Bacheloroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Institutt for elektroniske systemer

Gerd Åshild Ueland
Ida Westerheim

# TWAMP sammenlignet med tradisjonelle målemetoder for nettverksovervåkning

Bacheloroppgave i Elektronikk
Veileder: Mohammad Derawi
Medveileder: Håkon Gunleifsen

Mai 2021

NTNU
Kunnskap for en bedre verden

Gerd Åshild Ueland
Ida Westerheim

# TWAMP sammenlignet med tradisjonelle målemetoder for nettverksovervåkning

Bacheloroppgave i Elektronikk
Veileder: Mohammad Derawi
Medveileder: Håkon Gunleifsen
Mai 2021

Norges teknisk-naturvitenskapelige universitet
Institutt for elektroniske systemer

NTNU
Kunnskap for en bedre verden

NTNU
Kunnskap for en bedre verden

# TWAMP sammenlignet med tradisjonelle målemetoder for nettverksovervåkning

Gerd Åshild Ueland
Ida Westerheim

Bachelor i elektroingeniør
Innlevert: 20.05.2021
Hovedveileder: Mohammad Derawi

Norges teknisk-naturvitenskapelige universitet
Institutt for elektroniske systemer

NTNU
Kunnskap for en bedre verden

| Oppgavens tittel:<br>TWAMP sammenlignet med tradisjonelle målemetoder for nettverksovervåkning. | Dato: 20.05.2021<br>Antall sider: 125 | | | |
|---|---|---|---|---|
| | Masteroppgave: | | Bacheloroppgave | X |
| Navn:<br>Gerd Åshild Ueland<br>Ida Westerheim | | | | |
| Veileder:<br>Mohammad Derawi | | | | |
| Eventuelle eksterne faglige kontakter/ veiledere:<br>Håkon Gunleifsen | | | | |


**Sammendrag**

Prosjektet var gitt av Eidsiva Bredbånd, da de ønsker å finne en bedre måte å dokumentere kundeopplevd kvalitet på leverte tjenester. Dette for å kunne oppdage og lokalisere problemer i nettverket på en mer effektiv måte. For å gjøre dette ønsker de å undersøke om ny målemetode, Two-Way Active Mesurement Protocol (TWAMP), kan gjøre dette på en bedre måte enn de målemetodene bedriften bruker i dag.

For å sammenligne målemetodene har vi satt opp en testlab med mulighet for å gjennomføre målinger over fiber og over 4G med IPSec. Dette for å kunne gjennomføre målinger over nettverk med ulik kvalitet. Resultatene fra målinger gjennomført med TWAMP vil bli sammenlignet med resultatene fra målemetodene Eidsiva Bredbånd bruker i dag.

Etter å ha sammenlignet alle måleresultatene har vi kommet frem til at TWAMP gir merverdi dersom det brukes sammen med bedriftens nåværende målemetoder. Dersom TWAMP skal implementeres i nettverket som eneste måleverktøy vil det være nødvendig med en orkestrator. En orkestrator er en ekstern enhet som holder oversikt, kjører sesjoner og presenterer resultatet i grafer. Dette gjør at resultatene kan bli sett på i ettertid og vil bidra til å bedre dagens dokumentasjon. Noe som kan brukes til å kvalitetssikre og dokumentere de ulike leddene i nettverket.



**Stikkord**

| |
|---|
| Målemetoder |
| TWAMP |
| Forsinkelse |
| Nettverksovervåkning |
| Pakketap |


Ida Westerheim Gerd Åshild Ueland

Ida Westerheim Gerd Åshild Ueland

# Abstract

This project was provided by Eidsiva Bredbånd whose goal was to obtain a clearer way to documenting the user experienced quality within their delivered broadband service. In other words, this means to be able to discover and localize problems in any kind of network in a more efficient way than current measurement methods. To do this they want to inquire if a new measurement method, Two-Way Active Measurement Protocol (TWAMP), can be performed in a more effective and optimal way.

To compare the measurement methods, we have designed and developed a network lab with the possibility to conduct measurements over fibre and over 4G with IPSec. This is for being able to conduct measurements over a network with different quality. The result of the measurements done with TWAMP will be compared with the results from the monitoring tools Eidsiva Bredbånd uses today.

After the comparison of the measurements performed, we discovered that the TWAMP gives an added value if used together with the current measurement methods of the company. If TWAMP were to be implemented in the network as a separate measurement tool then we would necessary be needing an additional orchestrator. An orchestrator is an external unit that keeps an overview, run sessions and presents the result in graphs. The results can then be looked at afterwards and contribute to improve today's documentation. This can be used to ensure quality and to document the different parts in the network.

# Forord

Denne bacheloroppgaven er den avsluttende oppgaven av studiet elektroingeniør - elektronikk ved Norges Tekniske- og naturvitenskaplige universitet (NTNU) Gjøvik. Hvor vi fikk jobbe med en spennende oppgave gitt av Eidsiva Bredbånd. Kontaktpersonen for prosjektet var Håkon Gunleifsen, ansatt ved Eidsiva Bredbånd.

Prosjektet har vært en lærerik og spennende opplevelse, dette er mye takket være den friheten vi fikk til å løse oppgaven. Muligheten vi fikk til å vinkle oppgaven på en måte som vi syntes var spennende og som var relevant for bedriften var meget motiverende og ga stor arbeidslyst til å gjennomføre prosjektet på en god måte. Vi ønsket også å kunne bruke store deler av emnene som har vært en del av studiet, slik at fagnivået på utdanningen blir representert gjennom arbeidet. Vi ønsket også å arbeide videre på den kunnskapen som vi har tilegnet oss gjennom utdanningen, da dette vil bli aktuelt i arbeidslivet.

Vi vil først og fremst å uttrykke en stor takknemmelighet til vår veileder Mohammad Derawi for god veiledning og gode råd underveis i prosjektet. Vi vil også takke han for hans tilgjengelighet og fleksibilitet i forbindelse med veiledning på bacheloroppgaven.

Vi ønsker også å takke vår oppdragsgiver Eidsiva Bredbånd og ekstern veileder Håkon Gunleifsen for å ha stilt opp med gode råd og god hjelp under prosjektets forløp. Vi vil spesielt uttrykke stor takknemmelighet til Håkon Gunleifsen for god hjelp med faglig innhold og veiledning til å gjennomføre prosjektet

I tillegg er det ønskelig å takke Terje Uhlen for gode råd under planleggingen av designet som er brukt i prosjektet og konfigurasjon av utstyr.

# Innholdsfortegnelse

# Figur- og tabelliste

# Forkortelser

| | |
|---|---|
| ACK | Acknowledgement (number) |
| ADSL | Asymmetric Digital Subscriber Line |
| bps | bit per sekund |
| CLI | Command Line Interface |
| CPU | Central Processing Unit |
| CRC | Cyclic Redundancy Check |
| dB | Desibel |
| DNS | Domain Name System |
| DS | Differentiated Services |
| DSCP | Differentiated Services Code Point |
| DSL | Digital Subscriber Line |
| EB | Eidsiva Bredbånd |
| FPGA | Field-Programmable Gate Array |
| FW | Firewall |
| GUI | Graphical User Interface |
| ICMP | Internet Control Message Protocol |
| IMRAD | Introduction, Methods, Results and Discussion |
| IP | Internet Protocol |
| IPDV | Inter Packet Delay Variation |
| IPSec | Internet Protocol Security |
| IPv4/IPv6 | Internet Protocol version 4/ Internet Protocol version 6 |
| ISP | Internet Service Provider |
| LAN | Local Area Network |
| Laser | Light Amplification by the Stimulated Emission of Radiation |
| LED | Light-Emitting Diode |
| LUT | Look Up Table |
| MAC | Medium Access Control |
| MBZ | Must Be Zero |
| MTU | Maximum Transmission Unit |
| NAT | Network Address Translation |

| | |
|---|---|
| NTNU | Norges Teknisk-Naturvitenskaplige Universitet |
| OSI | Open System Interconnection Basic Reference Model |
| OWAMP | One-Way Active Measurement Protocol |
| PCP | Priority Code Point |
| PDV | Packet Delay Variations |
| QoE | Quality of Experience |
| QoS | Quality of Service |
| RAM | Random Access Memory |
| RFC | Request For Comments |
| RTP | Real-time Transport Protocol |
| RTT | Round Trip Time |
| SFP | Small Form-factor Pluggable transceiver |
| SNMP | Simple Network Management Protocol |
| SNR | Signal to Noise Ratio |
| SYN | Synchronize |
| TCP | Transmission Control Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| ToS | Types of Service |
| TTL | Time To Live |
| TWAMP | Two-way Active Measure Protocol |
| UDP | User Datagram Protocol |
| USB | Universal Serial Bus |
| VCE | Virtual Customer Edge |
| VLAN | Virtual Local Area Network |
| VPN | Virtual Private Network |
| WAN | Wide Area Network |
| WDM | Wavelength-Division Multiplexing |

# Definisjoner

**SFP**

Small Form-factor pluggable (SFP) transiver er en modul med en kontakt som kan kobles til for eksempel en svitsj. Denne brukes ofte for å unngå unødvendig bruk av ekstra overganger. Den brukes stort sett for å koble til fiber eller kobberkabel.

**Jitter**

Brukes for å kunne se forskjell i tidsforsinkelse mellom pakker. Dersom to pakker blir sendt og den første pakken bruker 3ms og den andre pakken bruker 7 ms vil det bli en jitter på 4ms.

**Ping**

En liten pakke blir sendt til en server som returnerer pakken. En vil da få beskjed om hvor mye tid som har gått fra pakken blir sendt, til den blir mottatt igjen. Det er ønskelig med lavest mulig ping-tid for å ha raskets mulig nett.

**Node**

En node er en fellesbetegnelse for ulike typer enheter i nettverk. En node kan bearbeide, motta eller sende data.

**Link**

En link er transmisjonsmediet mellom noder. Linken kan være både trådløs og kablet.

**Båndbredde**

Båndbredde er antall bits som kan sendes samtidig over transmisjonsmediet.

# 1 Introduksjon

I dette kapittelet vil vi se på årsaken til at Eidsiva bredbånd ønsker å se på en ny målemetode for å forbedre kvaliteten på nettverket sitt. Dette for å kunne gi en bedre kvalitetssikring av produktet de leverer. Det vil også bli presentert bakgrunnen for oppgaven, samt motivasjonen tilhørende prosjektet. Kapittelet vil blant annet inneholde prosjektets problemstilling, hvordan rapporten er bygd opp og de aktuelle begrensingene som gruppen har blitt enige om i samråd med oppdragsgiver.

## 1.1 Bakgrunn for oppgaven

Samfunnet er under en konstant utvikling når det kommer til teknologi. Eidsiva Bredbånd utvikler kontinuerlig den nettverkshastigheten de kan tilby kunden. Hos dem har dette utviklet seg fra Asymetrisk Digital Subscriber Line (ADSL), hvor kunden hadde så lav hastighet som 1 til 10 Mbps, til fiber hvor kundene kan få opptil 1 Gbps. Fibernettet har betraktelig bedre ytelse sammenlignet med de andre overføringsmetodene og kunden vil nesten alltid oppleve internettilkoblingen som hurtig og stabil. Selv om hastigheten har økt, har kundenes krav til kvalitet, stabilitet og forutsigbarhet på internettforbindelsene økt. Sluttbrukerne stiller også krav til at de får det produktet de betaler for.

Ett annet aspekt rundt opplevd tjenestekvalitet på bredbåndsleveranser er sluttbrukerens bevissthet av hva en bredbåndsmåling innebærer. Et betydelig antall enheter koblet til en bredbåndslinje spiser opp kapasiteten hvor resterende kapasitet er det som kan måles. Dette gjør at en kvalitetssikring både ovenfor leverandøren og sluttbrukeren vil bli mer aktuelt i utviklingen av kapasiteten som kan tilbys til en sluttbruker.

I henhold til Eidsiva Bredbånd, er det å finne ut av hvor feilen er, en av de største utfordringene i en feilsituasjon knyttet til reduserte hastighetsmålinger. Enda vanskeligere er det når en bredbåndslinje leveres av flere leverandør og kommunikasjonsnett. I en slik verdikjede kan for eksempel A-B være levert av NextGenTel, B-C kan være levert av Eidsiva Bredbånd og C-D kan være levert av Telenor. Her vil det være ressurssparende hvis en hadde funnet en metode

for å kunne overvåke og sikkert påpeke hvor i verdikjeden en feil oppstår, eller kapasiteten ikke er tilstrekkelig. Dette kan også bidra til en kvalitetssikring ovenfor kundene.

Nye målinger kan bidra til å øke kvalitetssikringen på den leveransen som Eidsiva Bredbånd tilbyr kundene. De målingene som ønskes gjennomført er blant annet punkt-til-punkt målinger, hastighetsmålinger, tapsmålinger og generelt alle målinger som er med på å gi et mer fullstendig bilde av hvordan kvaliteten på nettverket er. Eidsiva Bredbånd ønsker å bruke de målingene som blir gjort til blant annet å dokumentere kvaliteten på sambandsleveransen, verifisering av nettverkskonfigurasjon, overvåkning i tilfelle feil oppstår og kunne se endringer i kvaliteten på sambandet.

## 1.2 Problemstilling

Den valgte problemstillingen er:

*Eidsiva Bredbånd ønsker å forbedre internkontroll av kvaliteten på nettverket de leverer. For å kunne gjennomføre dette har de ett ønske om å ta i bruk Two-Way Active Measurment Protocol (TWAMP). Denne protokollen ønskes å bli brukt for å kunne dokumentere kvaliteten på sambandsleveransen, verifisere nettverkskonfigurasjon, overvåke nettverket i tilfelle feil oppstår og kunne se endringer i kvaliteten på sambandet. I dag bruker Eidsiva Bredbånd tradisjonelle målemetoder, ved bruk av Iperf, Ping og SNMP. Det er derfor en del av denne oppgaven å se på forskjellene mellom disse tradisjonelle målemetodene og nyere metoder.*

Forskningsspørsmål:

1. Er ny målemetode, TWAMP, bedre enn bedriftens nåværende målemetoder?
2. Hvordan kan innføring av TWAMP forbedre dokumentasjonen av kvaliteten på nettverket som blir levert over fiber?

## 1.3 Motivasjon

Etter hvert som teknologien utvikles, blir etterspørselen for bredbånd stadig større. De fleste husstander og bedrifter har nå tilgang til fiberbredbånd. Dette gjør at behovet for å kunne sikre kvaliteten på bredbåndet samt dokumentere statusen på linken til enhver tid også vil øke. I dagens samfunn så består hjemmene og arbeidsplassene våre av mange enheter som er avhengig av internett for å kunne fungere. Eksempler på slike enheter er videokonferanseutstyr, kameraovervåking, varmekabler, sikkerhetsalarm, datamaskiner og mobiler. Tidligere var det for det meste kun en datamaskin eller en telefon som var tilkoblet nettverket, men nå går så å si alt vi har på denne teknologien. Eidsiva Bredbånds kunder bruker derfor mer nett og har en større etterspørsel etter kvalitet og stabilitet.

Eidsiva Bredbånd har ett ønske om å kunne sikre og øke kvaliteten på det produktet de leverer til kundene sine. Da det er et stort forbedringspotensial i bedriftsmarkedet for Eidsiva Bredbånd når det kommer til de overvåkningsmetodene og testmulighetene de har aktivt i bruk per dags dato. Mange kunder kontakter bedriften med klager i forbindelse med pakketap, dårlig opplevelse av internett, utfall på linjen og ustabilitet på nettverket. Per dags dato er det ikke mulig å periodisk kvalitetssikre reel makshastighet. En konsekvens av dette er at det er vanskelig å dokumentere oppsatt hastighet i en tjenesteleveranse.

Bedriften bryr seg om kundeopplevd kvalitet på tjenestene. De tradisjonelle målemetodene er gjennomsnittsmålinger og har i hovedsak bestått av trafikkmengdemåling og responstidsmåling av fysiske sambandslinjer. Målingene gjøres ofte på målepunkter sentralt i nettverket, og møter ikke lengre kravene som stilles til dokumentasjon i forbindelse med kvalitetssikring av samband til enkeltkunder på en tilfredsstillende måte. Eidsiva Bredbånd ønsker i denne forbindelsen å se på en ny målemetode. Two-Way Active Mesurement Protocol (TWAMP) som definert i Request For Comments (RFC) 5357 *[1]* er en ny målemetode som bedriften ønsker å ta i bruk. Denne åpner for muligheten for å gjennomføre kontinuerlige målinger mellom alle målepunktene som er satt opp i nettverket. Denne ende-til-ende-målingen gir ett bedre bilde av hvordan kunden opplever kvaliteten. Bedriften ønsker å ta i bruk TWAMP i håp om at dette kan gi et bedre dokumentasjonsgrunnlag som kan brukes til å kvalitetssikre nettverkssambandet til kunden. Eidsiva Bredbånd ønsker å benytte protokollen i smarte Small Form-factor Pluggable (SFP) transivere for å enkelt kunne implementere og gjennomføre målinger i de ønskede endepunktene.

## 1.4 Oppdragsgiver

Eidsiva Bredbånd er et selskap som leverer bredbånd til store deler av innlandet. De tilbyr bredbåndstilgang til store deler av Norden, dette via deres samarbeidspartnere. Eidsiva Bredbånd ble startet opp i 2004 og er et datterselskap i Eidsiva konsernet. Konsernet består av tre datterselskap; Eidsiva Bioenergi, Eidsiva Bredbånd og Elvia. Eidsiva Bredbånd leverer internettilgang hjem til privatpersoner og til ulike bedrifter. Som en internettleverandør vil de være med på å knytte sammen samfunnet, både mellom næringer og enkeltindivider.

Som en nettverksleverandør, har Eidsiva Bredbånd vært igjennom en teknologisk utvikling. Siden oppstarten av selskapet på tidlig 2000-tallet har teknologien de benytter seg av blitt utvidet fra kun Digital Subscriber Line (DSL) til og i hovedsak benytte seg av fiberteknologi og koaksialkabel. I denne oppgaven fokuseres vi på fiberleveransene som er levert i bedriftsmarkedet.

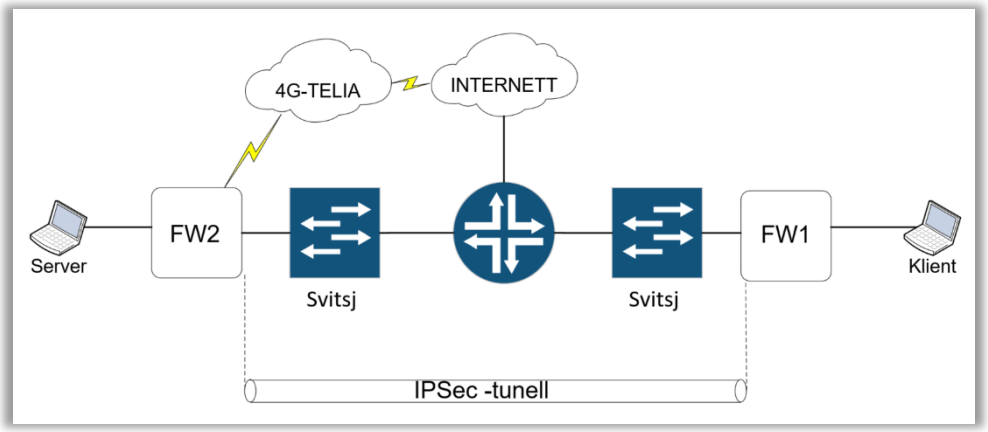
## 1.5 Rapportens oppbygning

Denne rapporten er bygd opp med bakgrunn i Introduksjon, Metode, Resultat og Diskusjon (IMRAD)-strukturen[2]. Derfor vil vi etter dette kapittelet, innledning, se på relevant teori til oppgaven. Denne teoridelen er vesentlig da den vil danne grunnlaget for å forstå resultatet og fremgangsmetoden. Etter teori vil det bli sett på den metoden som er brukt for å gjennomføre oppgaven. Det vil så komme ett kapittel om design og implementasjon, det vil her bli beskrevet de stegene vi må igjennom for å kunne gjennomføre målinger. Resultatet av de målingene som det er klargjort for i design og implementasjon vil deretter bli sett på, før vi går over til en evaluering. Denne evalueringen skjer med bakgrunn i teoridelen, og resultatene for målemetodene vil her bli evaluert, i tillegg til metoden og designet. Til slutt vil vi komme med en konklusjon som har bakgrunn i måleresultatene og evalueringen.

## 1.6 Begrensinger

Det er blitt satt noen begrensninger når det kommer til denne oppgaven. Disse begrensningene er satt med bakgrunn i oppgavens omfang, og er med på å begrense oppgaven til den ønskede vinklingen.

- Målingsteknologien skal kun brukes for bedriftsmarkedet.
- Aksessteknologi begrenses til å i hovedsak inkludere fiber, men med mulighet for redundans på 4G.
- Det er gitt føringer til å spesielt se på målemetode ved bruk av TWAMP.
- Føringer for plattform til TWAMP målinger er satt til leverandøren Accedian.
- Det finnes flere andre målemetoder som blir brukt i dag, men vi valgte å benytte oss av de målemetodene som Eidsiva Bredbånd bruker nå.
- Det er definert en nettverksprinsippskisse som inkluderer 2 laboppsett, en over 4G og en over fiber. Det vil være elementene i denne skissen (fig. 1.1) som vil danne grunnlaget for teori, design, test av målemetoder og evalueringer i denne oppgaven.



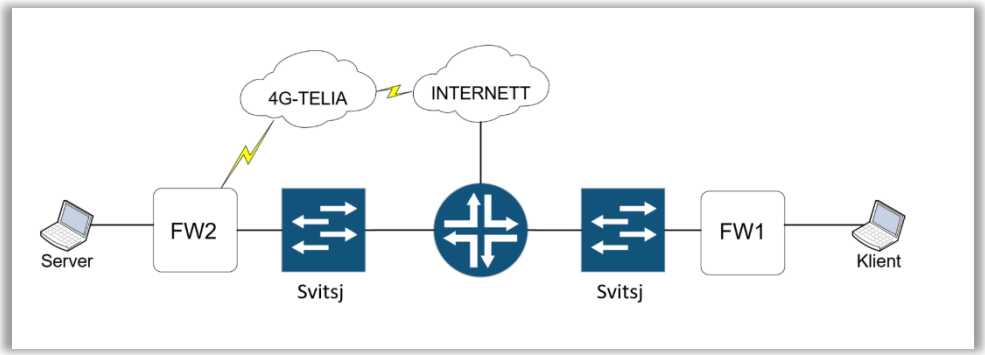

*Figur 1. 1: Nettverkprinsippskisse for laboppsett*

# 2 Teori

I tidligere kapittel har det blitt sett på bakgrunnen for at Eidsiva Bredbånd ønsker å forbedre målemetodene de benytter seg av. De ønsker å forbedre disse for å kunne gi en økt kvalitetssikring av nettverket og produktet de leverer. Den forbedrede målemetoden skiller seg fra tradisjonelle målemetoder ved at det kan gjennomføres dynamiske ende-til-ende målinger i stedet for punktmålinger. Dette kapittelet gir nettverksbakgrunnen som forklarer forskjellen på disse typene nettverksmålinger.

## 2.1 Nettverksoppbygning

I dette kapittelet skal det bli sett på de enhetene som brukes i generell nettverksoppbygning og som er de fysiske enhetene som brukes i prosjektets laboppsettet. Ett nettverk vil alltid være bygd opp av flere enheter og hver enhet vil bidra til at en pakke blir sendt imot destinasjonen. I figuren (fig.2.1) ser vi prinsippskissen til ett nettverksoppsettet, dette nettverksoppsettet inneholder FW1 (firewall 1) og FW2 (firewall 2) som er brannmurer (kap. 2.1.5). Videre har vi har to svitsjer (kap. 2.1.2), en ruter (kap. 2.1.3) og 4G modem via Telia (kap. 2.1.6). Her vil FW1 kontrollere trafikken som sendes til klienten og FW2 kontrollere trafikken som sendes til server. Svitsjene sender pakkene mot riktig destinasjon og ruteren vil sende pakkene mellom de to ulike nettverkene.




*Figur 2. 1: Nettverksprinsippskisse for overordnet laboppsett*

### 2.1.1 Node og kommunikasjonskanal

I ett nettverk er det mange enheter, og hver enhet kalles en node. Dersom en node har en IP-adresse, blir det kalt en nettverksnode. En node kan lage, lagre, sende eller motta data, dette gjør at noden kan være både mottaker, avsender eller et strategisk punkt pakken må sendes igjennom. Noen av de enhetene som kalles noder er rutere, servere og datamaskiner. [3]

For å koble sammen nodene brukes en kommunikasjonskanal, disse kalles ofte linker. En link kan både være en fysisk kabel som for eksempel fiber eller en kobberkabel eller den kan være trådløs. En link kan være en ende-til-ende-link som sender informasjon fra avsenderenheten til mottakerenheten, eller en kringkastingslink der informasjon går fra en enhet til alle enheter innen rekkevidde som ønsker å motta informasjonen. I figuren (fig. 2.1) kan vi se et eksempel på en fysisk kabel mellom FW2 og den ene svitsjen og et eksempel på en trådløs link mellom FW2 og 4G-TELIA. [4]

### 2.1.2 Svitsj

I figuren (fig. 2.1) kan vi se to firkantete, blå enheter med navnet svitsj. En svitsj er en enhet i ett nettverk hvor flere kommunikasjonslinker kobles sammen. Her vil pakkene som svitsjen mottar bli sendt videre i retning av destinasjonen. Svitsjen vil "pakke opp" pakkene den mottar i datalinklaget (kap. 2.2.3). Svitsjer har også den egenskapen å kunne determinere hvor en innkommende datapakke skal sendes videre. Dette vil lede til raskere overføringer og mindre fare for tap og høy forsinkelse på pakkene. Svitsjer kan kun koble sammen enheter i ett lokalt nettverk, derfor må pakkene som skal ut på et annet nettverk sendes via en ruter (kap. 2.1.3). [5]

### 2.1.3 Ruter

I figuren (fig. 2.1) kan vi se en ruter, dette er den runde enheten imellom de to svitsjene. Dette er en internettruter som vil si at enheten gir tilgang til internett og distribuerer internettpakker. En ruter er en enhet som har i oppgave å sende pakker mellom ulike datanettverk. Ruteren inneholder rutingtabeller (kap. 2.2.4) som den bruker for å sjekke hvor pakkene som passerer skal sendes videre utfra informasjonen som ligger i datapakken. En ruter vil altså rute pakkene i riktig retning og spiller en sentral rolle når det blant annet kommer til internett. [6]

### 2.1.4 Server

Til venstre i figuren (fig. 2.1) kan vi se en enhet som kalles server. En server, også kjent som en tjener, er en maskin som kjører en tjeneste som kan brukes av flere enheter. I en bedrift kan en server settes opp for blant annet e-post, lagring og utskrift av dokumenter slik at den er tilgjengelig for de som har behov for tilgang. En server kan også settes opp på internett, da kan den blant annet brukes til å være vert for nettsider. I dagligtale kan en si at ting "flyttes ut i skyen". [7]

### 2.1.5 Brannmur

I figur (fig. 2.1) refereres det til to brannmurer (FW1 og FW2). En brannmur har som oppgave å hindre uønsket trafikk i å komme inn i ett nettverk. Dette vil gjøres ved å kun åpne for ønskelig trafikk og sperre for uønsket trafikk. Dette vil for eksempel være å tillate pakker som kommer fra nettleser og e-post. En brannmur kan også settes opp til å bare slippe inn pakker som blir forespurt på innsiden av brannmuren, eller den kan se på innholdet i det som blir mottatt og vurdere om pakken er grei å sende videre i nettverket eller om den skal droppes. En brannmur kan være programvare og/eller maskinvare. [8]

### 2.1.6 4G

I figuren (fig. 2.1) ser vi en sky med navnet 4G-Telia. Denne skyen illustrerer den trådløse forbindelsen ett 4G-modem introduserer i oppsettet. Dette er også en av hoveddelene i det nettverket som vil kalles laboppsett 2 i design og implementasjon (kap. 4.1.1). 4G blir ofte implementert på de områdene det ikke er mulig eller praktisk å bruke kablet nettverk for deler av strekningen for internettleveransen. 4G kan enten brukes direkte av mobiler eller så kan det brukes av enheter koblet til en ruter med støtte for mobilt bredbånd. Det er flere faktorer som spiller inn på hvor god tilkobling 4G kan gi. Noen av faktorene som reduserer kvaliteten på 4G kan være om det er mange enheter koblet til samme sender, om det er langt unna senderen og plasseringen av utstyret. Det kan allikevel være et godt alternativ i områder der det ikke er fiber, eller som en redundant link om noe skulle skje med fiberen. [9-11]

## 2.2 Relevant nettverksteknologi

I det forrige delkapittelet så vi på enhetene i prinsippskissen for vårt nettverk. I dette delkapittelet tar vi for oss nettverksteknologien som disse nettverksenhetene bruker. Vi vil se på to ulike nettverksmodeller (kap. 2.2.1 og kap. 2.2.2) hvor vi spesielt tar for oss 2 elementer fra denne, henholdsvis datalinklaget (kap. 2.2.3) og nettverkslaget (kap. 2.2.4). Disse lagene er viktigere fordi de er med på å forklare teorien bak hastighetsbegrensede faktorer og forsinkelser (kap. 2.2.10) i datanettverk. Vi vil også se på andre faktorer som påvirker hastighet og opplevd kvalitet i ett nettverk som for eksempel Transmission Control Protocol (TCP) vindu (kap. 2.2.5), pakkefragmentering (kap. 2.2.6), Network Address Translation (NAT) (kap. 2.2.7) og IPSec (kap. 2.2.8). Til slutt vil vi vise hvordan jitter (kap. 2.2.11) er en måleenhet som kan si noe om slik nettverkskvalitet.

### 2.2.1 OSI-modellen

Det skal i dette underkapittelet bli sett på teori omkring oppbyggelsen av nettverk og de ulike laginndelingene vi har i nettverkssammenheng. I datakommunikasjon er det en modell som brukes som en referanse for laginndelingen i nettverk. Denne modellen kalles Open Systems Interconnection Basic Reference Model (OSI-modellen). OSI-modellen gir retningslinjer for hvordan kommunikasjonen skjer i et nettverk og det er en modell som består av 7 lag (fig. 2.2), de høyeste lagene representerer et høyere abstraksjonslag. De 7 lagene er [12]:

1. **Fysisk lag**: Her defineres utstyrets fysiske utforming og laget jobber på en link og overfører bit over disse. Laget kobler sammen noder i nettverket og bruker både kablet og trådløs overføring.

2. **Datalinklaget**: Står for overføring av data internt i nettverket og vil korrigere feil som oppstår på det fysiske laget. Laget inneholder flytkontroll og feildeteksjon og kan derfor forespørre om pakker kan sendes på ny om de går tapt i overføringen.

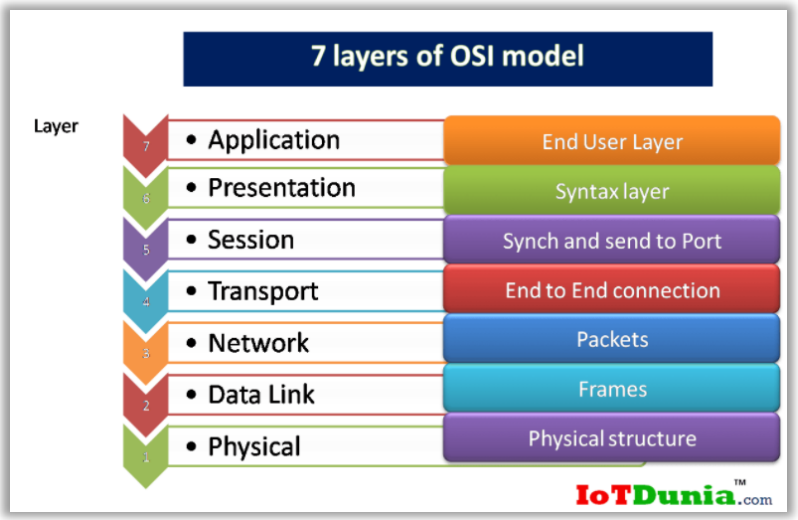
3. **Nettverkslaget**: Bestemmer ruten en pakke skal ta fra A til B og vil overordnet ta seg av overføringen av informasjon mellom endepunkter. Internet Protocol (IP) er en av de vanligste protokollene som kjører på dette laget. Laget vil koble sammen ulike nettverk ved hjelp av IP-adresser.

4. **Transportlaget**: Her lages det kanaler for ende-til-ende kommunikasjon. Laget sørger for at overføringen i nettverkslaget skjer uten feil og korrigerer eventuelle feil som oppstår. I dette laget brukes blant annet protokollene TCP og User Datagram Protocol (UDP).

5. **Sesjonslaget**: Står for dialog mellom hvert endepunkt.

6. **Presentasjonslaget**: Laget står for kryptering og komprimering data og vil sørge for at data vises korrekt.

7. **Applikasjonslaget**: Laget utfører tjenester slik at applikasjoner kan kommunisere seg imellom. Dette gjør at programvarer kan forholde seg direkte til hverandre uavhengig av hvordan hvert enkelt system fremstiller dataen.




*Figur 2. 2: De syv lagene i OSI-modellen sin oppbygning [1]*

OSI-modellen er en lagdelt modell som sier noe om hvordan kommunikasjon skjer og er en referansemodell som produsenter av utstyr kan referere til. Det er tydelige grenser mellom lagene og hvert av lagene har definerte arbeidsoppgaver. Når en pakke sendes fra applikasjonslaget, vil hvert lag den sendes via legge på en header på pakken. Dersom pakken blir mottatt vil headere bli pakket opp av enheter for å avgjøre hvor pakken skal sendes til. Informasjonen i headeren gjør at pakken vil bli sendt i riktig retning, mot destinasjonen. OSI-modellen blir ofte forenklet til Transmission Control Protocol/Internet Protocol (TCP/IP) modellen (kap. 2.2.2), da TCP/IP modellen er en mer protokollorientert tilnærming og en mer praktisk modell.

Siden lagene har definerte arbeidsoppgaver, så vil det være mulig å gjennomføre målinger på de ulike lagene. Da er det slik at målinger som gjøres på svitsjer vil være målinger på datalinklaget og målinger som gjøres på rutere vil være målinger på nettverkslaget. Det vil derfor ikke være mulig å oppdage hvor feil oppstår i på en svitsj som er mellom to rutere, dersom målingene gjøres på nettverkslaget. For å kunne måle over alle lagene i OSI-modellen ønskes det å kunne gjøre ende-til-ende målinger. Det vil da være nødvendig med nettverksutstyr, som en SFP (kap. 2.7), eller en applikasjon lastet ned på endepunktet. Det vil gjøre at målinger gjøres på hele strekket, mellom endepunktene og over alle lagene. Denne typen måling vil kunne gi et bedre bilde av hvordan sluttbrukeren opplever nettverkskvaliteten.

### 2.2.2 TCP/IP modellen

TCP/IP modellen (fig. 2.3) er en referansemodell i datakommunikasjon og er en sammensetting av protokoller som datamaskiner bruker for å kommunisere med hverandre. Protokollene kan koble sammen enheter på tvers av ulike nettverk, og brukes også i private nettverk til å overføre data. Den kan enkelt modifiseres og kan brukes sammen med alle operativsystemer, maskinvarer og nettverk. I TCP/IP modellen så snakkes det i hovedsak om 4 lag som er beskrevet under:

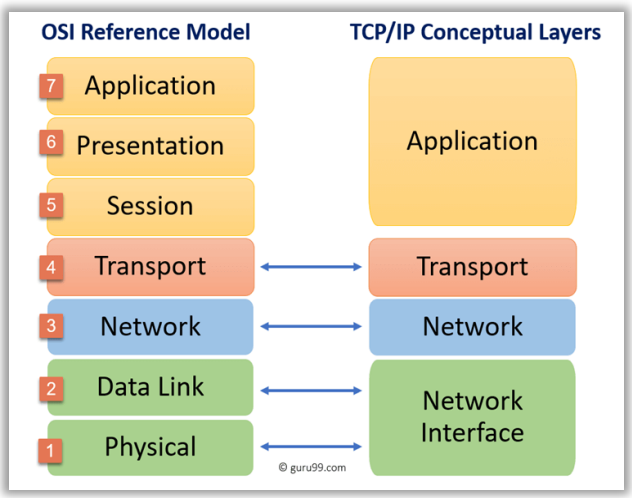
- **Applikasjonslaget** er en kombinasjon av sesjonslaget, presentasjonslaget og applikasjonslaget i OSI-modellen (kap. 2.2.1). Det er dette laget som programvarer kjører på og laget har ansvar for å standardisere den dataen som skal overføres. Her er det flere protokoller som brukes, noen av disse er Simple Network Management Protocol (SNMP) (kap. 2.5.4) som brukes til å overvåke og administrere enheter og TWAMP (kap. 2.6) for å måle mellom to enheter i nettverket. Laget har også ansvaret for kryptering.

- I **transportlaget** opprettes det forbindelse for ende-til-ende kommunikasjon over nettverket. Laget bruker TCP for mer pålitelig overføring og UDP for raskere overføring. Laget håndterer også fragmentering av pakker (kap. 2.2.6).

- **Nettverkslaget** i TCP/IP modellen vil tilsvare nettverkslaget i OSI-modellen. Den viktigste protokollen på dette laget er IP og denne protokollen vil blant annet rute og adressere pakker på internett. Det finnes også andre protokoller som kjører på dette laget

som for eksempel Internet Control Message Protocol (ICMP) (kap. 2.5.2) som sender feilmeldinger og IPSec som krypterer datapakker.

- **Nettverksgrensesnitt** har samme funksjon som datalinklaget og det fysiske laget i OSI-modellen. Laget jobber på link og vil koble sammen enhetene i ett nettverk. Laget vil bruke Media-Access Control MAC-adresser (kap. 2.2.3) til å adressere pakkene.




*Figur 2. 3: Sammenhengen mellom OSI-modellen og TCP/IP-modellen. [2]*

TCP/IP modellen er satt sammen av flere protokoller, de to viktigste i denne sammenheng er TCP og IP. TCP er en forbindelsesorientert protokoll, som lager kommunikasjonskanaler som pakkene kan bli sendt over. TCP deler også opp informasjon i mindre pakker før de sendes og vil sørge for å sette pakkene sammen på mottakersiden. IP bestemmer hvordan pakkene skal adresseres og hvordan de skal sendes for å komme frem til riktig sted. Hver ruter pakken passerer vil sjekke IP-adressen for å avgjøre hvor pakken skal sendes videre. [13, 14]

I målingene som gjennomføres på nettverket vil det i hovedsak være UDP som er benyttet og ikke TCP. Dette kommer av at jitter og pakketap bestemmer kvaliteten på TCP. TCP er ikke egnet som en egen målingsprotokoll, da den blant annet har opphopningskontroll i forbindelse med TCP-vinduet Dette gjør at den vil rette opp eventuelle problemer som oppstår under sendingen av pakker. Derfor må pakketapsmålinger og måling av jitter i hovedsak gjøres via UDP.

### 2.2.3 Datalinklaget

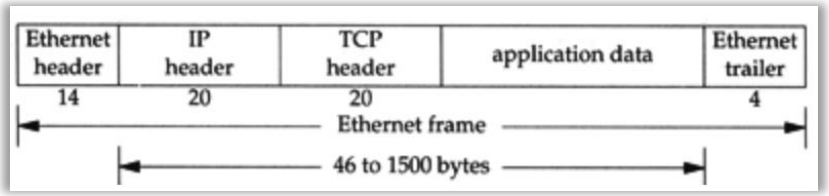
Datalinklaget sørger for at overføringen av data på det fysiske laget skjer tilnærmet feilfritt (kap 2.2.1). En av oppgavene til datalinklaget er å overføre informasjon mellom enheter i nettverket. Datalinklaget kan motta pakker fra nettverkslaget og fra det fysiske laget. Når datalinklaget mottar pakker fra nettverkslaget, vil det legge på en header med fysisk adresse til mottaker og avsender og sende pakken videre på det fysiske laget. Pakken vil også inneholde informasjon om når alle deler av en pakke er mottatt. Når datalinklaget mottar en pakke fra det fysiske laget vil denne pakken bli sendt videre til nettverkslaget dersom pakken ikke har noen feil eller mangler. Media-Access Control (MAC) er ett underlag av datalinklaget og vil bruke MAC-adresser til å definere destinasjonen til en pakke. Et annet underlag av datalinklaget står for å overføre pakkene på en sikker måte. Da må ofte pakkene fragmenteres og markeres. Dersom en av delene går tapt, sendes de på ny.

