja nämä paketit taas sisällytetään UDP segmenteiksi. Eli RTP on sovellustason protokolla, joka on rakennettu UDP:n päälle toimivaksi, sekä multicast tai

unicast palveluja varten. RTP sisältää ominaisuuksia kuten sekvenssinumerointi ja pakettien aikaleimaus, joten sitä voidaan ajatella osittain myös kuljetuskerroksen alakerrokseksi. Kuitenkin RTP:n käyttö täytyy koodata sisälle jokaiseen sitä käyttävään sovellukseen erikseen, joten se on käytännössä osa sovelluskerrosta. Vaikka RTP käytetäänkin usein erilaisissa mediasuoratoisto- sekä videokonferenssipalveluissa, se ei sisällä mitään mekanismeja datan saapumiselle ajallaan tai muita palvelujen laadun varmistusta varten. Se ei myöskään varmista pakettien saapumista alkuunkaan tai estä pakettien saapumista väärässä järjestyksessä. (*Introduction to the Real-time Transport Protocol (RTP)*, 2022) Tämä voi toisaalta myös toimia näistä puutteista huolimatta. Sovellukset voivat silti hyväksyä puutteellista tai virheellistä dataa. Esimerkiksi telekonferensseissa ääni- ja videopalveluita ei voida viivästyttää, jotta pakettien uudelleenlähettäminen kerkeäisi mukaan. (Koistinen, 1999)

