

Vedlegg A – nedkoblingsfasen, TCP-protokollen

Teoretisk sett vil nedkoblingsfasen gå som beskrevet under, men her tar vi ikke for oss alle de mange senarioene som kan forekomme (f.eks hvis begge parter sender FIN-pakker samtidig e.l.), vi ser på en typisk nedkobling mellom klient og tjener. Part A vil sende en FIN-pakke og gå stadiet «FIN_WAIT_1». Part B mottar da en FIN-pakke og går i stadiet «CLOSE_WAIT» og sender en ACK-pakke på FIN-pakken. Part A mottar ACK-pakken og går i «FIN_WAIT_2». Nå venter part A på at part B skal sende sin FIN-pakke, som er nøyaktig det part B gjør, og deretter går i «LAST_ACK» hvor den venter på en ACK-pakke fra part A. Part A mottar FIN-pakken til part B og sender en ACK-pakke og går i «TIME_WAIT». Part B mottar ACK-pakken fra A som bekrefter FIN-pakken dens, og går da i «CLOSED», B er nå ferdig nedkoblet. A derimot må vente i «TIME_WAIT» i en gitt tid (typisk noen minutter, men dette varierer fra implementasjon til implementasjon). Om A ikke får noen flere pakker fra B i løpet av den gitte tiden, vil også A gå til «CLOSED» og da være ferdig nedkoblet. Det er jo et senario at ACK-pakken til A (som kvitterer FIN-pakken til B) ikke kommer frem til B, da vil B prøve å sende FIN-pakken på nytt og A mottar da denne og vil sende ACK-pakken på nytt og resette tiden i «TIME_WAIT». (Kurose & Ross, 2017) (McKinney, 2021)

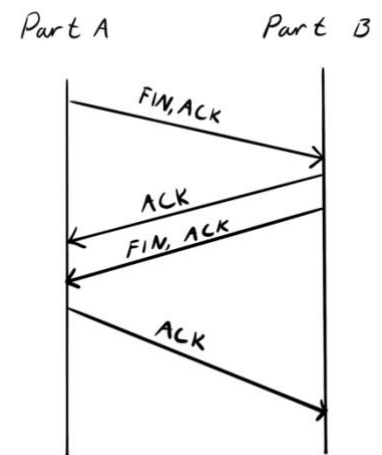


Figure 1: Illustrasjon av nedkoblingsfasen til TCP-protokollen.

Kort sagt vil man typisk se (f.eks i Wireshark, se Bilde 1) at man har pakke «mønsteret» FIN-ACK, ACK, FIN-ACK, ACK (se Figure 1).

TCP	76	49914 → 8080	[FIN, ACK]
TCP	76	8080 → 49914	[ACK] Seq=:
TCP	76	8080 → 49914	[FIN, ACK]
TCP	76	49914 → 8080	[ACK] Seq=:

Bilde 1: Skjermdump fra WireShark av nedkolbingen av en TCP-forbindelese.