Ett Local Area Network (LAN) er en sammenkobling av enheter på en bestemt lokasjon. Ved å ha ett LAN kan enheter i ett nettverk dele filer og nødvendige dokumenter med hverandre over en nettverkstilkobling. Når pakker skal flyttes internt i et nettverk, flyttes de via det fysiske laget og datalinklaget. Pakkene kan forflyttes seg i Virtual Local Area Network (VLAN) og via svitsjer. Dersom det brukes VLAN kan nettverksenheter kommunisere på tvers av geografiske områder. I datalinklaget kan pakker i hovedsak sendes på samme nettverket. Skal pakkene sendes ut av nettverket krever det mer komplekse handlinger enn hva som kan gjøres på datalinklaget. Pakkene vil da overføres til nettverkslaget (kap. 2.2.4), hvor pakkene kan overføres mellom ulike nettverk. [15-18]

### 2.2.4 Nettverkslaget

Nettverkslaget tar ansvaret for pakker som skal sendes ut av nettverket. Når en pakke forlater nettverket, gå den ut via Wide Area Network (WAN) porten. WAN kobler sammen flere LAN som befinner seg på ulike lokasjoner. Et eksempel på hvordan pakker sendes ut av nettverket er hvis man bruker internett og sender en forespørsel til en server. For å komme til riktig server må pakken gå gjennom flere rutere for å overføre pakken i riktig retning. For at nettverkslaget skal kunne sende pakken riktig brukes IP-adresser. Pakkene har både avsender og mottakeradresse, slik at mottaker vet hvor eventuelle svar skal bli sendt i retur. [19]

Nettverklaget kan håndtere store pakker, men dersom pakkene blir for store vil de bli fragmentert. Ulike nettverk kan håndere ulik størrelse og dette vil avhenge av hvor god overføringskapasitet det er der pakken skal sendes. En standard som blir mye brukt er at en Ethernet-ramme kan ha en Maximum Transmission Unit (MTU) opp mot 1500 bytes (fig 2.4). Dette er inkludert 20 bit er satt av til IP-headeren og dersom det brukes TCP har den en header på 20 bytes og brukes UDP har den en header på 8 bytes. Dette gjør at det kan overføres informasjon (payload) med en maks størrelse på 1460-1472 bytes avhengig om det sendes med UDP eller TCP. Her vil Ethernet headeren og Ethernet trailern være i tillegg til pakkestørrelsen på 1500 bytes. [20]



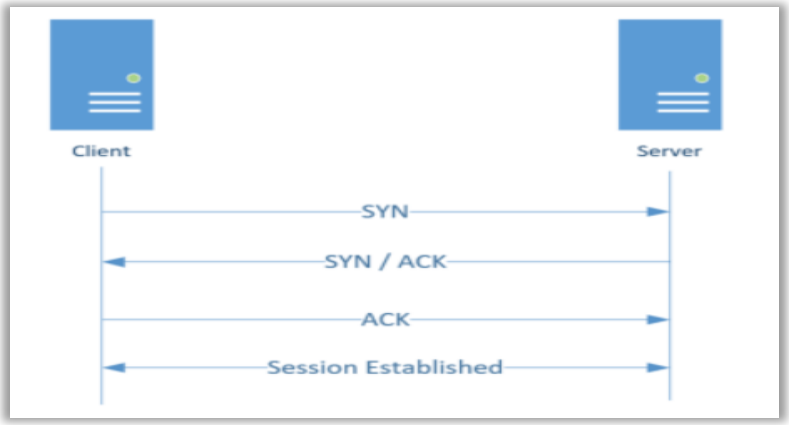

*Figur 2. 4: Illustrasjon som viser header fra de ulike lagene [3].*

I datalinklaget brukes ofte svitsjer til å sende pakker i samme nettverk, for at pakkene skal kunne sendes ut av dette nettverket vil det sendes til rutere på nettverkslaget. I nettverkslaget vil pakkene rutes i riktig retning i forhold til destinasjonen til pakken. Det er ingen definert måte å kunne dokumentere pakketap i mellom lagene, men destinasjonen vil kunne sette sammen fragmentene tilhørende en pakke og deretter oppdage om det mangler noen fragmenter. Siden det ikke er en definert måte å oppdage hvor ett pakketap oppstår mellom lagene er det en større viktigheten når det kommer til gode målemetoder og nettverksovervåkning. Det er via overvåkning av de ulike linkene i ett nettverk at det er mulig å definere hvor pakketapet oppstår for å kunne gjøre utbedringer av nettverket.

### 2.2.5 Transmission Control Protocol (TCP) vindu

En TCP-sesjon er den linken som opprettes mellom to enheter, eller to endepunkter i ett nettverk. Det oppstår en sesjon mellom endepunktene ved at endepunktene utfører det som kalles en «handshake». Vi kan se ett eksempel på denne opprettelsen av en TCP-forbindelse mellom en server og en klient i figuren (fig. 2.5). Her vil en klient etterspørre en sesjon mot en server ved å sende en Synchronize (SYN)-forespørsel, serveren vil da svare med en SYN og en Acknowledgement (ACK) tilbake, klienten vil til slutt svare serveren med en ny ACK. Dette

markerer at sesjonen er etablert. Ett TCP-vindu er derimot det antallet pakker som kan sendes før sender får en bekreftelse på at en av pakkene er mottatt. Dette vinduet vil bidra til en dataflyt-kontroll på kommunikasjonslinken. [21]




*Figur 2. 5: TCP-vindu som opprettes ved en «handshake» [4]*

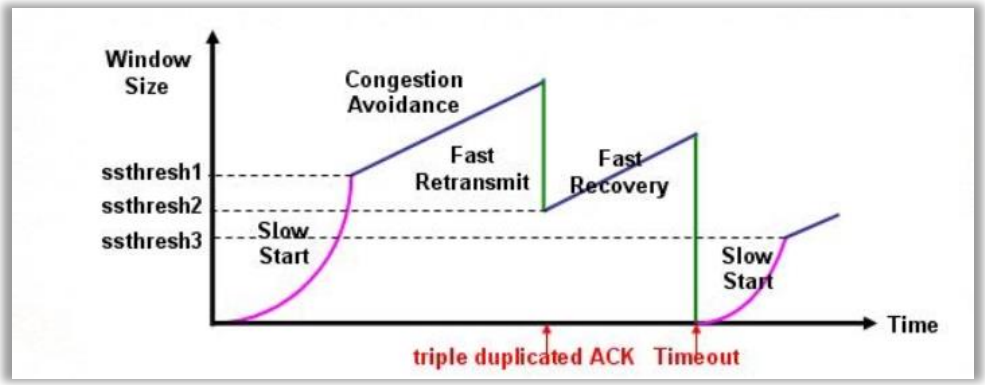
Ved sending av pakker brukes gjerne TCP, da denne protokollen er sikrere en UDP. Pakkene vil være pakket inn med en header, en TCP-header. Noen av de viktigste punktene på headeren er kilde port, som definerer senderens portnummer og destinasjonsport som er mottakerens portnummer. Videre har vi sekvensnummeret, som indikerer datamengden som blir sendt i en sesjon, og ACK nummer hvor mottaker vil spørre neste segment. En annen viktig verdi vi finner i headeren er en verdi som spesifiserer antall bytes som kan bli sendt eller mottatt. [22]

### Opphopningskontroll

Opphopningskontroll er viktig å ha på utstyret pakker sendes over. Dette er en kontroll som er med på å forhindre eventuelle pakketap eller timeout av pakker. Opphopningskontroll består av en algoritme som skal forhindre at pakker blir opphopet på enheter som mottar pakker raskere enn den klarer å sende dem. Når det oppstår opphopning blir det mer forsinkelse, noe som vil redusere ytelsen. Opphopningskontroll er en viktig del av TCP-vinduet og TCP-vinduet vil eksponentielt sende pakker med større og større antall mulige pakker, frem til den oppdager at ikke alle pakkene får et svar på at de har kommet frem. Da vil pakker sendes på nytt eller vinduet halveres, noe som vil minske muligheten for opphopning av pakker. Opphopningskontroll er altså en konsekvens av TCP-vindu.

Når det snakkes om opphopningskontroll, består denne av 4 algoritmer, «slow start», «congestion avoidance», «fast retransmit» og «fast recovery». Slow start og congestion

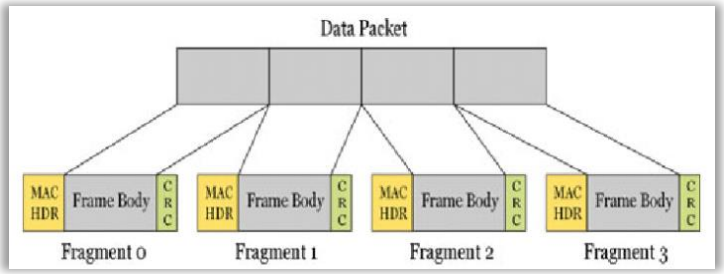
avoidance vil være implementert hos TCP senderen, slik at de kan kontrollere hvor mye data som sendes på utgående link. For at senderen skal måle disse vil to nye variabler legges til ved sending og mottak av pakker. Fast retransmit og fast recovery vil tas i bruk dersom en TCP mottaker sender en ACK når den mottar segmenter som ikke er i korrekt rekkefølge. Det vil detekteres eventuelle feil og hvorfor de har oppstått hos senderen og mottar senderen 3 ACK-meldinger innen kort tid vil den anta at dette segmentet er tapt og sende segmentet på nytt. Når de oppdager at en pakke er tapt vil TCP-vinduet sende pakken på nytt, fast retransmitt, for deretter å halvere vinduet og starte med fast recovery. Dette kan bli sett på figuren nedenfor (fig. 2.6). [23]




*Figur 2. 6: Illustrasjon av hvordan opphopningskontroll fungerer. [5]*

### 2.2.6 Pakke fragmentering

Til nå har vi sett på litt generelle protokoller og oppbygning av nettverk i forhold til OSI-modellen. For at informasjon skal kunne sendes over nettet, må den deles opp i mindre biter (fragmenter) (fig.2.7), dette kalles fragmentering. Dette er fordi nettverket har en begrensning i hvor store pakker eller hvor mye informasjon som kan sendes samtidig. Hvert fragment får en header som inneholder informasjon om hvem som er sender og mottaker av pakken, hvor høyt pakken skal prioriteters i sendingen, hvor lang den er og hvor lang levetid den har. Hvis tillatt pakkestørrelse til ulike linker er forskjellige eller hvis mer informasjon puttes inn i en datapakke, så må pakken deles opp i mindre biter.




*Figur 2. 7: Eksempel på hvordan en pakke kan bli fragmentert. [15]*

Siden et fragment kan adresseres vil den bli rutet igjennom nettverket via linker med minst mulig trafikk, for å redusere unødvendig forsinkelser for fragmentene. For å ha kontroll på alle fragmentene som hører sammen, markeres de med hvor mange pakker det er totalt og hvilket nummer i rekken de er. De har også noen bits på slutten som forteller mottaker at alle fragmentene er mottatt. Fragmentene blir sendt ut individuelt, slik at hver av dem kan velge den beste veien selv. Dette gjøres for å utnytte nettverket på best mulig måte. På mottakersiden må alle fragmentene settes sammen igjen i riktig rekkefølge. Om noen av fragmentene har forsvunnet underveis, kan mottaker be om å få sendt dem på nytt. Dette fører til at det tar lengre tid for hele beskjed å komme frem, og det må sendes ekstra trafikk over nettet. [24-26]

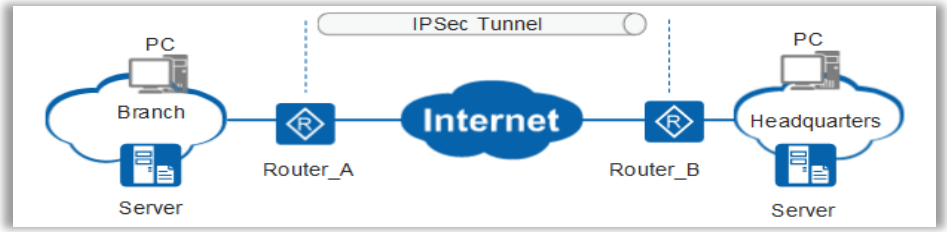
### 2.2.7 Network Address Translation (NAT)

I dette underkapittelet skal det bli sett litt mer på hvordan NAT fungerer. NAT er utviklet og tatt i bruk da det er ett begrenset antall IPv4-adresser. En NAT er en tabell som oversetter mellom nettverksadresser. Det vil derfor være mulig å ha ett privat nettverk innenfor en offentlig adresse og NAT vil da oversette mellom privat og offentlig adresse. Dette vil gjøre ende-til-ende måling vanskelig å gjennomføre, da det kan være mange endepunkter innenfor den offentlige adressen. En NAT tillater brukere med interne IP-adresser tilgang til internett. Det skal nå bli sett på ett eksempel for hvordan det vil fungere med sammenhengen mellom NAT, interne og offentlige IP-adresser. En enhet i nettverket vil sende forespørsel mot internett. Rutere i ett nettverk vil da reagere på dette og sende en forespørsel videre til nettverkets brannmur. Brannmuren vil gå innom NAT og gjøre om den interne adressen til den offentlige adressen som brannmuren bruker. Når brannmuren mottar svar på sin forespørsel, vil den benytte seg av NAT for å avgjøre hvilken enhet som skal motta dette svaret. Så når brannmuren mottar svar, vil den sjekke opp i NAT hvor denne forespørselen kom fra, og videresende resultatet dit. For enheten i nettverket vil dette oppfattes som en direkte link, men alle enhetene

i det private nettverket har lik offentlig adresse. Som beskrevet ovenfor legger en NAT til et ekstra ledd i nettverkskommunikasjonen noe som vil lede til mer forsinkelser når pakker skal sendes. [27, 28]

### 2.2.8 Internet Protocol Security (IPSec)

Det kan ved flere anledninger være ønskelig med en sikker tilkobling over internett, Internet Protocol Security (IPSec) tilbyr dette ved bruk av ett sett med protokoller. IPSec er en av de mest brukte overføringsmetodene for sikker overføring over internett i dag. Det vil gi mulighet til å kryptere og legge ved autentisering på pakkene som blir sendt. IPSec brukes for å sette opp Virtual Private Network (VPN). Illustrasjonene (fig. 2.8) viser at det blir satt opp en IPSec tunell for å opprette VPN mellom to rutere over det åpne internett. Den sørger for at uvedkommende ikke skal kunne få innsyn i dataen og dette gjør det mulig å dele konfidensiell informasjon uten at den kommer på avveie.



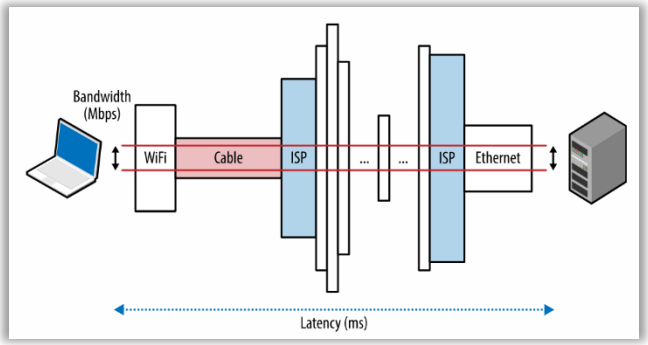

*Figur 2. 8: Illustrasjon av ett generelt oppsett av en IPSec-tunell [6]*

For å bruke VPN må både sender og mottaker ha samme krypteringsnøkler. Ved å bruke samme nøkler, kan meldingen låses hos avsenderen før den blir send og låses opp og leses hos mottaker. Hver av pakkene blir også markert med autentisering, slik at mottaker vet at pakken kommer fra en trygg avsender. Det at pakkene gjennomgår denne prosessen, fører til ytterligere forsinkelser i nettverket. [29]

IPSec vil lage en tunell som gjør at det vil være mulig å komme på innsiden av en NAT og det vil gjøre det mulig å gjennomføre ende-til-endemålinger. Ved å bruke en tunell vil det være mulig å måle helt til sluttbruker uten at det må gjennom flere ledd eller være usikkert hvilket endepunkt målingen skal gjøres mot. For at det skal være mulig å ha en god overføring ved bruk av IPSec er kapasiteten til overføringsmediet en stor faktor. Ved bruk av IPSec vil pakkene kreve større kapasitet og er derfor nødvendig med god nok båndbredde (kap. 2.2.9).

### 2.2.9 Bitrate, båndbredde og hastighet

Bitrate, båndbredde og hastighet er faktorer som henger tett sammen for å avgjøre hvor stor overføringskapasitet det er i nettverket. Bitrate er hvor mye data som kan overføres fra ett punkt i nettverket til et annet punkt på en bestemt tid og blir målt i bit per sekund (bps). Hvor stor overføringskapasitet det er påvirkes direkte av båndbredden og hastigheten dataen blir overført med. Båndbredden sier noe om hvor mange bits som kan overføres samtidig og hastigheten forteller hvor fort informasjon kan sendes inn på transmisjonsmediet. Hvor mye av overføringskapasiteten som faktisk kan brukes, avhenger av det punktet i nettverket med dårligst kapasitet. Dette punktet i nettverket kalles ofte en flaskehals. Figuren under (fig. 2.9) viser i dette tilfelle at selv om det trådløse nettet har bedre kapasitet, begrenser det kablede nettverket overføringen. Hvis det ønskes bedre nett her, er det tilkoblingen mellom Wi-Fi og internettleverandøren som må forbedres. Er overføringskapasiteten lav i forhold til mengde data som skal overføres, kan forsinkelsen til pakkene skape problemer for opplevd kvalitet på nettverket. [30, 31]




*Figur 2. 9: Illustrasjon som viser sammenheng mellom bitrate og forsinkelse. [7]*

### 2.2.10 Forsinkelse av pakker

Tidligere i kapittelet har vi sett på NAT, IPSec og pakkefragmentering, dette er alle faktorer som er med på å kunne gi forsinkelse på en pakke. En annen faktor som også kan introdusere forsinkelse er TCP-vindu (kap. 2.2.5) dette kommer av at hastigheten kan bli ujevn siden antallet pakker som blir sendt vil bli doblet frem til en pakke ikke kommer frem. Da vil antall pakker som sendes halveres. Hvor mange pakker som kan være under sending, uten å bli bekreftet mottatt bestemmes av sender eller mottaker avhengig av hvem av dem som kan sende eller motta minst. For å unngå pakketap kan det legges på litt forsinkelse ved sendingen av

pakkene, noe som også kalles trafikkforming. Dette gjøres for at avsender skal rekke å få bekreftelse på at pakkene er mottatt og derfor ikke trenger å begrense TCP-vinduet. Noe som igjen fører til jevnere overføring og bedre opplevd kvalitet på nettverket. Begrensningen på hvor mye som kan sendes gitt av størrelsen på TCP-vinduet er:

$$Gjennomstrømning = \frac{TCP\ vindu}{RTT} \quad (1)$$

Hvor stor gjennomstrømning det er vil være et direkte resultat av størrelsen på TCP-vinduet dividert med Round Trip Time (RTT) eller tiden det tar fra pakken blir sendt, til avsender får melding om at den er mottatt. Ved å innføre mer forsinkelse på pakkene i form av trafikkforming vil det sørge for at overføringen ikke overgår kapasiteten til linken. Båndbredden vil utnyttes bedre og pakkene blir totalt sett overført raskere. Det at pakkene sendes med jevne mellomrom fører til at en unngår "sagtannformet" overføring, noe som vil være med på å redusere pakketap. [32]

Når en pakke skal sendes mellom to enheter, er det flere andre ting som er med på å lage forsinkelser underveis. Det vil være en forplantingsforsinkelse der pakken bruker noe tid på å gå gjennom mediet. Hvor mye forsinkelse dette legger til, er avhengig av hvor langt pakken skal flytte seg og hvilket medium som blir brukt. Med fiber sendes det med tilnærmet lysets hastighet, mens overføring på kobber bruker lengre tid. Skulle en pakke sendes på en fiber langs ekvator uten å gå gjennom noen rutere eller møtt på andre faktorer som kan forsinke signalet, ville det brukt 200 ms RTT. For å beskrive forplantingsforsinkelsen kan det gjøres med følgende formel:

$$Forplantningsforsinkelse = \frac{D}{S} \quad (2)$$

Hvor D står for avstanden mellom mottaker og sender, og S står for forplantingshastigheten i mediet.

Det kan også oppstå en økt forsinkelse under overføringen av dataen. Her skal hver "bit dyttes" på kabelen. Hvor mye forsinkelse dette medfører påvirkes av dataraten til linken og pakkens lengde. Denne forsinkelsen kan beskrives med følgende formel:

$$Overføringsforsinkelse = \frac{L}{bps} \quad (3)$$

Hvor L er antall bits i en pakke og bps er antall bits per sekund som linken har kapasitet til å overføre.

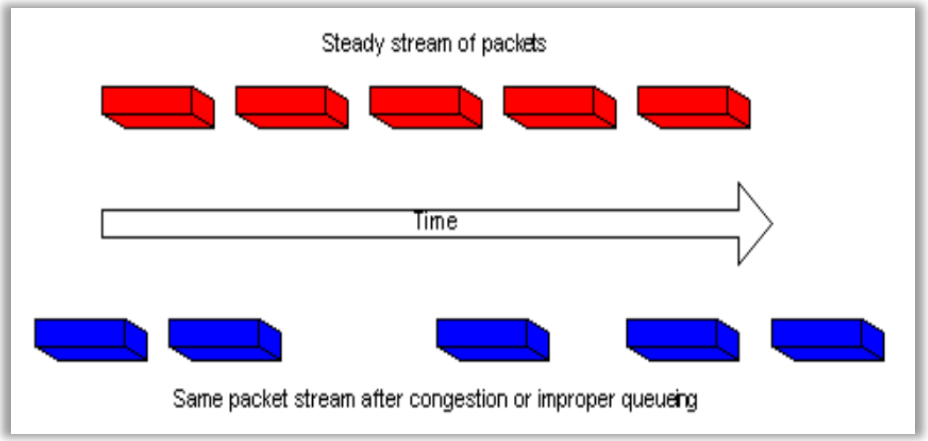
Eksempel:
Hvis en har en 10 Mb fil som skal sendes over en 1 Mbps linje, vil det ta 10 sekunder og sende pakken. Har en derimot 100 Mbps linje, vil det ta 0,1 sekunder å sende samme fil.

Bearbeidelsesforsinkelse oppstår dersom det er mange pakker som skal bli behandlet av en enhet og pakkene må vente for å bli overført. For hver ruter som pakken møter, må headeren leses og ruteren må finne ut hvor den skal sende pakken videre. Noen steder blir det også sett etter bitfeil.

Det er ønskelig med lavest mulig forsinkelse for å få en mest mulig sømløs opplevelse av nettverket. Ved å se på hvordan hver av disse delene kan forbedres, vil forsinkelsen kunne reduseres. Hvis det brukes fiber til å overføre signaler vil forplantningsforsinkelsen være minimal. Hvis en enhet har for mange pakker å håndtere vil det oppstå køer. Om enhetene rundt merker dette, kan de redusere hastigheten pakkene blir sendt med. Hver enhet pakken går igjennom må «pakke opp» pakken og se hvor den skal videresendes, svitsjer gjør dette mer effektivt enn rutere siden den operere på et lavere lag. Det som er den største faktoren i forsinkelsen er hvor langt pakkene skal sendes, fordi flere av disse forsinkelsene kan oppstå flere ganger. [33, 34]

### 2.2.11 Jitter

Jitter er endringen av forsinkelse over en periode og det kan være flere årsaker til dette. Ett eksempel på dette er om pakkene tar forskjellig rute i nettet, da vil pakkene kunne komme frem på forskjellig tid. En annen årsak kan være om en ruter har «brukt opp kvoten» over hvor mange pakker den har lov til å sende i en gitt tid. Ruteren vil da måtte mellomlagre resten av pakken til den får lov å sende mer. Som et resultat av dette, kan strømmen av pakker bli ujevn, noe som fører til variabel forsinkelse (fig. 2.10). Dette fører til at kvaliteten på internettleveransen oppleves dårlig.




*Figur 2. 10: Illustrasjon av jitter etter forsinkelser på grunn av kø. [8]*

Det er ønskelig med lavest mulig jitter. Om alle pakker som blir sendt fra A til B bruker like lang tid, vil det ikke oppleves noe jitter. Hvor mye sluttbrukeren vil merke av endringer i forsinkelse, er avhengig av hvilken type tjeneste de bruker på internett. Når det blir sett på film er det ikke like viktig med lite forsinkelse, da pakkene kan lastes ned før de skal brukes. Det er i motsetning viktig med lite forsinkelse om det gjennomføres en videosamtale. En måte å redusere jitter på, er å tilpasse hastigheten pakkene sendes med til ruten pakken skal sendes via. Store pakker kan også skape jitter og for å redusere dette deles pakkene opp i mindre pakker slik at de passer bedre til den tilgjengelige linkhastigheten. En siste faktor som kan bidra til en reduksjon av jitter vil være å prioritere viktige pakker i overføringen. [35-37]

## 2.3 Tjenestekvalitet og tjenesteprioritering

Nå har det blitt sett på pakketap, jitter og andre forhold som kan være med på å redusere den opplevde kvaliteten på nettverket. For å redusere konsekvensene av dette, kan det legges ulike prioriteringer på hvordan nettverksutstyr skal håndtere de ulike pakkene. I dette kapittelet blir det sett nærmere på hvordan Quality of Service (QoS) (kap. 2.3.1) og kundeopplevd forsinkelse (kap. 2.3.2). Til slutt vil det bli sett på nettnøytralitet (kap. 2.3.3) og hvorfor det er viktig å bevare dette ved prioritering av pakkene.

### 2.3.1 Quality of Service (QoS)

For å redusere forsinkelser, jitter og pakketap, brukes QoS til å kontrollere pakkene som blir sendt på nettverket. Da blir pakker som er mer kritisk at kommer frem, prioritert i overføringen. En måte å gjøre dette på er ved å lage flere parallelle køer ved rutere og svitsjer, der hver av køene har ulik prioritet. Da kan for eksempel Real-time Transport Protocol (RTP)-pakker som blant annet brukes i videokonferanser, bli prioritert da en forsinkelse på disse pakkene vil redusere den opplevde kvaliteten betraktelig. Mens om et dokument bruker litt lengre tid på å åpnes eller en e-post venter litt lengre med å bli sendt, ikke vil gå ut over brukeropplevelsen. Når pakkene blir gitt prioritet, blir de markert med denne informasjonen.

IP-pakker kan ha en byte med Types of Service (ToS) som rutere kan bruke for å se hvor høyt pakken skal prioriteres i køen. De første 3 bits i ToS forteller hvilken prioritering pakken har. Hvis noen pakker må kastes fordi køene blir for lange, blir de pakkene med lavest prioritet kastet først. De neste 4 bits brukes for å markere hvilken ToS pakken skal ha. Det kan være å ha minst mulig forsinkelse, høyest mulig gjennomstrømming, størst mulig pålitelighet eller velge korteste mulig vei. Det siste bit kalt Must Be Zero (MBZ) skal være 0. Etter hvert var det ønskelig å kunne dele pakker inn i flere prioriteringer enn ToS hadde mulighet til. Hele strukturen til ToS-bytet ble endret, og ble kalt Differentiated Services Field (DS field). I DS field brukes de 6 første bits til å velge hvilken prioritet pakken skal ha. Ved å bruke 6 bit gir det muligheten til å bruke opp til 64 forskjellige nivåer av prioritet. Dette kalles Differentiated Services Codepoint (DSCP), de to siste bits er ikke i bruk. Fordelen med å bruke DS er at da kan nettverksutstyret konfigureres til å sortere trafikken ut fra prioriteringen pakken har, istedenfor å måtte konfigurere alle svitsjene pakken skal gjennom til å prioritere pakker sendt på en gitt port eller til en gitt mottaker. [38, 39]

En enhet på datalinklaget har ikke tilgang til DSCP da dette ligger i IP-headeren på nettverkslaget (kap. 2.2.4). Standarden for å kunne tilby QoS på datalinklaget ligger under 802.1Q. Her er det 3 bit som ligger i Ethernet-rammens headeren og blir kalt Priority Code Point (PCP). PCP gir mulighet til 8 prioriteringer, der konfigurasjon av nettverket har høyest prioritet. Det at PCP ligger i headeren på ethernettrammen gjør at når pakken kommer til en ruter, vil denne informasjonen forsvinne i oppakningen. Hvis det er ønskelig at pakken skal ha samme prioritering på tvers av flere nettverk, må denne informasjonen lagres i ToS på nettverkslaget. [40]

### 2.3.2 Kundeopplevd forsinkelse

Det er flere ting som gjør at det oppleves forsinkelser fra kundens perspektiv. Ofte vil kundeopplevd forsinkelse oppstå dersom en enhet er overbelastet. Er enheten overbelastet vil den ikke kunne håndtere pakker best mulig og det vil være nødvendig å kunne finne ut hva som er årsaken dette. En årsak til kundeopplevd forsinkelse kan være at det er for mange enheter tilkoblet ruteren og at det sendes for mye trafikk på denne. Dette vil gjøre at ruteren kan ha problemer med å videresende pakker, og pakker kan bli droppet eller bli liggende i kø. En ruter som er overbelastet kan ha behov for å bli oppgradert til å kunne håndtere trafikkmengden. Det vil også kunne oppleves forsinkelse for kunden dersom båndbredden til kunden ikke er tilstrekkelig for de behovene kunden har. [41]

Quality of Experience (QoE) er en måte å måle opplevd kvaltitet sett fra sluttbrukerens ståsted. Der de fleste målemetoder av nettverkskvalitet ser på overføringen, ser QoE på hele prosessen fra dataen blir generert til den blir presentert for mottaker. I noen tilfeller kan det være mer pakketap, uten at det vil merkes. Men i andre tilfeller vil selv litt pakketap ødelegge for hvordan kvaliteten oppleves. Hvis det tar litt lenger tid å laste ned noe fra nett, vil det ikke være merkbart. Er videosamtalen hakkete, eller en får avvist en autentisering fordi det er for mye forsinkelser, påvirker dette sluttbrukeren i større grad. For å måle QoE kombineres passive og aktive målinger (kap. 2.5.1). [42]

### 2.3.3 Nettnøytralitet

Nettnøytralitet bygger på ett prinsipp hvor ingen leverandører av innhold skal prioriteres over andre. Begrepet brukes fortrinnsvis innen bredbåndstilgang. Det skal sørge for at alle innholdsleverandører har like god båndbredde på leveransen sin og prioriteres på likt grunnlag. I nettverket som er brukt i prosjektet vil det ikke være prioritering av noen leverandører over andre. Dette vil blant annet tilsi at hver enkelt yter av bredbåndet skal ha lik tilgang til nettverket og bredbåndet. I forhold til at all trafikk skal prioriteres på en liknende måte, slik at hver kunde har samme opplevelse og kvalitet. Abonnenten skal ha mulighet til å bruke nettverkstilgangen til det de ønsker, og ikke være usikker på om det er en begrensning på kvaliteten til deres ønskede innholdsleverandør. For at nettverket skal være nøytralt, er det viktig at QoS blir satt til å prioritere pakker ut fra type trafikk, og ikke ut fra hvem avsender er. [43]

## 2.4 Teori rundt årsaker til pakketap på fiber

I dette kapittelet blir det sett på hvordan fiberoptikk er bygd opp og hvordan signaler transmitteres over en fiber til fordel fra andre medium. Dette kommer av at fiber er det mediet som det skal måles hastigheter og pakketap over. Det vil her bli sett på hvordan fiberen er bygd opp og hvordan fiber benytter seg av laser for å sende signalet.

### 2.4.1 Optikk

For å kunne forstå hvordan signaler forplanter seg i fiber, er kunnskap om optikk en viktig. Optikk handler om hvordan lys oppfører seg og hvordan lys og materiale påvirker hverandre. Opprinnelig ble optikk bare brukt om lys som øyet kan oppfatte. Senere har også ultrafiolett- og infrarødt lys og mikrobølger også blitt definert som optikk. Lyset i en fiberkabel forplanter seg ved refleksjon i kjernen på kabelen. Dersom fiberkabelen får en for stor bøy, kan det føre til at signalet sendes ut av kabelen fremfor å bli reflektert, noe som ofte refereres til som brytning.

Når en stråle av lys treffer en jevn, reflekterende flate, vil strålen bli reflektert i samme vinkel som den traff flaten. Om en lysstråle går fra ett medium til ett annet, vil strålen endre retning i det nye mediet. Hvilken vinkel signalet vil få i det nye mediet er avhengig av refleksjonsindeksen materialet har. Denne formelen er kjent som Snell's lov. [44-46]

$$\frac{n_1}{n_2} = \frac{sin\alpha_2}{sin\alpha_1} \tag{4}$$

I formelen ovenfor er $n_1$ og $n_2$ refleksjonsindeksen til materialene som blir brukt som transportmediet. $\alpha_1$ er vinkelen lysstrålen treffer materialet på og $\alpha_2$ er vinkelen lysstrålen forflytter seg i det nye materialet. Siden $n_1$ og $n_2$ er konstanter for det mediet de representerer, må forholdet mellom sin $\alpha_2$ / sin $\alpha_1$ være konstant. Det betyr at så lenge det brukes samme bølgelengde og lysstrålen treffer det samme mediet med lik vinkel, vil refleksjonen være lik. [47]

Fysisk optikk kan bli sett på ulike måter. Det kan bli blant annet bli sett på som geometrisk optikk, bølgeoptikk og fiberoptikk, men det er ikke et skarpt skille mellom disse typene optikk. Geometrisk optikk ser på lys som en stråle og forklarer hvordan lysets brytning og refleksjon

er. Bølgeoptikk ser på lys som elektromagnetiske bølger og vil også kunne beskrive lysets brytning og refleksjon, men kan i tillegg forklare dispersjon (kap. 2.4.7), interferens (kap. 2.4.8) og spredning. Når lyset brukes til å sende informasjon over en fiberkabel, blir det omtalt som fiberoptikk. Fiberoptikk brukes i forbindelse med fiberteknologi for å overføre signaler med minst mulig risiko for støy.

### 2.4.2 Fiberteknologi

For overføringer over lange strekk har fiberoptikk erstattet kobber. Signalet kan forsterkes, men skal det sendes over svært lange avstander kan det være fordel å regenerere signalet. I dag brukes fiber i alt fra TV-signaler til nettverksoppbygning. Fiber har lang levetid og er vanskeligere for uvedkommende å avlytte siden det ikke skaper magnetfelt rundt kabelen, slik kobber kan gjøre.

For å sende signaler over fiber må de modulers fra elektroniske pulser til lyspulser. Signalet kan enten bli sendt med Light Emitting Diode (LED) eller laser som blir skrudd av og på. Laser er dyrere en LED, men gjør at signaler kan bli overført en del lenger. Lyspulsen forflytter seg lett i den optiske fiberen og avstanden i fiberkjernen tillater signalet å forplante seg via refleksjoner. Det er viktig at kjernen i fiberen er så klar som mulig, slik at lyspulsen kan fortsette uten forstyrrelser. Kortere fibertråder kan være av plast, men i all hovedsak er kjernen laget av glass.

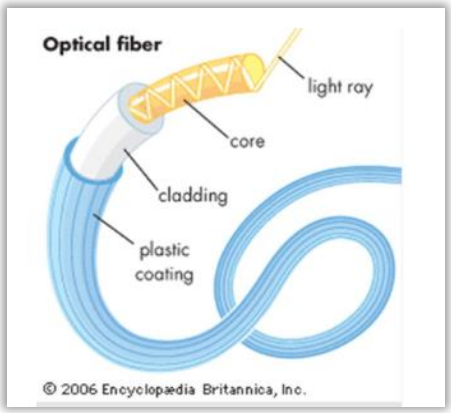
Signalet som sendes over fiber vil kunne bli dempet når det sendes. Sendes signalet over for lang avstand vil det være en fare for at signalet blir for dårlig. Signalet som sendes vil også bli påvirket om det er demping i skjøtene mellom fiberkablene. Dersom skjøten er dårlig vil det introdusert en type støy som påvirker hvor langt signalet kan sendes. Dette vil kunne resultere i at signalet endrer brytningsvinkelen og lyset kan forsvinne ut av fiberkabelen. Det kan også resultere i for mye refleksjon. For å oppdage dette kan fiberkabelen måles etter den er skjøtt og se om forventet demping tilsvarer målt demping. [48, 49]

For å kunne øke antall signaler som kan overføres på en fiber brukes det Wavelength Division Multiplexing (WDM). Da settes det sammen lysbølger med forskjellig bølgelengde. Det vil her være slik at en mottaker vil ha utstyr som klarer å skille disse bølgelengdene fra hverandre. WDM bruker to farger til multipleksingen, 1310 og 1550 nm. Fargene blir lagt sammen og sendt over fiberen som ett signal. Det kan også sendes flere signaler samtidig, men det avhenger

av hvor langt det er mellom de ulike kanalene. Hvis bølgelengdene ligger for nære hverandre, kan det være vanskeligere å skille dem på mottakersiden. [50, 51]

### 2.4.3 Fiberens oppbygning

Fiberen er bygget opp av flere lag for å beskytte kjernen der selve signalet går. Kjernen er bare en brøkdel av et hårstrå tykk, og trenger beskyttelse for lys utenfra, støt og støy som kan påvirke signalet.




*Figur 2. 11: Illustrasjon av hvordan fiber er bygd opp [9]*

For å unngå at lys skal reflekteres eller bli påvirket av utenforstående omstendigheter har fiber en unik oppbygning. Ved å ta utgangspunkt i figuren (fig. 2.11) vil det ytterste laget (plastic coating) være ett lag som gir beskyttelse fra ytre omgivelser. Den beskytter mot støt og bøyninger under montering, og mot støv som kan bidra til å forstyrre signalet underveis. En av flere fordeler med fiber er at siden de bruker lys til å overføre signalet unngår man at signalet «smitter over» på andre kabler i nærheten.

Det neste laget i figuren over, er det ytre reflekterende laget (cladding/kappen). Laget har en tykkelse som ett hårstrå (0,125 mm) og er oftest lagd av glass. Kappen har noe lavere refleksjonsgrad enn kjernen, dette for å reflektere signalet tilbake til kjernen.

Kjernen er det innerste laget i oppbygningen til fiber og i kjernen oppnår lysstrålen total indre refleksjon. Dette betyr at signalet kan forplante seg svært langt med lite endringer eller demping av signalet. Urenheter i glasset kan påvirke hvordan signalet forplanter seg i kabelen, noe som redusere drastisk avstanden signalet kan sendes. Hvis fiberkabelen blir bøyd for mye, kan

lysstrålen få for stor vinkel mot kappen at signalet blir reflektert ut av kjernen og signalet kan blir ødelagt. [49]

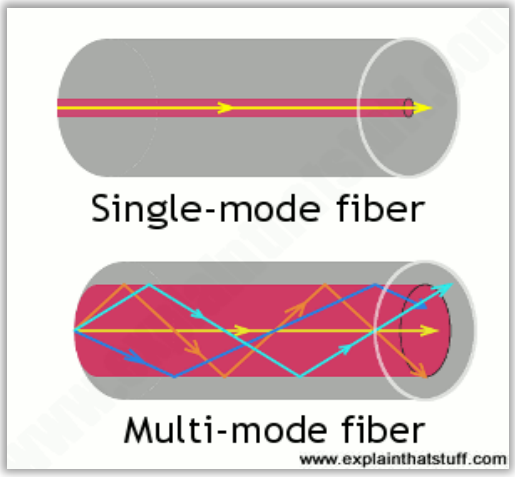
### 2.4.4 Laser

Signalet som overføres i fiberen, blir gjerne sendt med en laser. Laser er et akronym som står for «Light Amplification by Stimulated Emission of Radiation». Her betyr light amplification det å lage mer av det samme lyset med samme frekvens og fase, mens stimulated emmission er en fysisk prosess om hvordan lys blir produsert.

En laserstråle bruker en bestemt bølgelengde. Laserstrålen består av en enkel frekvens eller farge, og er koherent noe som vil si at bølgene ligger i fase med hverandre. Hvis det blir sendt signaler på stråler som ligger nær hverandre i frekvensspekteret, kan signalene overlappe hverandre, og gjøre det umulig å skille ut signalene på mottakersiden, også kalt interferens (kap. 2.4.8). Laserstrålen kan enten være en kontinuerlig lysstråle, eller en pulserende lysstråle. Innen fiberoptikk blir pulserende laserstråle brukt for å sende signaler. [52, 53]

### 2.4.5 Singelmodus og multimodus fiber

Når det kommer til fiberkommunikasjon er det flere forskjellige typer fiber, mest vanlig er singelmodus og multimodus fiber. En singelmodus fiber er som navnet tilsier en fiber som kun er designet for å bære ett lys og det kan forflytte seg i midten av fiberen uten å reflekteres i kabelen. Dette gjør at lyset får kortest mulig strekning å bevege seg på gjennom fiberen. Singlemodusen har liten diameter på kjernen, noe som gjør at lysstrålen som beveger seg ikke har noen forstyrrelser. Dette gir mindre sannsynlig for interferens mellom flere bølgelengder. Signalene sendes hovedsakelig på 1310 nm eller 1550 nm, og kan sendes 50 ganger avstanden sammenlignet med multimodus. Singelmodus er den metoden som har minst endringer av signalet og overføringshastigheten er større en andre overføringsmetoder med fiber. Figuren (fig. 2.12) viser hvordan lysstrålene beveger seg i singelmodus og i multimodus.




*Figur 2. 12: Forskjellen på singelmodus og multimodus fiber. [10]*

Multimodus fiber har en større diameter enn det singlemodus har. De ulike lysstrålene som beveger seg i denne har noe mindre presisjon, noe som gjør at det introduseres mer støy. Om det skal sendes lys over lange distanser med denne typen fiber, vil lyset kunne oppleve en større spredning. Dette fører til interferens og det vil kunne oppstå pakketap eller forsinkelser. I multimodusfiberen vil lyset bevege seg i flere vinklede ruter, hvor hvert av lyssignalene beveger seg i en bestemt rute utfra dens bølgelengde. [48, 54, 55]

### 2.4.6 Modulasjon

Modulasjon er å sette sammen et informasjonssignal sammen med en bærebølge som kan overføres på transmisjonsmediet. En bærebølge er ett signal eller en bølgeform som brukes til å transportere informasjonen. På mottakersiden blir informasjonen hentet ut fra det sammensatte signalet, denne prosessen kalles demodulasjon. Det finnes ulike måter og sette sammen informasjonssignalet og bærebølgen på. Om en har et analogt signal inn, kalles det en analog modulasjon. Er signalet digitalt, må en ha en digital modulasjon. Analoge signal blir i større grad enn digitale signaler påvirket av støy, og taper seg derfor over tid. Digitale signaler er derfor å foretrekke. Analoge signaler endrer seg gradvis hvor et eksempel på dette kan være en sinusbølge. Et digitalt signal vil endre seg stegvis for eksempel av eller på. For å sende signalet over fiberen vil informasjonssignalet bli modulert inn på bærebølgen. Bærebølgen er lysstrålen som sendes inn på fiberkabelen.