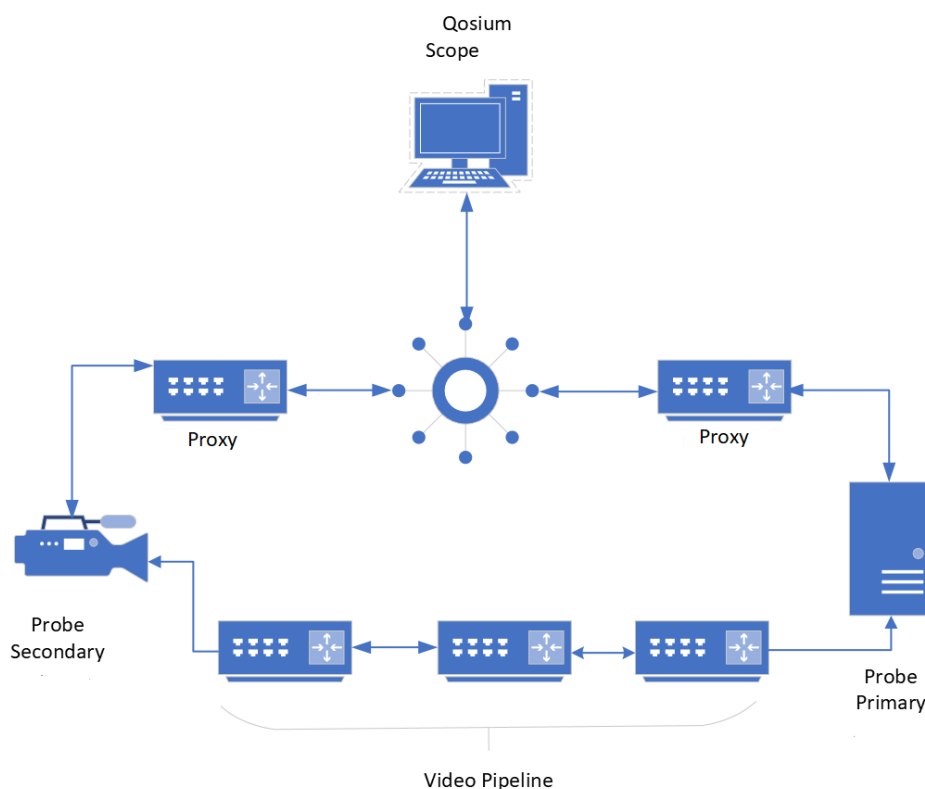
3.2.4 RTSP

Real Time Streaming Protocol toimii eräänlaisena kaukosäätimenä joko yhdelle tai useammalle audio- tai videolähetyksen lähteelle. Se ei siis itse kuljeta lähetyksen toistamiseen tarvittavia paketteja, vaan RTSP yhteyden luomisessa lähettää RTSP pyynnön esimerkiksi valvontakameralle, joka käynnistää kommunikaatiojärjestelmän perustamisen kameran ja ohjauslaitteen välille. Tämän jälkeen itse medialähetys toimitetaan RTP:n välityksellä. Ensimmäisen kommunikointipyynnön mukana komennon "Options" myötä ohjaavalle laitteelle välitetään käytettävissä olevat komennot. Käyttäjä voi näiden komentojen avulla mm. toistaa, pysäyttää, tarkastella lähetyksen tietoja tai sulkea yhteys kokonaan. Luettelo näistä komennoista löytyy liitteenä. Itse RTSP ei tue lähetyksen salausta tai uudelleenlähetystä, sillä nämä rajoitukset ovat itse käytettävän protokollan RTP luomia. Osittain näiden puutteiden ja muiden kilpailevien protokollien yleistymisen myötä RTSP on väistynyt yleisestä käytöstä. Tästä huolimatta edelleen valvontakameroissa ja monissa konferenssivideolaitteissa RTSP on säilynyt ensisijaisena protokollana.

Myös työssä käytettävässä PTZ-kamerassa RTSP on ainoa käytettävä protokolla, jota kamera tukee sisäänrakennettuna. Se ei kuitenkaan lopulta toiminut yhteen lopun mittauslaitteiston kanssa, joten RTSP:stä siirryttiin toiseen toteutukseen.

4 Testilaitteiston kasaus ja soveltuvuus selvityksen suorittaminen

Testauslaitteisto koostuu pääosin viidestä eri tekijästä, videosignaalin ulostulo (PTZ kamera), videosignaalia vastaanottava mediapalvelin, näiden kahden väliin rakennettu verkko, mittausohjelmisto, joka pyörii sekä kameralla että palvelimella sekä kone, jolla ohjataan mittareita näissä edellä mainituissa kahdessa päätteessä ja annetaan mittaukseen tarvittavat parametrit. Suunnitellusta kokonaisuudesta löytyy esimerkkikuva alla.



Kuva 6. Testilaitteiston kokonaisuutta havainnollistava kuva

Testien välillä vaihtuvana muuttujana toimii videosignaalia välittävän verkon arkkitehtuuri. Jokainen kytkin, reititin tai muuten siirrettävää data muuttava tai käytävä laite lisää vaihtelevan verran viivettä datan kulkuun päätelaitteiden

välillä. Esimerkiksi reitittimet vastaanottavat vaikkapa kameran UDP videon datapaketteja ja lukevat niiden sisältämästä otsikosta vastaanottajan osoitteen. Tämän luettuaan, reititin lähettää paketit eteenpäin oikeita portteja pitkin vastaanottajan laitteeseen. Vaikka reitittimien sisäinen viive on erittäin pieni, esimerkiksi Dellin valmistaman erään mallin viive on alle mikrosekunti (*N5860-48SC Switch Datasheet*, no date), reitittimien käyttö verkkojen ylityksessä sisältää monia viiveen lähteitä. Suurimmat näistä ovat: pakettien fyysinen siirto toiseen osoitteeseen, turvallisuustoimet kuten antivirus-ohjelmistot ja pakettien muuttaminen digitaalisesta formaatista käytettävään siirtotapaan (valokuidun tapauksessa muutos biteistä ja tavuista valoksi) ja toisinpäin. (*Latency in Ethernet Switches*, 2016)

Ensimmäisissä soveltuvuuseelvityksen testauksissa laitteet kytkettiin toisiinsa kiinni yhteen paikalliseen verkkoon reitittimen välityksellä. Näin tavoiteltiin testien toimivuuden varmistusta ennen laajempaan testiympäristöön siirtymistä.

