**Øving 2 – Nettverksprogrammering**

**Oppgave 1**

Løsningen består av vedlagt *Klient.java*, *Tjener.java* og *Main.java*. Koden har kommentarer som burde forklare hva som skjer.

**Oppgave 2**

Vi opprettet JavaSSLServer.java og JavaSSLClient.java (Ukjent, 2021). Deretter opprettet vi en jar-fil til hver av disse med følgende steg:



Lager .class-fil av java-fila.

Graphical user interface, text, website

Description automatically generated

Lager MANIFEST.MF-fil, og linker main-klassen og legger den i samme mappe som class-fila.

Text

Description automatically generated

Lager jar-fil i samme mappe. Gjør dette for både server og client.

Text

Description automatically generated

Oppretter en keystore i en egen mappe i prosjektet.

Text

Description automatically generated

Kjører deretter komandoen:

java -jar -Djavax.net.ssl.keyStore=examplestore -Djavax.net.ssl.keyStorePassword=123456 "bin/server/JavaSSLServer.jar"

Text

Description automatically generatedSom kjører jar-fila og bruker keystore. Starter da server og deretter client, og har derav kryptert meldinger gjennom TLS/SSL.

Figure : Client side

Text

Description automatically generated

Figur 2: Server side

Vi kjørte Wireshark, og observerte på loopback under kommunikasjonen. Vi observerte at innholdet var kryptert med TLS. Datafangsten ligger ved som vedlegg.

Graphical user interface, text

Description automatically generated with medium confidenceSer at port 8000 er i bruk som vi spesifikt satte opp i koden, og at dataen er kryptert.

# Works Cited

Ukjent. (2021, Feb 1). *Java example of SSL Server and Client, and how to generate keystore*. Retrieved from java-buddy.blogspot.com: http://java-buddy.blogspot.com/2016/07/java-example-of-ssl-server-and-client.html