For å kunne omforme et analogt signal om til et digitalt, må det tas målinger av signalet for å se hvilken verdi det skal erstattes med. Dette gjøres ved bruk av samplingsraten. Samplingsraten

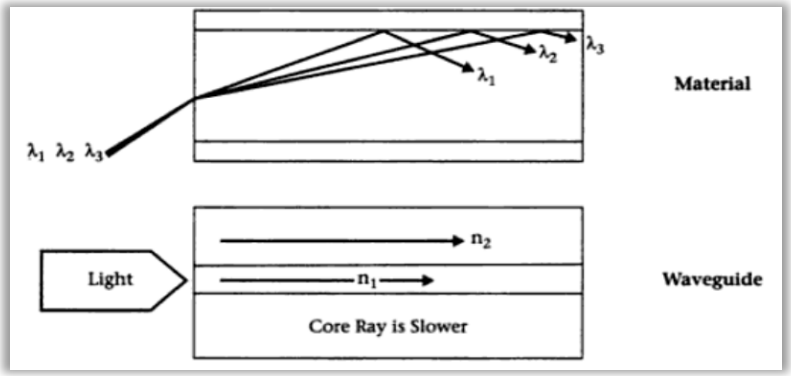
sier noe om hvor ofte det analoge signalet må sjekkes. For å være sikker på at signalet sjekkes ofte nok, må samplingsraten ($f_s$) være over to ganger høyere enn den høyeste frekvensen ($f_{max}$) som kan måles. [15]

Nyquist teoremet beskriver samplingsraten som:

$$f_s > 2 \cdot f_{max} \tag{5}$$

### 2.4.7 Bølgelengdedispersjon

Dispersjon innen optikk handler om lysstrålens spredning. Vi skal i dette kapittelet se på to typer dispersjon, material dispersjon og bølgelengdedispersjon. Material dispersjon er når en lysstråle som inneholder flere frekvenser treffer et materiale og de ulike frekvensene vil få litt forskjellig brytningsvinkel utfra hvilken bølgelengde de har. Da vil de ulike lysstrålene få litt forskjellig retning gjennom fiberkabelen og de vil bruke ulik tid på å komme frem til mottaker. I singelmodus fiber (kap. 2.4.5) er det bølgelengdedispersjon som påvirker signalet mest. Bølgelengdedispersjon er at deler av en lysstråle gå over i claddingen (kap. 2.4.3). Her vil signalet forflytte seg raskere enn i kjernen av fiberen på grunn av lavere brytningsindeks. Dette fører til at deler av signalet vil komme til mottakeren litt før resten av signalet. Hvis det sendes en puls, vil mottakersiden få en bredere puls en hva som ble sendt på grunn av denne forsinkelsen. Sendes signalet langt, vil bredden på pulsene øke og kunne overlappe hverandre. Da vil signalet bli vanskelig å tolke og det kan oppstår bitfeil. Illustrasjonen under (fig. 2.13) viser forskjellen på material og bølgelengde dispersjon. [54]




*Figur 2. 13: Forskjellen på bølgelengde og material dispersjon [11]*

Tidligere har det vært SFPene som avgjorde hvor langt et signal kan sendes, men ettersom overføringskapasiteten økes, vil signalene i større grad bli utsatt for dispersjon. Dispersjon vil

sette begrensninger på hvor langt signalet kan sendes. I WDM sendes det flere signaler over samme fiberkabel, dersom det er mange signaler i samme fiberkabel vil bølgelengden til disse ligge nært hverandre. For å begrense virkningene av dispersjon, må det være nok avstand mellom bølgelengder slik at signalene ikke sklir for mye inni hverandre. [55]

Dispersjon kan måles er ved å sammenligne støy og mottatt signal. Signal-støyforhold blir ofte beskrevet som Signal-to-Noise Ratio (SNR). Når det kommer til transmisjon av signaler så er det essensielt at det ikke er for mye støy på linja som signalet sendes over. Etter hvert som støymengden øker vil signalet rekke over kortere distanser, og det blir en høyere sannsynlighet for at signalet kan bli forvrengt eller at pakker blir tapt. Da vil sannsynligheten for at enkelte pakker ikke når frem til mottaker bli høyere, noe som vil lede til forsinkelser og dårligere opplevd hastighet og kvalitet for mottaker. SNR er generelt sett forholdet mellom signalstyrken og hvor mye støy det er. SNR verdien måles normalt i dB (Desibel). For å beregne SNR, er det mest vanlige å beregne ved bruk av effekt [watt], og spenning [volt]:

$$Beregning\ for\ effekt: SNR = 20 \cdot \log\left(\frac{S}{N}\right)\ [W] \tag{6}$$

$$Beregning\ for\ spenning: SNR = 10 \cdot \log\left(\frac{S}{N}\right)[V] \tag{7}$$

For å beregne støy blir det benyttet logaritmisk beregning ved bruk av log. Det blir her tatt logaritmen av signalstyrken (S) dividert med den støyen (N) som oppleves på signalet. Selv om det er lite støy ved overføring over fiber sammenlignet med andre metoder, kan det likevel være hensiktsmessig å beregne SNR for å vurdere hvor langt signalet kan sendes og om det bør forstekes underveis. [56, 57]

Som en konsekvens av støy kan det oppstå feil i overføringen. En mye brukt måte å se etter feil i overføringen, er cyclic redundancy check (CRC). Da legges det ved ekstra bits i overføringen. Det gjennomfører polynomdivisjon med dataen og kontrollbits. Om svaret ikke blir riktig, kan det sendes en forespørsel om å sende informasjonen på nytt. [58]

### 2.4.8 Støyfaktorer

**Støy** på et signal reduserer kvaliteten på signalet og kan i verste fall føre til at signalet ikke kan tolkes. Analoge signaler er mer utsatt for støy, og kvaliteten på signalet vil derfor reduseres gradvis. På digitale signaler skal mer til før kvaliteten påvirkes, men når det først har blitt introdusert for mye støy reduseres kvaliteten på signalet fort. Det er flere faktorer som kan skape forstyrringer på et signal, som etter hvert kan gjøre det umulig å hente ut det opprinnelige signalet. Fiber bli også bli påvirket av hvor mye støy det er på en strekning, for fiber vil dette redusere hvor lang signalet kan sendes.

**Forvrengning** er en annen støyfaktor som kan påvirke signalet. Det kan enten bli utsatt for lineære forvrenging, da vil signalet gradvis bli svakere etter som frekvensen øker/synker sammenlignet med originalsignalet. Eller så kan det bli utsatt for ikke-lineær forvrenging. Da kan det oppstå harmoniske frekvenser i tillegg til den originale frekvensen. Sendes mange frekvenser i samme kanal, kan det være vanskelig å skille harmoniske frekvenser fra signalfrekvenser. Forvrenging kan være et problem i multimodus fiber siden det sendes flere lysstråler inn på samme kabel, noe som gjør at signalene kan sendes betraktelig kortere enn i singelmodus. [59, 60]

**Interferens** skjer hvis to eller flere bølger påvirker hverandre. Har bølgene samme fase vil de legges sammen og amplituden dobles. Er bølgene i motfase vil de kansellere hverandre, og det vil oppstå en destruktiv interferens. Selv om det er lite sannsynlig at det oppstår interferens med lysbølger er det fortsatt mulig da lysbølgene på fiberen kan ligge tett. De kan da treffe på likt punkt og bølgene vil kunne doble eller utligne hverandre. [61]

## 2.5 Tradisjonelle teknikker for nettverks målinger

I dette kapittelet vil noen tradisjonelle målemetoder presenteres. Etter hvert som teknologien har utviklet seg har også behovet for målemetoder endret seg. I større nettverk er det mer sannsynlig at det kan oppstå pakketap og forsinkelser samtidig som det kan være vanskelig å avgjøre hvor i nettverket disse feilene oppstår. Det vil først bli sett på to ulike typer overvåkning, passiv og aktiv overvåkning (kap. 2.5.1). Det vil deretter bli presentert noen tradisjonelle målemetoder som blir brukt i prosjektet. Dette er ping (kap. 2.5.2), Iperf (kap.

2.5.3) og SNMP (kap. 2.5.4). Til slutt vil det bli sett på en begrensning som disse målemetodene har, som er muligheten til å oppdage microburst (kap. 2.5.5)

### 2.5.1 Passiv og aktiv overvåkning

Overvåkning av ett nettverk blir bare mer aktuelt etter hvert som teknologien utvikler seg. Nettverket blir overvåket for å kunne sikre kvaliteten på en nettverksleveranse. Det er flere parametere som overvåkes, noen av de vanligste er hvor mye av båndbredden som bli brukt, forsinkelse i nettverket, pakketap og QoS (kap. 2.3.1). Skulle det oppstå problemer eller bli for dårlig kapasitet på nettverket, kan overvåkning hjelpe til med å finne ut hvor feilen befinner seg og hvor utbedringer må gjøres for å øke eller bedre kvaliteten.

Ved passiv overvåkning er fokuset på overvåkningen av enhetene som er i nettverket. Her vil informasjonen om utstyret og linken til utstyret lagres og kan deretter hentes ut når det er ønskelig å sjekke om enheter i nettverket fungerer optimalt. Informasjon som kan bli lagret her vil blant annet kunne være benyttet båndbredde, hvilken trafikk som går igjennom enheten og eventuelt hvor lang forsinkelsestid det er på enheten. Informasjonen kan lagres i en database, og senere hentes ut data derfra. Denne metoden krever mye lagringsplass.

Ved aktiv overvåkning av ett nettverk vil det legges til ekstra trafikk på nettverket for å måle ønskede parametere. Den ekstra trafikken som legges på under målingen av nettverk gjør at vi kan måle parametere som forsinkelse, jitter og pakketap både fra sender til mottaker og mellom sender og mottaker (rundetiden). Trafikken som legges til skal etterligne vanlig trafikk på linjen, noe som gjør at målingene kan gjennomføres selv om det ikke går reell trafikk på nettverket under måleperioden. Målingen kan skje mellom alle målepunktene i nettverket, noe som gir oversikt over sanntidsmålinger over hele nettverket og kan oppdage feil med en gang de oppstår. [62]

### 2.5.2 Ping

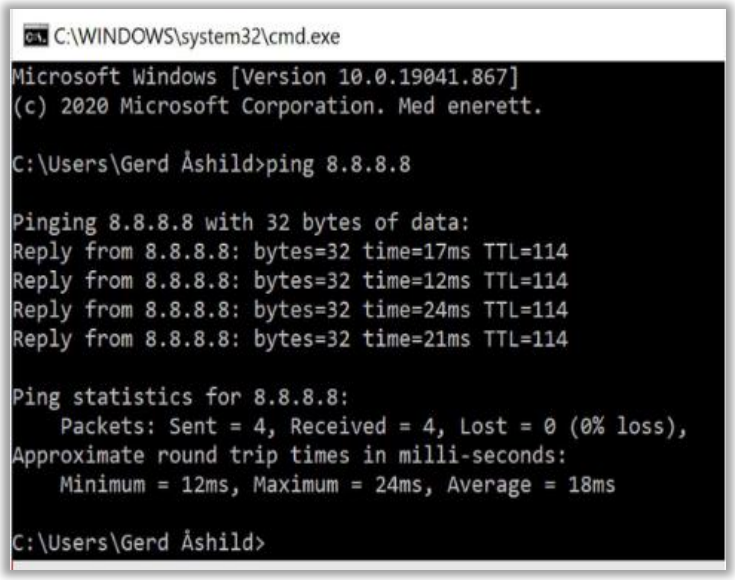
Ordet ping kommer opprinnelig fra ekkolodd som sendte ut en lydbølge og fikk lydbølgen i retur når den traff objekter. Så ser den på tiden lydbølgen brukte på å returnere for å avgjør avstand til objektet. Ping var navnet de brukte om lydbølgen. Litt på samme måte som ping blir

brukt i dag, der det sendes ut en forespørsel til utstyr på nettet, som kan svare på forespørselen om det er tilgjengelig.

Ping og forsinkelser er begreper som blir brukt mye om hverandre. Ping er egentlig applikasjonen som blir brukt til å sende ut en forespørsel som blir besvart, mens forsinkelsen er måleenheten som her er den gjennomsnittlige rundetiden gitt i millisekund. Forsinkelsen viser kvaliteten på nettverkstilkoblingen og har ikke noe med hastighet å gjøre, men sier noe om reaksjonstiden til nettverket.

Ping brukes primært til å sjekke om enheter er tilgjengelige eller for å teste rundetiden. Rundetiden er tiden fra forespørselen blir sendt til den blir besvart. Det brukes ICMP for å sende og besvare Ping-forespørselen. ICMP er en protokoll som også brukes til å sende melding om feil i nettverket. Det er ønskelig med lavest mulig pingtid, der under 30 millisekunder blir ansett som svært bra. For å pinge utstyr, brukes gjerne IP-adressen til enheten. ICMP og IP-adresser operere på nettverkslaget i OSI-modellen (kap. 2.2.1).

Ved å kjøre ping mot utstyr blir det også gitt en Time To Live (TTL) verdi. Dette sier noe om hvor mange rutere en pakke får lov til å passere før den kastes. Denne verdien bidrar til at feilsendte pakker ikke kan bli sendt rundt i nettverket i evig tid. Dette kunne skapt store problemer med unødvendig data på linjene. For hver ruter pakken møter, reduseres TTL-verdien med 1. Hvis pakken når 0, blir den kastet og en feilmelding blir sendt i retur. [35, 63]

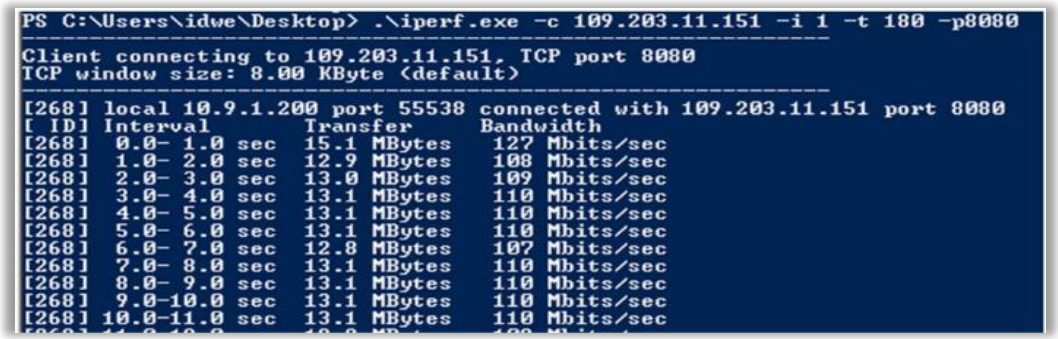



*Figur 2. 14: Eksempel på ping gjennomført i kommandovinduet.*

Som en demonstrasjon av hvordan ping fungerer, ble det i figur (fig. 2.14) sendt ut en ping-forespørsel til Google sin Domain Name System (DNS)-server ved hjelp av IP-adressen til serveren. Testen ble kjørt fra kommandolinjen som er innebygd i Windows. Figuren viser at den gjennomsnittlige pingtiden er på 18 ms, TTL er på 114 og at alle pakkene som ble sendt, kom frem.

### 2.5.3 Iperf

Den neste tradisjonelle målemetoden det skal bli sett på er Iperf. Denne metoden vil måle pakketap ved bruk av UDP og hastighet ved bruk av TCP. Målemetoden er basert på en klient-server-arkitektur og for at målingen skal kunne gjennomføres vil klienten kobles til serveren og det vil bli generert nettverkstrafikk. Trafikken som sendes gjør at klient og server kan måle hastighet eller pakketap. Under er det ett eksempel på hvordan denne målingen kan se ut når den blir kjørt (fig. 2.15), i dette tilfelle overvåkes hastigheten på linjen.




*Figur 2. 15: Eksempel på Iperf-måling gjennomført over nettverket til bedriften. (Godkjent målepunkt i forbindelse med personvern)*

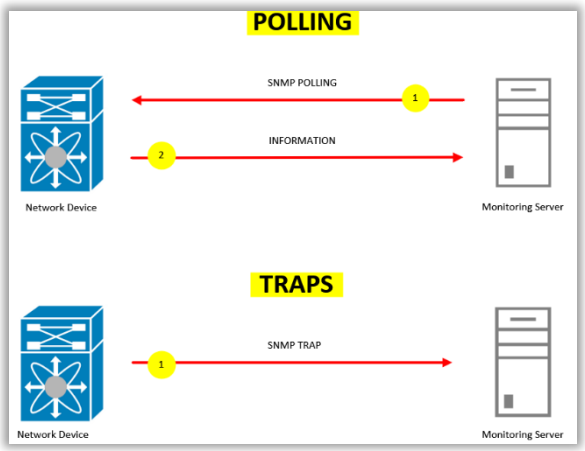
Noe som er spesielt med denne måleteknikken er at den tradisjonelt sett kun måler når en person manuelt initierer målingen fra en klient og mot en server. Dette gjør at teknikken i hovedsak vil gi øyeblikksbilder, og ikke gi en like stor mulighet i forbindelse med å gå tilbake å kunne se hvordan linken så ut en bestemt tid i forveien. Det vil alltid være nødvendig å kjøre en test i det øyeblikket det er ønskelig å se ett resultat.

For å gjennomføre denne typen måling, vil det være behov for å laste ned applikasjonen for måling med Iperf. Dette ligger tilgjengelig på internett og vil bli forklart i design og implementasjon (kap. 4.4.2).

### 2.5.4 Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) er en av de mer tradisjonelle målemetodene som brukes til å måle nettverkskvaliteten. SNMP er en protokoll som er definert på applikasjonslaget i OSI-modellen (kap. 2.2.1). Protokollen brukes både til å overvåke og konfigurer nettverksenheter. For å overvåke nettverket bruker SNMP stort sett UDP. SNMP er en enkel form for overvåkning da den har en ukomplisert oppbygning og kan brukes av de fleste aktører. Den er mer hensiktsmessig å implementere for større aktører, da det gir dem mulighet til å overvåke alle SNMP-enhetene i nettverket på samme plattform.

Protokollen har en klient-server-arkitektur hvor klientene har mulighet til å rapportere informasjon om minnebruk, prosessorkraft og hvor mye båndbredde som benyttes av nettverksenhetene. Hovedkomponentene som blir brukt i denne protokollen er SNMP agenter og SNMP managere. Enhetene som blir overvåket av SNMP kan være servere, rutere, svitsjer og brannmurer for å nevne noen. På disse enhetene er det en programvare, kalt SNMP agent, som kjører. Denne samler kontinuerlig inn data om ulike parametere tilhørende CPUen og hvor mye båndbredde som blir brukt. SNMP agenten sender denne informasjonen til SNMP manageren. SNMP kan settes opp på to forskjellige måter. Enten kan manageren sende ut forespørsel om informasjon som agentene svarer på, ellers kan agentene sende informasjon til manageren om det skulle skje noen endringer med den.



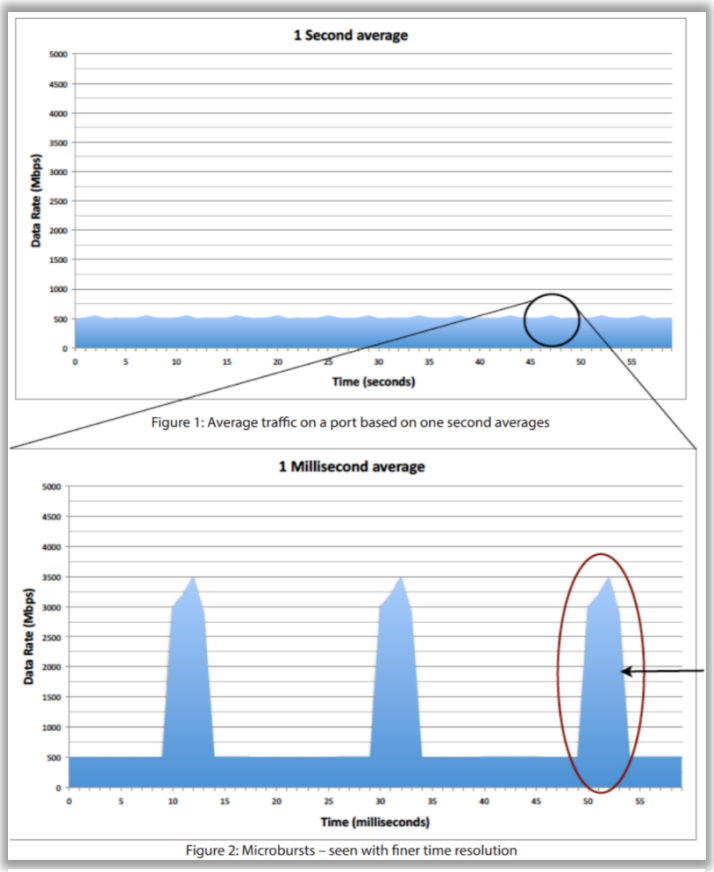

*Figur 2. 16: Illustrasjon av hovedfunksjonene med SNMP [12]*

Figuren (fig. 2.16) viser prinsippet bak hvordan de to forskjellige metodene fungerer. I den øverste metoden sender SNMP manageren ut en forespørsel som SNMP agenten svarer på. Her vil manageren med jevne mellomrom få tilbakemelding om enhetene som overvåkes. Skulle

det skje noe med en enhet, vil manageren ikke få vite om dette før neste gang den sender ut en forespørsel. Denne metoden kalles polling. Den andre metoden som brukes med SNMP er traps. Her sender SNMP agenten en traps-melding uten en forespørsel fra manageren. Den beste måten å overvåke nettverket på, er å kombinere de to metodene. Det vil i hovedsak brukes polling ved gjennomføringen av målinger i dette prosjektet. [64, 65]

### 2.5.5 Microburst

Målinger som er gjort over en periode, kanskje på 5 minutter eller ned mot ett sekund kan vise at det er en jevn trafikkflyt. Hadde målingene vært på noen mikrosekunder kan det være at trafikkflyten ville sett annerledes ut. Illustrasjonen under (fig. 2.17) viser først målinger per sekund, der omtrent 500 Mbps av båndbredden blir benyttet. Under er samme måling med høyere oppløsning, her med målinger per millisekund. Da viser grafen at i korte perioder sendes det 3,5 Gbps i en periode på 4,5 ms, disse målingene syntes ikke i utsnittet per sekund. Årsaken til slike plutslige hopp i benyttet båndbredde kan være at flere enheter er koblet til en server, og at hver av dem får en periode å sende og motta data på. For eksempel hvis enheten har mye den ønsker å overføre, vil den overføre mest mulig i det tidsrommet den får tildelt. Dette gjør at linken blir fult belastet i ett lite øyeblikk og vi får microburst. Dersom mye data blir sendt i ett kort tidsrom kan det føre til at enheter får for mange pakker å håndtere samtidig, noe som vil kunne gi pakketap og økt jitter. Hvis ett nettverk har mye microburst, vil det gå ut over den opplevde nettverkskvaliteten uten at det kan oppdages av målemetoder som benytter gjennomsnittsmålinger slik som for eksempel SNMP. [66]




*Figur 2. 17: Figuren viser et bilde av hvordan microburst i ett nettverk vil se ut [13]*

## 2.6 Two-Way Active Measurement Protocol (TWAMP)

I forrige kapittel ble det sett på mer tradisjonelle målemetoder som fortsatt er mye brukt i dagens nettverk. Disse metodene er ikke lengre komplekse nok for dagens nettverk og vil ikke kunne gi gode nok målinger for å dokumentere kvaliteten på nettverket. Dette er bakgrunnen for at TWAMP ble opprettet og blir tatt i bruk i stadig flere nettverk. I dette kapittelet skal det bli sett på de tjenestene som TWAMP åpner for og hvordan denne protokollen fungerer.

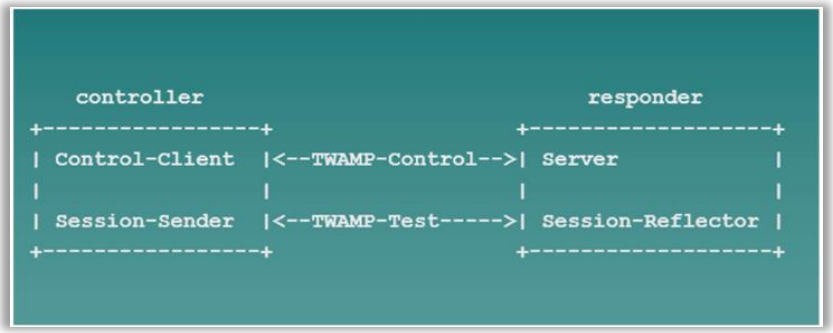
### 2.6.1 Introduksjon til TWAMP

TWAMP er en utvidelse av One-Way Active Measurment Protocol (OWAMP), som kun har mulighet til å måle en-veis forsinkelser og pakketap. TWAMP er en protokoll som åpner for aktiv overvåkning ved å måle ytelsen til nettverket mellom to enheter som begge må støtte protokollen. TW i TWAMP står for «two way» som vil si at denne protokollen har toveis kommunikasjon. Med TWAMP er det mulig å måle trafikken som går begge veier, og brukes ofte som et fjernmålingspunkt i ett nettverk. Her vil det ikke være nødvendig med synkronisering mellom målepunktene, noe som gjør det til en nyttig teknikk for å måle linker. I TWAMP vil ett målepunkt være en sender og det andre målepunktet være en reflektor og datastrømmen som brukes til måling vil sendes mellom disse. TWAMP er en nyere måleteknikk, noe som fører til at resultater og publikasjoner omkring denne protokollen er noe begrenset sammenlignet med for eksempel ping. [67]

TWAMP består av flere logiske roller. De logiske rollene er en kontroll-klient og en server som kjører TWAMP-kontrolleren mellom seg. Den består også av en Sesjons-sender og en Sesjon-reflektor som kjører TWAMP-testen mellom seg. En illustrasjon av hvordan dette fungerer kan sees i figuren (fig. 2.18). Kontroll-klienten og sesjons-senderen kalles gjerne en kontroller, og serveren og sesjonsreflektoren kalles en «responder». Når det kommer til hvordan protokollen fungerer i praksis så vil kontroll-klienten ha ansvar for å initiere og avslutte kommunikasjonen mellom kontroller og responder. Dette vil bli gjort ved å benytte TCP-forbindelser, som det har blitt sett på i kapittelet om TCP-vindu (kap. 2.2.5). Videre vil serveren som svare med en ACK på kontroll-klienten sin initiering av en TCP-forbindelse. Dette vil være det som i hovedsak

trengs for å kunne starte flyten av trafikk mellom server og klient. TWAMP vil bestå av to hovedprosesser, dette er TWAMP-kontroll og TWAMP-test. TWAMP-kontrollen har ansvaret for å initiere og starte TWAMP-sesjoner. TWAMP-test ha ansvaret for å sende testpakker for å teste linken mellom sesjonssenderen og reflektoren. [68]

Klienten vil være den som starter og stopper sesjoner. Denne kan inneholde sesjons-senderen som da lager pakkene som skal sendes til reflektoren, men senderen kan også være separat fra klienten. Serveren håndterer hver sesjon sammen med klienten og lytter etter beskjeder over TCP fra klienten. Reflektoren kan enten være separat, eller så kan den være en del av serveren. Den sender pakker med svar på testpakkene sendt av senderen. [69]




*Figur 2. 18: Generelt oppsett av TWAMP-protokollen. [14]*

TWAMP kan også fungere ved at kontrolleren (control-client) i figuren (fig. 2.18) trekkes ut som en sentral enhet, gjerne kalt en orkestrator. Dette gjør det mulig å ha en sentral server som kan ha oversikt over alle sesjoner som kjører og alle enheter som er implementert som en del av nettverket. Denne teknikken er en videreutvikling av hvordan TWAMP kan fungere i større skala enn gjennomføring av tester i korte øyeblikk.

### 2.6.2 TWAMP-kontroll og TWAMP-test

I dette delkapittelet skal vi gå litt mer inn på de hovedprosessene som finnes i TWAMP, TWAMP-kontroll og TWAMP-test. TWAMP-kontroll har ansvaret for å initiere og starte TWAMP-sesjonene slik at målinger kan bli foretatt. Før målingene kan gjøres må noen oppgaver løses under oppsettet av sesjonen. Det vil opprettes en sesjon mellom klienten og serveren på port 862. Ved opprettelsen av denne forbindelsen vil serveren respondere med en melding som inneholder nødvendig informasjon for opprettelse av forbindelsen. Denne vil blant annet inneholde ønsket autentiseringsmodus, hvor modus 1 for uautentisert, 2 for autentisert og

4 for kryptert. Det vil så oppstå en «set-up-response» melding, denne vil blant annet inneholde moduset som klienten ønsker i forbindelse med autentisering. Om klienten da har valgt modus 4 vil denne meldingen også inneholde nøkkelen til krypteringen. Når forbindelsen så er opprettet og begge endepunktene, klienten og serveren, er klare til å sende data, vil det bli sendt en «request» om å ha en to-veis-sesjon mellom dem. Dette vil foregå på en tilnærmet lik måte som en vanlig opprettelse av TCP-forbindelse.

Videre er det TWAMP-test som er ansvarlig for å overføre pakker som skal teste linken mellom to TWAMP enheter i nettverket. Her vil sesjons-reflektoren overføre testpakker til sesjons-senderen utfra hvor mange pakker den mottar. Noe som er unikt med denne måleteknikken er at både senderen og reflektoren i sesjonen er ansvarlig for at tidsstempelet på pakken er så nøyaktig som mulig. Når en reflektor mottar en pakke fra en sender må denne kopiere tidsstempelet som var på pakken og sekvensnummeret. Dette er fordi mye av målingene går ut på å se på hvor lang tid en pakke bruker på å bli sendt begge veier over en link. Og da brukes spesielt tidsstempelet til å avgjøre hvor lang tid sendingen av pakken tar for å kunne illustrere dette i plattformen som brukes for målingen. Da vil det bli sett på forsinkelse for den enkelte pakken, det vil bli sett på det maksimale antallet pakker som kan sendes og hvor lang tid det tar for mottak av disse pakkene. Dette vil igjen kunne gi indikasjoner på om det er en flaskehals eller ett punkt i nettet med mindre kapasitet enn resten av nettverket. [70]

### 2.6.3 Tjenester TWAMP åpner for

TWAMP åpner for nye måter å måle nettverk på. Der hvor tradisjonelle målemetoder kan måle fra server til klient A og server til klient B, kan TWAMP måle trafikken som går mellom A og B siden alle målepunktene kan settes opp til å gjøre målinger mot hverandre. Denne formen for måling gir et mye bedre bilde av hvordan kunden opplever nettverket og gir bedre dokumentasjonsgrunnlag for bedriften. Det å ha muligheten til å sette opp varsler på om deler av nettverket får lavere ytelse gir det internettleverandøren mulighet til å fikse problemer før de blir rapportert inn.

For å oppnå en kommunikasjon mellom reflektoren og sender så vil det være nødvendig å sende kontrollmeldinger. TWAMP vil tillegg bruke tidsstempelet for å kunne få sikrere og mer nøyaktige resultater enn hvis den skulle prøvd å synkronisere tiden med reflektoren.

TWAMP åpner for muligheten til å overvåke hvor mange pakker og Byte som blir sendt, og hvor mange som blir mottatt. Duplikater blir også telt, noe som gjør det er teoretisk mulig at flere pakker kommer frem en det som var sendt. Det blir målt både hvor mange pakker og bytes som blir sendt fra kontrollenheten til reflektoren, og hvor mange som sendes i retur.

Det kan også bli gjort målinger på hvor mange pakker som forsvinner. Pakkens maksimale og minimale størrelser kan også registreres. Det er flere andre ting som måles i tillegg til dette. Blant annet om pakken kommer frem senere en den skulle eller om pakkene har byttet rekkefølge. Hvis pakkene er forsinket, kan det være at de blir registrert på neste måling i forhold til hvor de opprinnelig hørte til. Skulle dette skje, blir pakken registrert som tapt i den måleperioden den tilhørte og tooLate i påfølgende måleperiode. Hvis en pakke blir merket med tooLate vil det si at antall pakker mistet i forrige måleperiode er for høyt da denne pakken kom frem en måleperiode senere enn den skulle. Det blir også registrert om pakken kommer frem korrupt, hvor mye av båndbredden som er i bruk og hvor mye båndbredde som er ledig. [67]

TWAMP gir muligheten til å teste forsinkelse på flere måter enn ordinære testmetoder. Den kan bruke blant annet Inter-Pakcet Delay Variation (IPDV) og Packet Delay Variation (PDV). IPDV sammenligner forsinkelsen på pakken ut fra forsinkelsen på forrige pakke. Ved hjelp av formelen under kan forsinkelsen til pakke i beregnet, der D(i) er forsinkelsen til sist ankommet pakke og D(i-1) er pakken som kom før.

$$IPDV(i) = D(i) - (Di - 1) \tag{8}$$

For å få forsinkelse en vei, brukes tidsstempel på når pakken er mottatt (M) minus når pakken er sendt (S). Ved å si at D(2) = M(2) - S(2) kan det utledes:

$$IPDV(2) = D(2) - D(1) = (M2 - S2) - (M1 - S1) = (M2 - M1) - (T2 - T1) \tag{9}$$

IPDV sier noe om hvor bra nettverket er til å holde jevn avstand mellom pakkene. Ofte vil snittverdien av IPDV være lik 0.

PDV er et mål på hvor mye forsinkelse det er i nettverket. Målingene blir gjort på enkeltpakker. Som vist i formelen under er det differansen mellom den pakken med minst forsinkelse D(min) og gjeldene pakke D(i). Verdien på dette kan være null eller positivt.

$$PDV(i) = D(i) - D(min) \tag{10}$$

PDV verdiene vil bli bedre etter at målingene har pågått en stund, fordi D(min) da vil være den laveste verdien. PDV kan ofte omtales som jitter (kap. 2.2.11), men det blir litt upresist da PDV beskriver variasjonene i forsinkelsen på pakker en vei. Jitter handler også om variasjon av forsinkelse på pakker, men er ikke konkret definert noe sted og betydningen kan variere avhengig av hvem som benytter begrepet. [67, 71]

### 2.6.4 Nettverksparametere som kan overvåkes

TWAMP-målinger kan gjøres over IP-nettverk mellom to enheter som støtter TWAMP-protokollen. Målingene kan gjøres både over trådløst og kablet nettverk. Det blir mer nøyaktige målinger sammenlignet med ICMP (Ping) og UDP Echo, og målingene vil ha høyere presisjon en tidligere målemetoder kan tilby. TWAMP kan også måle QoS (kap. 2.3.1), noe som stadig blir viktigere da kunden stiller større krav til kvalitet, blant annet fordi videomøter blir mer vanlig. TWAMP kan måle både forsinkelse en vei og rundetiden. Den er heller ikke låst til en spesifikk leverandør av utstyr for å kunne gjennomføre målingene, noe som gjør den mer fleksibel.

TWAMP kan måle mellom to endepunkter og kan gjennomføre både enveis og toveis måling. Enveismålinger krever at mottaker er synkronisert med avsender for at målingene skal bli riktige. Måles rundetiden er det ikke et krav om synkronisering. Det kan bli gjort målinger om forsinkelser, båndbredde og pakketap. TWAMP åpner for målinger som gir et mer fullstendig bilde av hvordan nettverket egentlig er og gir en oversikt over trafikken på nettverket, nettverksytelse og applikasjonsytelse. For å få bedre oversikt over målingene kan det brukes en orkestrator, som kan kjøre målinger mellom TWAMP-enhetene og presentere måleresultatene for hele nettverket i samme interface.  Dataen som blir innsamlet kan presenteres grafisk. I tester hvor det er sammenlignet måleresultater fra ICMP og TWAMP, blir resultatet for TWAMP mer likt den opplevde forsinkelsen på nettverket da TCP og UDP pakker blir prioritert høyere i overføringen. [70, 72]

## 2.7 Small Form-factor Pluggable (SFP)

I dette kapittelet blir det sett på hva en Small Form-factor Pluggable (SFP) er (kap. 2.7.1), hvordan de er bygd opp (kap. 2.7.2) og hva en smart SFP er (kap. 2.7.3). En SFP er en enhet som har vært på markedet i lang tid og den vil være leddet mellom nettverksutstyr og en innkommende kabel. Etter hvert som teknologien har utviklet seg har det i senere tid kommet en enhet som kalles smart SFP. Smart SFPer er enheter som kan kjøre egen software, noe som åpner for helt nye måter å overvåke ett nettverk på.

### 2.7.1 Hva er en SFP?

En SFP er en modul som både kan sende og motta ett signal. Denne kalles ofte en SFP transiver og vi inneholde en del som sender signalet og en del som mottar signalet. En SFP som brukes til fiber vil sende optiske signaler ved bruk av en laser. Det er lite støy og forsinkelser forbundet med denne overføringsmetoden. SFPer er fleksible i den forstand at de støtter flere former for kommunikasjonsstandart. SFPene er like i den enden som kobles til utstyret, men ulike modeller har ulike tilkoblingspunkter. Dette gjør at de kan brukes med kobber eller ulike typer fiberkontakter. En SFP har en liten størrelse, noe som gjør den lett å håndtere og plassere i en del utstyr. Den kan da for eksempel kobles til en svitsj, en mediakonverter eller en ruter.

En SFP vil kunne plasseres i ulikt utstyr, dette kan for eksempel være en svitsj i ett nettverk. I denne svitsjen vil det ofte være flere SFPer som sender og mottar signaler i forskjellige retninger. Hvis en SFP er ødelagt eller det skulle være ønskelig å oppgradere denne, vil det kunne gjøres uten å ta ned utstyret den står i. Dette kalles «hot-swappable», og er noe som gjør teknologien lønnsom for en bedrift. Det reduserer nedetiden som det potensielt kunne vært på utstyret.

Noen SFP benytter single-mode fiber som sender på 1310 nm og 1550 nm. Disse har tynnere kjerne og kan sende signaler opp mot 2 km til 12 km. Andre benytter multimode fiber, som har større kjerne og billigere optikk. De kan sende opp til 500 m. [73]

### 2.7.2 Oppbygning av en SFP

Vi skal nå se på oppbygningen av en SFP, og hva de ofte består av. En SFP er et innebygd system, noe som vil si at det er et hardware-system som har programvare innebygd. Den programvaren som systemet har, er vanligvis spesifikt for det enkelte systemet. SFPen har både prosessor og minne, noe som gjør at systemet alltid vil fungere som en fullstendig enhet. Data vil bli sendt og mottatt via kommunikasjonsportene.

Under kan vi se en figur (fig. 2.19) av hvordan en SFP ser ut innvendig. Denne SFPen er en type som Eidsiva Bredbånd bruker i sitt nettverk. Det er en Gigabit SFP som sender på 1550nm og mottar på bølgelengden 1310nm. Denne ville som oftest ha en SFP som sender og mottar på motsatt bølgelengder i motsatt ende av linken. [74]


*Figur 2. 19: figuren viser hvordan en SFP ser ut innvendig.*

### 2.7.3 Smart SFP

Etter hvert som teknologien har utviklet seg, har også de tradisjonelle SFPene utviklet seg. Det er nå mulig å legge inn konfigurasjon og programmere de SFPene som brukes i fibernettverk. Disse SFPene kalles smart SFPer og det kan kjøres egen software på dem. Denne softwaren kan ofte brukes i forbindelse med overvåkning av nettverket. SFPene brukt i dette prosjektet vil være smart SFPer og softwaren som kjører på smart SFPen vil være en TWAMP agent. Ved å koble SFPene til enheter i nettverket vil det være mulig å overvåke og se etter forsinkelser og pakketap på linjene. Det gjør at hastigheten som går på en linje kan måles mer nøyaktig enn det som har vært mulig tidligere. Ved tilgang på smart SFPer har det åpnet seg mange muligheter for kontinuerlig overvåkning av en link på en måte som tidligere har vært mer kompleks.

Når det kommer til smart SFPer brukes ofte Field-Programmable Gate Array (FPGA). En FPGA kan bli konfigurert av brukeren etter den er blitt produsert. Dette gjør at de bedriftene som benytter seg av disse, kan konfigurere dem til deres formål. FPGAen samarbeider og kommuniserer med de andre delene i SFPen, blant annet en Cental Processing Unit (CPU). Når det kommer til FPGA, så er denne teknologien under utvikling hele tiden, noe som gjør at kapasiteten fortsetter å øke, og den fysiske størrelsen blir mindre. Dette gjør teknikken mer håndterlig og gjør den får plass i en SFP. Det at FPGAene kan omprogrammeres gjør det lettere å kunne flytte på SFPene om behovet for hvor målinger skal gjøres skulle endre seg. En FPGA består av mange logiske blokker som kan programmeres. Den inneholder ett lookup table (LUT) som har en logisk kobling mellom utgangs- og inngangssignaler. [75, 76]

# 3 Metode

I forrige kapittel ble det sett på den teorien som trengs for å kunne forstå det som kommer videre i prosjektet. Det skal nå bli sett på hvordan metode som er brukt for å løse oppgaven og komme frem til en konklusjon (kap. 3.1). Videre vil det bli sett på metode for litteratursøk (kap. 3.2) og benyttet dataverktøy og fysisk utstyr (kap. 3.3). Til slutt vil det bli sett på etikk rundt oppgaven (kap. 3.4) og begrensninger gitt av korona (kap. 3.5).

## 3.1 Metodikk

Metoden som kommer til å bli brukt for å gjennomføre prosjektet består av flere deler. Først vil vi gjennomføre ett møte med oppdragsgiver for å gå igjennom det som skal gjøres og den vinklingen oppdragsgiver ønsket at oppgaven skal ha. Dette for å vite at oppgaven som skrives treffer oppdragsgivers intensjon og ønske med oppgaven. Med bakgrunn i møtet med oppdragsgiver ønsker vi å sette en problemstilling som skal løses gjennom perioden det arbeides med oppgaven. Underveis i prosessen vil vi konstant gå tilbake og vurdere om problemstillingen er relevant eller om det har dukket opp faktorer som gjør at vinklingen i problemstillingen bør endres.

Etter hvert som det formelle rundt oppgaven kommer på plass vil det bli begynt et litteraturstudium rundt nødvendig teori samtidig som vi vil sette opp en testlab for å kunne gjennomføre målinger. Dette for å se teorien i en praktisk sammenheng. Et laboppsettet vil så bli utformet og implementert og det vil bli kjørt tester over dette for å få resultater som kan vurderes og evalueres. Utfra resultatene vil det vurderes om laboppsettet bør endres for å kunne få bedre måleresultater å sammenligne. Deretter vil resultatene på det endelige laboppsettet bli evaluert før det på bakgrunn av dette vil bli gjort en konklusjon. Fremgangsmåten vil bli dokumentert på en slik måte at oppsett og resultater vil kunne gjenskapes av andre.