4.1.1 Qosium mittaussovellusten asennus

Erilaisten verkkojen laatumittauksia varten kehitetty Qosium ohjelmisto toimii passiivisena mittarina yhdessä tai useammassa pisteessä mitattavan verkkoon kytketyissä laitteissa. Ohjelmiston passiivisuus tulee sen minimaalisesta määrästä lähettämästä datasta mittauksen suorittamiseksi. Itse mittaukset suoritetaan käyttäen olemassa olevaa verkossa välitettävää datasiinaalia, jota Qosiumin yksittäiset mittauspisteet eli "Probet". Nämä Probet lähettävät vain pienen määrän kontekstietoa keskenään verkon rasiuksen minimoimiseksi. Tämä kontekstietä on mm. datapakettien aikaleimoja ja tunnisteita, jotta kaksi Probea pystyvät varmistumaan datan olevan samaa molemmissa päissä. Tämän avulla mittauksella voidaan reaaliaikaisesti tunnistaa verkossa olevia häiriöitä. Lisäksi Qosiumin avulla pystytään testata ja varmistaa verkon riittävä suorituskky käytettäviä sovelluksia ja palveluita varten. Proben vastakappaleena toimii Qosium Scope, joka on datan analysoija, joka ottaa yhteyden Probeen ja visualisoi käyttäjille halutun informaation. Sen avulla voidaan myös tallentaa ja viedä kerätyt mittaukset esimerkiksi Exceliin

jatkokäsittelyä varten. Scope asennetaan ohjaavalle tietokoneelle, joka sijaitsee samassa verkossa mitattavien kohteiden kanssa. Sekä Scope että Probet tukevat Windows-, MacOS/OSX sekä Linux alustoja samoilla ominaisuuksilla, riippuen asiakkaan käyttötarkoituksista. (UKK, 2022)

Mediapalvelin täytyi valmistella asentamalla mittausohjelmiston Qosium Probe -osa laitteelle pyörimään. Koska palvelimen käyttöjärjestelmä perustuu erääseen Linux- pohjaiseen järjestelmään, siihen sai melko kivuttomasti pääsyn SSH yhteydellä ja ohjelmiston asentaminen Kaitotekin tarjoamien ohjeiden avulla onnistui ongelmitta. Probe päätettiin asentaa laitteelle palveluna, eikä ns. "perinteisenä" ohjelmana, sillä näin sitä ei tarvitse joka kerta käynnistää erikseen, kun mittauksia halutaan suorittaa.

Kameralle Proben asentaminen toimi vielä helpommin, sillä kameran valmistaja Axis on rakentanut käyttöliittymään suoran tavan asentaa ja pyörittää kolmannen osapuolen sovelluksia. Kameroihin löytyy eri yhteistyökumppanien luomia ja Axiksen hyväksymiä luomuksia valmiista "sovelluskaupasta". Proben asennuspaketti saatiin suoraan Kaitotekiltä, sillä nämä asennuspaketit täytyy räätälöidä jokaiselle kameralle erikseen niiden käyttöjärjestelmän alustan perusteella. Sekä kameran, että mediapalvelimen Probe asennettiin käyttäen oletusasetuksia, sillä esimerkiksi porttien muutos ei ollut tarpeellista omiin tarkoituksiimme.

Qosium testaukset vaativat kahden mittaavan Proben lisäksi itse Probeja ohjaavan laitteen. Alussa käytimme kannettavaa tietokonetta asennuksen ja testailun helppouden vuoksi. Myöhemmin siirryttiin käyttämään Ubuntu-virtuaalikonetta, joka pyörii VMware vSphere ympäristössä. Virtuaalikonetta käyttäen voidaan varmistaa pidempien mittausten luotettava jatkuvuus ja mittauksien suorittaminen miltä vain virtuaaliympäristöön pääsevältä päätelaitteelta. Ohjausohjelmisto Qosium Scope ottaa yhteyden ensimmäiseen Probeen ja sen jälkeen sekundääriseen Probeen. Scope siirtää itse mittausasetukset ensin pääProbelle, joka taas lähettää halutut parametrit eteenpäin sekundäärille. Näin sekundäärillä ei tarvita erikseen suurta laskennallista tehoa, vähentäen Proben vaikutusta kameran ja mediapalvelimen

normaaliin käyttöön sekä varmistaa mittauksen mahdollisimman todenmukaisuuden. Tämä Scopea pyörittävä ja mittauksia ohjaava tietokone sekä fyysisessä että virtuaalikoneen muodossa toimi samaan aikaan myös NTP palvelimena. Koneen NTP palvelu konfiguroitiin Network Time Protocol Daemon (NTPD) palvelulla. Tämä palvelu taas synkronoi tietokoneen kelloa ulkoisen NTP palvelimen kanssa, tässä tapauksessa NTP Pool Projectin Suomen palvelimia käyttäen

