

Laboration 1

DV1668

Viktor Listi
20020227-3397
vili22@student.bth.se
2023-02-12

A3)

Det gick bra att köra programmen för TCP/UDP server och klient, stötte inte på några problem vid körning. Som server användes min stationära PC med ethernet koppling till routern och för klienten användes min laptop som anslöt sig via WI-FI. Vid körning av UDP programmet så var det ett meddelande till server och ett svarsmeddelande tillbaka till klienten, med TCP körningen var det mycket mer.

B2)

```
X-Varnish: 220899079
Age: 0
Via: 1.1 webcache2 (Varnish/trunk)
Accept-Ranges: bytes
Content-Length: 223
Connection: close

<html>

<head>
<TITLE>Tic-Tac-Toe</TITLE>
</head>

<body>

<b> Tic-Tac-Toe (3x3)</b><br>
<br><u>Before turn</u><br>x o x<br>o e o<br>e e x<br><br><u>After turn</u><br>e o x<br>o x o<br>e e x<br></body>
</html>
PS C:\Users\vikto>
```

B3)

HTTP GET REQUEST:

+-----+-----+-----+

16:09:06,613,192 ETHER

|0

|d8|33|b7|aa|d1|46|d8|bb|c1|05|e3|6d|08|00|45|00|00|86|84|01|40|00|80|06|00|00|c0|a8|00|55|2e|1e|d5|ae|c
a|84|00|50|23|00|dc|a4|d3|59|d7|28|50|18|04|02|c5|42|00|00|47|45|54|20|2f|74|69|63|74|61|63|74|6f|65|2f|
69|6e|64|65|78|2e|70|68|70|3f|62|6f|61|72|64|3d|78|6f|78|6f|65|6f|65|65|78|20|48|54|54|50|2f|31|2e|31|0d|
0a|48|6f|73|74|3a|77|77|77|2e|69|6e|67|6f|6e|6c|69|6e|65|2e|6e|75|0d|0a|43|6f|6e|6e|65|63|74|69|6f|6e|3a|6
3|6c|6f|73|65|0d|0a|0d|0a|

HTTP RESPONSE:

+-----+-----+-----+

16:09:06,636,584 ETHER

|0

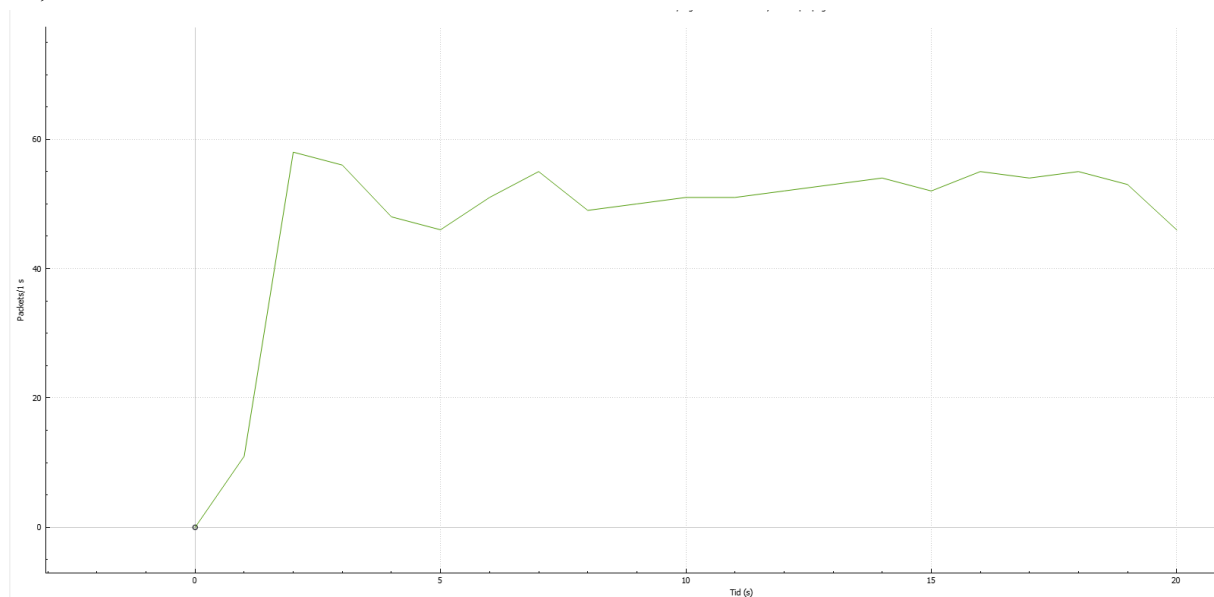
|d8|bb|c1|05|e3|6d|d8|33|b7|aa|d1|46|08|00|45|00|02|28|9e|65|40|00|37|06|de|a0|2e|1e|d5|ae|c0|a8|00|55|0
0|50|ca|84|d3|59|d7|28|23|00|dd|02|50|18|00|20|b2|77|00|00|48|54|54|50|2f|31|2e|31|20|32|30|30|20|4f|4b|
0d|0a|44|61|74|65|3a|20|54|68|75|2c|20|30|39|20|46|65|62|20|32|30|32|33|20|31|36|3a|30|39|3a|30|35|20|4
7|4d|54|0d|0a|53|65|72|76|65|72|3a|20|41|70|61|63|68|65|0d|0a|58|2d|50|6f|77|65|72|65|64|2d|42|79|3a|20
|50|48|50|2f|38|2e|30|2e|32|37|0d|0a|56|61|72|79|3a|20|41|63|63|65|70|74|2d|45|6e|63|6f|64|69|6e|67|0d|0
a|43|6f|6e|74|65|6e|74|2d|54|79|70|65|3a|20|74|65|78|74|2f|68|74|6d|6c|3b|20|63|68|61|72|73|65|74|3d|55|
54|46|2d|38|0d|0a|58|2d|56|61|72|6e|69|73|68|3a|20|32|37|38|31|33|35|37|32|35|0d|0a|41|67|65|3a|20|30|0
d|0a|56|69|61|3a|20|31|2e|31|20|77|65|62|63|61|63|68|65|32|20|28|56|61|72|6e|69|73|68|2f|74|72|75|6e|6b
|29|0d|0a|41|63|63|65|70|74|2d|52|61|6e|67|65|73|3a|20|62|79|74|65|73|0d|0a|43|6f|6e|74|65|6e|74|2d|4c|6
5|6e|67|74|68|3a|20|32|32|33|0d|0a|43|6f|6e|6e|65|63|74|69|6f|6e|3a|20|63|6c|6f|73|65|0d|0a|0d|0a|3c|68|7

4|6d|6c|3e|0d|0a|0d|0a|3c|68|65|61|64|3e|0d|0a|3c|54|49|54|4c|45|3e|54|69|63|2d|54|61|63|2d|54|6f|65|3c|2f|54|49|54|4c|45|3e|0d|0a|3c|2f|68|65|61|64|3e|0d|0a|0d|0a|3c|62|6f|64|79|3e|0d|0a|0d|0a|3c|62|3e|20|54|69|63|2d|54|61|63|2d|54|6f|65|20|28|33|78|33|29|3c|2f|62|3e|3c|62|72|3e|0d|0a|3c|62|72|3e|3c|75|3e|42|65|66|6f|72|65|20|74|75|72|6e|3c|2f|75|3e|3c|62|72|3e|78|20|6f|20|78|3c|62|72|3e|6f|20|65|20|6f|3c|62|72|3e|65|20|65|20|78|3c|62|72|3e|3c|62|72|3e|3c|62|72|3e|3c|75|3e|41|66|74|65|72|20|74|75|72|6e|3c|2f|75|3e|3c|62|72|3e|65|20|6f|20|78|3c|62|72|3e|6f|20|78|20|6f|3c|62|72|3e|65|20|65|20|78|3c|62|72|3e|3c|2f|62|6f|64|79|3e|0d|0a|3c|2f|68|74|6d|6c|3e|