Det ble gitt ganske frie rammer på hvordan vi skulle gjennomføre oppgaven, men det var et ønske fra bedriftens side å benytte Accedian til å gjennomføre TWAMP-målingene. Testlaben vil trolig bli mer kompleks underveis i prosjektet etter hvert som gjennomføring av målinger blir gjort, da det kan resultere i for få støyfaktorer på det første utkastet av laboppsettet. Det har

også blitt gjort vurderinger på hvordan målingene kan gjennomføres for å få like resultater på de ulike metodene. Underveis vil det mest sannsynlig være noe prøving og feiling, og vi vil arbeide strukturert for å finne løsninger på utfordringene som dukker opp.

## 3.2 Litteraturstudium

For å innhente informasjon for å kunne gjøre oppgaven, ble det utført litteratursøk på ulike plattformer. I begynnelsen brukte vi Oria for å søke i digitale bibliotek, her ble søkeord som «TWAMP» og «network» brukt for å lese oss opp på hvordan protokollen benyttes i praksis. Det ble også utført søk med andre søkeord, og lest abstrakt fra andre artikler, men de ble vurdert som ikke relevante til vår oppgave. Vi fant også bøker fra universitetsbiblioteket for å lese oss opp om protokollen.

Underveis brukte vi digitale leksikon som Store Norske Leksikon og Britannica for å lese oss mer opp på enkelttema som viste seg å være relevante. Vi i tillegg benyttet oss av dokumenter vi har fått tilsendt av leverandører, samt sider fra produsenter av tilsvarende utstyr som vi bruker. Dette for å lære mer om hva komponentene er og hvordan de brukes. Vi har også brukt ressurser som andre universitet har tilgjengeliggjort. Om søkeresultatene har blitt funnet er tvetydige eller mangelfulle, har vi arbeidet med å finne RFC dokumenter. RFC dokumenter beskriver blant annet protokoller og prosedyrer for hvordan detaljer rundt internett skal fungere.

## 3.3 Benyttede dataverktøy og fysisk utstyr

Under prosjektet ble benyttet ulike dataverktøy og nettverksutstyr. Mesteparten av programvarene som blir brukt er tjenester Eidsiva Bredbånd har lisens på og nettverksutstyr som de har liggende. Det ble også brukt open source programvare der det var hensiktsmessig.

### 3.3.1 SolarWinds Orion

SolarWinds Orion: 2017.3.5 SP5, WPM 2.2.1, SRM 6.6.0

Orion er plattformen Eidsiva bredbånd bruk til SNMP måling, i tillegg benytter plattformen ping til å gjennomføre noen av målingene. Ping mot kundens svitsj eller ruter brukes ofte for å

definere eventuelle forsinkelser og pakketap. Den måler linken i bestemte tidsintervaller og gir gjennomsnittsmålinger av nettverket. Dersom utstyr skulle gå ned vil det bli sendt SNMP traps slik at bedriften kan få varsler når utstyr slutter å sende.

### 3.3.2 Iperf 3.1.3

Iperf 3.1.3 (8 jun 2016 - 1.3 MiB for Windows Vista 64bits to Windows 10 64bits)

Iperf er et verktøy for å måle blant annet båndbredde og pakketap. Det settes opp en server og en klient for å måle kvaliteten mellom dem. Iperf brukes til live-målinger, så det gir et øyeblikksbilde og gir derfor ikke mulighet til å kunne gå tilbake for å se nærmere om en uforutsett hendelse skulle ha oppstått. Iperf er open source som sender TCP og UDP trafikk mellom målepunktene.

### 3.3.3 Microsoft Visio Plan 2 – versjon 2008

Build: 13127.21506

Visio er en av Microsoft sine lisensiert programvarer. Det brukes for å tegne diagrammer, blant annet nettverkskart og flytskjemaer. I forbindelse med prosjektet brukes dette til å illustrere nettverksoppsettet.

### 3.3.4 Juniper

Utstyret brukt er: EX-2300C (Compact)

Juniper er leverandør av ulike nettverksutstyr og programvare for nettverksadministrasjon. De begynte med å lage kjernerutere og hadde internettleverandører (ISP) som målgruppe, men har etter hvert utvidet varesortimentet. Svitsjene som har blitt brukt til å gjennomføre testene i dette prosjektet, er produsert av dem.

### 3.3.5 FortiGate

Utstyr brukt: FortiGate 80E

En FortiGate er nettverksutstyr fra leverandøren Fortinet. Her vil FortiGate 80E brukes som en brannmur. Iperf-klient og server til nettverket vil være koblet til hver sin brannmur.

### 3.3.6 Accedian - Skylight Orkestrator

BuildVersion: Skylight-orchestrator 20.05 GA 34 f2d5b96 Wed Jun 17 16:49:35 EDT 2020

Accedian er en plattform som brukes som en orkestator for overvåkning av nettverket. I prosjektets sammenheng ble orkestratoren brukt i forbindelse med TWAMP. Den holder oversikt over alle enheter som målinger kan kjøres mot og alle sesjoner som er kjørende eller kan kjøres.

### 3.3.7 Smart SFP

Smart SFP: nano 1310nm – interface type: IEEE 802.3z 1000 Base-LX

Vi brukte Accedian sine Smart SFPer som sensorer. Det kan bli lagt på software på disse, noe som gjør at de kan brukes som endepunkt i en måling. Smart SFPene kan konfigureres med IP-adresser og VLAN, noe som gjør at målinger kan gjøres direkte mot en SFP.

### 3.3.8 Datamaskin

Virtuelle maskiner med 8x2,4 GHz virtuelle prosessorer og 8 GB RAM og med Windows 2012r2.

Brukt i sammenheng med gjennomføring av målinger. Datamaskinene vil brukes som server og klient og er brukt som endepunkter for å gjennomføre målinger. De vil både kunne initiere og respondere på målinger. Datamaskinene er spesielt brukt i forbindelse med målinger som øyeblikksmåling på TWAMP, Iperf og Ping.

## 3.4 Etikk

Gruppen har ikke opplevd noen etiske problemstillinger under gjennomføringen av prosjektet. Selv om en av gruppedeltagerne er ansatt hos Eidsiva Bredbånd, opplevdes det ikke noen interessekonflikt mellom stillingen i bedriften og resultatene som er fremlagt i denne rapporten.

## 3.5 Begrensninger gitt av korona

Med de varierende restriksjonene som kan endre seg raskt, har vi sett på muligheter for å kunne jobbe videre uten å være avhengig av fysisk tilstedeværelse. Siden vi i stor grad kommer til å jobbe fra forskjellige geografiske lokasjoner, kommer vi til å jobbe en del over digitale plattformer uavhengig av smittesituasjonen. Møte med veiledere kan også gjennomføres digitalt. Selve oppsettet av testlaben krever fysisk tilstedeværelse, men begge hadde tidlig i prosessen tilgang på VPN slik at testene også kan gjennomføres virtuelt. Eidsiva Bredbånd sine lokaler bør benyttes minst mulig i forbindelse med fysisk oppmøte og bør kun brukes dersom det er høyst nødvendig. Det er også en risiko at en av oss skal bli smittet, for å redusere faren for dette vil vi sørge for å ha begrenset sosialt miljø i denne perioden. Selv lettere forkjølelse vil i denne perioden kunne hindre fysisk oppmøte noe som gjør det viktig å arbeide jevnt gjennom perioden og starte med gjennomføringen av tester så tidlig som mulig. Dette stiller ett høyere krav til at begge parter, involvert i prosjektet, har kunnskap til og informasjon om hvordan de kan gjennomføre målinger dersom den ene skulle være forhindret.

# 4 Design og implementasjon

I dette kapittelet skal vi se på hvordan vi utstyret er satt opp og hvordan vi har konfigurert dette. Utstyret er satt opp for at det skal være mulig gjennomføre testene på en best mulig måte og oppnå et best mulig resultat. Det vil i tillegg bli sett på hvordan vi har implementert konfigurasjon på SFPene og også se på hvordan de ulike målemetodene vil gjennomføres.

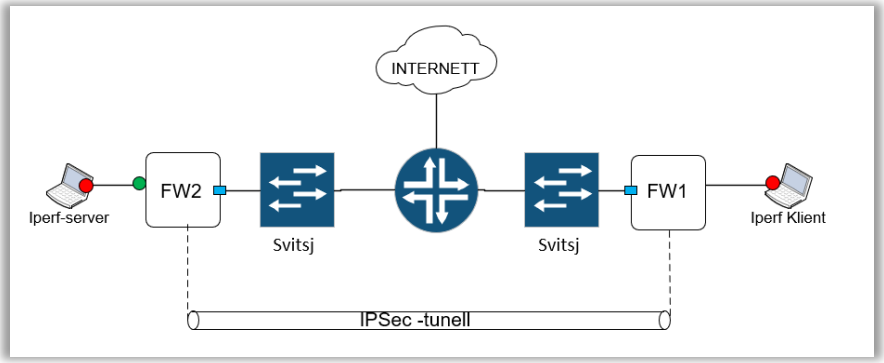
## 4.1 Oppsett av nettverk

I dette kapittelet skal det snakkes om hvordan laboppsettene som målingene skal gjennomføres over vil se ut. Det vil først bli sett på en prinsippskisse for hvert av nettverkene før det blir sett på mer avanserte nettverksskisser av laboppsett 1 (kap. 4.1.2) og laboppsett 2 (kap. 4.1.3). Dette er for å kunne gi en mer overordnet forklaring før vi går inn i detaljene til hvert nettverk. Til slutt vil det bli sett på bakgrunnen for det valgte oppsettet (kap. 4.1.4), i sammenheng med hvor komplekst et nettverksoppsett kan være. De målingene som gjøres over laboppsettene vil tilsvare målinger gjort over det aktive nettverket til Eidsiva Bredbånd.

### 4.1.1 Prinsippskisser

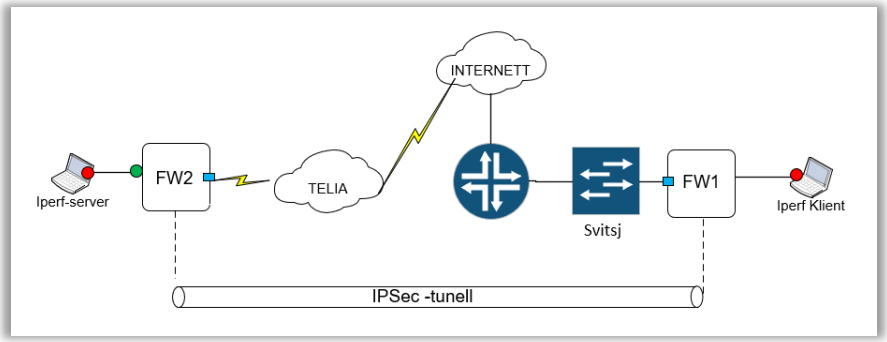
I dette underkapittelet skal vi se på hvordan oppsettet av labmiljøet vil se ut for de to ulike nettverkene vi benytter for å gjennomføre målinger. Det vil her bli introdusert to prinsippskisser av nettverket basert på figuren vist i kapittel 2 (fig. 2.1), men de er delt inn i to ulike laboppsett. Hvor laboppsett 1 er over fiberlinken og laboppsett 2 er over 4G linken. Dette er for å kunne skille de to ulike nettverkene fra hverandre både under gjennomføringene av målinger og når det kommer til de avanserte nettverkskissene. Hver FW vil ha to WAN porter, som går ut på nettverket via fiber og 4G og flere LAN porter som blant annet er koblet til Iperf Klienten og Iperf-server.

**Prinsippskissen for laboppsett 1 over fiber**:




*Figur 4. 1: Enkel prinsippskisse over laboppsett 1*

**Prinsippskisse til laboppsett 2 over 4G-modem**:




*Figur 4. 2: Enkel prinsippskisse over laboppsett 2*

**Viktige oppklaringer til prinsippskissene**:

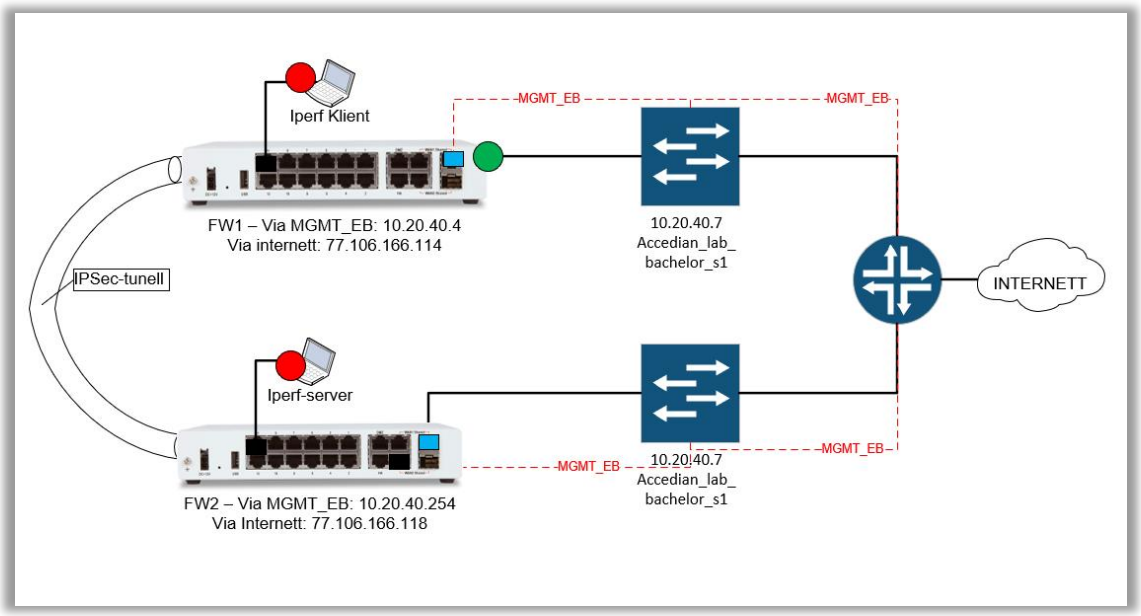
- Rød prikk viser mellom hvilke endepunkter Iperf-målingene gjennomføres.
- Grønn prikk viser SNMP-målingen og denne overvåker interfacet ut mot fiber og 4G.
- Blå prikk viser hvor TWAMP-målingene gjennomføres og hvor hver av SFPene er koblet til nettverket.
- Svitsj – Beskriver svitsjen som er konfigurert i dette prosjektet. Konfigurasjon som er lagt til kan bli sett i neste kapittel (kap. 4.2.1).
- FW, forkortelse for Fire Wall (brannmur) og er 2 FortiGater som er konfigurert i forbindelse med prosjektet. Derfor har vi FW1 og FW2. Konfigurasjonen av disse er

spesifisert i kapittel 4.2.2. Det er såpass lignende konfigurasjon på dem at det ikke er satt opp eget konfigurasjon kapittel til begge.

- Telia representerer linken over ett 4G-modem.
- Ruteren er en internettruter, dette blir illustrert ved at den er koblet til en «sky» (INTERNETT).

### 4.1.2 Laboppsett 1 - Fiber

Den første delen av nettverket er satt opp på en veldig tradisjonell måte, hvor store deler av trafikken vil gå via Eidsiva Bredbånd sitt kjernenett (her sett på som internett). Nettverket er satt opp slik at det kan gjennomføres målinger mellom to FWer, FW1 og FW2 via en internettruter. Dette ser vi illustrert nedenfor (fig. 4.3). Oppsettet gjør at vi kan gjennomføre mer reelle målinger via det fysiske nettverket til Eidsiva Bredbånd. Forhåpentligvis vil dette bidra til en større variasjon på resultatene. For å komme frem til dette oppsettet så brukte vi et eksisterende LAB-nett i bedriften. Her koblet vi opp vår konfigurerte svitsj, 10.20.40.7, og to FWer som også ble konfigurerte. I hver FW vil det videre sitte en SFP, blått rektangel, som vil bidra til å kunne gjennomføre målinger direkte via SFPen. I det utstyret vi har satt opp, er det en fibersnor mellom 10.20.40.7 og hver FW. Disse snorene vil ende i en SFP i hver ende.

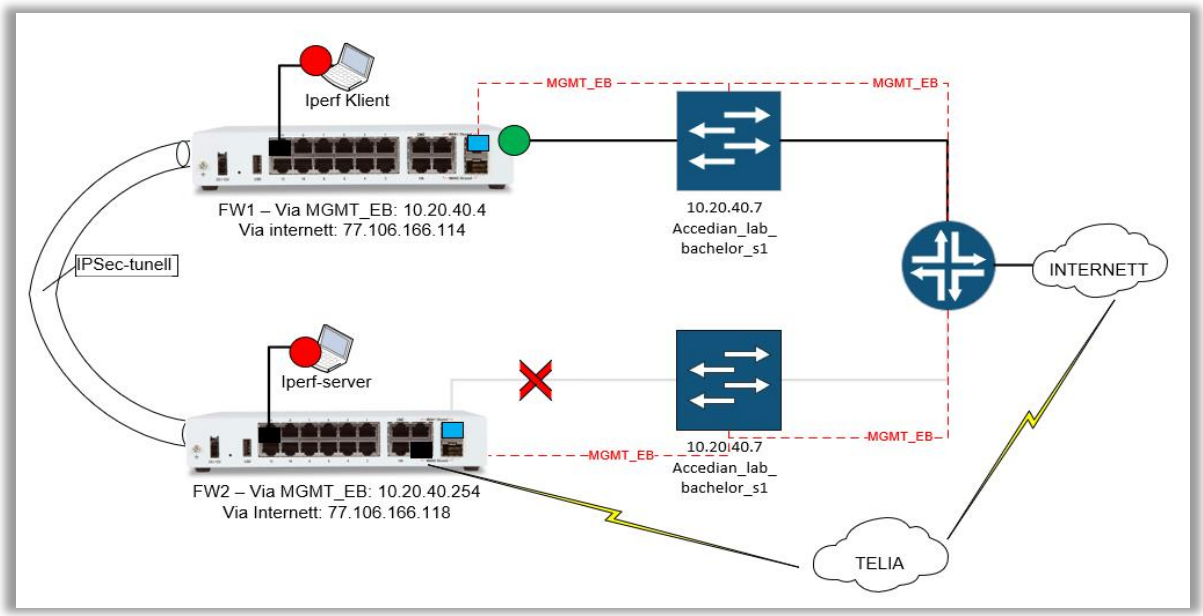



*Figur 4. 3: Avansert nettverksskisse for laboppsett 1.*

Oppsettet bak svitsjen (mot ruteren) er ikke satt opp i forbindelse med dette prosjektet. Dette er en del av det aktive nettverket til bedriften. Det var noen endringer som ble gjort i det aktive nettverket til Eidsiva Bredbånd. Dette var blant annet fremføringen av VLANene til kommunikasjonen via internett, VLAN 3666 og 3667. Det ble også lagt frem flere IP-adresser, herunder en ekstra adresse for hver av Iperf-klientene, i forbindelse med gjennomføring av rød måling. Det ble også satt opp en IPSec-tunell mellom hver FW, i håp om at denne skulle introdusere ekstra forsinkelse i nettverket.

### 4.1.3 Laboppsett 2 – 4G

Laboppsett 2 (fig. 4.4) består av kommunikasjon over 4G med en IPSec tunell mellom endepunktene, FW1 og FW2. Forskjellen på dette nettverket og labboppsett 1 med fiber er at denne vil sende trafikken via et 4G modem og videre til internett og internettruteren. Dette er satt opp på en måte slik at fiberlinken er primærlinken og 4G er den redundante linken. Slik at dersom fiberlinken blir brutt eller det blir mer kostbart å overføre via den, vil 4G linken via Telia bli den prioriterte linken. Her vil datatrafikken bli ført ut på en 4G-link (fra FW2) og vil nå den andre brannmuren (FW1) ved bruk av en IPSec-tunell. IPSec-tunellen er illustrert til venstre i figuren. Dette er satt opp for å kunne få resultater og målinger over ett nettverk hvor det er en større sannsynlighet for forsinkelser og reduksjon i hastighet og ytelse.



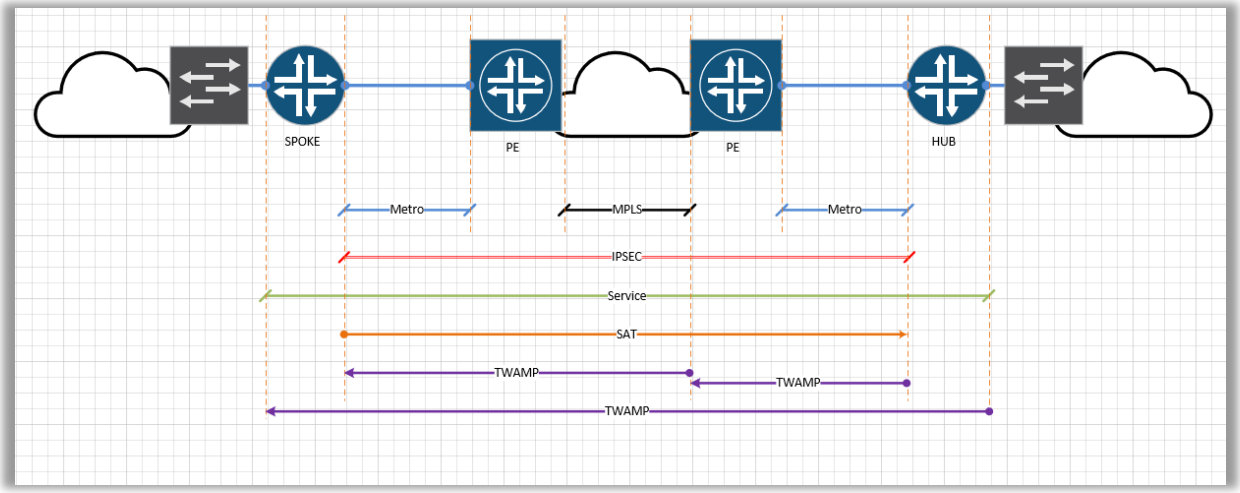

*Figur 4. 4: Avansert nettverkskisse over laboppsett 2.*

Det røde krysset ved FW2 illustrerer at fiberlinken er nede og trafikken må gå via et 4G-modem og linken til TELIA. Videre vil pakkene sendes til INTERNETT og internettruteren før trafikken dirigeres til FW1. For at det skal være mulig må det være en IPSec tunell mellom FW1 og FW2. Denne tunellen gjør at data kan overføres sikkert mellom to LAN.

### 4.1.4 Bakgrunnen for oppsettet

Dette underkapittelet er ment for å gi en illustrasjon (fig. 4.5) på hvor komplekse og kompliserte nettverk kan være. Det er for å kunne reflektere på i hvor stor grad ett nettverk har behov for bedre overvåkningsverktøy, siden ett nettverk består av mange deler. Nettverket til Eidsiva Bredbånd er bygd opp av tre hovednivåer, de har kjerne, distribusjon og aksess. I kjernenettet er det få enheter og de enhetene som befinner seg her er gjerne større rutere som gjør det mulig for Eidsiva Bredbånd å rute trafikken mot riktig destinasjon. Videre har de distribusjonsnettverket. Dette er den delen av nettverket som fører trafikk ut mot hver enkelt kunde og videre ut til svitsjene som fordeler trafikken mellom kunde og kjernenettet. Helt ytterst i Eidsiva Bredbånd sitt nettverk, har de aksessnivået. På dette nivået er det flest enheter. Nivået omfatter utstyr som ligger mellom det ytterste punktet på distribusjonsnettet og helt ut til det utstyret som står tilkoblet som endepunkt hos kunden.

Bakgrunnen for at vi har valgt oppsettet slik som vi har for laboppsett 1 og laboppsett 2 er for å få en likhet til den kompleksiteten ett nettverk kan ha for Eidsiva Bredbånd. Nettverket er i tillegg satt sammen av flere teknologier og nettverksarkitekturer. Det brukes blant annet IPSec, VLAN og LAN. Et generelt eksempel på sammensetningen i ett nettverk kan vi se på figuren (fig. 4.5). Noe som gjør at det kan oppstå mange kilder for forsinkelser og pakketap i nettverket. Det er derfor ønskelig med en måleteknikk som kan gjennomføre målinger på flere ledd, slik TWAMP kan, da dette kan bidra til å avgjøre hvilket ledd eventuelle forsinkelser eller pakketap oppstår.




*Figur 4. 5: Hvordan ett nettverk kan bestå av flere tjenester mellom hvert ledd og på tvers av de ulike enhetene i nettverket.*

Denne figuren (fig. 4.5) viser et bilde av hvordan et oppsett generelt kan se ut ved bruk av TWAMP og det er ett av utgangspunktene til hvorfor vi har valgt det oppsettet som skal bli sett på i de neste delene av dette kapittelet. Figuren viser at TWAMP kan både gjennomføre ende-til-ende målinger, men også målinger mellom hver enhet i nettverket. Som figuren illustrerer så vil TWAMP kunne gjennomføre målinger begge veier i nettverket og det er også en mulighet å gjennomføre målinger mellom hvert enkelt ledd med denne protokollen.

## 4.2 Konfigurasjon av utstyr

Vi skal i dette underkapittelet se på konfigurasjonen av utstyret vi trenger for å kunne gjennomføre målinger med TWAMP protokollen. Dette er konfigurasjon som trengs i forhold til det oppsettet som vi har bestemt oss for. Vi vil konfigurere en svitsj og vi trenger også å konfigurere to FortiGate 80E som vil benyttes som brannmurer. Disse vil videre bli kalt FW om vi snakker om generell konfigurasjon som er i begge boksene og FW1 eller FW2 dersom det snakkes om konfigurasjon på en av de spesifikke boksene.

### 4.2.1 Konfigurasjon av svitsj

Vi benytter en svitsj levert i fra Juniper Networks av typen EX-2300C. Svitsjen er konfigurert for at det skal opprettes forbindelse mot smarte SFPer, og for at det skal være mulig å gjennomføre de ønskede testene som skal kjøres via SFPene. Den er også konfigurert for at det

skal kunne gå internettrafikk over nettverket og for at det skal kunne gjennomføres SNMP målinger ved bruk av bedriften sitt verktøy for dette (Orion).

**Konfigurasjonen til svitsjen**

Det er mye konfigurasjon som ble satt på svitsjen i dette prosjektet. Vi har dermed valgt å ta med den konfigurasjonen som var spesifikk for prosjektet, og ikke ta med det som er mer generell konfigurasjon som Eidsiva Bredbånd bruker. Fokuset vil derfor være på den overordnede konfigurasjonen deretter vil det bli gått dypere inn i den litt mer spesifikke konfigurasjonen (fig. 4.6-4.8). Det settes konfigurasjon når det kommer til autentisering og innlogging på utstyret og systemkonfigurasjon, deretter vil det bli satt en IP-adresse på svitsjen for å kunne logge inn på den og endre konfigurasjon på et senere tidspunkt. Denne IP-adressen er 10.20.40.7. Deretter ble det konfigurert ett interface (ge-0/0/0) som går videre ut i Eidsiva Bredbånd sitt nettverk, mot ruteren i figurene for laboppsett (fig. 4.1 og 4.2). Dette interfacet kalles gjerne en «uplink» for svitsjen. Det ble også satt konfigurasjon relatert til dette interfacet, herunder VLAN som blir brukt på FW1 og FW2. Hvilke VLAN dette er kan bli sett i figuren (fig. 4.7). Det ble videre konfigurert VLAN 20 til alle aktive interface, dette VLANet blir kalt EB_MGMT og er brukt for management på utstyret. Det ble også satt hvilken MTU størrelse det skulle være for hvert interface som er i bruk. Dette sier noe om hvor store pakkene som sendes har lov til å være.

I tabellen (tab.1) nedenfor kan vi se den konfigurasjonen som ble satt spesielt relatert til FW1 og FW2:

*Tabell 1: Konfigurasjon relatert til FW1 og FW2 på svitsjen*

| | **FW1** | **FW2** |
|---|---|---|
| **Koblet til interface på svitsj** | ge-0/1/0 | ge-0/1/1 |
| **Internett VLAN** | 3666 | 3667 |
| **LAN_INT VLAN** | 31 | 32 |
| **SFP VLAN** | 29 | 29 |

VLAN konfigurasjonen som blir satt og hvilket interface denne konfigurasjonen tilhører kan vi se i figuren (fig. 4.6). Her ser vi at VLAN 20 settes for EB_MGMT, VLAN 29 settes for management for SFP (description EB_MGMT_CPE). Her ser vi også de to VLANene som er satt for internett. Dette vil henholdsvis være VLAN 3666 for FW1 og VLAN 3667 for FW2.

```
idwe@no0517-accidian_lab_bachelor-s1> show configuration vlans | display set
set vlans vlan-20 description EB_MGMT
set vlans vlan-20 vlan-id 20
set vlans vlan-20 interface ge-0/1/0.20
set vlans vlan-20 interface ge-0/0/0.20
set vlans vlan-20 interface ge-0/1/1.20
set vlans vlan-20 l3-interface irb.20
set vlans vlan-29 description EB_MGMT_CPE
set vlans vlan-29 vlan-id 29
set vlans vlan-29 interface ge-0/1/0.29
set vlans vlan-29 interface ge-0/0/0.29
set vlans vlan-29 interface ge-0/1/1.29
set vlans vlan-31 vlan-id 31
set vlans vlan-31 interface ge-0/1/0.31
set vlans vlan-31 interface ge-0/0/4.31
set vlans vlan-31 interface ge-0/0/0.31
set vlans vlan-32 vlan-id 32
set vlans vlan-32 interface ge-0/1/1.32
set vlans vlan-32 interface ge-0/0/5.32
set vlans vlan-32 interface ge-0/0/0.32
set vlans vlan-3666 description I_idabachelor
set vlans vlan-3666 interface ge-0/1/0.3666
set vlans vlan-3666 interface ge-0/0/0.3666
set vlans vlan-3667 vlan-id 3667
set vlans vlan-3667 interface ge-0/0/0.3667
set vlans vlan-3667 interface ge-0/1/1.3667
```

*Figur 4. 6: VLAN konfigurasjonen på svitsjen*

Det vil også settes konfigurasjon til hvert interface som vi kan se i figuren (fig. 4.7). Det vil også bli satt konfigurasjon på to andre interface, ge-0/1/0 som leder mot FW1 og ge-0/1/1 som leder mot FW2. På disse interfacene vil også all nødvendig konfigurasjon bli lagt på. I tillegg kan vi se i figuren (fig. 4.7) at det er lagt på konfigurasjon på interface ge-0/0/4 og ge-0/0/5, dette er interface som er knyttet mot LAN-portene på hver FW (for å kunne gjennomføre målinger kablet direkte til svitsjen). Her vil ge-0/0/4 tilhøre FW1 sitt LAN og ge-0/0/5 tilhøre FW2 sitt LAN.

```
idwe@no0517-accidian_lab_bachelor-s1> show configuration interfaces | display set
set interfaces ge-0/0/0 description "*** noc_prosjektlab -s1 ***"
set interfaces ge-0/0/0 flexible-vlan-tagging
set interfaces ge-0/0/0 mtu 9192
set interfaces ge-0/0/0 encapsulation extended-vlan-bridge
set interfaces ge-0/0/0 unit 20 vlan-id 20
set interfaces ge-0/0/0 unit 29 vlan-id 29
set interfaces ge-0/0/0 unit 31 vlan-id 31
set interfaces ge-0/0/0 unit 32 vlan-id 32
set interfaces ge-0/0/0 unit 3666 vlan-id 3666
set interfaces ge-0/0/0 unit 3667 vlan-id 3667
set interfaces ge-0/0/4 description "*** FG1 LAN_INT ***"
set interfaces ge-0/0/4 flexible-vlan-tagging
set interfaces ge-0/0/4 native-vlan-id 31
set interfaces ge-0/0/4 mtu 1600
set interfaces ge-0/0/4 encapsulation extended-vlan-bridge
set interfaces ge-0/0/4 unit 31 vlan-id 31
set interfaces ge-0/0/5 description "*** FG2 LAN_INT  ***"
set interfaces ge-0/0/5 flexible-vlan-tagging
set interfaces ge-0/0/5 native-vlan-id 32
set interfaces ge-0/0/5 mtu 1600
set interfaces ge-0/0/5 encapsulation extended-vlan-bridge
set interfaces ge-0/0/5 unit 32 vlan-id 32
set interfaces ge-0/1/0 description "accedian sfp - bachelor oppgave"
set interfaces ge-0/1/0 flexible-vlan-tagging
set interfaces ge-0/1/0 native-vlan-id 3666
set interfaces ge-0/1/0 mtu 9192
set interfaces ge-0/1/0 encapsulation extended-vlan-bridge
set interfaces ge-0/1/0 unit 20 vlan-id 20
set interfaces ge-0/1/0 unit 29 vlan-id 29
set interfaces ge-0/1/0 unit 31 vlan-id 31
set interfaces ge-0/1/0 unit 3666 vlan-id 3666
set interfaces ge-0/1/1 description "accedian sfp - bachelor oppgave"
set interfaces ge-0/1/1 flexible-vlan-tagging
set interfaces ge-0/1/1 native-vlan-id 3667
set interfaces ge-0/1/1 mtu 9192
set interfaces ge-0/1/1 encapsulation extended-vlan-bridge
set interfaces ge-0/1/1 unit 20 vlan-id 20
set interfaces ge-0/1/1 unit 29 vlan-id 29
set interfaces ge-0/1/1 unit 32 vlan-id 32
set interfaces ge-0/1/1 unit 3667 vlan-id 3667
set interfaces irb unit 20 description Management
set interfaces irb unit 20 family inet address 10.20.40.7/24
```

*Figur 4. 7: Interface-konfigurasjonen på svitsjen*

Videre ble det også satt konfigurasjon omkring SNMP funksjonaliteten (fig. 4.8).

```
idwe@no0517-accidian_lab_bachelor-s1> show configuration snmp | display set
set snmp location "noc-rack fakkelgÅ¥rd kjellerlab"
set snmp contact "noc@eidsiva.net"
set snmp community qwerty authorization read-only
set snmp community qwerty clients 10.0.0.0/8
set snmp community qwerty clients 82.147.36.0/27
set snmp community qwerty clients 0.0.0.0/0 restrict
set snmp trap-options source-address lo0
set snmp trap-group JUNIPER categories authentication
set snmp trap-group JUNIPER categories chassis
set snmp trap-group JUNIPER categories link
set snmp trap-group JUNIPER categories remote-operations
set snmp trap-group JUNIPER categories routing
set snmp trap-group JUNIPER categories startup
set snmp trap-group JUNIPER categories services
set snmp trap-group JUNIPER targets 82.147.36.4
```

*Figur 4. 8: SNMP-konfigurasjon på svitsjen*

### 4.2.2 Konfigurasjon av FortiGate 80E - Firewall (brannmur)

Når en FortiGate 80E skal konfigureres er det to vanlige måter å gjøre dette på. Den som skal konfigurere kan enten logge inn på FortiGatens Graphical User Interface (GUI), eller det kan bli logget inn ved å benytte kommandovindu. Men før disse to metodene for konfigurasjon kan benyttes, må enheten konfigureres med en management-IP via seriell kabel direkte kablet til boksen. FortiGaten som er konfigurert her vil brukes som en brannmur og kalles derfor en FW.

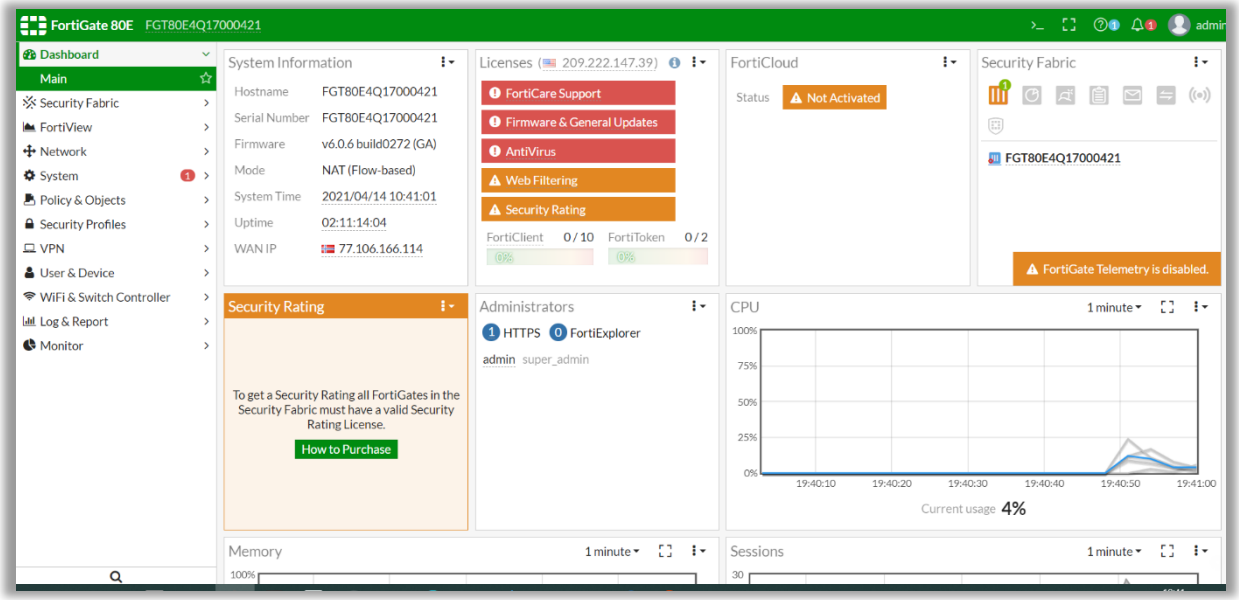
Det første som ble gjort ved konfigurasjon av FW var at den ble konfigurert med en management-IP i VLAN 20 som er Eidsiva Bredbånd sitt management VLAN (EB_MGMT). Videre ble det lagt til konfigurasjon som internettadresse og et VLAN som trafikken skulle sendes over. Under (tab. 2) kan vi se noen overordnede punkter med karakteristikk for konfigurasjonen av de forskjellige FW-boksene:

*Tabell 2: De viktigste konfigurasjonene på FW*

| | **FW1** | **FW2** |
|---|---|---|
| **Managementadresse** | 10.20.40.4 | 10.20.40.254 |
| **Internettadresse** | 77.106.166.114 | 77.106.166.118 |
| **VLAN til internettadresse** | 3666 | 3667 |
| **Hvilke WAN interface brukt** | WAN1 (fiber) | WAN1 (fiber) og WAN2 (4G) |

Managementadressen gjør at FW kan konfigureres uten at det er nødvendig å være fysisk til stede. For at denne skal kunne kommunisere og koble til Internett trengs den også en internettadresse som kan bli sett ovenfor (tab. 2). For at denne skulle ha tilgang til internett var det nødvendig å samtidig legge frem ett internett VLAN som også var lagt frem til svitsjen.

Når FW skal konfigureres ved å logge inn på GUI, vil det se ut som på figuren nedenfor (fig. 4.9) ved innlogging. Som vi kan se her vil det være mulig å konfigurere alle nødvendige innstillinger, for ett endepunkt. Det vil her være mulig å gå i menyen på venstre side i figuren for å sette innstillinger som det er ønsket av FW skal ha. Det vil også være mulig å sette samme innstillinger ved å bruke ett kommandovindu.




*Figur 4. 9: Grensesnittet på en FortiGate (FW) brukt i lab-oppsettet.*

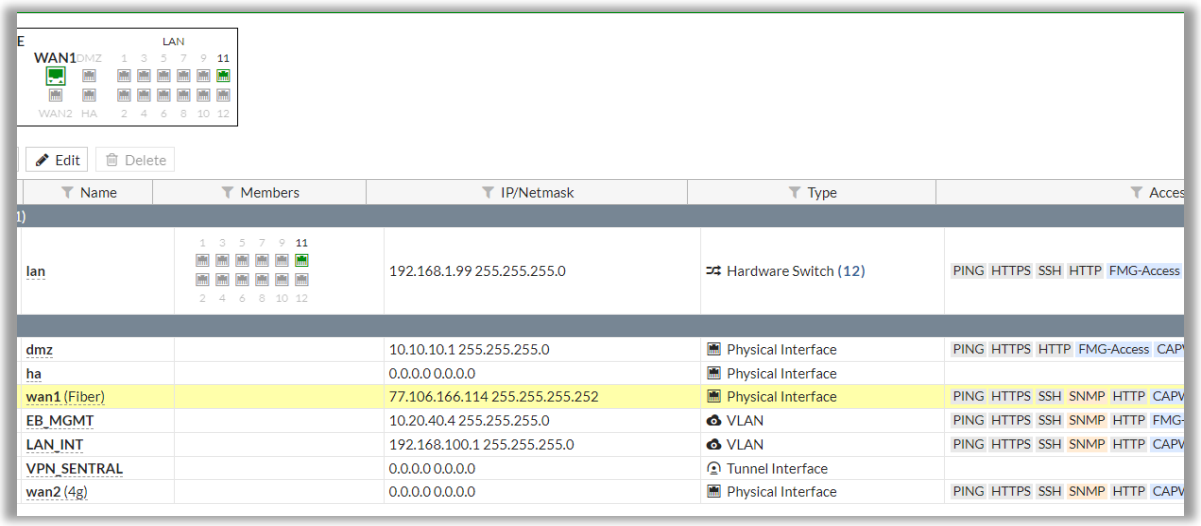
Det vil også være mulig å konfigurere den ved å benytte kommandovinduet. Under (fig. 4.10) kan vi se hvordan det ser ut ved konfigurasjon via kommandoer, ofte kalt Command Line Interface (CLI).

```
idwe@helmsdeep:~$ ssh admin@10.20.40.4
admin@10.20.40.4's password:
FGT80E4Q17000421 # config system interface
```

*Figur 4. 10: Innlogging for konfigurasjon og kommando for interface konfigurasjon.*

**Konfigurasjon av WAN-portene på FW1**

Videre konfigurerte vi interfacene på FW1 som var i bruk i prosjektet. I FW1 ble interface WAN1 konfigurert til å ha en ønsket IP-adresse, 77.106.166.114, som er internettadressen til enheten. Det ble i tillegg lagt opp statisk rute for denne slik at trafikken skulle gå i riktig retning. Videre ble det satt opp andre innstillinger for at WAN1 skulle fungere optimalt.