4.1.2 Ensimmäinen testi

Soveltuvuusselvityksessä mediapalvelin konfiguroitiin vastaanottamaan videosuoratoistoa RTSP muodossa (oletusmuoto PTZ kameralle), jolloin videoesityksen pystyttämiseen vaadittiin vain kameran IP-osoite ja porttinumero. Video laitettiin pyörimään läheiselle monitorille, jotta voitiin varmistaa laitteiston toimivuus ja huomata helposti, jos yhteydet ovat katkenneet syystä tai toisesta. Mittaukset suoritettiin lähettämällä 1920 x 1080p resoluutiollista ja 60hz taajuudellista videota kameralta ilman todentamista ("katsojalta" vaadittavaa salasanaa ja käyttäjätunnusta). Oikeassa reaalitytilanteen laitteistossa tämä lähetys todennäköisesti olisi suojattu tavalla tai toisella, etenkin jos yhteydet ovat rakentuneet langattomien verkkojen ylitse.

Scope tunnisti molemmat Probet heti ja mitään erityistä vianselvitystä ei tarvinnut suorittaa. Sinänsä tämä ei tullut yllätyksenä, sillä kaikki laitteet sijaitsivat samassa verkossa, samaan kytkimeen kiinnitettyinä. Noin kahden minuutin ajalta kerättyjen tulosten mukaan suurin ja pienin ajoittainen latenssi asettui välille $\approx +2.5$ ms ja ≈ -1.4 ms. Eli kokonaisvaihtelu latenssissa oli yhteensä noin 4 millisekuntia. Käytännössä viiveetön tiedonsiirto, sillä ihmisoperaattori ei pysty käsittelemään noin pieniä viiveitä. MIT suorittaman tutkimuksen perusteella ihminen minimissään huomaa tai voi "prosessoida" kuvia 13 millisekunnin ajalla. Näiden tulosten perusteella voidaan todeta, että itse verkon puolesta video on käytännössä viiveetön aluksen ohjaajalle, kun käytetään samassa paikallisessa verkossa toimivia laitteita. Käytännössä tähän tulee vielä lisäksi itse kameran sekä palvelimen sisäinen latenssi, jota

nykyisellä mittalaitteistolla ei voida mitata. Videokuvaa seuratessa monitoreilta tämä ns. glass to glass (lasista lasiin, eli videokameran edestä näytölle) latenssi tuli selvästi esille, tuoden esille tämän tutkimuksen yksistä heikkouksista, koska sitä ei voitu tutkia rakennetulla mittauslaitteistolla.

5 Viimeistelty mittausjärjestelmä

Ongelmana nousi esille PTZ kameran suoratoiston RTSP- muoto ja siltauksen suorittavien laitteiden tuen puute kyseiselle formaatille. Sillä formaatti oli eri kamerasta ulostullessa (RTSP) kuin mediapalvelimelle mennessä (MPEG-TS), Scope ei tunnistanut videosignaalia samaksi. Videoformaattia muuttaessa kontekstiedot ja pakettien sisältämän datan identifiointiheaderit muuttuivat, estäen pakettien tunnistamisen samoiksi alku ja loppupäässä.

Tästä johtuen kameraan täytyi asentaa erillinen kolmannen osapuolen lisäosa, joka tekee erillisen videosuoratoiston jonka formaattia ja muita konfiguraatioita voidaan säätää manuaalisesti. Tämän lisäosan avulla mahdollistettiin MPEG-TS muotoinen lähetys alku ja loppupäähän, joten Scope onnistui tunnistamaan datafeedit saman lähetyksen kahtena puolena. Toisena mahdollisuutena olisi ollut SRT formaatti, mutta mediapalvelinta ei syystä tai toisesta saatu toimimaan tämän protokollan kanssa. Tämä olisi ollut mielenkiintoista testata lisäksi, sillä käyttäen SRT:tä, olisi mahdollistettu protokollan pysyminen samana koko matkalta kamerasta palvelimelle. Näin minimoitaisiin protokollan muuntamisen tuoma mahdollinen ylimääräinen viive. Jos reitillä tapahtuu useita formaatin muutoksia, myös tämä konvertointi vie resursseja ja tuo lisäprosessointia johtaen latenssin kasvuun.

