

IKN Obl. AFl 1

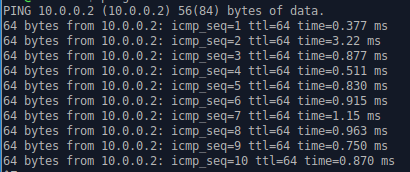
Gruppe 12 – 05-04-2018



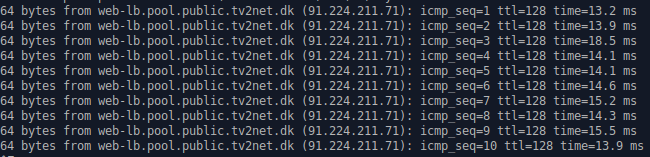
|  |  |
| --- | --- |
| Navn | Studienr |
| Frederik Kastrup Mortensen | **201607221** |
| Søren Schou Mathiasen | 201605264 |
| Martin Haugaard Andersen | 201605036 |
| Alexander Lichtenstein Davidsen | 201608479 |

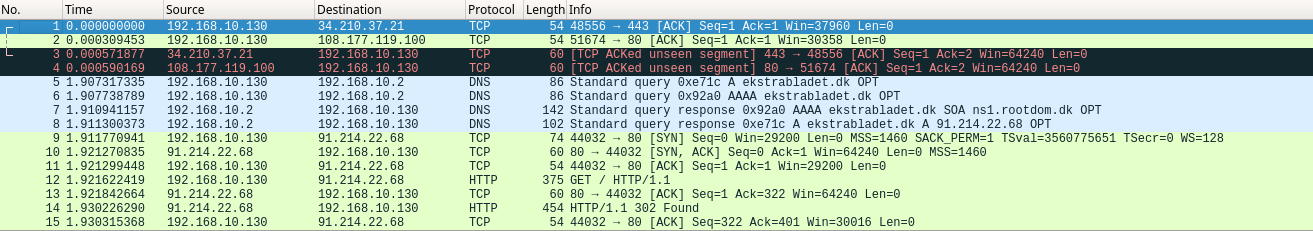
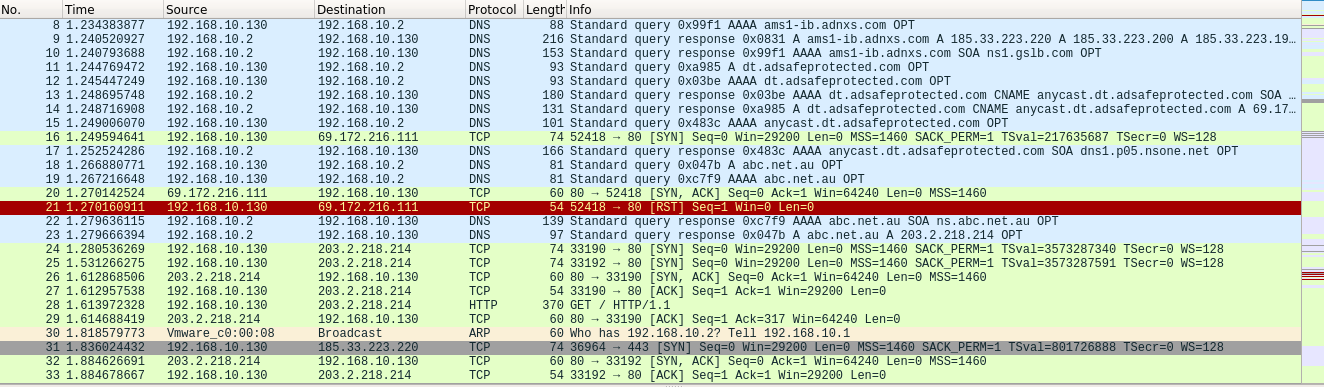
# Øvelse 3

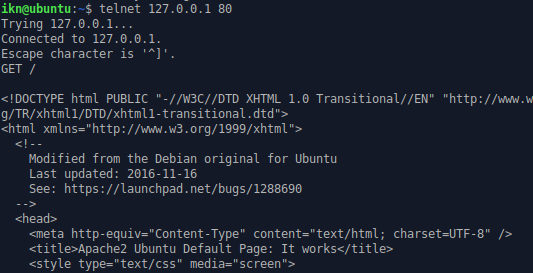
1. Mål den tid der går fra en ping kommando startes i H1 til ping-respons fra H2 modtages i H1.
2. Mål minimum-/maksimum-/gennemsnits-forsinkelsestider og standardafvigelsen for 10 på hinanden følgende ping-kommandoer, udført som i punkt 1.

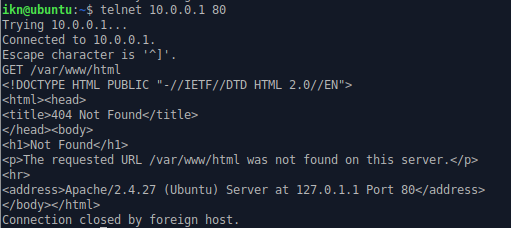


1. Mål den tid der går fra kommandoen ping –c 1 www.tv2.dk startes i H1 til ping-respons fra web-serveren www.tv2.dk modtages i H
2. Mål minimum-/maksimum-/gennemsnits-forsinkelsestider og standardafvigelsen for 10 på hinanden følgende ping-kommandoer, udført som i punkt 3.