*Figur 4. 11: Brukergrensesnitt for Interface på FW1*

På FW1 ble det kun lagt på spesifikk konfigurasjon på WAN1. Dersom konfigurasjonen og oppsettet av interfacet skal gjennomføres via brukergrensesnittet, vil den bli opprettet ved å trykke på «create new» på figuren ovenfor (fig. 4.11). Det var også tre andre interface som er brukt på FW1 her vil den viktigste være LAN_INT som gjør at WAN1 også kan fungere som en LAN-port.

Konfigurasjon lagt på WAN1 - FW1 (fig. 4.12):

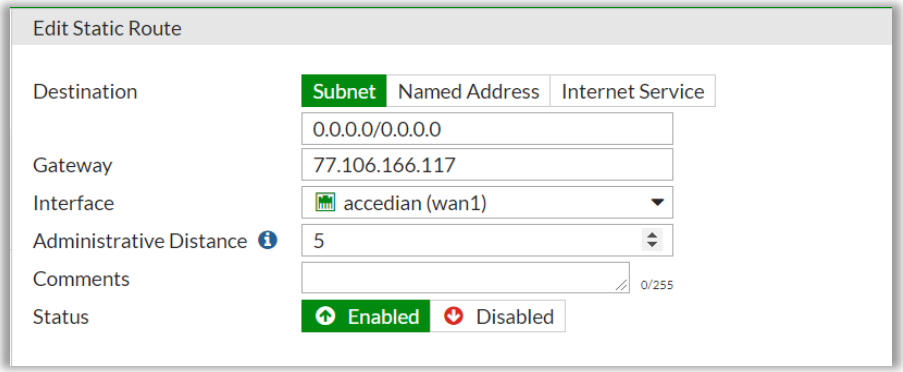
```
FGT80E4Q17000421 # config system interface

FGT80E4Q17000421 (interface) # edit wan1

FGT80E4Q17000421 (wan1) # show
config system interface
    edit "wan1"
        set vdom "root"
        set ip 77.106.166.114 255.255.255.252
        set allowaccess ping https ssh snmp http capwap
        set type physical
        set alias "Fiber"
        set role wan
        set snmp-index 1
    next
end

FGT80E4Q17000421 (wan1) # end

FGT80E4Q17000421 # config router static

FGT80E4Q17000421 (static) # edit 1

FGT80E4Q17000421 (1) # show
config router static
    edit 1
        set gateway 77.106.166.113
        set distance 1
        set weight 50
        set priority 10
        set device "wan1"
    next
end
```

*Figur 4. 12: Konfigurasjon til WAN1 på FW1*

For at det skal være mulig å koble til utstyr på LAN portene var det nødvendig å sette opp en NAT på FW1, hvor denne bruker interne adresser på sine LAN-porter og en offentlig adresse for å sende trafikken ut på internett. Dette var viktig å konfigurere da vi er nødvendig til å bruke LAN portene i forbindelse med målingene som skal gjøres. Spesielt når det kommer til Iperf og annet tilkoblet utstyr på LAN siden.

**Konfigurasjon av WAN-portene på FW2**

På FW2 ble det lagt på spesifikk konfigurasjon på WAN1 og WAN2, hvor WAN1 er fiberlinken til enheten og WAN2 er kablet til ett 4G modem. For å skille mellom hvilken av disse linkene trafikken skal bli sendt over, er det nødvendig å skille mellom hvilken prioritet de ulike linkene har. I prosjektet er dette satt opp slik at fiberlinken via WAN1 er primærlinken og 4G modemet er redundantlinken. For at 4G modemet skal bli prioritert må fiberlinken få nedsatt sin prioritet. Figuren (fig. 2.13) viser hvordan dette ble endret i prosjektet. Dersom fiberlinken skulle være prioritert ble Administrative Distance satt til 5 og dersom fiberlinken skulle bli nedprioritert og trafikken skulle føres over 4G modemet, ble denne satt til 15.




*Figur 4. 13: Static route for FW2. Administrativ Distance er viktig her.*

Det skal nå bli sett på den konfigurasjonen som er på hvert av WAN interfacene i FW2.

Konfigurasjon lagt på WAN1 – FW2 (fig. 4.14):

```
FGR60D4614000490 # config system interface

FGR60D4614000490 (interface) # edit wan1

FGR60D4614000490 (wan1) # show
config system interface
    edit "wan1"
        set vdom "root"
        set ip 77.106.166.118 255.255.255.252
        set allowaccess ping https ssh snmp http
        set type physical
        set alias "accedian"
        set role wan
        set snmp-index 1
    next
end

FGR60D4614000490 (wan1) # end

FGR60D4614000490 # config router static

FGR60D4614000490 (static) # edit 1

FGR60D4614000490 (1) # show
config router static
    edit 1
        set gateway 77.106.166.117
        set distance 5
        set device "wan1"
    next
end
```

*Figur 4. 14: Konfigurasjon til WAN1 på FW2*

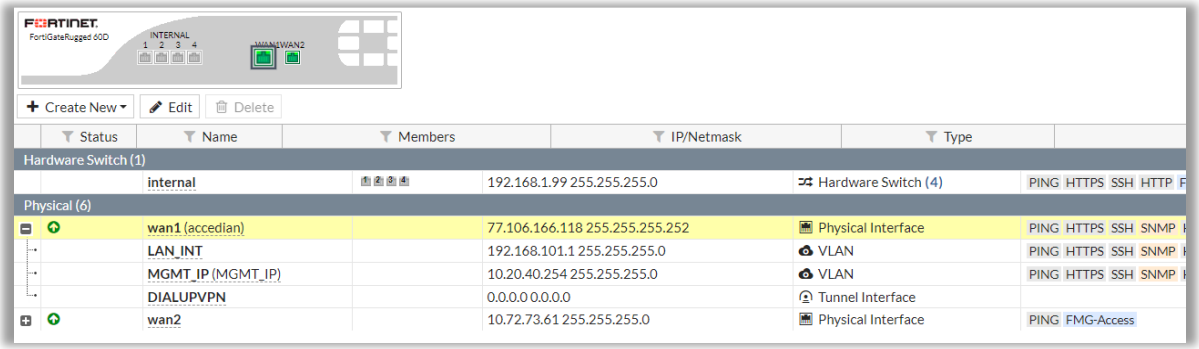
Konfigurasjon lagt på WAN2 – FW2 (fig. 4.15):

```
FGR60D4614000490 # config system interface

FGR60D4614000490 (interface) # edit wan2

FGR60D4614000490 (wan2) # show
config system interface
    edit "wan2"
        set vdom "root"
        set mode dhcp
        set distance 100
        set allowaccess ping fgfm
        set type physical
        set role wan
        set snmp-index 2
    next
end

FGR60D4614000490 (wan2) # end

FGR60D4614000490 # config router static

FGR60D4614000490 (static) # edit 8

FGR60D4614000490 (8) # show
config router static
    edit 8
        set gateway 10.72.73.1
        set distance 11
        set priority 11
        set device "wan2"
    next
end
```

*Figur 4. 15: Konfigurasjonen til WAN2 på FW2*

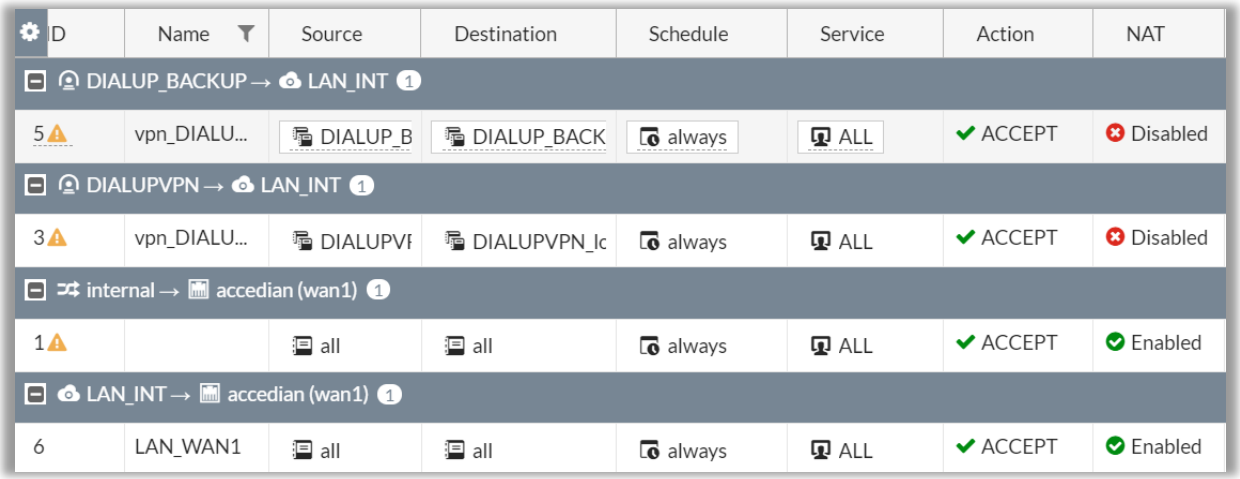
Figuren nedenfor (fig. 4.16) vil vise de ulike interfacene som er benyttet. Der er flere interface her enn WAN1 og WAN2 som benyttes. Det var også tre andre interface. Her har vi LAN_INT som gjør at WAN1 også kan fungere som en LAN-port. Dette fungerer ved at trafikken som skal måles eller sendes, først sendes inn i boksen igjen og deretter ut på ønsket WAN port for å kunne gjennomføre målinger på SFPene. Videre har vi DIALUPVPN som er IPSec tunellen for

WAN1. Under WAN2 har vi også en IPSec tunell som heter DIALUP_BACKUP. I figuren ser vi også hvilke IP-adresser interfacene har og hvilken type de er.




*Figur 4. 16: Interfacene i brukergrensesnittet for FW2*

Det ble også satt opp filter til brannmurene FW1 og FW2. Disse filtrene ble satt opp ved bruk av en funksjon på enhetene som kalles IPv4 policy. Dette er filtre som avgjør hvilken trafikk som kan gå igjennom enhetene og den definerer to IPSec-tuneller og hvordan all trafikk fra LAN skal gå via kryptert nett. Den definerer også hvordan trafikken vil gå ut på internett. Dette kan vi se i figuren nedenfor (fig. 4.17).

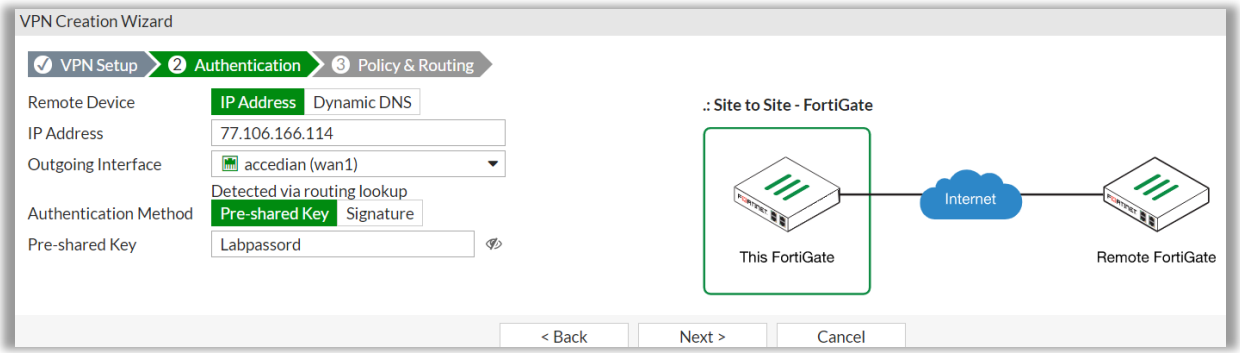



*Figur 4. 17: Filtrene som er satt opp på FW2*

### IPSec-tunell

For å kunne føre trafikk over 4G-modemet på FW2 og til enheten bak FW1. Så er det nødvendig med en IPSec-tunell. Denne krypterer pakkene som sendes og gjør sendingen tryggere og at det da er mulig å overføre data mellom to lokale nettverk. Når det skal sende trafikk over fiber så vil det ikke være nødvendig å bruke IPSec-tunellen for å overføre data. Men testene er gjennomført både over fiber med og uten IPSec-tunell.

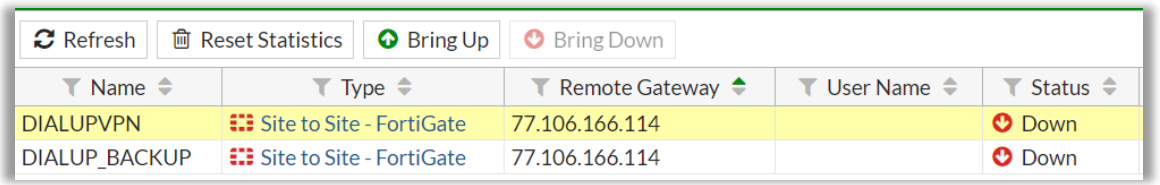
Det ble på begge FW konfigurert slik at de skulle ha en IPSec tunell mellom seg. I figuren (fig. 4.18) kan vi se hvordan dette oppsettet blir gjort for FW2, her vil enheten kalles en FortiGate, da det er utstyret som er brukt i prosjektet. Som vi kan se er dette en forenklet prosess, hvor utstyret har en automatisering av standardoppsettet for en IPSec-tunell. Det vil legges inn IP-adressen som i dette tilfellet hører til FW1, og det vil settes en nøkkel (Pre-shared Key) som må være lik på begge sider av tunellen. Det er denne nøkkelen som gjør det mulig for enhetene å pakke opp dataen som sendes imellom dem. I tillegg må det konfigureres hvilket interface som dataen skal sendes ut på. I figuren (fig. 4.18) vil dette være IPSec tunellen for fiberlinken (WAN1). Det vil også konfigureres en tunell på lik måte for 4G-linken. Bakgrunnen til at vi kun har tatt med dette bildet er fordi det i VPN-setup settes navn på tunellen og policy er forklart tidligere i kapitelet (fig. 4.17).




*Figur 4. 18: Oppsett av en IPSec-tunell for FW2*

Aktivering og deaktivering av IPSec-tunell på FortiGate:

Når vi endrer hvilken link som trafikken skal sendes over på FW2 så er det også nødvendig for oss å endre hvilken IPSec-tunell som skal være aktiv. Her er DIALUPVPN den som tilhører fiberlinken og DIALUP_BACKUP den som fører trafikk over 4G. Her vil det kun være nødvendig å trykke på Bring Up for å starte opp IPSec-tunellen.




*Figur 4. 19: Aktivering og deaktivering av IPSec-tunell*

## 4.3 Konfigurasjon av en SFP

I kapittel 2 (kap. 2.7.1) så har vi sett på smart SFPer og hvordan det kan legges en software på disse. Vi skal i dette underkapittelet se på hvordan vi faktisk legger på konfigurasjon på en SFP og hvilken konfigurasjon vi kan endre på eller legge til. I denne oppgaven valgte vi å bruke moduleDOCK (fig. 4.20) fra Accedian til å konfigurere SFPene. Denne er spesielt designet for disse SFPene da den inneholder en Universal Serial Buss (USB)-port, en sensor hvor SFP blir koblet til og en sensor til som kalles en modul kobling, dette er en Ethernetport. I denne modulen vil vi sette inn en av de SFPene som skal konfigureres, videre vil vi bruke Ethernet for å koble oss til moduleDOCKen. For at dette skal være mulig trenger vi å sette nettverksinnstillingene på maskinen sin Ethernet-inngang slik at dette samsvarer med den IPen som er satt som default innstilling på den enheten vi skal logge inn på, dette er i rangen til 192.168.7.0/24. Dette vil si at adressene i dette tilfellet vil være en adresse mellom 192.168.7.0 og 192.168.7.254.


*Figur 4. 20: Illustrasjonsbildet av ModuleDOCK brukt til konfigurasjon av SFPen*

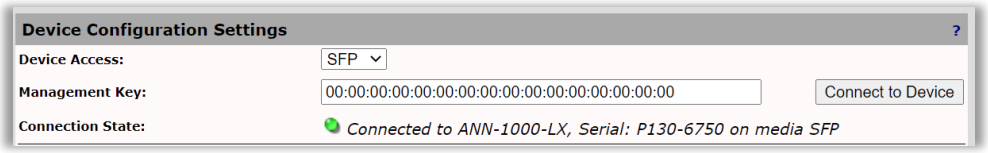
De hovedinnstillingene som blir konfigurert inn på SFPene er hvilken IP SFPene skal ha, om det skal være en IPv6 eller IPv4, hvilket VLAN vi må bruke eller ønsker å bruke og hvordan type VLAN vi skal ha. I vårt tilfelle vil den spesifikke konfigurasjonen være (tab. 3):

*Tabell 3: De viktigste konfigurasjon av SFPene*

| | **SFP 1** | **SFP 2** |
|---|---|---|
| **VLAN** | 29 | 29 |
| **C-VLAN** | Ja | Ja |
| **IP** | 10.29.40.9 | 10.27.40.11 |

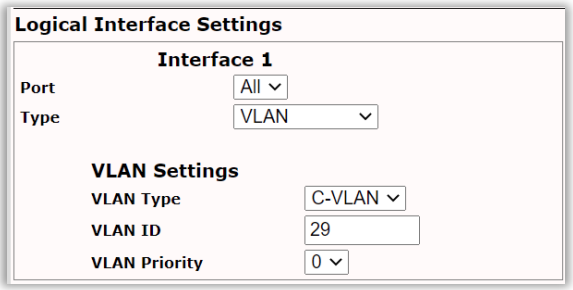
Det er flere innstillinger som det er mulig for oss å konfigurere, men i denne oppgaven er dette de viktigste innstillingene og de eneste nødvendige for å kunne gjennomføre våre ønskede målinger.

Det skal nå vises hvordan grensesnittet ved bruk av moduleDOCKen vil se ut i forbindelse med konfigurasjonen som ble satt. Innlogging er mulig dersom nettverkskonfigurasjonen på datamaskinen er satt med korrekt IP og Subnettmaske. Når vi logget inn måtte vi koble til enheten. Dette gjøres ved å trykke på “Connect to Device” i figuren (fig. 4.21).



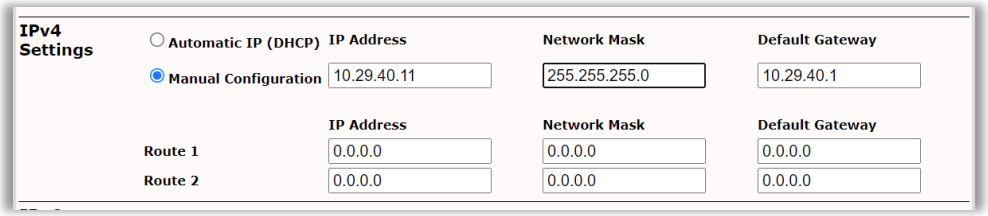

*Figur 4. 21: Innlogging på SFPen, oppretter forbindelse ved å trykke på «Connect to Device»*

Deretter vil VLAN konfigurasjonen bli satt (fig. 4.22):



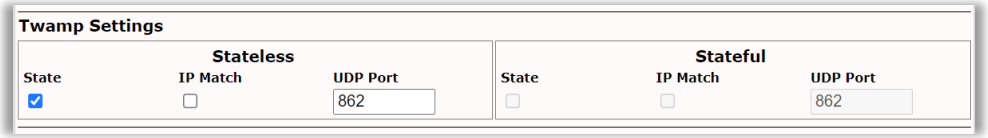

*Figur 4. 22: VLAN-konfigurasjon på SFP*

Deretter blir IP og subnettmaske satt (fig. 4.23):




*Figur 4. 23: SFPen sin konfigurasjon med IP-adresse.*

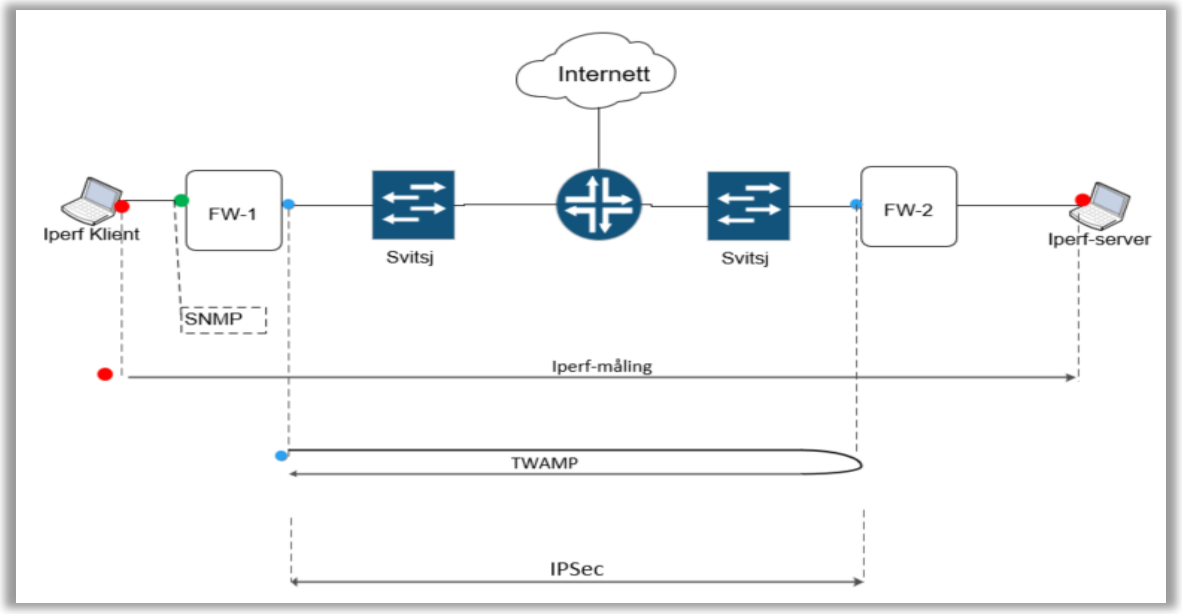
Til slutt er det nødvendig å sette en port. Port 862 er standardporten for gjennomføring av TWAMP-målingene (fig. 4.24):




*Figur 4. 24: Konfigurasjon av porten*

## 4.4 Fremgangsmåte for målinger

I dette kapittelet vil vi se på flere ulike måter å gjennomføre målinger. Samtidig gå gjennom de ulike verktøyene som brukes for å overvåke bredbåndsleveransen slik som situasjonen er nå og med den teknologien Eidsiva Bredbånd bruker. Kapittelet skal gi generell informasjon om verktøyene, hvordan de konfigureres eller settes opp og eventuelle eksempler på den informasjon som det vil være mulig å hente ut når det ønskes å kvalitetssikre eller dokumentere leveransen.




*Figur 4. 25:Prinsippskisse som illustrere de målingene som skal gjøres og mellom hvilke ledd de vil gjøres*

Det skal nå bli sett på hvordan implementasjon og oppsett av de ulike måleverktøyene er blitt gjennomført. I figuren (fig. 4.25) kan vi se en illustrasjon av de målingen som skal gjøres og mellom hvilke ledd de vil måles.

### 4.4.1 SNMP måling med Orion

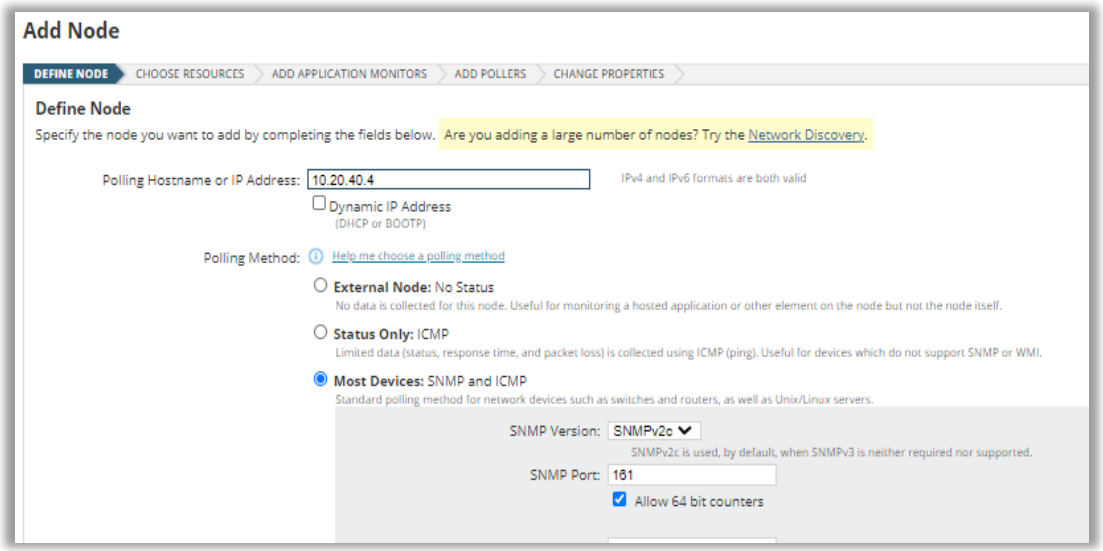
For å gjennomføre SNMP målinger benytter Eidsiva Bredbånd seg av en plattform kalt SolarWinds Orion. Det vil med denne plattformen være mulig å se hvilke hastigheter det er på ett punkt i nettet eller på det aktuelle interfaces som overvåkes. Det vil også være mulig å se annen relevant informasjon som responstid og pakketap, men de sistnevnte blir hentet ut ved bruk av ping. I Orion vil teknikken som benyttes være SNMP og ping, det vil sendes

pingforespørsler på gitte tidsintervaller. Der ser plattformen om det er noe tap mot svitsjens IP fra Eidsiva Bredbånd sin server og responstiden mellom serveren og endepunktet. Er det tap vil dette registreres i måleverktøyet og grafen som vises i Orion. På plattformen er det også mulig å se responstiden til utstyret.

Plattformen vil hente ut informasjon i bestemte tidsintervaller. Dette kan eksempelvis være hvert 5 minutt. Det vil derfor ikke være mulig med SNMP målinger å måle kontinuerlig. Samtidig som det vil komme en varsling om et interface går ned, noe som gjør at det blir varslet raskt og utbedring kan begynne.

**Essensielt oppsett på Orion**

Her vil hver node bli lagt inn manuelt, som vi har gjort i figuren (fig. 4.26). Plattformen krever at vi må inn og sjekke svitsjen vi har lagt inn etter eventuelle pakketap eller maksing av linje, og annen aktuell informasjon som vi ønsker å hente ut. Noden legges til ved bruk av IP og bruker SNMPv2. Den IP-adressen SNMP-målingen ble gjort mot var 10.20.40.254 og det ble gjort på fiberinterfacet. Da plattformen ikke ville gitt detaljerte resultater ved måling over 4G.




*Figur 4. 26: Hvordan SNMP-måling settes opp i bedriftens verktøy for SNMP-måling.*

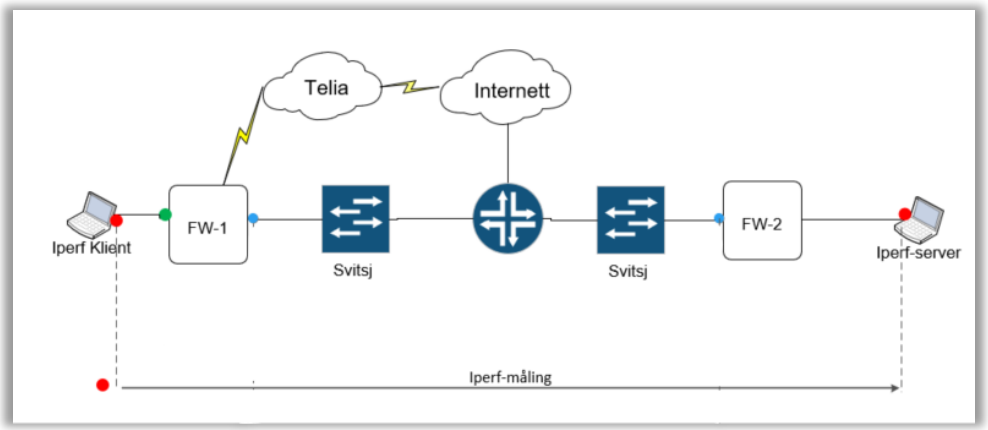
Deretter vil vi sette hvilke interface det ønskes å overvåke trafikken på, i dette tilfellet vil vi kun velge fiber. For å hente ut målingene gjort med SNMP vil vi logge inn på plattformen som kalles Orion, der utstyret er lagt inn til overvåkning. Der vil vi kunne søke på IP-adressen eller navnet på utstyret for å komme inn på siden som viser frem grafer basert på de gjennomførte målingene.

### 4.4.2 Iperf-målinger

For å ha mulighet til å gjennomføre Iperf målinger, var det behov for å sette opp en klientside og en server side. Dette er for at klienten skal ha en server å gjennomføre målingene mot. Det vil da sendes UDP eller TCP forespørsler mellom dem. For å sette opp både klient og serversiden må programmet for Iperf lastes ned fra nettside. Vi lastet ned fra denne siden:

https://iperf.fr/iperf-download.php (15042021)

For å kjøre dokumentet må det lastes ned i henhold til det operativsystemet som enheten det skal brukes på kjører, her var det Windows enheter. Programmet ble så pakket ut av en .zip-fil og iperf.exe var så tilgjengelig for å kunne kjøre i den aktuelle mappen som Iperf programmet ble lagt. For å kunne kjøre denne testen er det nødvendig å benytte seg av kommandovinduet til enheten det skal kjøre på.




*Figur 4. 27: Prinsippskisse med Iperf-målingne som gjennomføres*

Figuren (fig. 4.28) viser hvordan trafikken vil bevege seg under gjennomføringen av en måling ved bruk av Iperf. Målingene kan gjennomføres på begge laboppsettene. Måling over 4G må kjøres med bruk av en IPSec tunell mellom enhetene, mens målingene tatt over fiber, kan kjøres både med og uten IPSec tunell. Den er derfor ikke tegnet inn i prinsippskissen ovenfor.

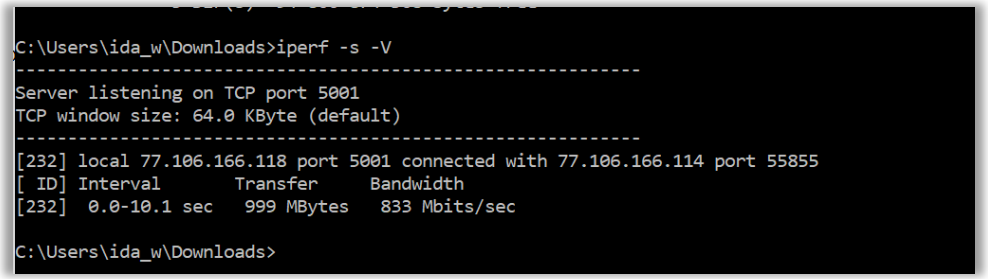
Endepunktene kjørt som Iperf-klient og Iperf-server var virtuelle maskiner med 8x2,4GHz virtuelle prosessorer og 8GB RAM. Med Windows 2012r2.

### Oppsett av Iperfserveren

Det må så settes opp en statisk internettaksess for å få mulighet til å gjennomføre målinger mot FW. Det nettet som ble kalt ett Iperf nett og består av et subnett som på FW1 er 192.168.100.2/30 og på FW2 er 192.168.101.2/30 er adressen til endepunktet.

Bakgrunnen for at testen ble valgt å kjøre på denne måten var for å kunne få flere reelle resultater, og det var ønskelig å teste over flere ledd, fremfor å kun teste via en svitsj. Det ble derfor lagt opp slik at all kommunikasjonen mellom enhetene i nettverket, her Iperf-server og Iperf-klienten skulle gå via en internettruter. Dette var for å få tilnærmet likt utgangspunkt ved målinger over fiber og over 4G. Det må også legges opp statiske ruter som trafikken skal følge. Dette ble konfigurert på begge FW.

I prosjektet er det valgt en enhet til å være serversiden, denne enheten vil da bli kjørt som en server i Iperf- testen. Dette gjøres ved å benytte –s (server) ved kall på testen (fig. 4.29).




*Figur 4. 28: Kommandoen som blir kjørt for å kjøre server og hvordan dette vil se ut når tester skal gjennomføres i kapittel 5.*

### Oppsett av Iperfklient

For å kunne kjøre måling ved bruk av Iperf her, er det nødvendig å bruke kommandovinduet på enheten til å komme seg inn i mappen hvor iperf.exe filen ligger. Deretter ble denne plasseringen brukt for å kjøre kommandoer. Kommandoene fra klienten ble rettet mot serveren ved å bruke IP adressen til serveren når det ble kjørt en måling.

**Initieringen av målingene blir gjort med følgende kommandoer**:

Serverside:

Hastighet: `Iperf3.exe -s -i 2`

Pakketap: `Iperf3.exe -s -u -p8080 -i 2`

Klientside:

Hastighet: `Iperf3.exe -c 192.168.101.2 -i 2 -t 180 -b 1000000000`

Pakketap: `Iperf3.exe -c 192.168.101.2 -u -b10M -i 2 -t 3000 -p8080`

For å avgjøre om enheten programmet kjørte på skal være server eller klient, brukte vi –s for server eller –c for klient. Målingene blir forespurt fra klientsiden, og kommandoen der inneholder også IPen til serveren for å si noe om hvor testen skal kjøres mot. TCP testen måler maksimal hastighet. Det var ikke nødvendig å spesifisere dette da det er standard for Iperf. Både på serversiden og klientsiden blir det satt 2 sekunders intervaller på når resultatene skal hentes ut og kommandoen for det var –i «ønsket tid i sekund» som i vårt tilfelle ble –i 2. For å få en bedre oversikt over måleresultatene blir testene kjørt i ulik lengde, dette med –t «ønsket tid i sekunder», denne kommandoen trengs kun å kjøre på klientsiden.

For å måle pakketap kjøres det en test med UDP, for å kunne gjøre dette ble kommandoen –u kjørt på både server og klientside. Testen blir kjørt på port 8080 med kommandoen –p8080. Ved UDP-tester kan det defineres hvor fort pakkene skal sendes. For å gjøre dette i denne testen brukte vi 10 Mbps båndbredde på målingene ved å bruke kommandoen –b10M hvor «b» står for båndbredde.

### 4.4.3 Måling gjort med TWAMP

Målinger som er gjort med TWAMP skulle opprinnelig gjøres med en sentral orkestrator. Da ville målingene kunne gjennomføres over tid slik at det ble en kontinuerlig måling. Etter mye problematikk omkring dette programmet ble det valgt å gjennomføre en øyeblikksmåling med et program som er tilgjengelig uten nødvendighet for linsens. Dette vil brukes for å illustrere to ulike måter som TWAMP kan benyttes. Både den kontinuerlige målingen og

øyeblikksmålingen bygger begge på grunnprinsippene av TWAMP, mens den kontinuerlige vil også ha en separat kontroller som kontrollerer alle sendere og respondere.

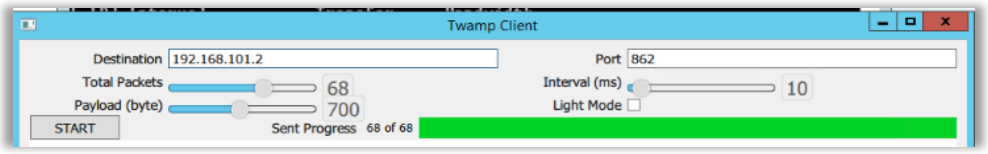
**Øyeblikksmåling**

Ved denne målingen vil det lastes ned ett program for klient og ett for server på enhetene som er koblet til LAN-porten på FW1 og FW2. Dette er enhetene som har IP 192.168.100.2 og IP 192.168.101.2.

Programmet er lastet ned via denne linken:

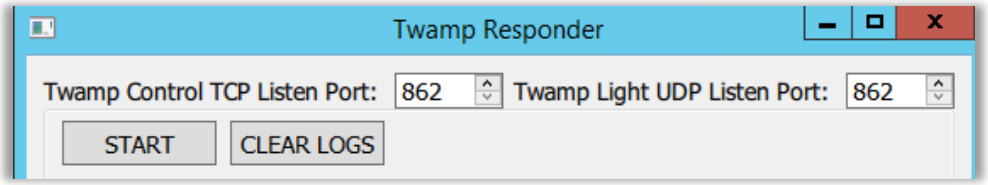
- https://github.com/demirten/twamp-gui

Enheten som kjører som klient vil være enheten 192.168.100.2 som vil kjøre på klienten bak FW1. Her stilles det inn innstillinger som tidsintervallet testen skal kjøres over, hvor mange pakker som skal overføres og antall byte som pakken skal ha i størrelse. I tillegg vil det legges inn IPen på destinasjonen som i dette tilfellet er 192.168.101.2. Videre vil det kun være nødvendig å trykke START for å gjennomføre målingen.




*Figur 4. 29: Innstillinger satt på TWAMP klienten*

Enheten som kjører som server vil være enheten 192.168.101.2, denne vil være responderen. Her vil det kun være nødvendig å starte serveren ved å trykke START.




*Figur 4. 30: Innstillinger satt på TWAMP-responder*

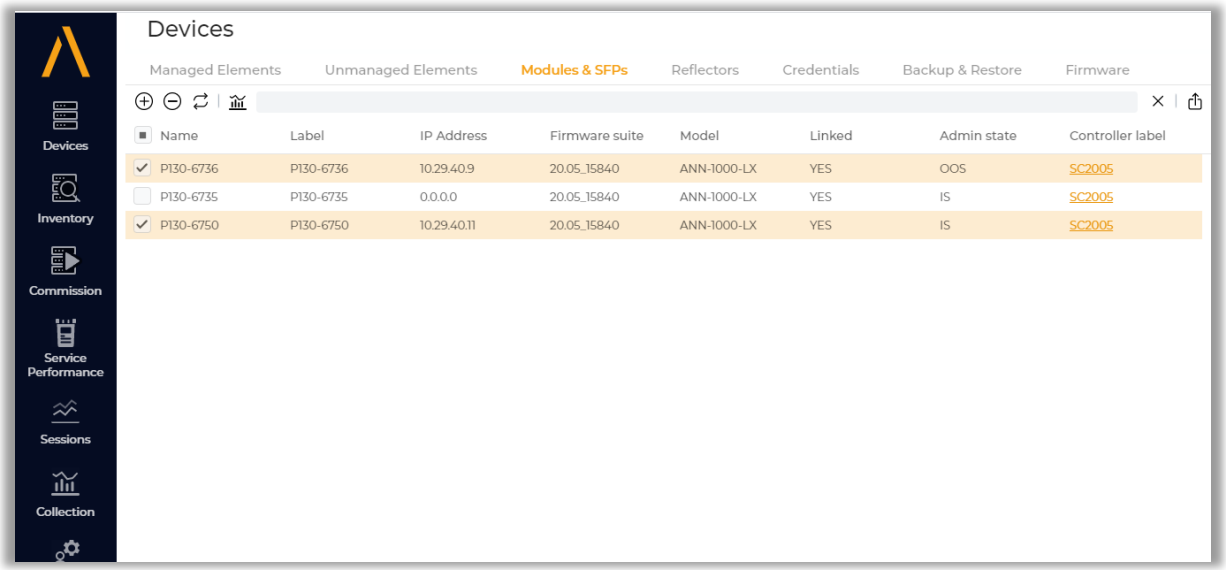
**Kontinuerlig måling**

Den kontinuerlige målingen er gjort med en plattform som kalles Accedian. Det ble her gjennomført målinger fra en smart SFP til en annen. Hvor den ene SFPen er programmert til å

være sender og den andre er programmert til å være reflektor. SFPer med software gjøres spesielt lite i dagens teknologi og er ganske nylig blitt mer tilgjengelig. Vi har tidligere i kapittelet sett på konfigurasjonen av de to SFPene som benyttes i prosjektet, hvor en står koblet i FW1 og den andre er koblet i FW2.

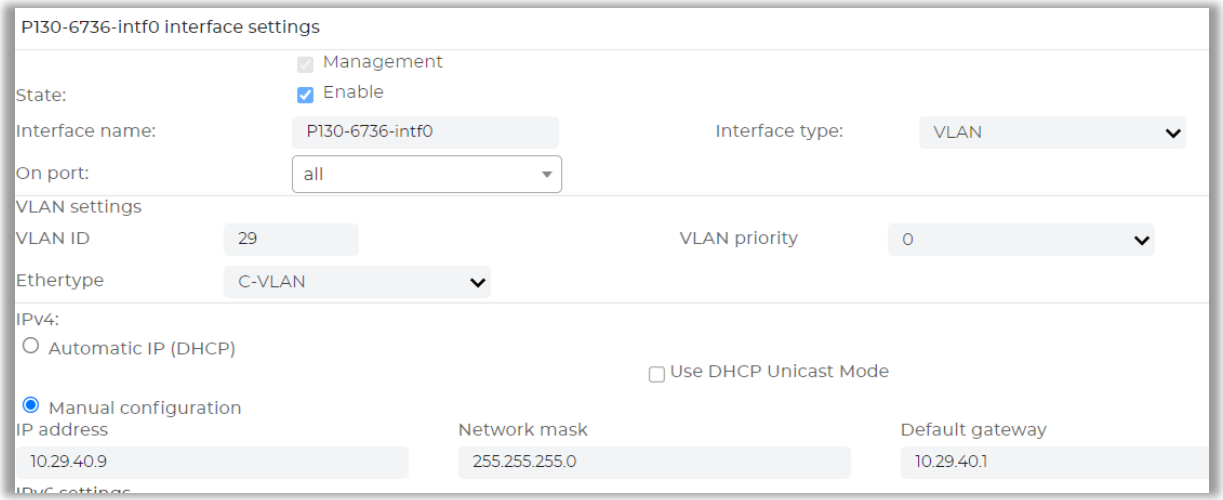
Oppsettet som denne plattformen benytter seg av vil være en sentral orkestrator. En orkestrator er en sentral server som administrerer multiple klienter. Orkestratoren vil holde oversikt over alle moduler og initiere tester i gitte tidsintervaller. Dette er tidsintervaller som settes opp ved hver målesesjon. Her vil orkestratoren også holde en oversikt over alle sesjonene som kjøres eller er tilgjengelige for å kjøres. Her vil det i hovedsak være fokus på orkestratoren opp mot sesjonssenderen og sesjonsreflektoren.