Näiden mittausten erona PoC testaukseen on kameran ja mediapalvelimen erotus erillisiin verkkoihin. Näiden verkkojen väliin asetettiin erilaisia siltoja toimimaan ratkaisuna videon siltaamiseen. Ensimmäisessä mittauksessa siltaaminen tehtiin kahdella laitteella, jotka muuttivat videoprotokollan MPEG-TS muodosta ensin SRT protokollaan laitteiden välille ja lopulta taas takaisin MPEG-TS muotoon mediapalvelimelle.

Sillä Probet kommunikoivat pieniä määriä paketteja kontekstietojen vuoksi, tätä datan siirtoa varten täytyi muodostaa oma reittinsä. Videokuvan kulkema reitti tukee vain pelkästään UDP over TS muodossa olevan pakettien siirtämistä, joten kommunikointipaketit eivät pysty kulkeutumaan siinä samalla. Jos nämä kaksi Probea olisivat sijainneet samassa lähiverkossa, mitään

toimenpiteitä ei olisi tarvittu, mutta tämä vaadittu verkon ylitys toteutettiin käyttäen rinetd palvelua Linux pohjaisella Bastion-hostilla. Käytännössä rinetd toimii yksinkertaisena proxyna tcp tiedonsiirrolle. Tällaista palvelua pyörittävä Bastion-host kuuntelee asetettujen ohjeiden mukaan tiettyä porttiaan ja saadessaan paketteja siihen, ohjaa ne eteenpäin parametriensä mukaan toiseen osoitteeseen. Sitä voi ajatella liikenteenohjaajana viittomassa reittiä datapaketeille, jotka kulkevat kahden erillään olevan verkon välillä.

Tämän luomisen myötä Scopessa täytyi vaihtaa valmiina olevat portit toisiin. Scopen käyttöliittymän monipuolisuuden vuoksi asia ei kasvanut ongelmaksi vaan muutos saatiin tehtyä pikaisesti. Jos mittaus olisi suoritettu yksinkertaisemmassa verkossa, erillistä proxytusta ei olisi tarvinnut tehdä. Suurimmassa osassa Qosiumilla suoritetuista palvelunlaatumittauksista tehdään samassa verkossa ollessa, joten verkon yksitasaisuus mahdollistaa sekä mittauskommunikoinnin Probeta Probelle, että videosignaalin lähetyksen ilman erillisiä kikkailuja.

Periaatteessa tämä Qosium Proben eriytetty toiminta (kommunikointi ja video erillisillä verkoissa) kahteen verkkoon muuttaa mittatuloksia, mutta käytännössä kontekstietojen vaikutus mitattavaan datankulkuun on merkityksetön

Kun kaikki vaadittavat siltaukset ja reititykset saatiin luotua, itse mittaus onnistui täysin samalla tavalla kuin alustavat PoC mittaukset. Huomattava ero näkyi vasta tuloksia tarkastellessa. Kun ensimmäiset samassa verkossa suoritettut mittaukset pyörivät 4 millisekunnin paikkeilla, lopullisessa mittauksessa latenssin mediaani oli noin 385 millisekuntia. Tämä alkaa olla jo merkittävä aikaviive videosignaalin, ja tämän vaikutuksen huomaa heti jos kameraa voidaan ohjata suoraan. Tämä myös on yksi isoimmista kehityksen kohteista. Verkkojen ja tämän videosignaalin käsittelyn optimointi lähes varmasti parantaa suorituskykyä ja minimoi viivettä. Lähes jokaisella ammattikäytössä olevalla kytkimellä on jokin tapa priorisoida liikennettä esimerkiksi joko VLAN tasolla tai varaamalla suuremman kaistanleveyden tiettyjen porttien välillä kulkevalle datalle. (Zoeller, 2013)

