Øvelse 3 *29.08.2019/MA*

Delay, Wireshark og HTTP Client/Server

Formålet med øvelsen er at anvende Wireshark til analyse af netværksfunktionalitet og at undersøge HTTP-protokollens funktionalitet.

I de følgende punkter (1-2) skal der vha. af testapplikationen *ping* måles den tidsforsinkelse der opstår, når to virtuelle maskiner kommunikerer indbyrdes i din PC.

1. Mål den tid der går fra en ping kommando startes i H1 til ping-respons fra H2 modtages i H1.
Hint:
#ping –c 1 10.0.0.2 (ping-kommandoen udføres én gang, respons-tiden udskrives)
#ping –c 5 10.0.0.2 (ping-kommandoen udføres fem gange, respons-tiden udskrives)
#ping 10.0.0.2 (ping-kommandoen gentages kontinuerligt, respons-tiden udskrives for hver ping)

2. Mål minimum-/maksimum-/gennemsnits-forsinkelsestider og standardafvigelsen for 10 på hinanden følgende ping-kommandoer, udført som i punkt 1.

I de følgende punkter (3-7) skal der måles den tidsforsinkelse der opstår, når din virtuelle maskine via eth0 kommunikerer med en server, som er placeret et sted på Internettet.

3. 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 H1.

4. Mål minimum-/maksimum-/gennemsnits-forsinkelsestider og standardafvigelsen for 10 på hinanden følgende ping-kommandoer, der tester *www.tv2.dk*

5. Prøv at anvende ping-kommandoen til at teste forsinkelsestiden til *www.dmi.dk*
Det viser sig, at det ikke er muligt – *www.dmi.dk* og mange andre web-servere responderer ikke på en ping-måling. Hvis tidsforsinkelsen skal måles kan Wireshark anvendes til dette formål.
Anvend Wireshark til måling af tidsforsinkelsen til f.eks. www.au.dk (PS www.au.dk understøtter ping respons, men måling af tidsforsinkelsen i dette punkt SKAL foregå med Wireshark). Anvend din Web Browser som client.

Undgå at måle på en https-baseret web-side, dette vil medføre at målingen bliver unødvendig kompliceret!

Fra https://en.wikipedia.org/wiki/Transmission_Control_Protocol#CONNECTION-ESTABLISHMENT:

To establish a connection, TCP uses a three-way handshake. Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections: this is called a passive open. Once the passive open is established, a client may initiate an active open. To establish a connection, the three-way (or 3-step) handshake occurs:

1. SYN: The active open is performed by the client sending a SYN to the server. The client sets the segment's sequence number to a random value A.
2. SYN-ACK: In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number i.e. A+1, and the sequence number that the server chooses for the packet is another random number, B.
3. ACK: Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value i.e. A+1, and the acknowledgement number is set to one more than the received sequence number i.e. B+1.

At this point, both the client and server have received an acknowledgment of the connection. The steps 1, 2 establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps 2, 3 establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

Respons tiden er altså fra SYN er afsendt fra klient til SYN-ACK er modtaget fra serveren, hvilket kan/skal identificeres i Wireshark!

6. Mål vha. Wireshark 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.

Undgå at måle på en https-baseret web-side, dette vil medføre at målingen bliver unødvendig kompliceret!

7. Er der forskel på tidsforsinkelses-målingerne i punkt 5 og punkt 6?
Hvis der er forskel, hvad kan årsagen være til at der er denne forskel?

MÅLING PÅ NETVÆRKSAKTIVITET VED DOWNLOAD AF WEB-SIDER FRA EKSTERN WEB-SERVER

8. Undersøg vha. Wireshark hvad der sker, når denne web-side hentes vha. en Web Browser:
http://i4prj.ase.au.dk/I4IKN
Hint: fokuser på følgende 'Protocol'-hændelser: tidsforsinkelse, DNS, TCP, HTTP

INSTALLATION OG OPSÆTNING AF EGEN WEB-SERVER, ANVENDELSE AF HTTP-CLIENT's

9. Installer en web-server (apache, nyeste stabile version) i din virtuelle maskine (H1).
Fremgangsmåde:
Installer Apache-serveren vha. denne kommando:
 # apt-get install apache2 (tryk 'Enter' hvis der stilles spørgsmål under installationen. Enter=default)
Efter installationen ligger Apache-serveren i folderen */etc/apache2*

10. Etabler en LAN-forbindelse mellem en web-serveren (H1) og en web-client (H2)
I første omgang anvendes en simpel, telnet baseret web-client i H2.

```
telnet syntax: telnet hostname port
Default port is 23 if not specified!

You might test yor telnet by typing:
# telnet rainmaker.wunderground.com 23
```

Fremgangsmåde:
Test først lokalt at web-serveren er installeret på H1 vha. denne terminalkommando (som udføres i H1):
 # telnet 127.0.0.1 80 (alternativt kan der skrives *http* i stedet for 80)
GET / <enter>
(Der skal nu udskrives en tekst på skærmen, som svarer til indholdet af den html-fil der ligger i mappen */var/www/html* i H1.

Test derefter, at grundlaget for kommunikationen mellem H1 og H2 er til stede ved at verificere at H1 og H2 kan ping'e indbyrdes.

Test at H2 kan hente test-web-siden fra H1 via LAN-forbindelsen vha. kommandoen:
 # telnet <H1's IP-adresse> 80
og derefter:
GET / <enter>

Mere om Telnet:

Liste af BBS guides: https://www.telnetbbsguide.com/bbs/connection/telnet/list/detail/

11. Test protokollerne: HTTP 0.9, HTTP 1.0 og HTTP 1.1.vha. telnet med fokus på oprettelse/nedlukning af TCP-connection og på persistent/non-persistent HTTP-kommunikation vha. HTTP-protokollen (uden/med pipelining).
NB: Husk at anvende 2 * enter når HTTP version 1.0 og HTTP version 1.1 anvendes. Hvis dette ikke gøres returnerer Apache serveren ingen respons. Ved anvendelse af HTTP version 1.1 skal Request Header "host" anvendes. Parameteren til host kan f.eks. være: *host:10.0.0.1*
Spørgsmål:
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?

12. Anvend Firefox web-browser som web-client i H2 sammen med Apache web-server i H1
Fremgangsmåde:
Opret en web side med 3 .jpg billeder i H1 (disse kan passende ligge i directory */var/www/html*). Billederne skal have "read"-permission (sættes under "properties").
Tjek at web-siden ser ud som forventet vha. din Web Browser.
Analyser relevante hændelser på LAN-segmentet vha. Wireshark. Herunder: beskriv den anvendte HTTP-version, request / response headers, persistent/non-persistent connection, pipelining/no pipelining.

Mere om HTML:

Du kan læse mere om HTML format på: https://www.w3schools.com/