Med denne målingen vil orkestratoren være ett sentralt punk i nettet, hvor den har oversikt over alle måleenhetene som er i nettverket. Dette kan være alt fra SFPer til rutere og svitsjer som er kompatible til å konfigureres til å benytte TWAMP. I dette prosjektet vil fokuset være på å benytte seg av SFPer for å gjøre dette. Figuren nedenfor (fig. 4.32) viser hvordan første siden på orkestratoren vil se ut med oversikt over alle modulene i nettverket. Her ser vi tre moduler, men det er kun to av dem vi benytter oss av. De vi benytter er de som er gule og de har IP: 10.29.40.9 og 10.29.40.11. Dette er management adressen til SFPene og når det gjennomføres en måling vil de ha en virtual customer edge (VCE)-adresse.



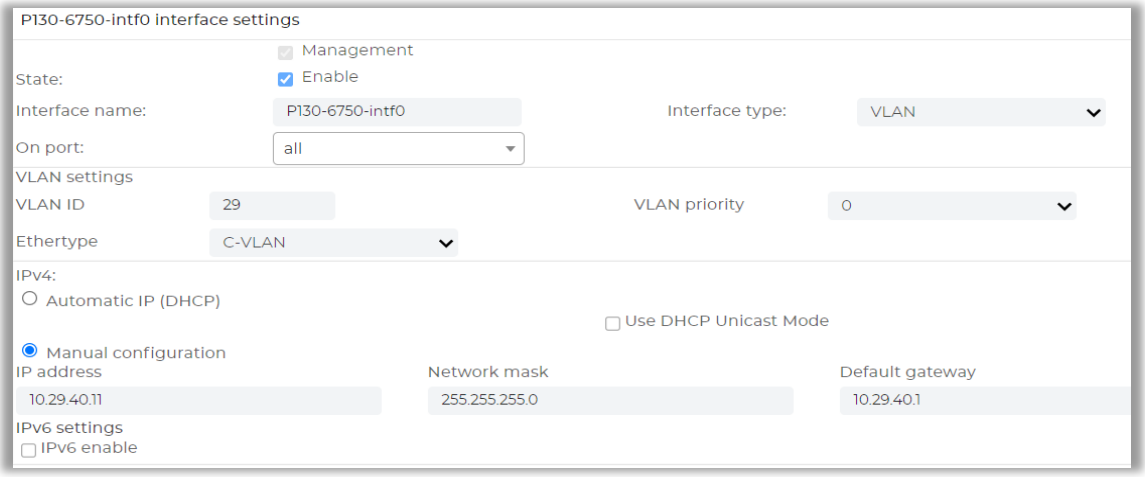

*Figur 4. 31: Devices på Accedian sin plattform.*

På denne plattformen vil det også settes innstillinger på SFPene, her vil SFP1 ha innstillingene vi kan se i figuren (fig. 4.33).



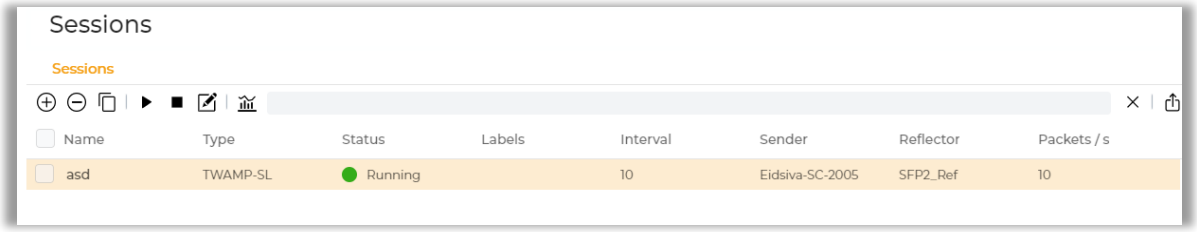

*Figur 4. 32: Innstillingene satt på SFP1*

Videre vil det også bli satt innstillinger på SFP2. Som vi kan se i figuren (fig, 4.34).



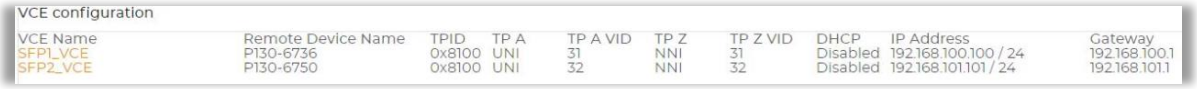

*Figur 4. 33: Innstillingene satt på SFP2*

Her vil den ene SFPen settes opp som en reflektor og den andre vil settes opp som en sender. Dette er viktig for at TWAMP skal kunne kjøre en måling da den avhenger av å ha en sender og en reflektor til å reflektere pakkene. Målingene som gjøres her bygger på de målingene som ble gjort med øyeblikksmåling, det er samme prinsipp, bare at i dette tilfellet er kontrollenheten trekket ut (orkestratoren) for å kunne ha en sentral enhet til å styre målingene som gjøres. Det vil settes opp ny sesjon ved å trykke på (+) tegnet i figuren nedenfor (fig. 4.35).



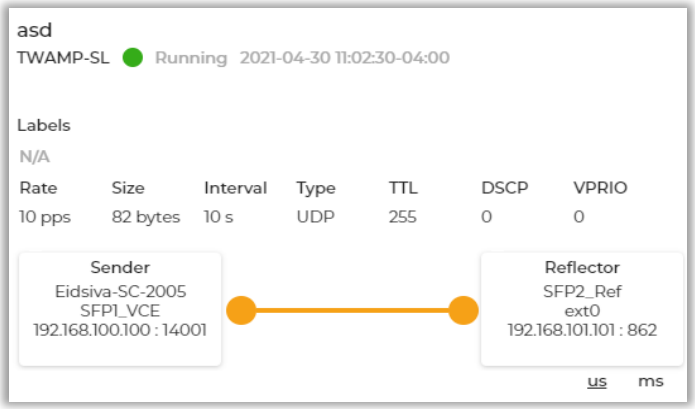

*Figur 4. 34: Sesjoner i Accedian.*

De IP-adressene som er gitt tidligere angir management adressene til enhetene. For at testen skal kunne gjennomføres vil de få en VCE-adresse (fig. 4.36).




*Figur 4. 35: VCE-konfigurasjonen til SFP1 og SFP2.*

I figuren nedenfor (fig. 4.37) vil vi kunne se at en sesjon vil ha en sender og en reflektor. Her kan vi også se hvilken SFP som er senderen og hvilken som er reflektor. Vi kan se at sender ha en IP-adresse på 192.168.100.100 dette tilsvarer SFP1 med managementadresse: 10.29.40.9. SFP1 er koblet i FW1. Tallet vi ser bak IP-adressen 14001 er porten som sendingen vil bli sendt over, denne har Accedian bestemt at må være mellom 14000 og 16000. For reflektoren vil den ha IP-adresse 192.168.101.101 og det vil tilsvare SFP2 med managementadresse: 10.29.40.11. Denne SFPen står koblet i FW2. Videre vil vi se ett tall bak IP-adressen. Dette tilsvarer port 862, denne porten er standard for reflektor i TWAMP. Samme port som er brukt ved øyeblikks måling.




*Figur 4. 36: En ferdig oppsatt sesjon som kjører.*

# 5 Test og resultat

I forrige kapittel ble det sett på flere målemetoder og hvordan disse er konfigurert eller implementert. Her vil det bli sett på resultatet disse målemetodene over laboppsettene. Strukturen i kapittelet vil bygge på at resultatet for en målemetode blir presentert først, før det vil presenteres en oppsummering av resultatene.

## 5.1 Ping

Det ble sendt ping-forespørsel fra den ene klienten i nettverket til den andre ved å bruke kommandoen: `ping «IP-adressen til klienten det måles mot»`.

Ping-forespørselen ble kjørt fra kommandolinjen som er forhåndsinstallert på aller Windows-maskiner. Testene ble kjørt med samme belastning og mot samme adresse. Målingene ble gjort både på fiber uten IPSec og over 4G med IPSec, vi har ikke tatt med resultat for fiber med IPSec da resultatet her ble identisk til resultatet for fiber uten IPSec.

**Over fiber uten IPSec**

Resultatet for målingen (fig. 5.1) viser at det ble sendt 32 bytes i hver av pakkene, og at alle pakkene brukte 3 millisekunder på å returnere. Det var ingen pakketap i denne målingen.


*Figur 5. 1: Resultat for Ping over fiber.*

**Over 4G med IPSec**

Resultatet for målingen (fig. 5.2) som ble gjort over 4G med IPSec, viser at pakkene bruker betraktelig lengre tid på å returnere sammenlignet med over fiber, det er også større variasjon i tiden de bruker på å returnere. Snittiden vil være på 98 ms, mens den tregeste pakken brukte 152 ms på å returnere. Det var ingen pakketap på målingen.


*Figur 5. 2: Resultater for Ping over 4G med IPSec*

### 5.1.1 Oppsummering av Ping-resultater

I tabellen (tab. 4) kan vi se en oppsummering av måleresultatene for Ping. Dette for å kunne sammenligne målingene på en bedre måte.

*Tabell 4: Oppsummerer av Ping-resultater*

| | **Fiber uten IPSec** | **4G med IPSec** |
|---|---|---|
| **Pakketap** | 0% | 0% |
| **Minimumstid på rundetid for en pakke** | 3 ms | 75 ms |
| **Gjennomsnittstid på rundetiden for pakkene** | 3 ms | 98 ms |
| **Maksimumstid på rundetiden for pakkene** | 3 ms | 152 ms |

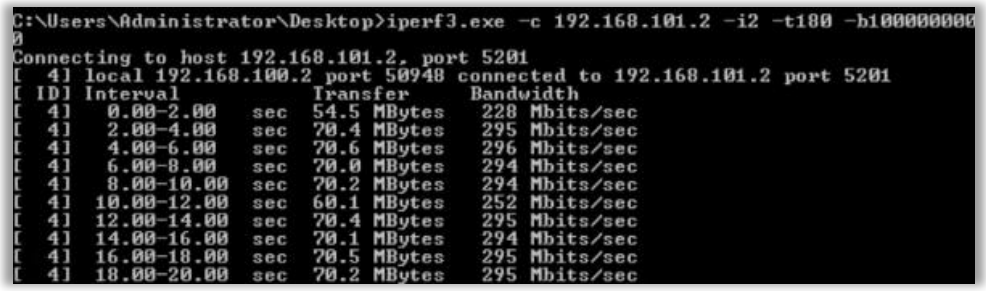
## 5.2 Iperf

Iperf-målingene som ble gjennomført, ble gjort både over UDP og TCP. Her vil TCP stå for hastighetsmålingen og UDP stå for målingen av pakketap. Det skal nå bli sett på de resultatene vi fikk med fiber uten IPSec, fiber med IPSec og 4G med IPSec, når det kommer til båndbredden på linjen og eventuelle pakketap.

### 5.2.1 Resultat Iperf

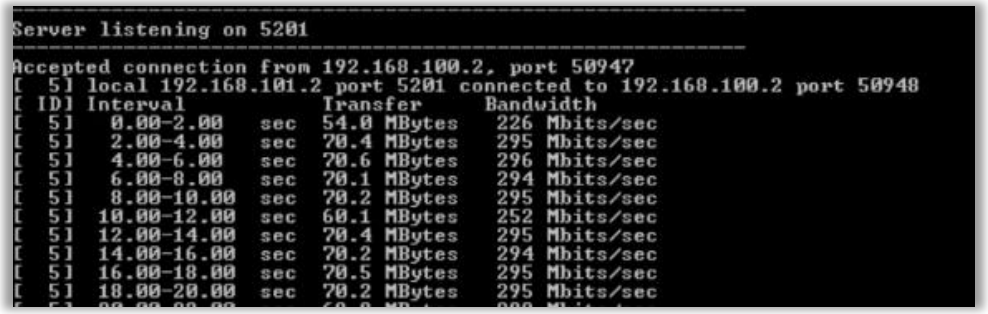
**Resultat på måling med Iperf over fiber uten bruk av IPSec**

I resultatet under (fig. 5.3) er Iperf-test kjørt med TCP fra klientsiden. Resultatet viser antall megabytes overført per andre sekund og hvor mye båndbredde («Bandwidth») som ble brukt. «Transfer» er hvor mye data som har blitt overført i en periode. Resultatene er hentet ut hvert 2 sekund og viser de 20 første sekundene målingene ble gjennomført. Som vi kan se på resultatet var det jevnt over en båndbredde rundt 295 Mbps.



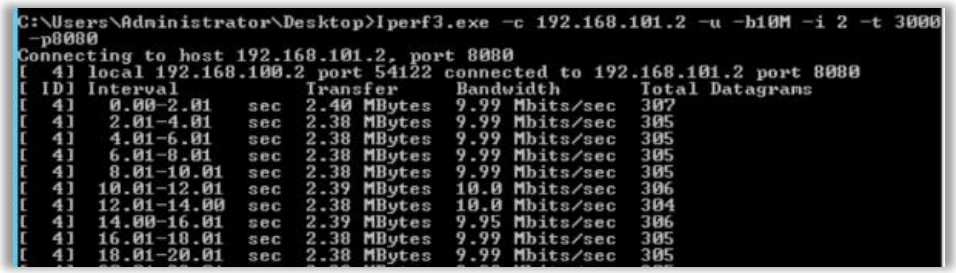

*Figur 5. 3: Iperf-resultat for fiber uten IPSec; hastighet (klientsiden)*

Resultatet under (fig. 5.4) viser hastighetsmåling sett fra serversiden. Det er tilnærmet ingen avvik fra resultatene på klientsiden. Dette gjør at videre resultat fra hastighetsmåling ikke vil tas med fra både server og klient.



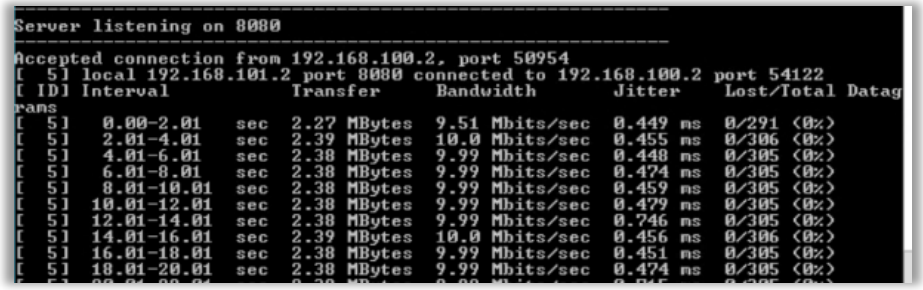

*Figur 5. 4: Iperf-resultat for fiber uten IPSec: hastighet (serversiden)*

Resultatet fra klientsiden (fig. 5.5) viser måling av pakketap ved bruk av UDP. Som vi kan se under så holdt båndbredden seg stabilt på 9,99 Mbps. Men dersom denne ses opp mot hastighets-målingen sitt resultat så overføres det en god del mindre data.




*Figur 5. 5: Iperf-resultat for fiber uten IPSec: pakketap (klientsiden)*

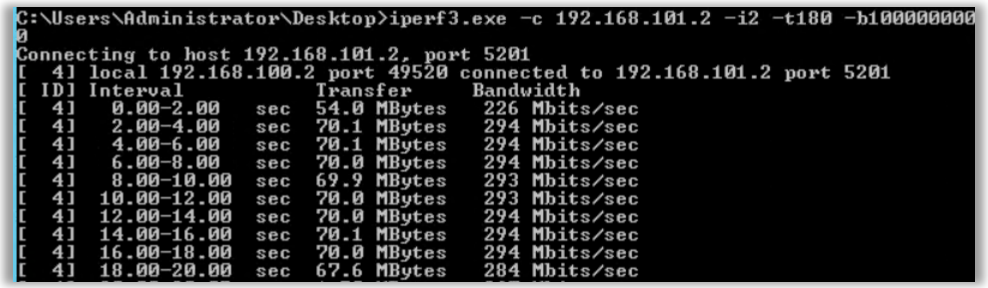
Resultatene fra serversiden (fig. 5.6) kjørt med UDP under viser lite avvik fra klientsiden. Jitter-verdiene er lave, og det er lite variasjon i dem. Som vi kan se under Lost/Total Datagrams så var det 0% pakketap. Noe som vil tilsi at ingen pakker ble droppet eller borte under overføringen. Det vil videre kun bli tatt med resultatet for pakketap sett fra serversiden, da resultatene fra klient og serversiden er tilnærmet identiske, men serversiden vil inneholde antall pakker tapt.




*Figur 5. 6: Iperf-resultat for fiber uten IPSec: pakketap (serversiden)*

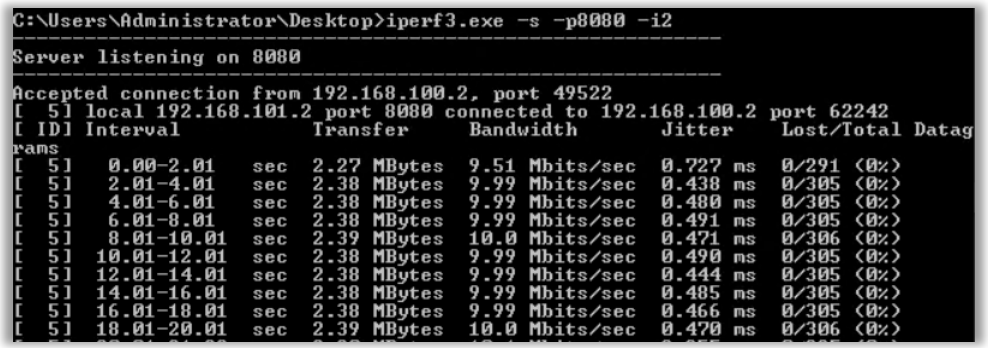
**Resultater ved måling over fiber med IPSec**

Resultatene ved gjennomføring av hastighetsmåling over fiber med IPSec (fig. 5.7) ga det lignende resultater som kunne bli sett på fiber uten IPSec. Resultatet var jevnt over på 294Mbps som båndbredde.




*Figur 5. 7: Iperf-resultat for fiber med IPSec: Hastighet*

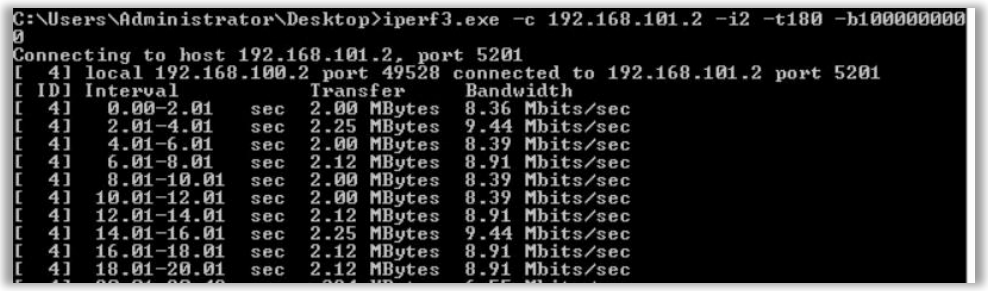
Det ble deretter hentet ut ett resultat for måling av pakketap (fig. 5.8). I resultatet kan vi se at båndbredden varierte i hovedsak mellom 9,99 Mbps og 10 Mbps. Videre kan vi se Jitterstørrelsen på de overførte pakkene og den var på maksimalt 0,777 ms og den laveste jitteren var på 0,444 ms. Til slutt kan vi se antall datagram tapt og mottatt, og her er alle datagram mottatt og ingen tapt, derav 0% pakketap.




*Figur 5. 8: Iperf-resultat for fiber med IPSec: pakketap*

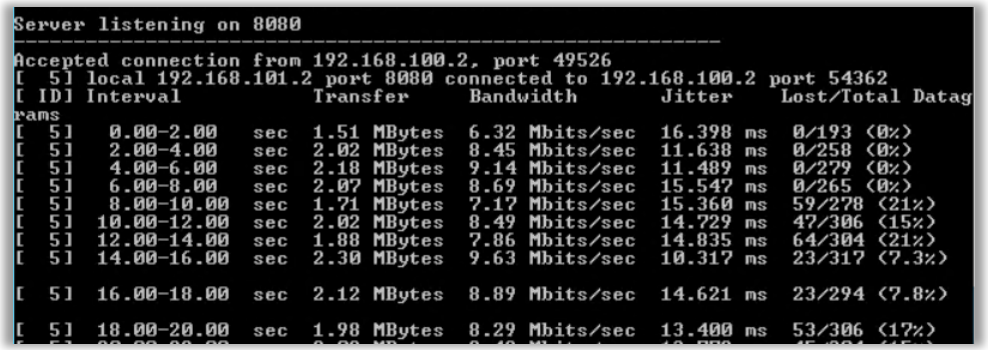
**Til slutt har vi resultatene ved måling gjort over 4G med IPSec**

Resultatet ved gjennomføring av hastighetsmåling over 4G med IPSec (fig. 5.9) ga varierende båndbredde. Her varierer båndbredden mellom 8,36 Mbps og 9,44 Mbps. Antall bytes overført vil her være rundt 2 MBytes hvert andre sekund.




*Figur 5. 9: Iperf-resultat for 4G med IPSec: Hastighet*

Det siste resultatet til 4G med IPSec er måling av pakketap (fig. 5.10). Som vi kan se på resultatet vil det være noen pakker som går tapt under sendingen. Det vil være en overføring på mellom 1,5 MBytes og 2,30 MBytes. Som vi kan se i resultatet vil det på oppstå pakketapp etter 8 sekunder være det første tapet, her vil 59 av 278 bli tapt, noe som vil resultere i et pakketap på 21%. Etter denne pakken vil det også være pakketap i resterende tidsintervaller. Her vil pakketapet variere fra 7,3% til 21%.




*Figur 5. 10: Iperf-resultat for 4G med IPSec: pakketap*

Til slutt har vi ett resultat på en brutt link (fig. 5.11). Dette resultatet er lagt ved som et eksempel på hvordan linjen vil se ut ved brudd på kabel. Mens testen kjørte, bøyde vi på kabelen. Testen viser mye større variasjon i jitter og pakketap. Det er omtrent en femtedel av det som blir sendt som kommer frem til serveren.

```
C:\Users\ida_w\Downloads\iperf>iperf.exe -s -u -p8080 -i 2
------------------------------------------------------------
Server listening on UDP port 8080
Receiving 1470 byte datagrams
UDP buffer size: 64.0 KByte (default)
------------------------------------------------------------
[360] local 77.106.166.118 port 8080 connected with 193.181.246.221 port 48321
[ ID] Interval       Transfer     Bandwidth        Jitter   Lost/Total Datagrams
[360]  0.0- 2.0 sec   472 KBytes  1.93 Mbits/sec   4.470 ms    0/  329 (0%)
[360]  2.0- 4.0 sec   484 KBytes  1.98 Mbits/sec   4.244 ms    0/  337 (0%)
[360]  4.0- 6.0 sec   561 KBytes  2.30 Mbits/sec   4.816 ms    0/  391 (0%)
[360]  6.0- 8.0 sec   531 KBytes  2.18 Mbits/sec   4.487 ms    0/  370 (0%)
[360]  8.0-10.0 sec   570 KBytes  2.33 Mbits/sec   4.560 ms    0/  397 (0%)
[360] 10.0-12.0 sec   379 KBytes  1.55 Mbits/sec  14.029 ms  319/  583 (55%)
[360] 12.0-14.0 sec   408 KBytes  1.67 Mbits/sec   4.751 ms    8/  292 (2.7%)
[360] 14.0-16.0 sec   515 KBytes  2.11 Mbits/sec   4.668 ms 1025/ 1384 (74%)
[360] 16.0-18.0 sec   620 KBytes  2.54 Mbits/sec   4.408 ms 1526/ 1958 (78%)
[360] 18.0-20.0 sec   567 KBytes  2.32 Mbits/sec  27.563 ms 2615/ 3010 (87%)
[360] 20.0-22.0 sec   567 KBytes  2.32 Mbits/sec   9.313 ms 2451/ 2846 (86%)
[360] 22.0-24.0 sec   482 KBytes  1.98 Mbits/sec   4.492 ms    0/  336 (0%)
[360] 24.0-26.0 sec   504 KBytes  2.06 Mbits/sec   3.797 ms 1332/ 1683 (79%)
[360] 26.0-28.0 sec   597 KBytes  2.45 Mbits/sec   3.851 ms 1365/ 1781 (77%)
[360] 28.0-30.0 sec   620 KBytes  2.54 Mbits/sec  74.292 ms 2127/ 2559 (83%)
[360] 30.0-32.0 sec   570 KBytes  2.33 Mbits/sec   7.191 ms 1740/ 2137 (81%)
[360] 32.0-34.0 sec   505 KBytes  2.07 Mbits/sec  17.712 ms 1172/ 1524 (77%)
[360] 34.0-36.0 sec   659 KBytes  2.70 Mbits/sec   2.601 ms 1138/ 1597 (71%)
[360] 36.0-38.0 sec   755 KBytes  3.09 Mbits/sec   2.628 ms 1028/ 1554 (66%)
[360] 38.0-40.0 sec   718 KBytes  2.94 Mbits/sec   3.929 ms 2287/ 2787 (82%)
[360]  0.0-41.3 sec  11.2 MBytes  2.28 Mbits/sec  25.906 ms 20442/28435 (72%)
```

*Figur 5. 11: Iperf-resultat for fiber med defekt kabel*

Som vi kan se her gjør denne testmetoden det kun mulig å måle et øyeblikksbilde av hvordan linjen til endepunktet ser ut der og da. Dette gjør at vi ikke vil kunne tilsi om det er pakketap eller dårlig hastighet som har skjedd ved spesielle klokkeslett, eller i løpet av den siste tiden, vi vil kun ha mulighet til å se om det er pakketap i det øyeblikket vi har initiert målingene.

### 5.2.2 Oppsummering av Iperf-resultater

**TCP-resultater:** I tabellen under (tab. 5) er resultatene for TCP-målingene over de ulike nettverkene samlet. Det er bare tatt med resultatene fra klientsiden, da serversiden viser tilsvarende resultater.

*Tabell 5: Oppsumering av Iperf hastighetsmåling*

| | Fiber uten IPSec | Fiber med IPSec | 4G med IPSec |
|---|---|---|---|
| **Maksimum data overført** | 70,6 MB | 70,1 MB | 2,25 MB |
| **Minimum data overført** | 54,5 MB | 54,0 MB | 2,00 MB |
| **Maksimum båndbredde** | 296 Mb | 294 Mb | 9,44 Mb |
| **Minimum båndbredde** | 228 Mb | 226 Mb | 8,36 Mb |

**UDP-resultater:** Tabellen under (tab. 6) viser resultatene vi fikk med UDP-målingene. Resultatene er fra serversiden, da den inneholdt informasjon om jitter i tillegg til informasjonen som var på klientsiden.

*Tabell 6: Oppsummering av Iperf tapsmåling*

| | Fiber uten IPSec | Fiber med IPSec | 4G med IPSec | Fiber med ødelagt kabel |
|---|---|---|---|---|
| **Maksimum data overført** | 2,39 MB | 2,39 MB | 2.30 MB | 11,2 MB |
| **Minimum data overført** | 2,27 MB | 2,27 MB | 1,51 MB | 379 KB |
| **Maksimum båndbredde** | 10,0 Mb | 10,0 Mb | 9,63 Mb | 3,09 Mb |
| **Minimum båndbredde** | 9,51 Mb | 9.51 Mb | 6,32 Mb | 1,55 Mb |
| **Maksimum jitter** | 0,746 ms | 0,727 ms | 16,398 ms | 74,292 ms |
| **Minimum jitter** | 0,448 ms | 0,438 ms | 10,317 ms | 2,601 ms |
| **Maksimum pakketap** | 0 % | 0 % | 21 % | 87 % |

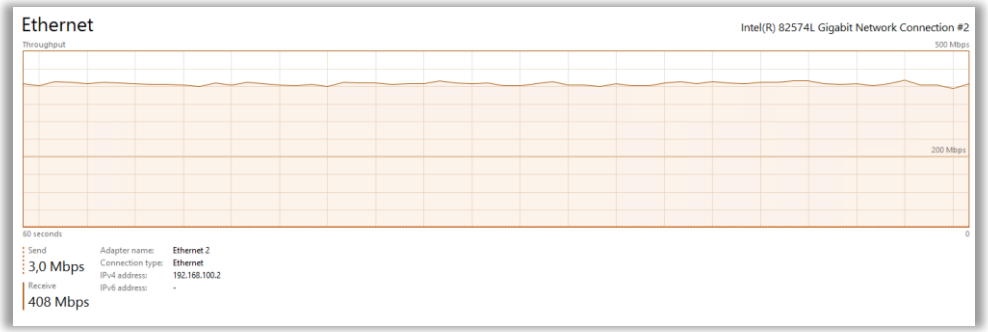
## 5.3 SNMP

De måleresultatene som blir presentert i denne oppgaven er bare gjort over fiber. Dette fordi vi mener at det gir et godt nok innblikk i hvordan SNMP-målinger fungerer, og at målinger over 4G ville gitt tilsvarende resultater.

For å gjennomføre målingene ble det lastet ned store pakker for å belaste nettverket. I den første testen ble det lastet ned en fil som brukte omtrent ett minutt på å laste ned fra internett. Under den andre testen ble det lastet ned flere store filer for å belaste nettverket over en lengre periode. Testene ble gjennomført med noe tid mellom, slik at måleresultatene kan skilles fra hverandre.

**Faktisk brukt båndbredde ved nedlastning**

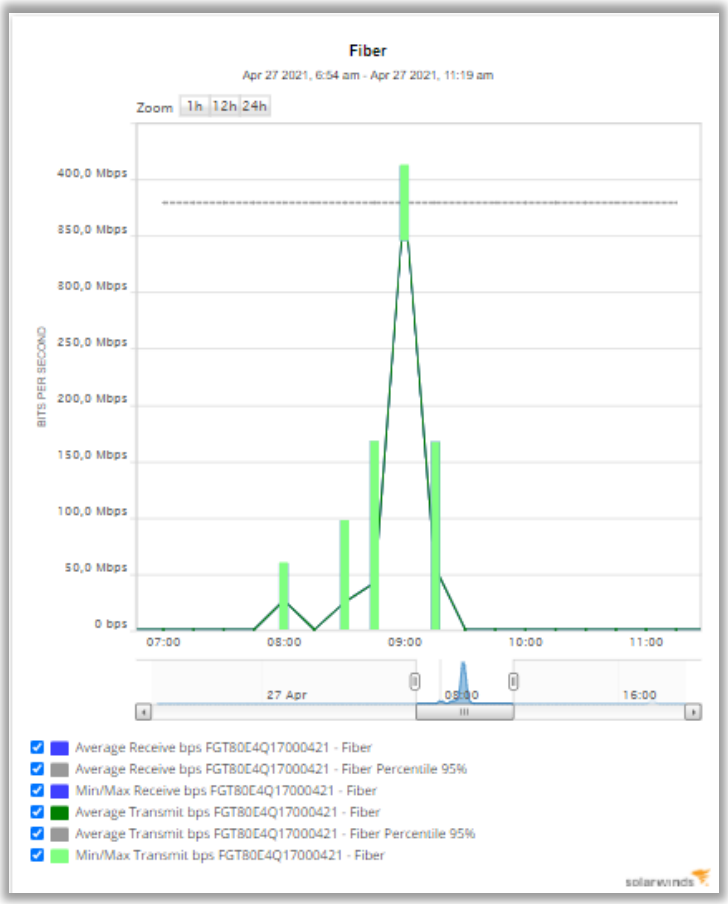
For å generere nettverkstrafikk og fremprovosere økt trafikk i gitte tidsrom ble det lastet ned en eller flere ISO-filer. Hver fil var på en størrelse av 2,5GB. Dette ble gjort for å kunne gi ett innblikk til hvordan verktøyet brukt til å måle med SNMP (Orion). For å få trafikk på ett kort tidsrom ble det kun gjort nedlastning av en slik fil, og tiden for nedlastningen var da på rundt ett minutt. Filene brukte omkring 400Mbps i båndbredde ved nedlastning. Ved generering av nettverkstrafikk over ett større tidsrom (fig. 5.12) ble flere slike filer lastet ned samtidig. Dette resulterte i en varighet på 15-20 minutter for å fullføre nedlastningen av alle filene, og den benyttede båndbredden var da på 400Mbps, slik som vi kan se det også var på nedlastning over et kortere tidsrom.




*Figur 5. 12: Figur som viser utsnitt av brukt båndbredd under nedlastning for SNMP*

### 5.3.1 Resultat SNMP

Resultatet for SNMP kan bli sett i grafen (fig. 5.13). I denne grafen vil vi fokusere på de lysegrønne søylene som viser minimum og maksimumsverdien på overføringen av fiber. De to første søylene i grafen viser resultatet av nedlastningen av en fil, på ett minutt, på to forskjellige tidspunkter. Resultatet viser at nettverket hadde en maksimal belastning på 60 Mbps ved første nedlastning og en maksimal belastning på i underkant 100 Mbps ved den andre nedlastningen. Dette er til tross for at filene nedlastet benyttet en båndbredde på 400 Mbps. De tre siste søylene viser nedlastningen av flere filer samtidig der nedlastningene tok 15-20 minutter til sammen. Resultatet viser at det ble brukt en maksimal båndbredde på over 400 Mbps.




*Figur 5. 13: SNMP-resultater ved bruk av en SNMP-applikasjon*

### 5.3.2 Oppsummering av resultater med SNMP

Oppsummering av resultatet (tab. 7) viser verdiene som kan leses ut fra SNMP-målingene opp mot faktisk benyttet maksimal båndbredde i samme tidsrom. Verdiene i tabellen er omtrentlige da de er lest ut av grafer.

*Tabell 7: Oppsummering SNMP-måling*

| | **08:00** | **08:15** | **08:30** | **08:45** | **09:00** | **09:15** |
|---|---|---|---|---|---|---|
| **Maksimum båndbredde målt med SNMP** | 60 Mbps | 0 Mbps | 100 Mbps | 170 Mbps | 410 Mbps | 170 Mbps |
| **Reelt maksimum båndbredde** | 350-400 Mbps | 0 Mbps | 350-400 Mbps | 400-420 Mbps | 400-420 Mbps | 400-420 Mbps |

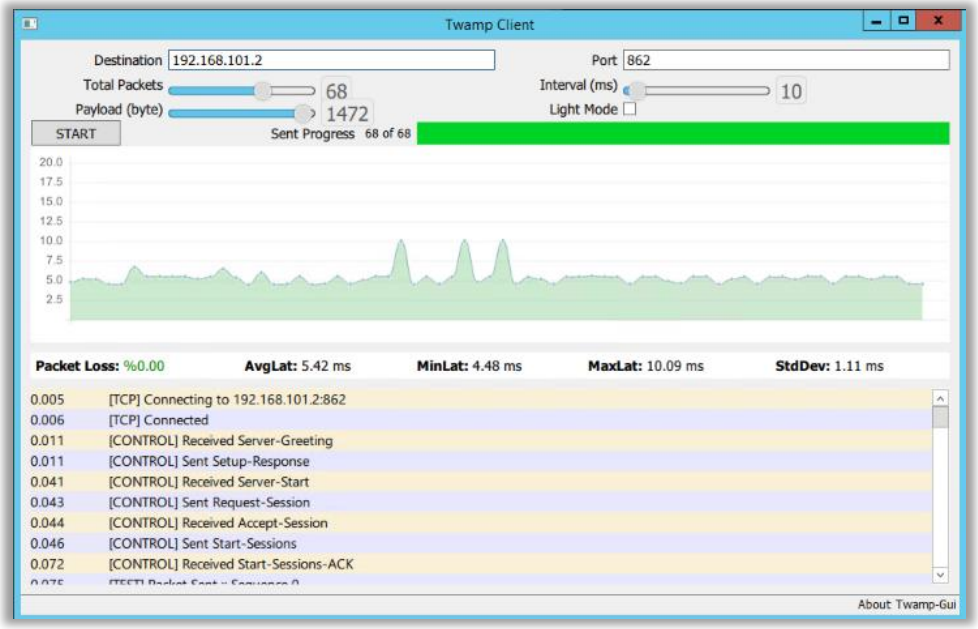
## 5.4 TWAMP

Ved gjennomføringen av TWAMP målinger ble det gjort med to forskjellige pakkestørrelser. Både maksimal pakkestørrelse på 1472 Bytes og med halv pakkestørrelse på 700 Bytes. Pakkene ble sendt med 10 ms intervaller, som var det laveste intervallet som kunne velges.

### 5.4.1 Resultat TWAMP - Øyeblikksmåling

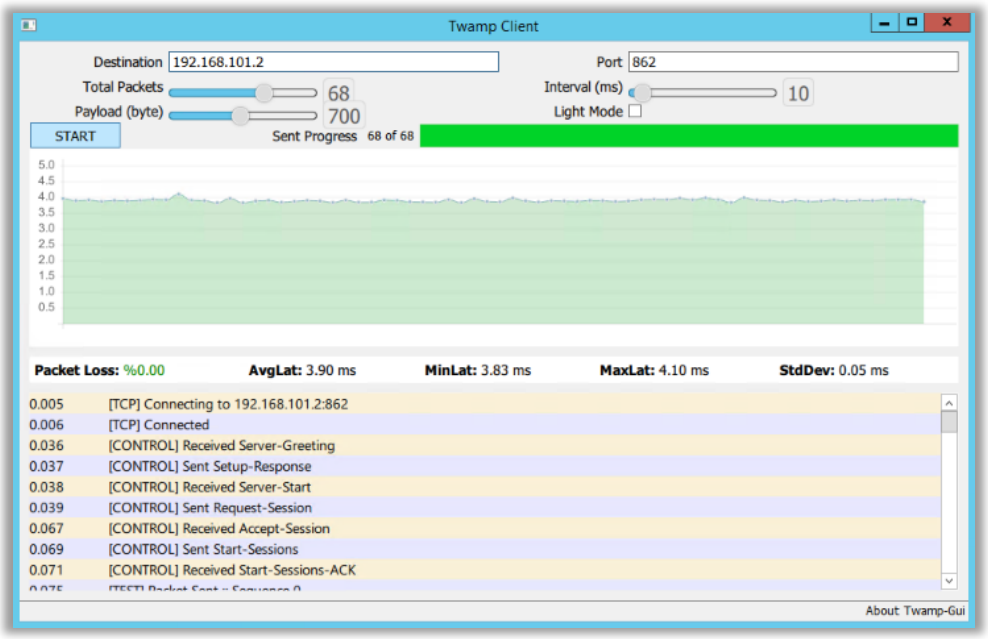
**Resultat over fiber**

Med maksimal pakkestørrelse over fiber uten IPSec (fig. 5.14) kan vi se at det er en gjennomsnittlig forsinkelse på levering av pakkene på 5,42 ms. Standard avviket ligger på 1,11 ms med en maksimal forsinkelse på 10,09 ms og en minimal forsinkelse på 4,48 ms.




*Figur 5. 14: TWAMP-resultat over fiber uten IPSec med full payload*

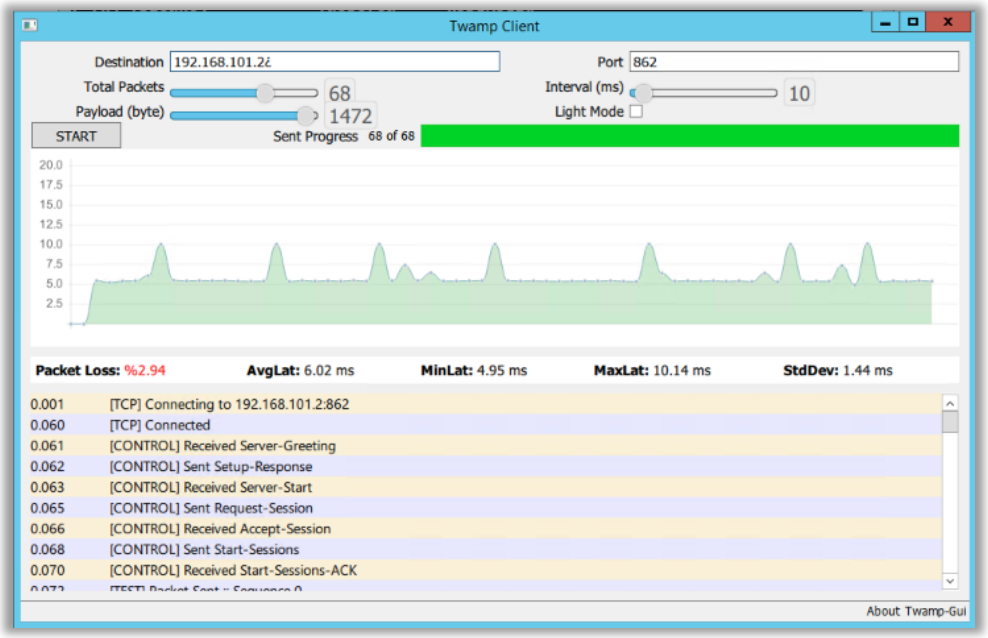
Dersom pakkestørrelsen reduseres til 700 Bytes så vil det ikke være en like stor variasjon på resultatene av de 68 pakkene som blir sendt (fig. 5.15). Det vil her kun være en gjennomsnittlig forsinkelse på 3,90 ms og standard avviket ligger på 0,05 ms. Maksimal og minimal forsinkelse vil her være 4,10 ms og 3,83 ms.