6 Vaikuttajien analysointi

Kuten edellisessä kappaleessa huomattiin, paikallisessa verkossa mediasiiirron latenssi on lähestulkoon nolla. Käyttäjälle osoittautuvassa kokonaiskuvassa kuitenkin tilanne ei läheskään ole näin ruusuinen. Testauksissa käytössä oleva mediapalvelin omaa n. 17 millisekunnin sisäisen latenssin sen ohjausohjelmiston mukaan ja kamera lisää tähän jonkin verran lisää, riippuen käytetäänkö kamerassa alkuperäisestä muodossa dekoodattua RTSP lähetystä vai laajennuksen avulla luotua MPEG-TS lähetystä. Tässä erillisen laajennuksen käyttö luontaisesti luo enemmän latenssia. Siirryttäessä lopulliseen mittausjärjestelmään, mediapalvelimen ja kameran välille lisätyt videosignaalin siltauslaitteet lisäsivät datankulkuun vähintään 150 millisekuntia, joka oli SRT protokollalla suositeltu latenssiarvo näille laitteille.

Toinen huomattava piirre tuloksissa oli latenssin "pieneminen" ajan myötä. Arvot lähenivät nollaa ja minimilatenssi kasvoi suuremmaksi negatiiviseksi luvuksi koemittausten edetessä. Todennäköisin syy tälle on kahden Proben aikaero millisekuntitasolla. Sillä mittauksia tuloksetkin ovat näin tarkassa mittaskaalassa, pienikin kellojen ajautuminen laitteissa johtaa sykkroinnin menettämiseen. Tämä taas vaikuttaa suoraan tuloksiin, sillä tuloksien luontia varten tarvitaan paketin aloituslähtökohta sekä tarkka aika loppukohteessa. Kun näiden kahden kohteen sisäiset kellot eroavat toisistaan, ero ilmenee tuloksissa suoraan. Tässä tapauksessa siis mahdollisesti kameran kello siirtyi eteenpäin, palvelimen kello taaksepäin tai hieman molemmat sattuiivat. Valitettavasti tähän ei suuremmin ole ratkaisua, sillä kameran tai palvelimen kellon NTP synkronoinnin taajuutta ei pystytty muuttamaan. Pahimmassa tapauksessa laite voi luulla olevan vakaasti ajassa yhtenä hetkenä ja asettaa seuraavan synkronointiajankohdan vaikkapa minuutiksi eteenpäin. Perinteisen tietokoneen kellon kanssa tämä ei vaikuttaisi laitteen käyttöön lähes mitenkään, mutta tutkimuksen kaltaisessa tilanteessa tämä synkronoinnin menetys heikentää tulosten luotettavuutta. Käytännössä tämä tarkoittaa, että mittauksen alkuvaiheet ovat lähimpänä todellisuutta latenssiarvoissaan.

7 Lopuksi

Raportin tavoitteena ollut keskeisten latenssitekijöiden tunnistus onnistui mielestäni tyydyttävästi. Vaikka yksittäisten laitteiden erottelua ei suoritettu jokaiselle osiolla, latenssin suurimmat ja staattiset vaikuttajat saatiin silti identifioitua. Käytetty ohjelmisto mahdollisti yllättävän tehokkaasti mittauksen eri osioilla vaikka sen käytönottaminen halutulla tavalla osoittautuikin vaikeammaksi mitä alussa ennustettiin. Tässä suurena apuna toimi Kaitotekin yhteyshenkilöt, ilman heidän apuaan mittaukset olisivat tyssänneet jo aivan alkutekijöissä.