B4)

Jag lyckades skapa kommunikationen och skicka en HTTP request och sedan få en HTTP response med den ombedda sidans html kod genom att först ansluta sig mot hemsidan (www.ingonline.nu) på port 80 och sedan skicka ett meddelande till den med en GET request där jag frågar efter innehållet på sidan ([/tictactoe/index.php?board=xxxoeoeex](http://tictactoe/index.php?board=xxxoeoeex)) med versionen HTTP/1.1 och där jag specificerar att kontakten med servern ska stängas efter att jag har fått min HTTP response. Jag stötte inte på några problem under försöken att få HTTP responsen.

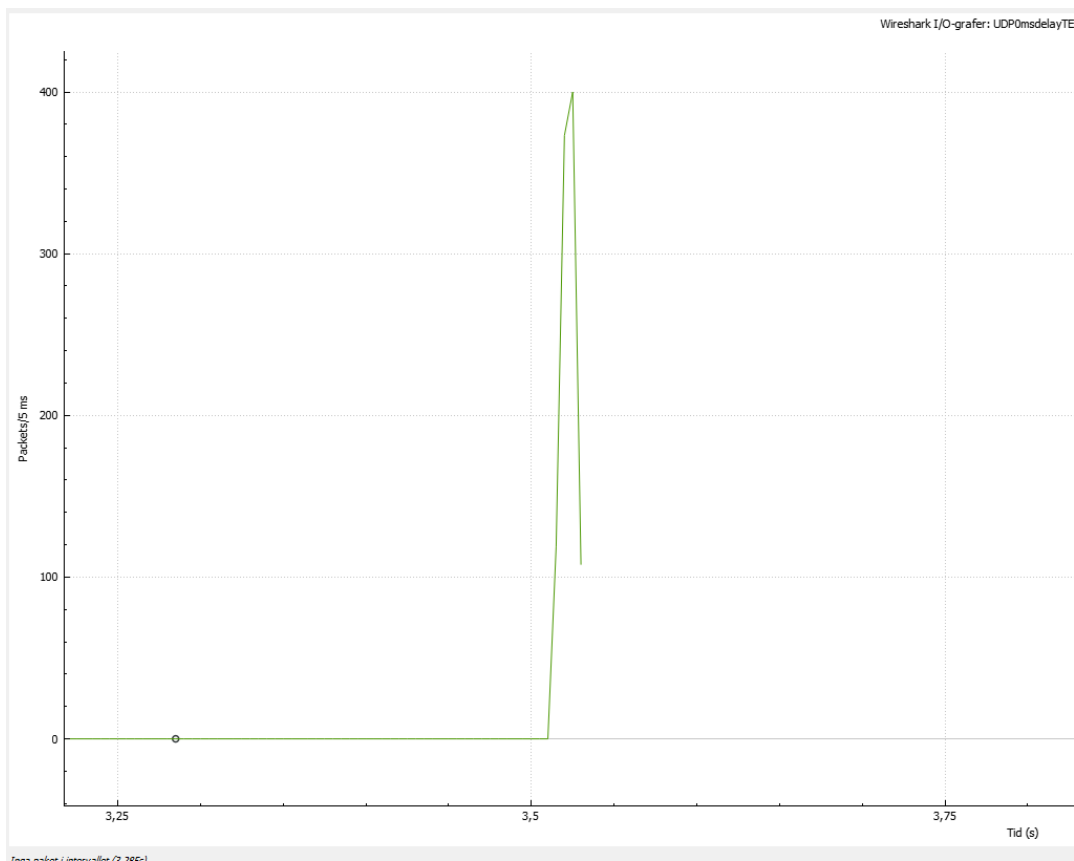
C3)