*Figur 5. 15:TWAMP-resultat over fiber uten IPSec med halv payload*

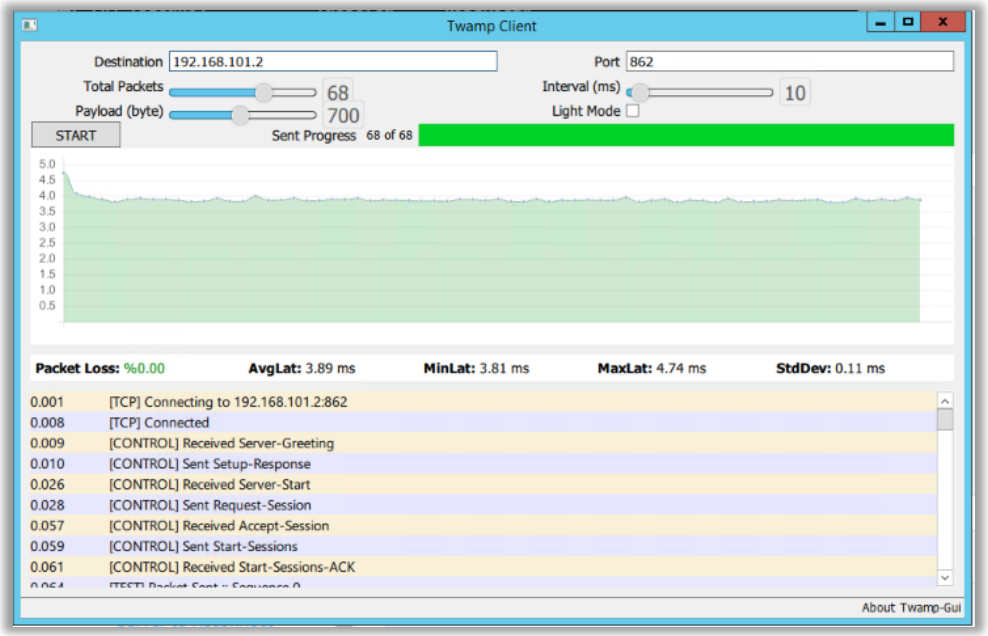
### Resultat over fiber med IPSec

Ved pakker send med en maksimal pakkestørrelse (fig. 5.17), her vil den gjennomsnittlige forsinkelsen ligge på 6,02 ms og ha et standard avvik på 1,44 ms. Den maksimale forsinkelsen vil ligge på 10,14 ms og minimale forsinkelsen vil ligge på 4,95 ms. Vi ser også 2,94 % pakketap.




*Figur 5. 16:TWAMP-resultat over fiber med IPSec med full payload*

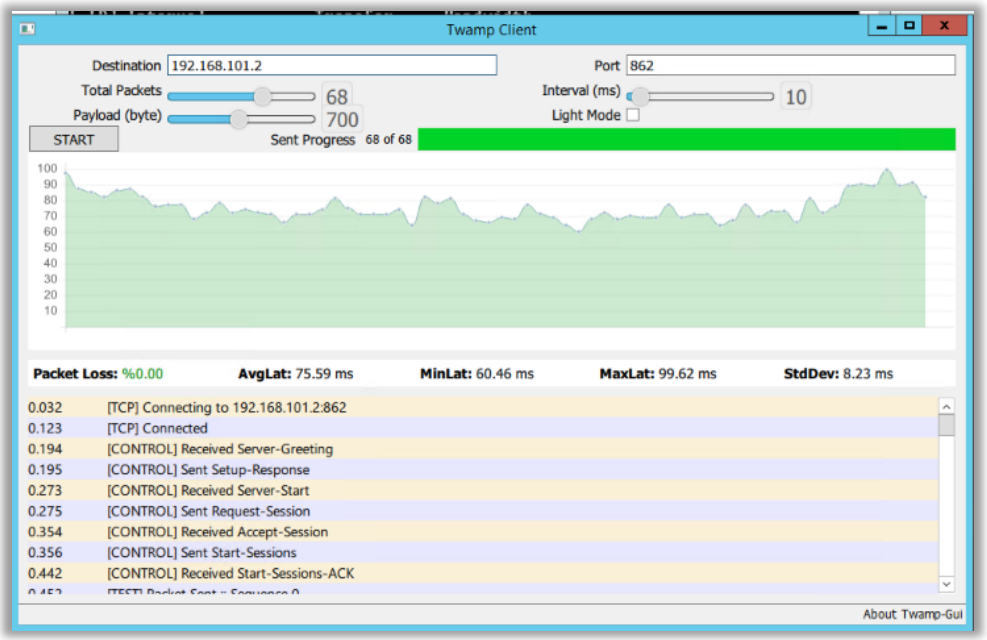
Når vi kjørte måling med en mindre pakkestørrelse over IPSsec (fig. 5.18) vil vi ha en gjennomsnittlig forsinkelse på 3,89 ms og et standard avvik på 0,11 ms. Maksimal og minimal forsinkelse vil ligge på 4,74 ms og 3,81 ms.




*Figur 5. 17: TWAMP-resultat over fiber med IPSec med halv payload*

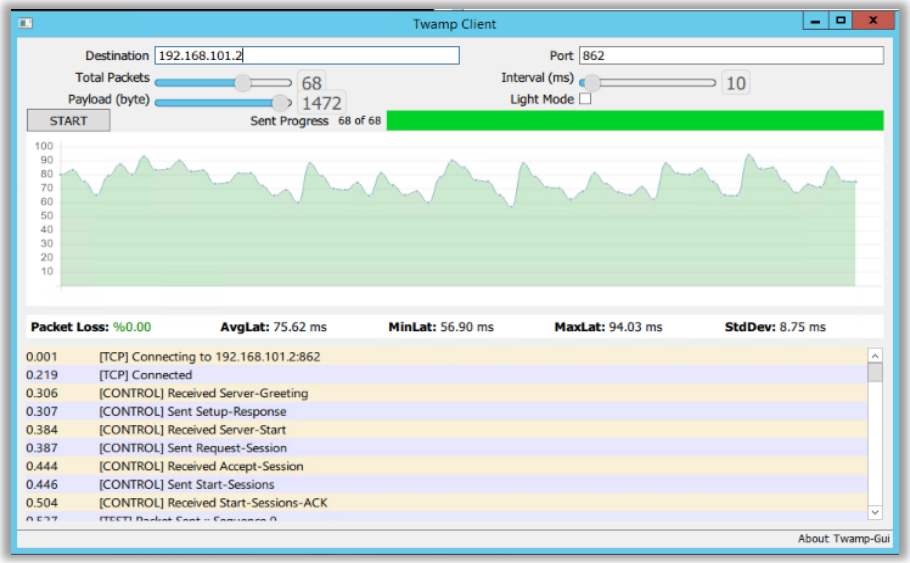
**Resultat over 4G med IPSec**:

Det ble gjennomført flere målinger over 4G enn det ble gjort over fiber. Dette er fordi vi opplevde større variasjoner når vi fikk resultatene underveis i målingene. Siden den største andelen av resultatene stammer fra måling med maksimal pakkestørrelse, begynner vi med å presentere resultatet fra målingene gjennomført med en pakkestørrelse på 700 Bytes (fig. 5.18). Her vil vi ha en gjensidig forsinkelse på 75,59 ms og et standard avvik på 8,23 ms. Den maksimale forsinkelsen for en pakke var 99,62 ms og den minimale forsinkelsen var 60,46 ms.



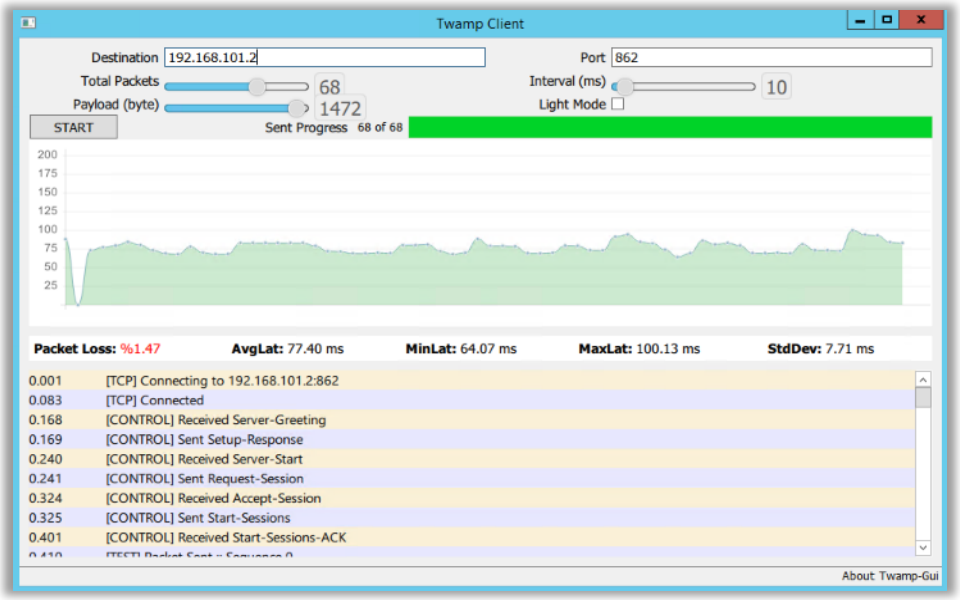

*Figur 5. 18: TWAMP-resultat over 4G med IPSec med halv payload*

Videre skal det bli sett på de resultatene vi fikk ved gjennomføring av målinger med en pakkestørrelse på 1472 Bytes. De første resultatene med måling av TWAMP over 4G kan vi se nedenfor (fig. 5.19). Og som vi kan se ut fra resultatet så varierer det en del mer enn hva det gjør med fiber. Resultatet vil ha en gjennomsnittlig forsinkelse på 75,62 ms og et standard avvik på 8,75 ms. Videre vil den ha en maksimal forsinkelse på 94,03 ms og en minste forsinkelse på 56,90 ms.



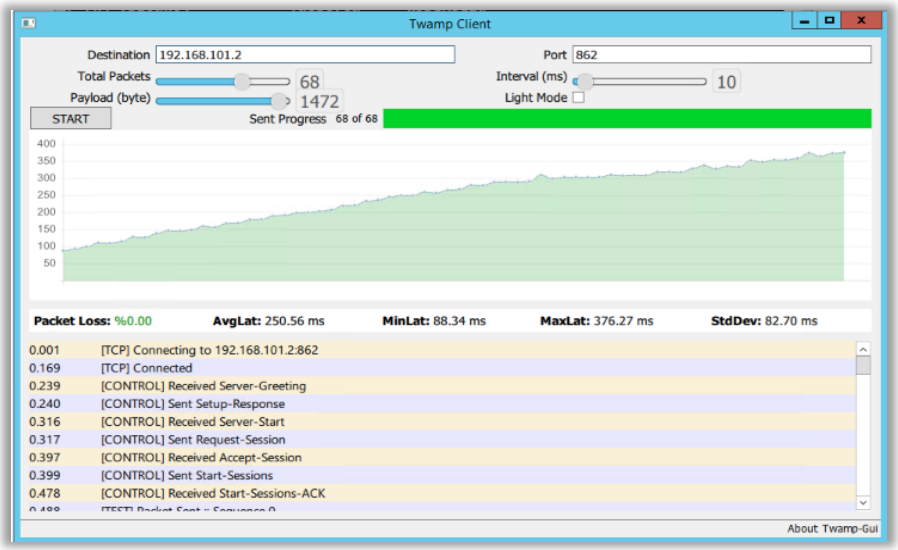

*Figur 5. 19: TWAMP-resultat over 4G med IPSec med full payload: varierende forsinkelse*

Figuren nedenfor (fig. 5.20) viser et resultat fra en måling gjennomført i samme tidsrommet som målingen ovenfor. Som vi kan se her er det en gjennomsnittlig forsinkelse på 77,40 ms, det er et standard avvik på 7,71 ms. Videre kan vi se at det er en maksimal forsinkelse på 100,13 ms og en minste forsinkelse på 64,07 ms. Som vi kan se på resultatet var det ett tap på 1,47%.



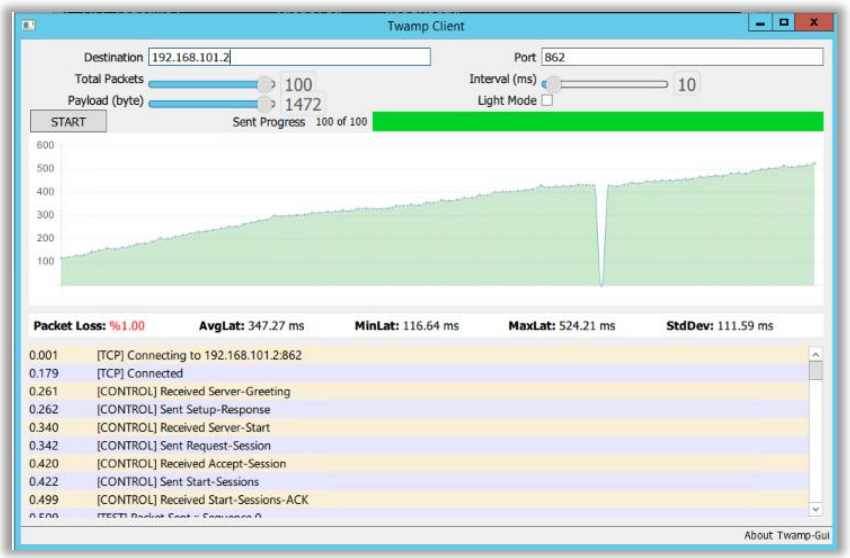

*Figur 5. 20: TWAMP-resultat over 4G med IPSec med full payload: med pakketap*

Under kan vi se et annet resultater fra en måling gjennomført over 4G med IPSec (fig. 5.21). Her kan vi se at resultatet starter på en forsinkelse og øker jevnt frem til siste pakke blir sendt. Det vil her være en maksimal forsinkelse på 376,27 ms og minste forsinkelsen vil være på 88,34 ms. Den gjennomsnittlige forsinkelsen vil være 250,56 ms og et standard avvik på 82,70 ms.




*Figur 5. 21: TWAMP-resultat over 4G med IPSec med full payload: stigende forsinkelse*

Til slutt har vi ett resultat over en måling hvor det ble sendt 100 pakker ovenfor (fig. 5.22). Vi ser at det er 1% pakketap noe som vil tilsvare ett tap på en pakke. Det vil her være en gjennomsnittlig forsinkelse på 347,27 ms og et standard avvik på 111,59 ms. Det vil da være en maksimal forsinkelse på 524,21 ms og en minste forsinkelse på 116,64 ms.




*Figur 5. 22: TWAMP-resultat over 4G med IPSec med full payload: stigende forsinkelse og pakketap*

Vi har nå sett på alle resultatene gjort med TWAMP ved bruk av dette verktøyet. Dette verktøyet har gjennomført en øyeblikksmåling av nettverket, hvor klienten også er orkestratoren.

### 5.4.2 Oppsummering av resultater med TWAMP – Øyeblikksmåling

**TWAMP med full payload**

Under (tab. 8) ser vi en oppsummering av resultatene for TWAMP-målingene gjort med maksimal payload (pakkestørrelse) satt opp for de ulike nettverkene målingene ble gjort over. Alle resultatene er for 68 pakker sendt med unntak av den ene målingen over 4G med IPSec hvor det er spesifisert at det ble gjort med 100 pakker.

*Tabell 8: Oppsummering TWAMP - øyeblikksmåling med full payload*

| | **Fiber uten IPSec** | **Fiber med IPSec** | **4G med IPSec** | | | |
|---|---|---|---|---|---|---|
| | | | 68 pakker | | | 100 pakker |
| **Pakketap** | 0 % | 2.94 % | 0 % | 1,47 % | 0 % | 1,00 % |
| **Gjennomsnittlig forsinkelse** | 5,42 ms | 6,2 ms | 75,62 ms | 77,40 ms | 250,56 ms | 347,27 ms |
| **Korteste forsinkelse** | 4,48 ms | 4,95 ms | 56,90 ms | 64,07 ms | 88,34 ms | 116,64 ms |
| **Lengste forsinkelse** | 10,09 ms | 10,14 ms | 94,03 ms | 100,13 ms | 376,27 ms | 524,21 ms |
| **Standardavvik** | 1,11 ms | 1,44 ms | 8,57 ms | 7,71 ms | 82,70 ms | 111,59 ms |

### TWAMP med halv payload

Tabellen (tab.9) under viser resultatene med halv payload eller 700 bytes-pakker sendt over de ulike nettverkene.

*Tabell 9: Oppsummering TWAMP - øyeblikksmåling med halv payload*

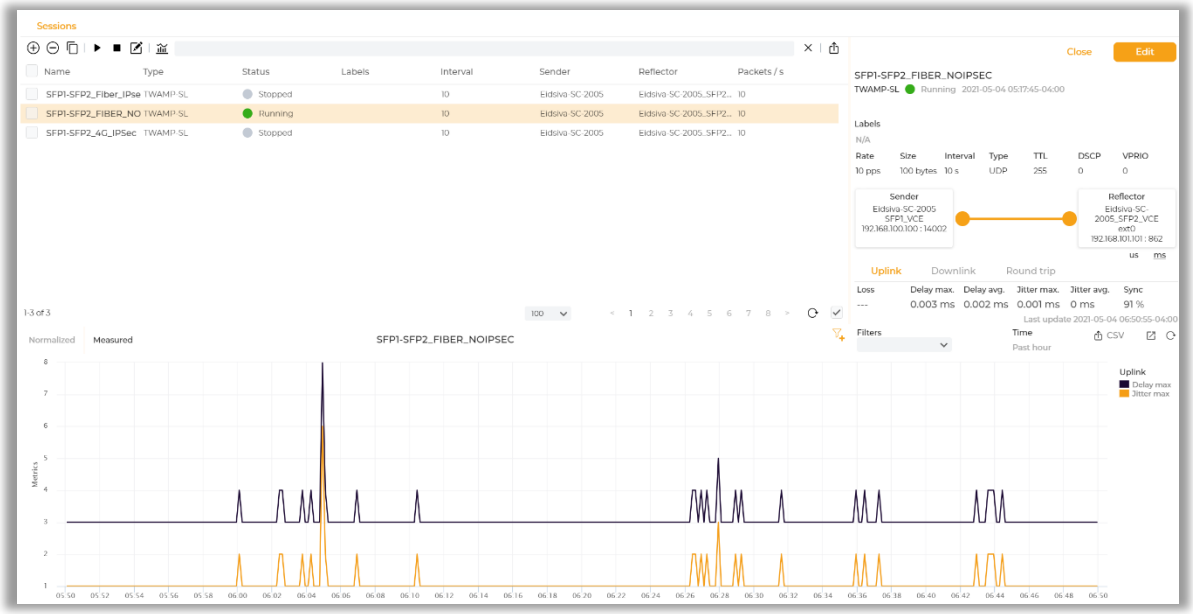
| | **Fiber uten IPSec** | **Fiber med IPSec** | **4G med IPSec** |
|---|---|---|---|
| **Pakketap** | 0 % | 0 % | 0 % |
| **Gjennomsnittlig forsinkelse** | 3,90 ms | 3,89 ms | 75.59 ms |
| **Korteste forsinkelse** | 3,83 ms | 3,81 ms | 60,46 ms |
| **Lengste forsinkelse** | 4,10 ms | 4,74 ms | 99,62 ms |
| **Standardavvik** | 0,05 ms | 0,11 ms | 8,23 ms |

### 5.4.3 Resultat TWAMP – Accedian

TWAMP målingene gjennomført ved bruk av Accedian gir oss en kontinuerlig overvåkning av nettverket. Det gir også en plattform som kan brukes for å se tilbake på målinger som er gjort tidligere for å se hvordan forsinkelser eller pakketap som har vært på en link. Her vil alle resultatene være fra målinger kjørt i en time.

**Resultat med TWAMP over fiber**

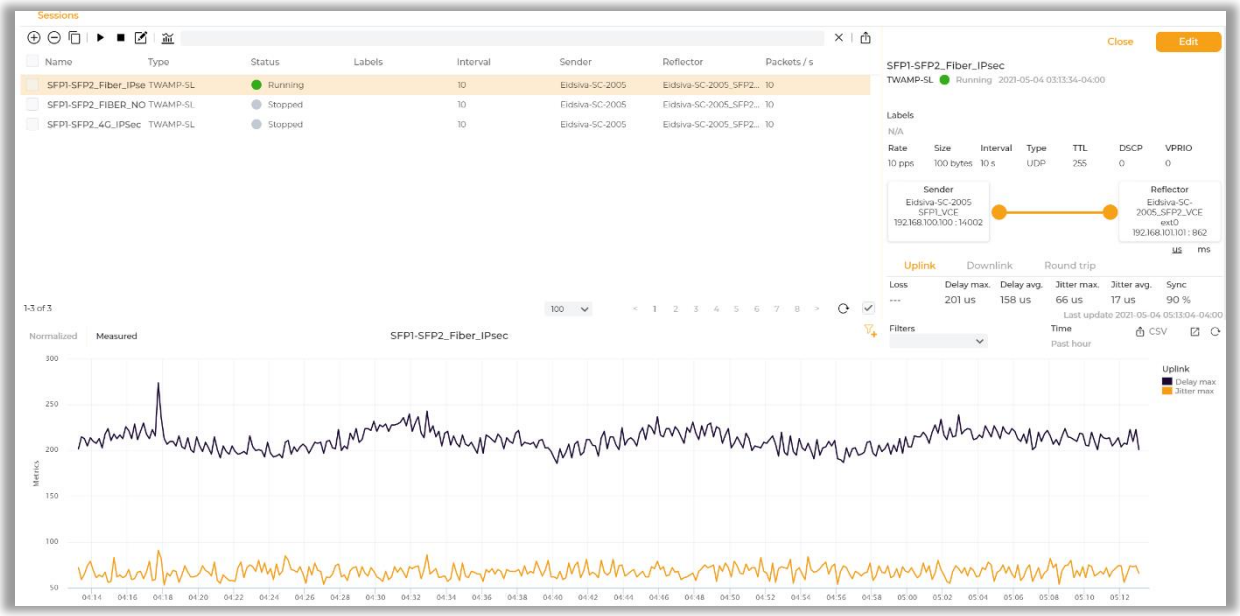
I figuren (fig. 5.23) kan vi se resultatene fra måling gjort med Accedian sin plattform. Som vi kan se vil det ikke være noen store pakketap eller forsinkelser. Det er lave verdier for jitter, den gjennomsnittlige jitteren er på 0 ms, noe som er ett veldig bra resultat over nettverket. Videre vil også den maksimale og gjennomsnittlige forsinkelsen være lave, på under 0,002 ms og 0,003 ms.




*Figur 5. 23: TWAMP-resultat med Accedian over fiber*

**Resultat med TWAMP over fiber med IPSec**:

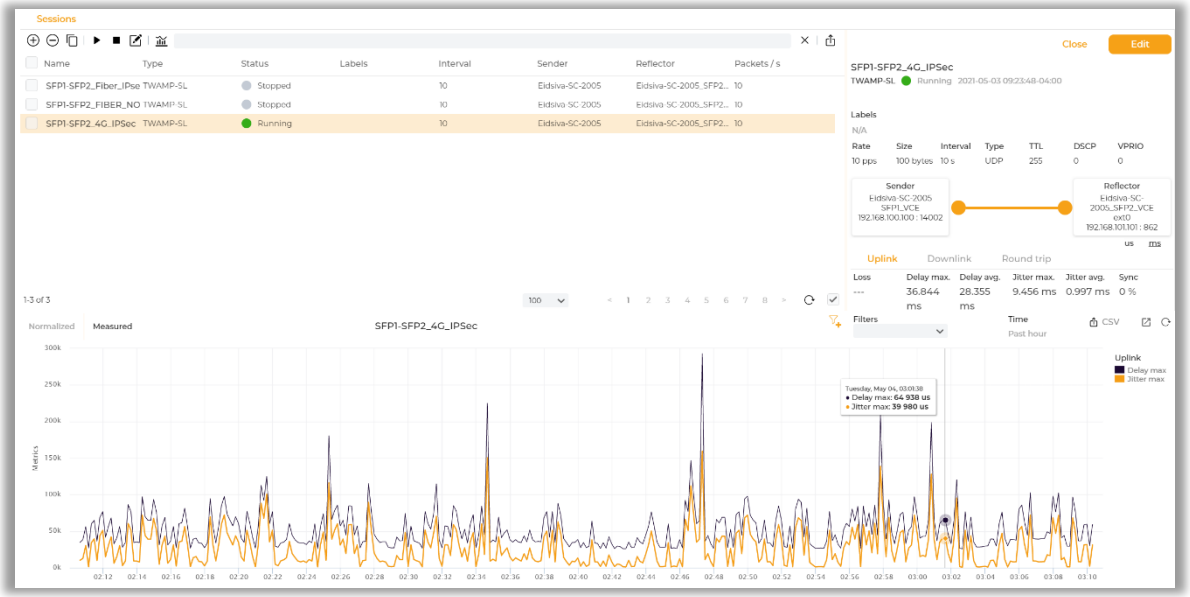
I figuren (fig. 5.24) ser vi resultatet fra målingen gjennomført over fiber med IPSec. Her er det mer forsinkelse og jitter enn hva det var over fiber uten IPSec. Her vises resultatene i mikrosekunder (us), dette vil omgjøres til ms og i tabellen med oppsummeringen av resultatene (tab. 10) vil resultatet skrives som ms. Her vil det være en maksimal forsinkelse på 201 us (0,201 ms), en gjennomsnittlig forsinkelse på 158 us (0,158 ms). Videre vil det også være en gjennomsnittlig jitter på 17 us (0,017 ms) og en maksimal jitter på 66 us (0,066 ms).




*Figur 5. 24: TWAMP-resultat med Accedian over fiber med IPSec*

**Resultat med TWAMP over 4G med IPSec**:

I resultatet over 4G med IPSec (fig. 5.25) kan vi se at det er en endring når det kommer til forsinkelse og jitterstørrelsen. Det vil her være en gjennomsnittlig forsinkelse på 28,355 ms og en maksimal forsinkelse på 36,844 ms. Det vil også være en høyere jitter enn hva det var på resultatet over fiber. Gjennomsnittlig jitterstørrelse vil være på 0,977 ms og den maksimale jitterstørrelsen er 9,456 ms.




*Figur 5. 25: TWAMP-resultat med Accedian over 4G med IPSec*

### 5.4.4 Oppsummering av resultater med TWAMP – kontinuerlig måling

Det er videre oppsummert resultatene fra TWAMP med kontinuerlig måling i tabellen (tab. 10) nedenfor.

*Tabell 10: Oppsummering av TWAMP-resultater med kontinuerlig måling*

| | **Fiber uten IPSec** | **Fiber med IPSec** | **4G med IPSec** |
|---|---|---|---|
| **Gjennomsnittlig forsinkelse** | 0.002 ms | 0,201 ms | 28,355 ms |
| **Maksimal forsinkelse** | 0,003 ms | 0,158 ms | 36,844 ms |
| **Gjennomsnittlig jitter** | 0 ms | 0,017 ms | 0,997 ms |
| **Maksimal jitter** | 0,001 ms | 0,066 ms | 9,456 ms |
| **Tap** | 0% | 0% | 0% |

# 6 Evaluering

Denne rapporten har til nå hatt flere fokusområder, vi har sett på det teoretiske grunnlaget til prosjektet, hvordan laboppsettet har blitt bygd opp og hvordan de tradisjonelle målemetodene kan gjennomføre målinger. Disse målemetodene har også gitt oss flere resultater som nå skal bidra til en evaluering. I dette kapittelet skal det bli gitt et innblikk i hvordan disse fokusområdene henger sammen. Kapittelet vil begynne med å evaluere metoden som har blitt brukt for å gjennomføre prosjektet (kap. 6.1), videre vil vi evaluere resultatene på de tradisjonelle målemetodene (kap. 6.2) og TWAMP (kap. 6.3) før vi sammenligner de tradisjonelle målemetodene mot TWAMP (kap. 6.4-6). Til slutt vil vi se på hvilke kostnader det er ved å implementere TWAMP i nettverket (kap. 6.7) og hvilket fremtidig arbeid (kap. 6.8) som må gjøres for at bedriften kan benytte TWAMP som målemetode.

## 6.1 Vurdering av metode og løsning

Metoden som ble brukt for å gjennomføre denne bacheloroppgaven begynte med å sette rammene for oppgaven før det ble startet med design av laboppsett og litteraturstudium. Det ble lagt veldig mye fokus på litteraturstudiet til å begynne med, noe som gjorde at det ble lagt mye arbeid inn i dette. Dette gjorde at fokuset til å begynne med ble tatt vekk fra den praktiske gjennomføringen av bacheloroppgaven, noe som resulterte i at vi kom litt senere i gang med målinger enn hva som var ønsket. Det var lenge satt en begrensning om at det var Accedian sin plattform som skulle brukes til å gjennomføre TWAMP målinger. Ettersom den lisensen som Eidsiva Bredbånd hadde på plattformen hadde utløpt måtte vi forespørre en ny lisens. Dette tok lengre tid enn vi hadde forutsett og var ikke på plass før sent i forløpet. Når vi så at det tok lang tid å få tildelt en ny lisens burde vi ha begynt å se på alternative målemetoder tidligere enn hva vi gjorde. Dette førte til at vi fikk kortere tid enn ønsket til å gjennomføre og evaluere måleresultatene.

Vi fant etter hvert en grei erstatning for å få teste konseptet rundt TWAMP. Etter at disse målingene var gjennomført, fikk Eidsiva Bredbånd endelig tilgang på lisensen fra Accedian. Rapporten vil derfor inneholde noen måleresultater fra Accedian, men det meste av diskusjonen vil være rundt open source versjonene av TWAMP.

### Laboppsett

Til å begynne var planen å sette opp et enkelt laboppsett, men vi kom fort frem til at det ikke ville være hensiktsmessig da det ikke ville være noe støy eller forsinkelse av betydning. Etter hvert utviklet det seg til å være ett nettverk som går via en av Eidsiva Bredbånds internettruter. Denne er koblet til internett, men siden begge endepunktene er koblet til kjerneruteren vil pakkene bli sendt videre uten å faktisk gå ut på internett for målingene over fiber.

For å kunne introdusere ytterligere forsinkelser ble det satt opp 4G som redundans. 4G er en mindre stabil tilkobling sammenlignet med fiber, og ytelsen en får er avhengig av blant annet hvor mange andre som bruker linken samtidig. Dette er faktorer som vi ikke har kunnet styre og måleresultatene kan derfor variere. 4G-modemet ble plassert innenfor to murvegger, noe som også er med på å redusere kvaliteten på signalet. For å kunne gjennomføre målingene over 4G måtte vi sette opp IPSec, for å kunne nå endepunktet. Siden alle pakkene i overføringen da må krypteres og dekrypteres fører dette til ytterligere forsinkelser i tillegg til at pakkene blir litt større på grunn av den ekstra informasjonen krypteringen krever. Siden vi ikke kan kontrollere alle parameterne rundt 4G, gjør dette det litt vanskeligere å sammenligne resultatene med målinger gjort over fiber. Dette er fordi fiberlinken i disse målingene ikke vil ha noe særlig støy. På tross av dette synes vi likevel det var relevant å ta med disse målingene, både for å kunne se mer støy og forsinkelse, men også fordi i en reell setting vil linjen være mer belastet. Disse måleresultatene vil trolig ligge nærmere de måleresultatene en vil få hvis en måler over en link om er i bruk.

Siden målingene er gjort i Eidsiva Bredbånd sitt nettverk, vil ikke alle parameterne som vi har tatt høyde for i denne oppgaven kunne gjenskapes da deres interne nett ikke er åpent for alle. Utenom det har vi prøvd å dokumentere fremgangsmåten for målingene vi har gjort slik at andre skal kunne komme frem til lignede resultater.

### Målemetoder

Målemetodene som er brukt i denne oppgaven er valgt på bakgrunn av hva Eidsiva Bredbånd bruker i dag, for å kunne vurdere om TWAMP til kunne gi merverdi sammenlignet med de målemetodene de bruker nå. De bruker SNMP som er en passiv overvåkningsmetode som jevnlig henter ut informasjon om enhetene i nettverket. Målemetoden forteller noe om ytelsen til enhetene som blir overvåket. Iperf er en aktiv overvåkingsmetode som må initieres. Den kan

gjøre både taps- og hastighetsmålinger som ende-til-ende måling i nettverket. Målingene er gjennomsnittsmålinger ofte per sekund, noe som gjør måleresultatene kan se fine ut selv om det ikke er reelt. Det er heller ikke mulig å gå tilbake å se på tidligere hendelser. Ping blir mye brukt, men stort sett for å vite om enheter er tilgjengelig over gjeldene nettverk, eller om klienten du jobber fra har kontakt med internett. Ping er en rask og enkel test, men kan ikke brukes til å overvåke ett nettverk alene.

## 6.2 Tradisjonelle metoder

**Ping**

Ping er en øyeblikks test som gir måling av enkeltpakker. Målingen gir en indikasjon på hvor lang tid destinasjonen bruker på å svare på en forespørsel og deretter få pakkene sendt i retur. Målemetoden vil gi en rask og god indikasjon på om utstyr er oppe eller om en enhet har tilgang til internett. En annen fordel med Ping er at kommandovinduet målingen kjøres fra er tilgjengelig på datamaskinen og gir svar innen ett minutt fra en ønsker å kjøre testen. Ping er en applikasjon som kan gi svar på om det er forbindelse mellom to punker. Derfor brukes det ofte i forbindelse med implementering av ett laboppsett. Får ikke pakkene svar, vil det gi en indikasjon på om det konfigurasjonsfeil i nettverket eller om det er andre grunner til at det ikke var forbindelse. Dersom Ping ga svar vil det kunne være en god indikasjon på om nettverket var korrekt satt opp og alle statiske ruter fungerte optimalt. Noen ulemper med Ping er at testen må initieres og vil ikke kunne gi en varsling på om det er pakketap i bestemte tidsrom eller om det oppstå andre feil i nettverket. Det vil heller ikke være mulig å gå tilbake for å se på tidligere hendelser. En begrensning for Ping er at resultatet den kan tilby vil være begrenset til noen parametere som pakketap og forsinkelse. Testen vil heller ikke kunne gi et godt innblikk i hvor mye nettverket er belastet.

Ping er en av de mest tradisjonelle målemetodene og den ble i prosjektet kjørt over fiber og over 4G med IPSec. Resultatet viser at pakkene har en forsinkelse på i gjennomsnitt 30 ganger mer over 4G med IPSec enn over fiber. Noe av grunnen til dette er fordi målinger gjort over 4G vil først sende pakkene trådløst ut på internett, mens pakkene som sendes over fiber vil gå til interenettruteren før de blir rutet videre til riktig svitsj. Ett 4G-modem med trådløst nettverk vil sjeldent kunne være like stabilt som en kablet link, spesielt sammenlignet med fiber. Dette

kommer av at fiberlinken er mye mer stabil og tilbyr en høyere hastighet og kapasitet, noe som vil gi mindre forsinkelser enn det 4G gir. En annen årsak til forsinkelsen som vil oppstå på 4G er bruken av IPSec. IPSec introduserer forsinkelse i forbindelse med tiden det tar å kryptere og dekryptere pakkene. 4G vil bli mer påvirket av denne forsinkelsen enn hva fiber vil på bakgrunn av den lavere maksimale hastigheten som 4G kan tilby.

**Iperf**

Det ble gjennomført flere målinger med Iperf. Det ble gjennomført målinger over fiber uten IPSec, fiber med IPsec og over 4G med IPSec. Som vi kan se i resultatene ble de drastisk dårligere når det ble gjennomført målinger over 4G med IPSec. Det ble gjennomført både hastighetsmålinger og tapsmålinger med de tre ulike utgangspunktene. Det skal først bli sett på resultatene vi fikk etter måling med hastighet. Ved måling over fiber både med og uten noen IPSec fikk vi gode verdier, ingen antydning til dårligere hastighet eller ytelse, noe som var forventet over ett godt fibernettverk med gode kabler. Det var ingen grunn for at disse målingene skulle bli lavere enn maksimal kapasitet på båndbredden. Som vi kan se i målingene lå resultatet rundt 295 Mbps. Dette anses som gode verdier utfra utgangspunktet og det utstyret som ble brukt og oppsettet av nettverket.

Vi kan se at resultatet på hastighet over 4G med IPSec er veldig lav sammenlignet med fiber. For 4G med IPSec var det en maksimal båndbredde på 9,44 Mbps noe som er et drastisk lavere resultat enn hva vi fikk over fiber. Den lave båndbredden kan stamme fra flere ting, en av årsakene kan være at linken over 4G er mindre stabil og båndbredden vil aldri kunne bli høyere enn hva den minste linken kan tilby. En trådløs forbindelse, slik 4G er, vil også ha en større fare for at pakker blir tapt og må retransmitteres fordi størrelsen på pakken blir for stor i forhold til hva linken kan overføre til forskjell fra fiber. IPSec vil her bidra til at tiden pakkene bruker på å bli sendt vil bli enda tregere, det vil også gjøre at pakkene blir større og mindre data kan sendes av gangen, siden det legger til ekstra header og tail som må krypteres og dekryptes. IPSec vil være nødvendig under overføringen av data over 4G, på bakgrunn av at overføringen av data må være sikker for å kunne overføres mellom to LAN via 4G.

Videre skal vi se på resultatene som vi fikk ved gjennomføring av tapsmåling. Det ble her sendt UDP-pakker for å gjennomføre målinger. UDP pakker er mindre sikkert enn hva TCP er og en mindre avansert protokoll. Målingene vi fikk over fiber både med og uten IPSec var helt uten tap, dette var å forvente. Dersom det hadde vært store mengder tap eller høye verdier på jitter

her, ville det heller vært en antydning på en kabel som har strekk eller er bøyd. Fiberteknologi er en teknologi der det tilbys større hastigheter og stor båndbredde. Det vil derfor være mindre sannsynlighet for pakketap og forsinkelser på leveringen av pakker. Dersom vi videre ser på resultatet vi har fått ved målinger over 4G med IPSec ser vi at det er drastiske endringer. Her ser vi mye større tap og mer forsinkelser. Vi kan se at det er nokså høye tap av pakker underveis i sendingen. Dette kommer av at det er flere ledd som må passeres for at trafikken skal kunne komme fra det ene endepunktet til det andre. Det kan også sees i resultatet for pakketap at jo flere pakker som sendes så vil det oppstå mer pakketap. Når vi kommer opp mot totalt 278 datagram så mistet 59 av pakkene underveis i overføringen. Over 4G er det større variasjoner med hvilken kvalitet som kan tilbys. 4G-modemet i denne testen stod inne i et murbygg, noe som gjør at det er mindre sannsynlig å kunne oppnå god forbindelse. Dette kan være en årsak til hvorfor vi opplevde såpass drastiske forskjeller på måleresultatene, siden det kun var over 4G det opplevdes tap. Det er viktig for 4G med god dekning for å kunne få en god ytelse. Og når det da i tillegg ble lagt på en IPSec-tunell blir også forsinkelsen i sendingen av pakker generelt større enn hva det ville vært uten.

Pakketapet som oppleves kan sees i sammenheng med at Jitter-størrelsen ligger på opp mot 15 ms og høyere. Noe som er en drastisk forskjell fra fiber hvor Jitter-størrelsen ligger jevnt på under 0,7 ms, med unntak av den siste pakken hvor vi har brutt målingen og derfor er noe høyere. Jitter gir oss en indikasjon på hvor lang tid pakkene bruker på å sendes ut fra en referansetid som er den som er ønskelig. Vi kan se ved denne målingen at resultatene for jitter varierer mellom fiber og 4G. Antallet bytes som overføres er veldig varierende for 4G sammenlignet med fiber. På fiber vil denne holdes stabilt rundt 2,38 MBytes, men for 4G vil denne variere mellom 1,5 MBytes og 2,3 MBytes. Og vi kan se at disse varierer i takt med båndbredden som linken tilbyr. Jo lavere båndbredde, desto lavere antall Bytes kan overføres.

Alle iperf-resultatene og målingene som ble gjort med denne metoden gir kun et øyeblikksbilde av hvordan nettverkets kvalitet er når målingen blir gjennomført. Testen må initieres og har noe skjedd ved et tidligere tidspunkt, vil den ikke gi oss mulighet til å gå tilbake å vurdere dette. Denne testmetoden vil kunne gi en gjennomsnittsmåling over linken trafikken sendes på mellom to endepunkter. Den vil gi oss tilstrekkelige verdier i forhold til hastigheten på linken og hvor mange pakker som ikke kommer frem. Noe som taler imot denne målemetoden er at det kun vil gi ett øyeblikksbilde, noe som ikke er veldig nøyaktig, i og med at den gir en gjennomsnittsmåling over det tidsintervallet den kjører på. Den vil ikke kunne kjøre på et lavere

intervall enn 1 sekund. Dette gjør at vi med denne målemetoden ikke vil vite hvilke pakker som går tapt eller om det er en årsak til at dette skjer. Den samme problemstillingen vil oppstå i forbindelse med jitterverdien. Måleresultatene kan derfor se bra ut, uten at det er reelt. En fordel med den er at verktøyet ligger tilgjengelig på internett, noe som gjør at alle kan ha tilgang til dette måleverktøyet.

Som det kan sees fra måleresultat med god fiberkabel sammenlignet med en fiberkabel som har blitt skadet, kan en se viktigheten av kvaliteten på fiberen. En fiberkabel vil være veldig mottakelig for bøy eller knekk da den ikke er like solid som andre kabler. Og en bøy vil kunne gjøre at lyset vil bli reflektert ut av kabelen, slik at det ikke når frem til endepunktet. Dette er en av svakhetene til en fiber kabel.

**SNMP**

SNMP-målingene blir gjort direkte mot enheter i nettverket, og målingene viser hvor mye trafikk som har gått gjennom det punktet. Målingene som blir gjort med SNMP går kontinuerlig, dette gjør det mulig å gå tilbake å vurdere verdiene etter en hendelse har skjedd. Verdiene blir hentet ut hvert femte minutt og hvis det er ujevn belasting i løpet av dette tidsintervallet, vil ikke det synes på målingene.

Målingene vi tok viser begrensningene med SNMP. Nettverket ble belastet med 400 Mbps i omkring ett minutt, mens måleresultatet til SNMP viser at det gikk trafikk på underkant av 100 Mbps i det tidsrommet. Grunnen til dette er at SNMP viser gjennomsnittsmålinger av den båndbredden som er benyttet i løpet av ett gitt tidsintervall. Det kan gi en grei oversikt over hvor mye trafikk som går på nettverket, men det vil ikke kunne oppdages om det går mye trafikk i kortere perioder. SNMP sier heller ikke noe om hvor mye jitter det er i nettverket, noe som kan ha mye å si for hvordan brukeren opplever kvaliteten. Når vi belastet nettet jevnt over tid, stemmer de tallene vi ser på SNMP-målingene bedre overens med hvor mye belasting nettverket faktisk hadde. I disse målingene viser Ethernet-ytelsen og SNMP-målingene at det blir brukt i overkant av 400 Mbps av båndbredden.