Työssä tehtyjen mittausten arvo tuotekehityksessä on suuri ja kohdistuu yhteen etäohjauksen suurimpaan tämänhetkiseen haasteeseen. Merenkulun seuraavalle teknologiselle harppaukselle, videolagenssin seurannan ja ohjauksen viiveen tarkempi hallinta on kriittistä turvallisen tulevaisuuden kehittämiseksi. Työn aikana syntynyttä mittausrunkoa voidaan käyttää etäohjausaseman teknologisen jatkokehityksen suuntaamisen tärkeänä tiedonlähteenä. Opinnäytetyön jälkeen päätettiin ostaa lisenssi Qosiumin jatkotutkimusta varten, mutta tekemäni mittaukset saatiin suoritettua kokeiluersiota käyttäen. Ottaen huomioon, että työn tuloksia tullaan käyttämään jatkossa tuotekehityksessä, projektin kustannukset ovat pienet sen tuomiin oppeihin verrattuna. Oppimani tiedon määrä opinnäytetyön aikana on erittäin suuri. Verkkotopologiasta mediaprotokolliin ja -formaatteihin, Linux-pohjaisten käyttöjärjestelmien hengenelämään ja virtualisoinnin perusteisiin. Opinnäytetyöni toimi suunnattomana mahdollisuutena ja motivoijana oman tietotaidon karttamisessa ja urakiinnostuksen ohjaamisessa.

Lähteet

Al-Mualla, M.E., Canagarajah, C.N. and Bull, D.R. (2002) 'Error-Resilience Video Coding Techniques', in *Video Coding for Mobile Communications*. Elsevier, pp. 205–229. Available at: <https://doi.org/10.1016/B978-012053079-3/50011-1>.

Autonomous Shipping (2022) Kongsberg Maritime. Viitattu 10.7.2022 <https://www.kongsberg.com/maritime/support/themes/autonomous-shipping/>

Conrad Eric, Misener Seth and Feldman Joshua (2012) *CISSP Study Guide*. 2nd edn. Elsevier. Available at: <https://doi.org/10.1016/C2009-0-61065-5>.

Demo & Use Case (2021) Autoship. Available at: <https://www.autoship-project.eu/demo-use-case/> (Accessed: 8 June 2022).

Emmet, D. (2017) 'OSI Model and Network Protocols', in *CompTIA Network+ N10-007 Exam Cram*. 3rd edn. Pearson Education.

Fortinet (2022) *User Datagram Protocol (UDP)*, Fortinet.

IMO (2021) *Outcome of the regulatory scoping exercise for the use of Maritime Autonomous Surface Ships (MASS)*.

Introduction to the Real-time Transport Protocol (RTP) (2022) Mozilla. Available at: https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API/Intro_to_RTP (Accessed: 17 October 2022).

Koistinen, T. (1999) 'Protocol overview: RTP and RTCP', *Teknillinen korkeakoulu* [Preprint].

Latency in Ethernet Switches (2016). Available at: https://networking.report/Resources/Whitepapers/196c5828-cb30-4664-8269-c524c18b55eb_Latency-in-Ethernet-Switches.pdf (Accessed: 17 October 2022).

N5860-48SC Switch Datasheet (no date). Available at: <https://img-en.fs.com/file/datasheet/n5860-48sc-switches-datasheet.pdf> (Accessed: 17 October 2022).

Nikols, L. (2022) *Video Encoding Basics: Live Video Streaming Protocols for Broadcast Contribution and Distribution*, Haivision.

Noergaard, T. (2010) 'The Fundamentals in Understanding Networking Middleware', in *Demystifying Embedded Systems Middleware*. Elsevier, pp. 93–190. Available at: <https://doi.org/10.1016/B978-0-7506-8455-2.00004-2>.

Saarni Jouni, Nordberg-Davies Sini and Saurama Antti (2018) *Outlook on the transition towards autonomous shipping*.

Technologies (2021) *Autoship*. Available at: <https://www.autoship-project.eu/technologies/> (Accessed: 8 June 2022).

Tran, D.T. *et al.* (2005) *Scps-Tp, Tcp, And Rate-Based Protocol Evaluation*.

Trattnig, A. (2022) *The Definitive Guide to Container File Formats [2022]*, Bitmovin.

UKK (2022) *Kaitotek*. Available at: <https://kaitotek.com/fi/resources/faq> (Accessed: 15 July 2022).

Zoeller, C. (2013) 'How Does QoS Work?', *Packet Pushers*. Available at: <https://packetpushers.net/how-does-qos-work/> (Accessed: 10 September 2022).