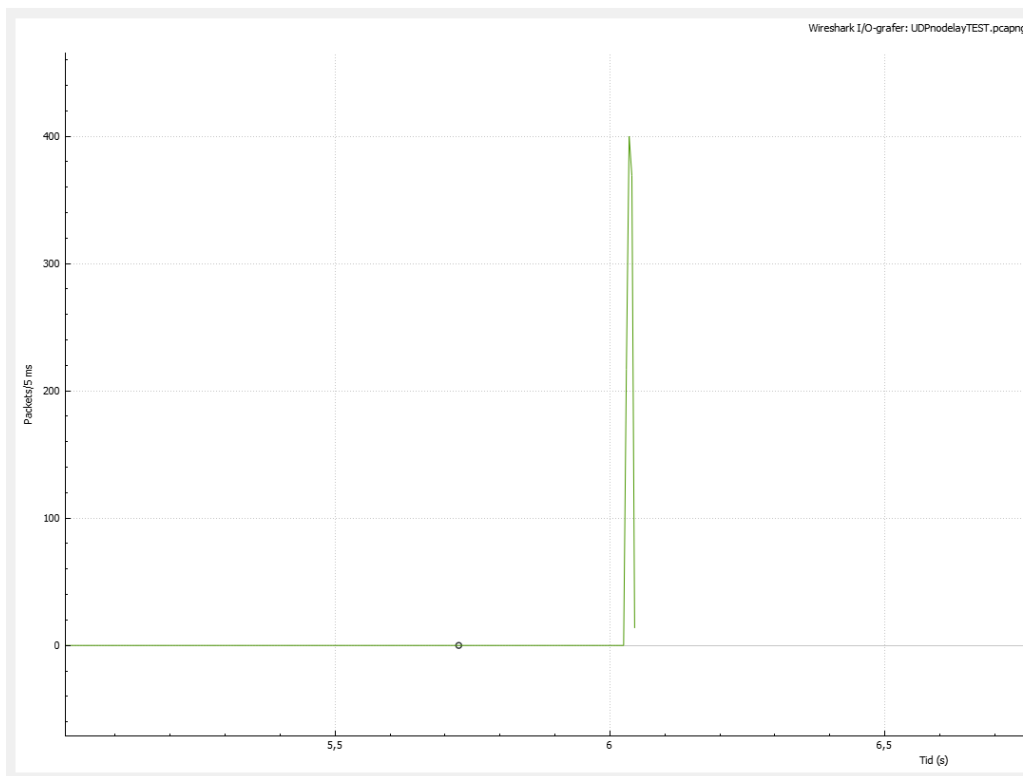
(UDP, 20 ms delay, 1 s intervall)

Grafen stämmer överens med förväntat resultat då den levererade alla 1000 paketen som skickas under 20 sekunder.

C4)