SNMP bruker en plattform som sørger for å hente ut verdier i gitte intervaller noe som gjør at verktøyet har den fordelen at man kan gå tilbake i tid og evaluere hendelser som har skjedd i nettverket. Verktøyet kan også settes opp til å varsle om noe skulle skje i deler av nettverket. Dette gjør det enklere for de som drifter nettverket å gå inn for å sjekke om alt er som det skal.

I motsetning til de andre tradisjonelle målemetodene, gir SNMP-protokollen mulighet til å endre på konfigurasjonen på enhetene den overvåker.

## 6.3 TWAMP

**Øyeblikksmåling**

Det ble gjennomført flere tester med en open source applikasjon for TWAMP. Denne applikasjonen baserer seg på grunnprinsippene omkring TWAMP og dens toveis måling. Som vi kan se i resultatet vil denne applikasjonen vise forsinkelsen på hver enkelt pakke som sendes, den vil også vise gjennomsnittstiden, standard avviket, maksimumstiden og minimumstiden på pakkene som er sendt over tidsrommet målingen gjennomføres over. Målingene over fiber og 4G med IPSec ble alle gjort med både 700 Bytes og 1472 Bytes som pakkestørrelse (payload).

Resultatene vi fikk ved målingene over fiber uten IPSec med 1472 som payload, dette er også maksimal payload for testen, viser at de fleste pakkene brukte rundt 5,4 ms på å returnere. Utfra resultatet kan vi se at noen pakker brukte det dobbelte av denne tiden. En grunn for dette kan være at pakkestørrelsen gjorde at pakken måtte fragmenteres før den kunne bli sendt videre. Det kan også være en microburst selv om det ikke er så sannsynlig med tanke på at pakkene ble sendt med 10 ms intervall. Da nesten alle pakkene vil ha rukket å nå frem før neste pakke vil bli sendt. Dette fører til at det ikke vil oppstå kø ved ruteren. Forsinkelsen vil heller ikke kunne oppstå på bakgrunn av at pakken har valgt en lengre rute, da labmiljøet målingen ble gjennomført over ikke hadde noen alternative ruter. Den neste måling over dette nettverket ble gjennomført med en payload på 700 Bytes. Her ble resultatet at alle pakkene brukte tilnærmet lik tid under overføringen med ett standard avvik på 0,05 ms. Dette kan tyde på at grunnen til at noen av pakkene i den første målingen brukte lengre tid, er på grunn av størrelsen på pakken. Dersom vi ser bort ifra de tre pakkene med større forsinkelser i det første resultatet, vil det stadig være en større variasjon i pakkene sendt med maksimal payload enn pakkene sendt med lavere payload.

Neste gjennomføringen av testen ble gjennomført over fiber med IPSec. I målingen med full payload kan vi se på resultatet at 7 av pakkene brukte opp mot dobbel så lang tid som de andre pakkene overført som en del av målingen. Gjennomsnittstiden for pakkene er også 0,6 ms høyere enn målingene uten IPSec. En av grunnene til at flere av pakkene bruker lengre tid

sammenlignet med resultatene ved fiber uten IPSec er at pakker sendt med IPSec må krypteres og dekrypteres, noe som ikke er nødvendig uten IPSec. Pakkene vil også få med ekstra krypteringsinformasjon, dette vil gjøre pakkene større og sannsynligheten for at noen pakker har blitt fragmentert i flere deler er større. De fleste pakkene brukte like lang tid på å bli sendt. I denne testen gikk to pakker tapt. Det kan være at forbindelsen mellom endepunktene hadde problemer med å opprettes, noe som resulterte i at pakken ikke kom frem, dette var da ikke ett problem for de resterende 66 pakkene som ble sendt. For målingen over IPSec med halv payload er resultatet relativt likt som målingene uten IPSec med halv payload. Eneste forskjellen er at den første pakken brukte litt lengre tid på å komme frem enn de andre pakkene. Vi kan også se utfra resultatet at kapasiteten og ytelsen ikke vil bli begrenset av bruken av IPSec over fiber.

Det ble til slutt gjennomført TWAMP målinger over 4G med IPSec. Her ble den første målingen gjennomført med en pakkestørrelse på 700 Bytes. Resultatet av målingen viser at pakkene bruker forskjellig tid på å returnere og det er stort sett en gradvis endring i forsinkelsen. Det ble deretter gjort flere målinger med full payload. Bakgrunnen for at det ble gjort flere målinger her var fordi resultatene varierte i stor grad. Noe som er forventet over en 4G link. Her ser vi tendenser til at en pakke bruker rundt 90 ms på å returnere før de neste 5-6 pakkene blir gradvis raskere til den raskeste pakken som bruker omtrent 60 ms før neste pakke bruker rundt 90 ms igjen. Dette gjenspeiler den ustabiliteten ett 4G modem og den trådløse linken tilfører ett nettverk.

De siste to målingene som ble også gjort over 4G med IPSec. På begge målingene øker tiden pakkene bruker på å returnere gradvis. Begge målingene har pakkestørrelse på 1472 Byte. I den første målingen ble det sendt 68 pakker, mens i den andre ble det sendt 100. Dette gjør sammenligningen av resultatet noe vanskeligere. Den første pakken i den første målingen brukte 88,3 ms på å returnere, mens den første pakken i den andre målingen brukte 116,6 ms. Det ser ut som økningen i tiden pakkene brukte på å returnere var lik i begge testene. De første pakkene bruker tilsvarende tid som pakkene i de første målingene over 4G med IPSec. Vi vet at IPSec er med på å belaste nettverket, så grunnen til at pakkene bruker stadig lengre tid på å returnere er trolig at det blir flere IPSec-pakker i nettverket som må håndteres. Noe som igjen fører til at de resterende pakkene må vente enda lengre og få ytterlige forsinkelser.

Denne open source plattformen å kjøre TWAMP-målinger på gir mulighet til å se hvor lang tid den enkelte pakken bruker. Dette representeres i en graf der høyden varierer utfra måleverdiene. Her har vi lov til å kjøre opp mot 100 pakker i en måling med 10 ms som kortest intervall. For

å få ett bedre bilde av ytelsesevnen til nettverket, burde det vært en mulighet å sende flere pakker med lavere intervall. I testene gjort over fiber, har de fleste pakkene rukket å komme tilbake før neste pakke sendes. Målingene vil da bli gjort i ett nettverk som i praksis ikke har noe belastning. Det eneste som kan introdusere mer forsinkelse i nettverket over fiber, er hvis internettruteren har mye trafikk å håndtere.

Der det i dag bare er mulig å måle fra server til punkt A og server til punkt B vil TWAMP gi muligheten til å måle strekket mellom A og B som er den linjen kunden faktisk bruker. Lisensierte plattformer gir også muligheten til å sette opp varslinger om endringer i nettverket. Det at TWAMP kan kombinere ende til ende målinger og å analysere hver pakke som går gjennom en node, gjør at den kan måle både nettverksytelsen og hvordan brukeren opplever kvaliteten på nettverket. En kan derfor si at TWAMP måler QoE og QoS bedre enn de andre målemetodene.

**Kontinuerlig måling med Accedian**

Det ble i tillegg gjennomført TWAMP-målinger med Accedian sin plattform. Dette er en plattform som tilbyr mer kontinuerlige målinger enn den metoden som ble brukt for øyeblikksmålingen, men målingen med Accedian plattformen vil ha samme grunnlaget under gjennomføringen av tester. Plattformen vil kunne gjøre målinger i ønskede intervaller slik at det er mulig å få en dokumentasjon av hvordan nettverket vil se ut og en oversikt over alle sesjoner på samme plattform. Det vil her være en mulighet å se på tidligere resultater, noe som er en stor fordel ved en slik nettverksovervåkning. Dette vil bidra til å kunne støtte opp under den kvaliteten som kunden måler eller opplever, slik at bedriften kan observere det kunden har opplevd ved en bestemt tid. Målemetoden vil også gi en sentral plattform som gir god oversikt.

Utfra resultatene kan vi se at disse er vist som en graf. Målingene ble for prosjektets formål kun målt over en time da dette ble ansett til å gi nok dokumentasjon til prosjektet. Resultatet vil heller ikke være helt kompatibelt med de resultatene vi fikk ved øyeblikksmålingen da det sendes pakker med en lavere byte, 100 Bytes, opp mot 700 Bytes og 1472 Bytes ved øyeblikksmålingen. Dette vil by på ett resultat som viser lavere forsinkelser, lavere jitter og mindre pakketap. Pakkene for begge målingene vil selv om størrelsen er forskjellig sendes med samme intervall på 10 ms. Noe som vil gjøre at resultatene fortsatt kan sammenlignes og diskuteres opp mot hverandre. Det vil uansett være en lavere forsinkelse både maksimalt og gjennomsnittlig på den kontinuerlige målingen og mindre sannsynlighet for å oppdage pakketap

da det ikke vil være en stor belastning av nettverket. Dette kan bli sett på som en begrensning ved denne metoden for gjennomføring av TWAMP. Den vil ikke gi en like god indikasjon på hvor mye linjen tåler i forhold til maksimal belastning. Dette resultatet viser ikke hvordan kapasiteten til kunden vil være dersom nettverket belastes mye i en kort periode.

Til tross for denne begrensningen vil det være mer aktuelt å måle med en lavere pakkestørrelse dersom dette er en teknikk som skal bli satt i operativ drift i nettverket til Eidsiva Bredbånd. Da det ikke er ønskelig å oppta kapasiteten til kunden for å gjennomføre kontinuerlige målinger. Måles det med større pakkestørrelse vil det bidra til lavere kvalitet for sluttbrukeren og en mindre opplevd hastighet. Ett resultat med en mindre pakkestørrelse vil kunne gi en like god indikator på nettverkets kapasitet og forsinkelse på pakkene. Dersom forsinkelsen ville økt ville det kunne bli identifisert som en potensiell feil i nettverket.

Resultatet vil gi ett svar på parametere som maksimal forsinkelse, gjennomsnittlig forsinkelse, maksimal jitter og gjennomsnittlig jitter. Dette er noen viktige punkter for en nettverksleverandør å kjenne til for å kunne dokumentere sitt nettverk. Dette er faktorer som vi ser vil variere utfra hvor god en link er. For fiber vil disse resultatene være lave, det vil ikke være høye verdier for jitter eller forsinkelser over den timen målingen ble gjennomført. Derimot vil vi se at når målingen ble gjennomført over 4G med IPSec var det høyere verdier for forsinkelser og jitter i forhold til pakkestørrelsen og det resultatet som ble oppnådd over fiber. Det vil her være over 100 ganger så høy forsinkelse og det vil være en maksimal jittertid på 9 ms. Dette viser oss at denne målemetoden vil gi en god indikasjon på kvaliteten til nettverk mellom Eidsiva Bredbånd og kunden de leverer til. Målemetoden vil på bakgrunn av forskjellene i resultatene på de ulike nettverkene gi informasjon om kvaliteten til nettverket selv om pakkestørrelsen under målingene ikke tester den maksimale ytelsen på nettverket.

Denne plattformen vil også gjøre det mulig å ha en større oversikt over alle sesjoner som kjøres i nettverket. Med bakgrunn i den orkestratoren som blir introdusert i denne plattformen, vil Eidsiva Bredbånd ha en større mulighet for å oppdage eventuelle feil og mangler. Og dette kun ved å gå inn på orkestratoren. Orkestratoren vil i tillegg holde en oversikt over alle enheter som målingene kan gjennomføres mellom, noe som vil gjøre det mulig å ha flere testsesjoner kjørende på samme linken for å avdekke hvor eventuelle feil vil befinne seg mellom to endepunkter i ett nettverk.

## 6.4 Ping vs. TWAMP

Måleresultatene for Ping ligger så langt unna de måleresultatene for kontinuerlig måling med TWAMP at det ikke ble sett på som hensiktsmessig å sammenligne disse. De måleresultatene som er evaluert i dette kapittelet er derfor Ping mot øyeblikksmåling ved bruk TWAMP.

Både TWAMP og Ping gir resultat per pakke. Begge forteller tiden til den raskeste og den tregeste pakken. Vi får også vite om alle pakkene ble besvart og begge metodene vil måle pakkens rundetid. For målinger over fiber uten IPsec var gjennomsnittstiden for Ping 3 ms og TWAM 3,8 ms i testen med halv payload og 4,5 ms med full payload. Grunnen til dette er trolig fordi Ping sender pakker på 32 Bytes, mens TWAMP brukte 700/1472 Bytes i de målingene vi utførte. Ping vil bruke ICMP til å gjøre testene. Denne protokollen blir nedprioritert i køer og er blant de første pakkene som blir kastet hvis køene er overfylte. Målingene for Ping ble kjørt motsatt vei av TWAMP-målingene, men når det måles rundetid vil resultatene bli tilnærmet det samme. Men dette vil variere på kvaliteten til målemetoden.

Målingene ble også gjort over 4G med IPSec. Her vil den første pakken på Ping og på den nest siste målingen for TWAMP over 4G med IPSec returnere relativt likt. De to neste pakkene som returnerte med Ping brukte lavere tid, mens de to neste pakkene i TWAMP hadde en jevn økning i tiden. Den siste pakken til Ping brukte noe lengre tide enn den fjerde pakken i den første TWAMP-målingen. Lest ut fra grafen brukte den 110-120 ms mot Ping sin 152 ms. De målingene som ble gjort med Ping her, har for lite utsnitt til å kunne si noe om hvordan forsinkelsen endrer seg over tid på 4G med IPSec. Vi valgte å bruke Ping uten å endre på parameterne som hvor mange pakker som skal sendes og hvor store de er.

Ping kan brukes mot alle enheter med IP-adresse som ikke er satt opp til å blokkere ICMP-forespørsler, mens TWAMP kan brukes opp mot enheter som er satt opp til å svare på TWAMP-forespørsler. Ping er en enkel og rask måte å teste forsinkelser ulike steder, og teste om nettverksenheter er tilgjengelige. Ping er et mye brukt verktøy, men kan ikke brukes til å overvåke ett nettverk alene. Den open source testplattformen som ble brukt til å gjøre TWAMP-målingene, kan gi litt mer informasjon om overføringene av pakkene. Målingene gjøres i et kort tidsrom og kan kun settes opp mot en responder av gangen. Målingen må initieres for hver gang den kjøres noe som gjør at heller ikke den plattformen alene brukes til nettverksovervåkning selv om den gir ett bedre bilde av hvordan trafikkflyten på nettverket er.

## 6.5 Iperf vs. TWAMP

Når det kommer til sammenligningen mellom Iperf og TWAMP som målemetode, vil det være mer hensiktsmessig å starte med å sammenligne resultatene brukt under øyeblikksmålingen med TWAMP. Dette er fordi denne er mer lik testen som ble gjennomført med Iperf. Til tross for at det vil være en mulighet for kontinuerlige målinger ved bruk av TWAMP.

En av fordelene med TWAMP sett opp imot Iperf er at vi får en grafisk fremstilling av resultatet noe som gjør at vi kan se forsinkelsen til hver enkelt pakke som sendes i testperioden. Dette vil vi ikke se når det kommer til Iperf. Iperf vil gi oss den gjennomsnittlige forsinkelsen på pakkene sett opp mot det intervallet som det skal rapporteres på. I prosjektet er dette på hvert andre sekund. Dette intervallet kunne også settes til hvert sekund, men det er ikke mulighet for raskere rapportering enn dette. Iperf vil heller ikke gi en god indikasjon på gjennomsnittlig jitterstørrelse eller maksimal jitter og heller ikke en god oversikt over maksimal forsinkelse. Dette gjør at Iperf ikke vil kunne gi en oversikt over den generelle kvaliteten til nettverket gjennom hele testperioden, den vil heller ikke gi et resultat på hvordan nettverket ser ut for hver pakke. Den vil kun fokusere på hvor mange bytes som er overført i de to sekundene som blir rapportert. Dette vil ikke kunne gi ett svar på om det er en pakke som har brukt lang tid som forårsaker forsinkelsen og jitterstørrelsen eller om det generelt er den kapasiteten som resultatet viser jevnt over. Dette er noe som TWAMP verktøyet vil vise. Samtidig er denne bidirektional, noe som gjør at den måler begge veier i nettverket mellom klienten og reflektoren.

En stor fordel med Iperf sett opp mot TWAMP vil være at denne vil kunne gi en direkte indikasjon på hastigheten (båndbredden) til linken målingene gjennomføres over. Dette vil vi ikke se med TWAMP med mindre dette blir beregnet utfra de parameterne som vi har fått som resultat. TWAMP fokuserer i all hovedsak på pakketap, jitter og forsinkelse, mens Iperf tilbyr en enklere metode for å måle hastighet. Dette vil være mer krevende å finne ett resultat på med TWAMP og krever enten ett verktøy som gjør denne beregningen, om ikke kreves det en plattform/server som kan gjennomføre dette underveis som målingene kjøres.

En annen forskjell på disse måleverktøyene er muligheten til å gjennomføre kontinuerlige målinger og for å kunne gå tilbake å se på tidligere resultater. Det vil med en slik kontinuerlig måling være mulig å kjøre målinger med jevne mellomrom for å kunne sikre dokumentasjon på den linken som kunden har. Det vil også gjøre det mulig å detektere feil og mangler i så fort

disse oppstår. Ulempen med dette er at for å få et godt verktøy og en god plattform til å gjennomføre disse målingene vil det kunne bli kostbart. Dette kommer av kostnaden det er for hver enkelt SFP når disse skal kunne kjøre egen software og også da de flere plattformer for gjennomføringen av slike målinger vil i stor grad kreve lisenser.

## 6.6 SNMP vs. TWAMP

SNMP måler trafikkflyt gjennom en enhet og presenterer dataen som gjennomsnittsverdier for en tidsperiode, mens TWAMP måler hver enkelt pakke som blir sendt mellom to enheter. Målingene vi gjorde viser unøyaktigheten til SNMP da reell trafikk i tidsrommet var flere ganger høyere enn hva SNMP-målingene viste. Dette kan være problematisk hvis det ikke er en jevn trafikk. Målingene kan se greie ut, mens i virkeligheten kan kapasiteten være sprengt i kortere perioder og pakker kastes fordi de nettverket ikke klarer å håndtere pakkene. TWAMP viser måleresultat pakke for pakke, hvis nettverket er ujevnt belastet vil det kunne leses av i grafen. Den open source plattformen vi har brukt til å gjøre målinger med, lar oss sende opptil 100 pakker. Siden SNMP måler båndbredde og TWAMP måler forsinkelse og pakketap, gjør det vanskelig å sammenligne måleresultatene.

SNMP er satt opp på en plattform som kontinuerlig overvåker enhetene i nettverket. Den kan varsle om noe skjer med enhetene i nettverket og en kan gå tilbake å se på tidligere hendelser. Lisensierte versjoner av TWAMP har også muligheten til å gjøre dette. Den kan settes opp til å varsle om noen strekninger har mer pakketap en normalt, eller om forsinkelsen skulle øke. Dette kan gjøre det lettere å finne ut hvilke deler av nettverket problemer oppstår og kan bidra til å gjøre feilsøkingstiden kortere. Hvis bedriften kan bruke kortere tid på å lete etter feil og mer tid på å rette opp feilene, kan det være lønnsomt å implementere på tross av kostnadene det er ved å betale lisens på en ny plattform og kjøpe nettverksutstyr som kan kjøre som TWMAP-agent. TWAMP gir også mulighet til å oppdage microburst i nettverket. Ved å finne ut hvilke enheter som er årsaken til at det blir microburst på nettverket, vil det kunne gjøres nødvendige tilpassinger for å begrense dette.

Både SNMP og TWAMP målinger gjort over Accedian sin orkestrator, er målinger som kontinuerlig henter ut status på nettverket. SNMP ser enhetene i nettverket og måler båndbredden som blir benyttet. TWAMP er en ende-til-ende måling som forteller om

forsinkelse, pakketap og jitter. Disse metodene utfyller hverandre godt, og gir et bredt bilde av hvordan kvaliteten på nettverket er. Fordelen med TWAMP er at den gjør målinger oftere, noe som gjør at den kan oppdage microburst. Dette kan skape store problemer på nettverket, uten at SNMP har mulighet til å fange det opp.

## 6.7 Kostnad ved implementasjon av TWAMP

Det vil være noen kostnader ved å implementere nye målemetoder. Bedriften må ha en lisens til plattformen som skal kjøre testene, og hvis de ønsker å bruke Accedian må de også kjøpe smarte SFPer som er en del dyrere enn de vanlige SFPene. For å gjøre målingene riktig, må SFPene stå på LAN-siden i nettverket. Det var ikke mulig å få til fysisk i testene våre da endepunktene vi brukte ikke hadde mulighet til å koble til SFPer i LAN-port. I prosjektet ble dette løst ved å sett opp WAN-porten til å også fungere som LAN-port. Dette vil ikke være hensiktsmessig å gjøre i et reelt nettverk, siden en risikerer å sende pakker feil. SFPen må stå på LAN siden det er ønskelig å måle både primærlinken og sekundærlinken, men utstyret som er i bruk i bedriften har sjeldent SFP-porter på LAN-siden.

## 6.8 Fremtidig arbeid

For at dette skal bli ett aktuelt verktøy å implementere for bedriften som en målemetode så vil det være behov for fremtidig arbeid rundt løsningen. Det å ta i bruk en ny metode for overvåkning av ett nettverk krever mye forberedelser og utfra de resultatene vi har fått i dette prosjektet, blir det sett på som ett behov å se på alternative løsninger før implementasjon av målemetoden. I forbindelse med Accedian sitt system og plattform opplevde vi store problemer. Det var vanskelig å kommunisere med deres kundebehandlere og det tok en periode på flere måneder å få tilgang til en lisens. Derfor vil en del av fremtidig arbeid, før implementasjon eventuelt gjøres, være å se på andre leverandører av en plattform med sentral orkestrator. Dette for å kunne sammenligne de forskjellige plattformene og observere om det finnes plattformer som kan tilby en bedre løsning.

Videre vil det være aktuelt å analysere resultater mottatt fra ulike plattformer. For å se hvordan resultatene er opp mot hverandre og hvor korrekte resultatene er i forhold til den faktiske

kvaliteten på nettverket som overvåkes. Så vil det også være en mulighet i fremtidig arbeid å se på automatisering av analysene. De resultatene som mottas med TWAMP plattformen brukt i prosjektet, vil gi lite informasjon. Det vil være aktuelt å analysere flere resultater og finne en teknikk for en god automatisering av en slik analyse.

Skal det være mulig å ha en fullstendig overvåkning av ett nettverk med TWAMP bør det bli sett på gjennomføring av målinger på flere ledd. Her bør det blant annet bli sett på hvordan TWAMP kan implementeres i de svitsjene og ruterne som Eidsiva Bredbånd bruker i sitt nettverk. For å se hvordan resultatene for målinger over flere ledd vil bli. For å kunne sikre at resultatene er gode nok bør det genereres nettverkstrafikk med et definert pakketap og en definert forsinkelsestid på linken. Dette for å se å kunne kvalitetssikre om TWAMP tilbyr en måling som er nøyaktig nok.

Før dette skal implementeres vil vi anbefale Eidsiva Bredbånd å teste det mot en bedrift som er deres kunde over en lengre periode. Dette for å kunne kvalitetssikre om verktøyet gir merverdi i forbindelse med overvåkning av en kundeaksess. Dersom dette viser seg å være tilfellet, vil vi videre anbefale Eidsiva Bredbånd å se på en automatisering av prosessen for leveranse og hvordan konfigurasjon av SFP skal skje og eventuell utrullingen av disse. Samtidig bør det gjøres en vurdering på kostnad for implementasjon og kostnaden for tjenesten for bedriftene sett opp mot den nytteverdien målemetoden gir.

Denne siden er blank

# 7 Konklusjon

Det har blitt gjennomført målinger som vi har hentet ut resultater fra og det er blitt gjort en evaluering på bakgrunn av disse resultatene. Dette har ledet inn mot en konklusjon på den problemstillingen som prosjektet baserer seg på. På bakgrunn av dette har vi kommet frem til at TWAMP ikke vil være ett godt nok verktøy til å erstatte de målemetodene som bedriften bruker. For at TWAMP skal kunne ha en merverdi for bedriften, må det være en sentral orkestrator som gjennomfører målingene. Denne orkestratoren gjør det mulig å dokumentere kvaliteten på nettverket på en tilfredsstillende måte. Den gjør også at det kan oppdages microburst i nettverket, som er en av de største begrensingene til de tradisjonelle målemetodene. Dokumentasjonen som kommer ved bruk av TWAMP med sentral orkestrator vil gi en reell kvalitetsstempling av linjen mellom målepunktene.

Accedian er en plattform som har mulighet til å tilby en slik orkestrator. Utfra den implementasjonsprosessen som har vært i forkant av målingene vil implementasjonen av målemetoden kunne by på en stor kostnad. Dette er en viktig faktor for om TWAMP er en ideell løsning for bedriften. Mange plattformer som bruker TWAMP vil være lisensierte og det vil også være nødvendig å investere i ett stort antall smart SFPer. For at det skal kunne gjennomføres målinger over en primær og sekundær link bør smart SFPen være koblet til en LAN-port på endepunktet. Av de endepunktene som brukes i dagens nettverk vil få ha denne muligheten. For at TWAMP skal bli aktuell å implementere i et nettverk vil vi anbefale bedriften å se på alternative metoder.

Utfra resultatene i dette prosjektet vil TWAMP være et godt supplement til tradisjonelle målemetoder dersom det brukes en orkestrator. Dersom TWAMP brukes sammen med SNMP vil det være mulig å oppdage pakketap og hvordan hastighet som er i bruk, samtidig som TWAMP vil tilby muligheten for å se microburst, forsinkelse og jitter. TWAMP tilbyr en mer detaljert overvåkning av nettverket i perioden det blir gjennomført målinger. Den vil også gi en god indikasjon på hvordan kvaliteten er på nettverket. Orkestratoren er nødvendig for at TWAMP skal gi bedre dokumentasjon enn de tradisjonelle målemetodene.

# Litteraturliste


[1] R. K. K. Hedayat, A. Morton, K. Yum, J. Baiarz. (2008). *RFC 5357: A Two-Way Active Measurement Protocol (TWAMP)*. Available: https://datatracker.ietf.org/doc/html/rfc5357, Hentet: [03.04.21]

[2] E. Hem. (2020). *IMRAD*. Available: https://sml.snl.no/IMRAD, Hentet: [13.05.21]

[3] K. Gerwig. (2016). *network node*. Available: https://searchnetworking.techtarget.com/definition/node, Hentet: [13.04.21]

[4] *Telecommunications link*. Available: https://en.wikipedia.org/wiki/Telecommunications_link, Hentet: [13.04.21]

[5] E. Rossen. (2020). *svitsj*. Available: https://snl.no/svitsj, Hentet: [13.04.21]

[6] I. M. Lister. (2020). *ruter (i datanettverk)*. Available: https://snl.no/ruter_-_i_datanettverk, Hentet: [13.04.21]

[7] E. Rossen. (2020). *tjener (IT)*. Available: https://snl.no/tjener_-_IT, Hentet: [19.04.21]

[8] E. R. Tom Heine Natt. (2019). *brannmur*. Available: https://snl.no/brannmur, Hentet: [23.04.21]

[9] *Vi bygger 4G for fremtiden*. Available: https://www.telenor.no/dekning/4g/, Hentet: [19.04.21]

[10] *4G*. Available: https://no.wikipedia.org/wiki/4G, Hentet:

[11] B. E. Loftås. (2017). *Inntil 90 megabit i sekundet på mobilnettet*. Available: https://dinside.dagbladet.no/data/inntil-90-megabit-i-sekundet-pa-mobilnettet/67420448, Hentet: [12.04.21]

[12] H. Øverby. (2020). *OSI*. Available: https://snl.no/OSI, Hentet: [23.02.21]

[13] (2019). *TCP/IP*. Available: https://www.britannica.com/technology/TCP-IP, Hentet: [22.03.21]

[14] (2020). *What is the TCP/IP model and how it works?* Available: https://afteracademy.com/blog/what-is-the-tcp-ip-model-and-how-it-works, Hentet: [24.03.21]

[15] B. A. Fofouzan, Data communications and networking: Alan R. Apt, 2007. [Online]. Available: http://zai.lecturer.pens.ac.id/Kuliah/Komunikasi%20Data/Buku%20Referensi/Data%20Communications%20and%20Networking%20By%20Behrouz%20A.Forouzan.pdf.

[16] V. Beal. *MAC Layer – Media Access Control Layer*. Available: https://www.webopedia.com/definitions/mac-layer/, Hentet: [26.03.21]

[17] T. Slattery. (2019). *VLAN (virtual LAN)*. Available: https://searchnetworking.techtarget.com/definition/virtual-LAN, Hentet: [02.04.21]

[18] *What is a LAN?* . Available: https://www.cisco.com/c/en/us/products/switches/what-is-a-lan-local-area-network.html, Hentet: [21.02.21]

[19] (2018). *Network Layer* Available: https://www.javatpoint.com/network-layer, Hentet: [12.04.21]

[20] W. T. Stevens. (1993). *TCP/IP illustrated | Encapsulation*. Available: https://flylib.com/books/en/3.223.1.18/1/, Hentet: [29.04.21]

[21] C. Greer. (2017). *TCP series #4: TCP receive window and everything you need to know about it*. Available: https://accedian.com/blog/tcp-receive-window-everything-need-know/, Hentet: [27.03.21]

[22] *TCP Header*. Available: https://networklessons.com/cisco/ccie-routing-switching-written/tcp-header, Hentet: [03.04.21]

[23] V. P. M. Allman. (2009). *RFC 5681: TCP Congestion Control*. Available: https://tools.ietf.org/html/rfc5681, Hentet: [14.04.2021]
[24] S. Heap. (2017). *Ruter og svitsj*. Available: https://ndla.no/nb/subject:25/topic:1:193108/topic:1:1:179564/topic:1:1:186780/resource:1:83330?filters=urn:filter:d97809a8-47b6-4d26-ae5c-1839f4c27940, Hentet: [26.03.21]
[25] *What are Data Packets?* . Available: https://whatismyipaddress.com/data-packets, Hentet: [26.03.21]
[26] H. Øverby. (2020). *datapakke* Available: https://snl.no/datapakke, Hentet: [26.03.21]
[27] *What is Network Address Translation?* Available: https://whatismyipaddress.com/nat, Hentet: [16.04.21]
[28] *What is a subnet? | How subnetting works*. Available: https://www.cloudflare.com/learning/network-layer/what-is-a-subnet/, Hentet: [16.04.21]
[29] *What is IPsec? | How IPsec VPNs work*. Available: https://www.cloudflare.com/learning/network-layer/what-is-ipsec/, Hentet: [27.03.21]
[30] *IT Explained: Bandwidth*. Available: https://www.paessler.com/it-explained/bandwidth, Hentet: [26.03.21]
[31] J. Hecht. (2016). *The bandwidth bottleneck that is throttling the Internet*. Available: https://www.nature.com/news/the-bandwidth-bottleneck-that-is-throttling-the-internet-1.20392, Hentet: [28.03.21]
[32] Kary. (2014). *Understanding Throughput and TCP Windows*. Available: https://packetbomb.com/understanding-throughput-and-tcp-windows/, Hentet: [09.05.21]
[33] I. Griogrik. (2013). *Primer on Latency and Bandwidth*. Available: https://hpbn.co/primer-on-latency-and-bandwidth/, Hentet: [27.03.21]
[34] *What are the different types of network delay?* Available: https://www.educative.io/edpresso/what-are-the-different-types-of-network-delay, Hentet: [09.05.21]
[35] (2020). *Ping vs latency vs jitter*. Available: https://www.ghostgb.co.uk/ping-vs-latency-vs-jitter/, Hentet: [23.03.21]
[36] (2016). *What is Acceptable Jitter?* Available: https://medium.com/@datapath_io/what-is-acceptable-jitter-7e93c1e68f9b, Hentet: [24.03.21]
[37] (2006). *Understanding Jitter in Packet Voice Networks (Cisco IOS Platforms)*. Available: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/18902-jitter-packet-voice.html, Hentet: [24.03.21]
[38] *Introduction to QoS (Quality of Service)*. Available: https://networklessons.com/cisco/ccna-routing-switching-icnd2-200-105/introduction-qos-quality-service, Hentet: [29.04.21]
[39] *IP Precedence and DSCP Values*. Available: https://networklessons.com/cisco/ccna-routing-switching-icnd2-200-105/ip-precedence-dscp-values, Hentet: [29.04.21]
[40] *IEEE P802.1p*. Available: https://en.wikipedia.org/wiki/IEEE_P802.1p, Hentet: [29.04.21]
[41] B. Mitchell. (2021). *What Causes Network Lag and How to Fix It*. Available: https://www.lifewire.com/lag-on-computer-networks-and-online-817370, Hentet: [31.03.21]
[42] J. Nurminen. (2020). *Are QoS and QoE the same thing?* Available: https://segron.com/are-qos-and-qoe-the-same-thing/, Hentet: [19.04.21]

[43] (2020). *Hva er nettnøytralitet?* Available: https://www.nkom.no/internett/nettnoytralitet/hva-er-nettnoytralitet, Hentet: [17.02.21]
[44] T. H. Johannes Skaar, Trygve Holtebekk. (2020). *Optikk*. Available: https://snl.no/optikk, Hentet: [10.02.21]
[45] B. J. Thompson. *Optics*. Available: https://www.britannica.com/science/optics, Hentet: [10.02.21]
[46] G. Stark. (2020). *Light*. Available: https://www.britannica.com/science/light, Hentet: [10.02.21]
[47] (2020). *Snell's law*. Available: https://www.britannica.com/science/Snells-law, Hentet: [25.03.21]
[48] *Brief over view of fiber optic cable advantages over copper*. Available: https://www.arcelect.com/fibercable.htm, Hentet: [10.02.21]
[49] (2020). *Fiber optics*. Available: https://www.britannica.com/science/fiber-optics, Hentet: [10.01.21]
[50] *HVA ER BØLGELENGDEMULTIPLEKSING*. Available: https://www.fossfiberoptikk.no/hva-er-bolgelengdemultipleksing, Hentet: [07.04.21]
[51] *xWDM - FLERE SIGNALER OVER SAMME FIBER*. Available: https://www.fossfiberoptikk.no/x-wdm-flere-signaler-over-samme-fiber-, Hentet: [07.04.21]
[52] J. Hecht. (2020). *Laser*. Available: https://www.britannica.com/technology/laser, Hentet: [09.03.21]
[53] *How Fiber Lasers Work*. Available: https://www.orc.soton.ac.uk/how-fibre-lasers-work, Hentet: [10.03.21]
[54] (2009). *WHAT IS OPTICAL FIBER DISPERSION?* Available: https://www.fiberoptics4sale.com/blogs/archive-posts/95052678-what-is-optical-fiber-dispersion, Hentet: [10.05.21]
[55] T. Weible. (2013). *WHAT TO CONSIDER WHEN DOING 10G OVER CWDM ?* Available: https://www.flexoptix.net/en/blog/2013/10/what-to-consider-when-doing-10g-over-cwdm/, Hentet: [11.05.21]
[56] *Shannon's Law | What does Shannon's Law mean?* Available: https://www.techopedia.com/definition/14558/shannons-law, Hentet: [03.03.21]
[57] (2020). *What is Signal to Noise Ratio and How to calculate it?* Available: https://resources.pcb.cadence.com/blog/2020-what-is-signal-to-noise-ratio-and-how-to-calculate-it, Hentet: [03.03.21]
[58] *Cyclic Redundancy Check (CRC)*. Available: https://www.techopedia.com/definition/1793/cyclic-redundancy-check-crc, Hentet: [01.04.21]
[59] S. Paonessa. *Reducing Signal Noise in Practice*. Available: https://www.predig.com/whitepaper/reducing-signal-noise-practice, Hentet: [25.03.21]
[60] F. Lied. (2018). *forvrengning*. Available: https://snl.no/forvrengning, Hentet: [01.04.21]
[61] T. Holtebekk. (2019). *interferens*. Available: https://snl.no/interferens, Hentet: [01.04.21]
[62] K. Gold. (2019). *The role of active and passive monitoring in virtual networks*. Available: https://www.exfo.com/en/resources/blog/active-passive-network-monitoring/, Hentet: [08.05.21]
[63] *Ping*. Available: https://www.paessler.com/it-explained/ping, Hentet: [23.03.21]

[64] *What is Simple Network Management Protocol (SNMP)?* Available: https://www.thousandeyes.com/learning/techtorials/snmp-simple-network-management-protocol, Hentet: [26.04.21]
[65] *SNMP Polling vs Traps* Available: https://cordero.me/snmp-polling-vs-traps/ Hentet: [26.04.21]
[66] (2016). *Microbursts, Jitter and Buffers*. Available: https://www.arista.com/assets/data/pdf/TechBulletins/AristaMicrobursts.pdf, Hentet: [05.05.21]
[67] A. networks. (2015). *AN153 V-NID Suite Metrics | Application Note*. Available: Tilsendt dokumentasjon som ikke er tilgjengelig på nett, Hentet: [05.04.21]
[68] (2021). *Understanding Two-Way Active Measurement Protocol on Routers*. Available: https://www.juniper.net/documentation/us/en/software/junos/flow-monitoring/topics/concept/twamp-overview.html, Hentet: [25.02.21]
[69] (2019). *Two-Way Active Measurement Protocol (TWAMP) Overview*. Available: https://www.aticara.com/twamp.html, Hentet: [02.02.21]
[70] M. Ruotsalainen. (2018). *L3 Latency in Regional Networks | Preparing 5G Launch*. Available: https://www.theseus.fi/bitstream/handle/10024/151488/Thesis_BSc_Ruotsalainen_Markus_v1-01.pdf?sequence=1&isAllowed=y, Hentet: [05.04.21]
[71] B. C. A. Morton. (2009). *RFC 5481: Packet Delay Variation Applicability Statement*. Available: https://www.hjp.at/(st_a)/doc/rfc/rfc5481.html, Hentet: [05.04.21]
[72] C. Kocak. (2016). *PERFORMANCE ANALYSIS OF IP NETWORK USING TWO-WAY ACTIVE MEASUREMENT PROTOCOL (TWAMP) AND COMPARISON WITH ICMP (PING) PROTOCOL IN A SATURATED CONDITION*. Available: https://www.researchgate.net/publication/313881274_PERFORMANCE_ANALYSIS_OF_IP_NETWORK_USING_TWO-WAY_ACTIVE_MEASUREMENT_PROTOCOL_TWAMP_AND_COMPARISON_WITH_ICMP_PING_PROTOCOL_IN_A_SATURATED_CONDITION, Hentet: [05.04.21]
[73] H. Khalid. (2017). *SFP Transceivers Explained*. Available: https://ourtechplanet.com/sfp-transceivers-explained/, Hentet: [22.03.21]
[74] B. Lutkevich. (2020). *embedded system*. Available: https://internetofthingsagenda.techtarget.com/definition/embedded-system, Hentet: [24.03.21]
[75] B. B. Larsen. (2020). *FPGA*. Available: https://snl.no/FPGA, Hentet: [02.04.21]
[76] Accedian. (2019). *The VCX Story* Available: https://accedian.com/wp-content/uploads/2015/05/Accedian-The_VCX_Story-2015-2Q-r3.pdf, Hentet: [02.04.21]

# Figurkilder


[1] «What is OSI model? 7 end to end layers in OSI model,» 2020. [Internett]. Available: https://iotdunia.com/7-end-to-end-layers-in-osi-model/.

[2] «TCP/IP vs OSI Model: What's the Difference?,» [Internett]. Available: https://www.guru99.com/difference-tcp-ip-vs-osi-model.html.

[3] «TCP/IP Illustrated | Encapsulation,» 1993. [Internett]. Available: https://flylib.com/books/en/3.223.1.18/1/.

[4] «Network Basics: TCP Handshake,» 2019. [Internett]. Available: https://www.itnmore.be/2018/12/14/network-basics-tcp-handshake/.

[5] «TCP Congestion control algorithm,» [Internett]. Available: https://sites.google.com/site/projectcodebank/computer-engineering-notes/tcp-congestion-control-algorithm.

[6] «Using IPSec VPN to Implement Secure Interconnection Between LANs,» 2019. [Internett]. Available: https://support.huawei.com/enterprise/en/doc/EDOC1100041799/f2298f86/using-ipsec-vpn-to-implement-secure-interconnection-between-lans.

[7] «Primer on Latency and Bandwidth,» 2013. [Internett]. Available: https://hpbn.co/primer-on-latency-and-bandwidth/.

[8] «Understanding Jitter in Packet Voice Networks (Cisco IOS Platforms),» 2006. [Internett]. Available: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/18902-jitter-packet-voice.html.

[9] «Fiber optics,» [Internett]. Available: https://www.britannica.com/science/fiber-optics.

[10] «Fiber optics,» 2021. [Internett]. Available: https://www.explainthatstuff.com/fiberoptics.html.

[11] «WHAT IS OPTICAL FIBER DISPERSION?,» 2009. [Internett]. Available: https://www.fiberoptics4sale.com/blogs/archive-posts/95052678-what-is-optical-fiber-dispersion.

[12] «SNMP Polling vs Traps,» 2020. [Internett]. Available: https://cordero.me/snmp-polling-vs-traps/.

[13] «Microbursts, Jitter and Buffers,» 2016. [Internett]. Available: https://www.arista.com/assets/data/pdf/TechBulletins/AristaMicrobursts.pdf.

[14] «TWAMP Features – Reflect OCTETS draft,» 2009. [Internett]. Available: http://www.ietf.org/proceedings/76/slides/ippm-2/ippm-2_files/ippm-2.pptx.

[15] «Packet-fragmentation-process,» [Internett]. Available: https://www.researchgate.net/figure/Packet-fragmentation-process_fig14_258165851.

Gerd Åshild Ueland / Ida Westerheim