

TCP en UDP

D. Leeuw

1 december 2023

v.0.5.0



Dit werk is uitgegeven onder de Creative Commons BY-NC-SA Licentie en laat anderen toe het werk te kopiëren, distribueren, vertonen, op te voeren, en om afgeleid materiaal te maken, zolang de auteurs en uitgever worden vermeld als maker van het werk, het werk niet commercieel gebruikt wordt en afgeleide werken onder identieke voorwaarden worden verspreid.

Over Dit Document

0.1 Leerdoelen

Dit document geeft een korte introductie TCP/IP. TCP/IP is het protocol dat gebruikt wordt op het Internet en de meeste lokale netwerken om computers met elkaar kunnen laten praten.

De leerdoelen zijn:

- kennis van IP-adressen
- kennis van subnet-masks
- kennis van (default) gateway
- kennis van TCP en UDP als transport protocollen
- kennis van het routeren van data over het Internet

0.2 Voorkennis

Voordat je aan deze les begint is er geen voorkennis vereist.

- Nog geen kennis nodig

Inhoudsopgave

Over Dit Document	i
0.1 Leerdoelen	i
0.2 Voorkennis	i
1 Data integriteit	1
1.1 TCP	1
1.2 UDP	2
2 De eindbestemming	3
3 Opdrachten	5
3.1 nmap	5

Hoofdstuk 1

Data integriteit

Op ons lokale netwerk (LAN) kunnen we er redelijk vanuit gaan dat data ongeschonden aankomt, maar hoe weten we zeker dat als we data van de ene kant van de wereld naar de andere kant sturen dat onze data dan goed is angekommen. Het zou toch heel vervelend zijn als we een e-mail versturen dat er dan enkele zinnen missen. Het controleren dat alle pakketten die we verzonden hebben ook daadwerkelijk angekommen zijn is de verantwoordelijkheid van de Transport-layer (OSI layer 4).

1.1 TCP

TCP is het transport protocol dat gebruikt wordt met IP. Vandaar dat we de protocol-stack kennen als TCP/IP. TCP staat voor Transport Control Protocol en het is verantwoordelijk voor het controleren van de verzonden en ontvangen data.

Een e-mail wordt niet in zijn geheel in een packet gestopt en dan over het netwerk gestuurd, maar het wordt opgehakt en allemaal kleine packetjes die stuk voor stuk verstuurd worden en aan de andere kant dus weer tot een e-mail verwerkt moeten worden. De reden achter dit systeem is heel simpel, stel iemand wil een film kijken over het netwerk en deze film duurt 1,5 uur. Als dan de film verstuurd zou worden als 1 packet dan kan er gedurende 1,5 uur niemand gebruik maken van het netwerk. Om dat te voorkomen wordt de film opgehakt in hele kleine stukjes die over het netwerk verstuurd worden. Tussen de kleine packetjes in kan dus ook iemand anders nog gebruik maken van het netwerk. Zo wordt het netwerk gedeeld, de data wordt gesegmenteerd en de data op laag 4 van het OSI-model heten dan ook segmenten.

Bij het versturen van segmenten moeten ze genummerd worden. Door

ze te nummeren weten we of we alle segmenten hebben en of ze in de juiste volgorde staan. Het is aan TCP om dit allemaal te verzorgen.

Een andere taak van TCP is om de verzendende partij te laten weten dat alles goed is aangekomen. Voor elk ontvangen segment stuurt TCP een bevestiging terug. De verzendende partij weet dan dat het goed is aangekomen. Mocht het na een bepaalde tijd geen bevestiging van ontvangst krijgen dan kan het dat segment opnieuw versturen. Op deze manier weten we zeker dat de complete e-mail bij de ontvanger aankomt en dat er geen woorden of zinnen missen.

1.2 UDP

TCP is behoorlijk omslachtig. De verzendende partij moet in de gaten houden of het voor elk verzonden segment een ontvangstbevestiging heeft ontvangen en de ontvangende partij moet nagaan of alle data is aangekomen en in de juiste volgorde staat. Toch is dit de enige manier om data op de juiste manier over een onbetrouwbaar netwerk te versturen.

Maar wat nou als de integriteit van de data niet zo belangrijk is en dat snelheid bijvoorbeeld van veel groter belang is?

Laten weer de film van 1,5 uur nemen. Als we die op hakken in hele kleine segmenten, dan is het misschien niet zo heel erg als we een segment missen. Dat levert misschien een kleine vervorming op op het scherm, maar de film gaat gewoon verder. Als we zouden moeten wachten tot het segment opnieuw verstuurt is dan zou het een freeze van het scherm kunnen opleveren omdat we zitten te wachten op dat ene segment. Bij een film is het dus belangrijker dat de film doorgaat dan dat hij volledig elke pixel laat zien.

Voor dit soort data overdracht is UDP bedacht. UDP werkt ook op de transport-laag, maar het controleert niet of de data correct is aangekomen en het kijkt ook niet of de segmenten wel in de juiste volgorde staan. Hogere lagen in de protocol-stack kunnen natuurlijk altijd nog deze functies uitvoeren, maar UDP doet het niet.

UDP wordt dan ook veel gebruikt als de integriteit van data niet zo belangrijk is, maar snelheid en reactietijd wel. Denk hierbij aan telefonie over Internet, videoconferencing, maar ook het spelen van netwerk-games.

Hoofdstuk 2

De eindbestemming

SPORT		DPORT	
SEQ#			
ACK#			
Length	Res	Flags	Window
Checksum		Pointer	
Options		Padding	
Data			

SPORT Source Port

DPORT Destination Port

SEQ# Sequence Number

ACK# Acknowledgement Number

Length Length of the header

RES Reserved bits

FLAGS Flags that can be set

Window Sliding window-size

Checksum TCP checksum

Pointer Urgent pointer

Options 0 of meer opties

Padding 0 of meer padding bits

Data Data afkomstig van hoger gelegen lagen

SPORT	DPORT
Length	Checksum
Data	

SPORT Source Port

DPORT Destination Port

Length Length of the header

Checksum UDP checksum

Data Data afkomstig van hoger gelegen lagen

Hoofdstuk 3

Opdrachten

3.1 nmap