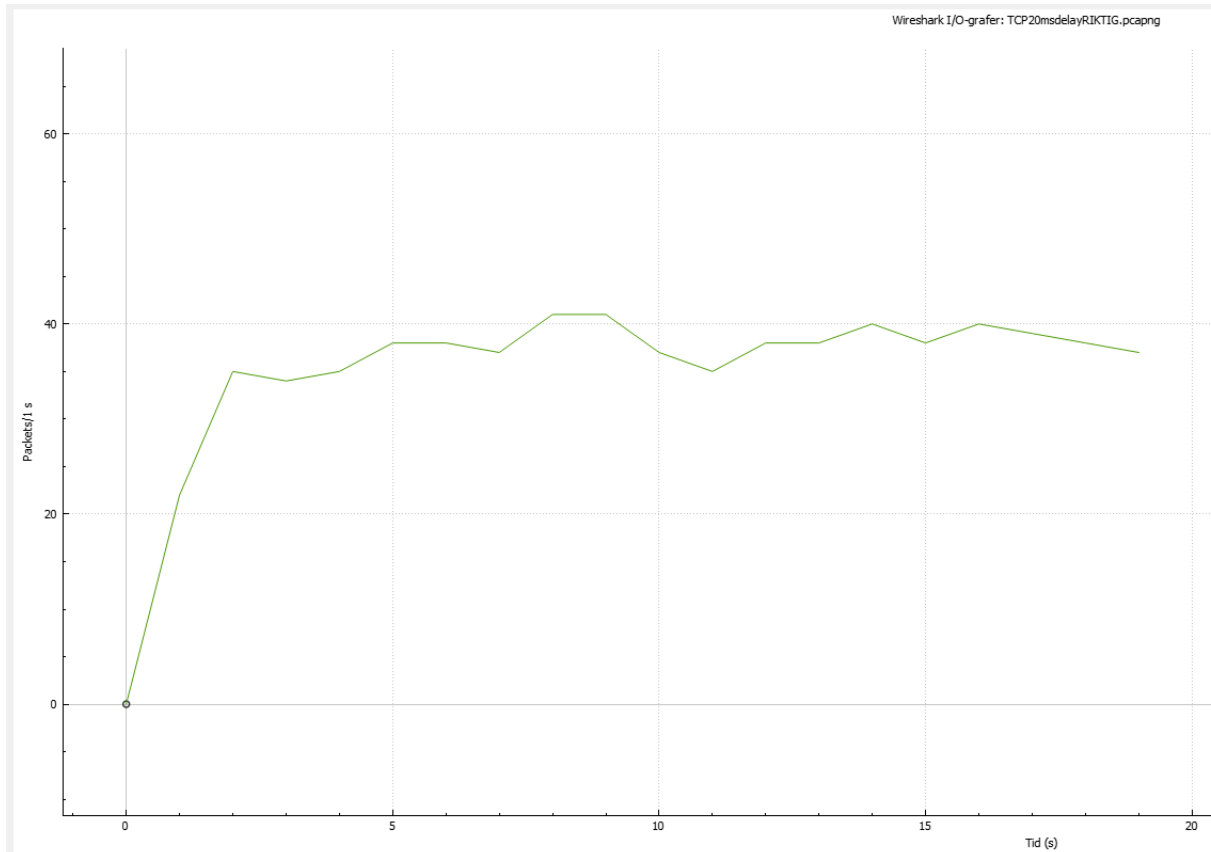
(UDP, 0 ms delay, 5 ms intervall)



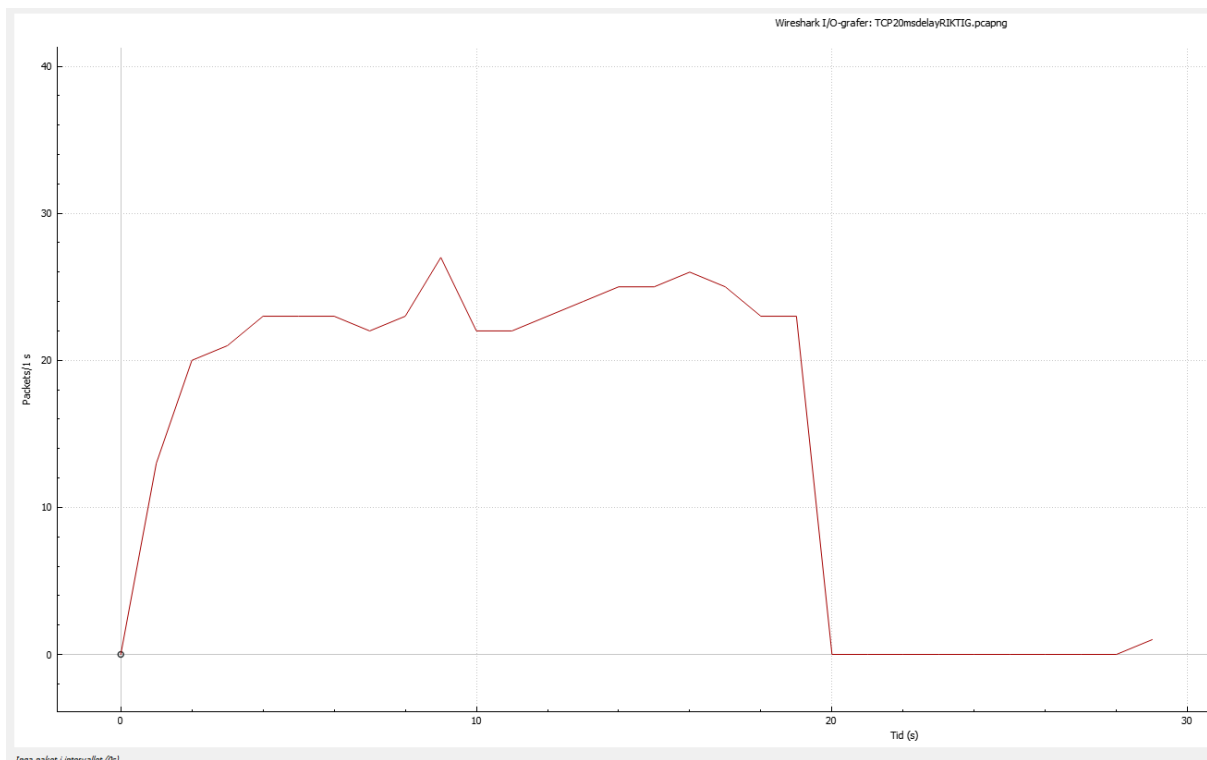
(UDP, no delay, 5 ms intervall)