1. Mål den tid der går fra en web-side i en dansk web-server ønskes modtaget i H1, til web-serveren responderer (anvend ikke længere www.tv2.dk, men en anden dansk web-side, som vælges tilfældigt ). Anvend din Web Browser som client. Anvend wireshark til måling af tidsforsinkelsen. Vi anvender dette måleværktøj til tidsmåling i stedet for pingkommandoen, da det ikke er givet, at den server vi vil anvende til tidsforsinkelses-målingen giver respons på en ping-kommando (f.eks. svarer wwww.dr.dk og www.dmi.dk ikke på en ping-kommando)  
   
2. Mål vha. samme målemetode den tid der går fra en australsk web-side (som er tilfældigt valgt) ønskes modtaget i H1 web-serveren, til web-serveren responderer. Anvend din Web Browser som client. Webserveren, som indeholder den australske web-side, befinder sig sandsynligvis fysisk meget længere væk end den danske web-sides web-server gør.  
   
3. Er der forskel på tidsforsinkelses-målingerne i punkt 5 og punkt 6? Hvis der er forskel, hvad kan årsagen være til at der denne forskel?
4. Undersøg vha. wireshark hvad der sker, når denne web-side hentes vha. en Web Browser:
5. Lokal forbindelse fra H1.

  
Forbindelse fra H2 til H1 server.



1. Test protokollerne: HTTP 0.9, HTTP 1.0 og HTTP 1.1.vha. telnet med fokus på oprettelse/nedlukning af TCP-connect

**Lukkes TCP-forbindelsen straks når HTTP 0.9 anvendes?   
  
Lukkes TCP-forbindelsen straks når HTTP 1.0 anvendes?   
Lukkes TCP-forbindelsen straks når HTTP 1.1 anvendes?   
Hvis TCP-forbindelsen ikke lukkes umiddelbart, lukkes den så automatisk lidt senere?  
 Hvor lang tid går der?  
 Hvad er fordelen ved at nedlukningen af TCP-forbindelsen udskydes?   
Er det Web-server eller WEB-client, der lukker TCP-forbindelsen?   
Hvilken version af Apache serveren anvendes?**

# Øvelse 6

Øvelsen består af et server/klient system, hvor en klient kan anmode om en fil, hvorved serveren sender filen tilbage.

## Server

Der er i denne del af øvelsen udviklet en server, der skal modtage en tekststreng fra en klient. Tekststrengen indeholder et filnavn, der beskriver hvilken fil, klienten ønsker at modtage samt serverens IP adresse. Hvis filen ikke eksisiterer, vil klienten få en fejlbesked. Hvis filen herimod eksisterer, vil serveren sender filen til klienten i bider af max 1000 bytes indtil hele filen er sendt.

## Klient

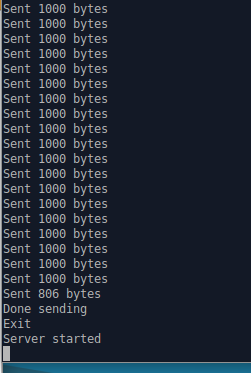
Klienten sender indledningsvis en tekststreng der indeholder et filnavn og serverens IP adresse. Klienten modtager filens størrelse hvorefter klienten modtager filen fejlfrit fra serveren.

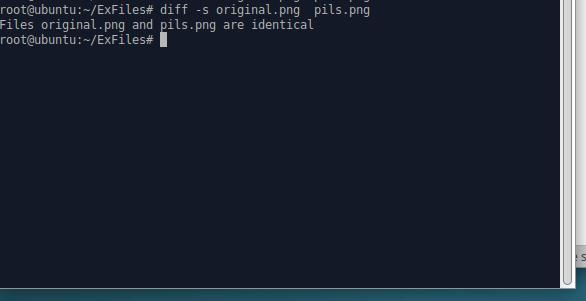
## Resultat

Der er sendt en jpg fil fra serveren til klienten. Den sendte fil sammenlignes med originalen, ved brug af diff funktionen.

Det ses her at klienten anmoder om at modtage filen kaldet ”pils.png”. Herefter modtage klienten filstørrelsen og begynder at modtage filen fra serveren. I nederste del af billedet ses det at klienten har modtaget filen ”pils.png” og gemt den i ”ExFiles” mappen.

## 

Her ses fil-transaktionen fra serverens synspunkt. Her kan det ses at serveren sender i pakker af 1000 bytes og at serveren fungerer iterativt. Efter serveren er færdig med at sende, er den her klar til at sende endnu en fil, som det ses i bunden af billedet.   
  
På nedenstående billede, sammenlignes den sendte fil med originalen, ved brug af diff funktionen. Det meddeles her at filerne er identiske.



# Øvelse 7

1. I denne del af øvelsen skal der designes og implementeres en UDP-server med support for en client af gangen. Serveren skal kunne håndtere beskeder der sendes fra en klient. Hvis serveren modtager et ”U”, skal informationen i filen /proc/uptime returnerer til klienten.  
   Hvis serveren modtager et ”L”, skal informationen fra /proc/loadavg returneres til klienten. Serveren skal kunne håndtere både store og små bogstaver fra klienten.
2. Der skal her designes og implementeres en UDP-klient, som kan sende en besked til en server. Beskeden skal være ”U”, ”u”, ”L” eller ”l”, svaret fra serveren skal her udskrives til operatøren af klienten.