Jag tror att de faktorer som avgör hur snabbt paketen kan skickas när vi inte har någon delay på hur snabbt paketen får skickas är de sakerna som sitter mellan klienten och servern. t.ex så sitter det en switch och en router mellan klienten och servern under dessa testerna, båda dessa variabler kan sakta ner och leda till paket förluster beroende på hur snabbt de kan hantera paketen. Klienten kommer skicka ut paketen så snabbt som den kan men sen kan många av paketen fastna i dörrarna mellan server och klient (switchen och routern).

C7)



(TCP, 20 ms delay, 1 s intervall, filter ip.dst==192.168.0.4 (destination = server))



(TCP, 20 ms delay, 1 s intervall, filter ip.src==192.168.0.4 and tcp (source = server))

Resultaten verkar stämma överens med förväntningarna av hur dessa grafer skulle se ut.

D1)

UDP och TCP har sina olika styrkor och svagheter. UDP är snabbt men har ingen garanti att paketen kommer fram, medans TCP är lite långsammare men har system på plats för att garantera att alla paketen kommer fram. Båda har sina olika tillfällen där den ena är att föredra över den andra. Under laborationen så observerade jag att vid höga hastigheter så hade UDP problem med tappade paket som då eftersom den inte har någon garanti aldrig kom fram. Vid lite långsammare hastigheter kom oftast alla paketen fram. En annan anmärkning om UDP är att alla paketen levereras hos servern i den ordningen dem skickades alltså om man går genom paketen en för en så ser man att dem ligger i nummerordning. Vid körning av TCP kom alltid alla paketen fram vid både låg hastighet och när vi test körde programmet med 0 ms delay. En anmärkning dock är att TCP paketen väldigt ofta kommer i oordning även vid låga hastigheter såsom 50 paket/s. Vilken ledde till att den att koden som kollade efter paketen i oordning markerade väldigt tidigt. Alla paketen vid TCP kom fram till slut.

D2)

Uppgiften som helhet var väldigt bra för att lära sig om hur paketen faktiskt tar sig fram då vi kunde bevaka dem via wireshark och I/O graferna. Laborationen har gett insyn på de praktiska delarna på mycket av det vi har lärt oss hittills i kursen vilket är bra. Under testerna så stötte jag på många problem med tiden som programmet tar på sig att köra och leverera alla paket till fullo, dessa problem löstes tillslut med lite förenkling och genom att testa runt med tiden på sleep funktionen. Ytterligare ett problem jag stötte på var att sleep funktionen inte funkade som den skulle göra vid mina tester då en sleep på 0.0155 sekunder tog ungefär 5 sekunder längre på sig att köra än en sleep på 0.016 sekunder vilket logiskt inte ska stämma, detta löstes till slut så att körningen av programmet hamnade runt 20 sekunder.