LAB 08 IP

Daniel Eide, Jørgen Lybeck Hansen, Jonas Dam, Elaine Sajets, Christian Fredrik Thorne, Bastian Strang

Dato: 15.03.2015

Versjon 1.0

Kurs: IS-105

**Teorioppgaver**

1)

# Version

Spesifiserer formatet på IP-pakkens header.

# Internet Header Length (IHL)

Spesifiserer lengden på IP-pakkens header, i 32-bit ord. Minsteverdi for en gyldig header er 5.

# Type Of Service (TOS)

# Total Length (TL)

Inneholder lengden på datagrammet.

# Identification

Blir brukt til å identifisere fragmentene til et datagram, ved å bruke fragmenter fra et annet datagram. Den opprinnelige protokoll-modulen av et internet-datagram setter et identifikasjons-felt som må ha unik verdi for source/destination-paret, samtidig som protokollen må ha samme verdi så lenge datagrammet er aktivt i internett-systemet.

# Flags

Et 3-bit field som brukes til å kontrollere eller identifisere fragmenter. Rekkefølgen er:

* Bit 0: Reserved; must be zero.
* Bit 1: Don’t fragment (DF)
* Bit 2: More Fragments (MF)

Hvis DF-flagget blir satt, og fragmentering er nødvendig for å rute pakken, blir pakken droppet.

# Fragment offset

Brukes til å dirigere sammensetningen av et fragmentert datagram.

# Time To Live (TTL)

Et timer-field som brukes til å spore levetiden til datagrammet. Når TTL-feltet er dekrementert til 0, blir datagrammet forkastet.

# Protocol

Dette feltet spesifiserer den neste innkapslede protokollen.

# Header Checksum

Brukes til å feilsjekke headeren. Når en pakke ankommer en ruter, kalkulerer ruteren headerens checksum, og sammenligner det med checksum field. Hvis verdiene ikke matcher, forkaster ruteren pakken.

# Source Address

IP-addressen til avsender.

# Destination Address

IP-addressen til mottaker.

2) **Bildet over viser en sammenligning av en IPv4 og en IPv6 header. Forklar hvorfor source address og destination address beslaglegger mye større plass i IPv6 i forhold til IPv4.**

IPv6 har større addresser, og krever derfor 128bit i stedet for 32bit. Med dette blir det også flere IP-addresser å dele ut, siden det er heksadesimaler. Det vil da ikke gå tom for IP-addresser like raskt som IPv4.

**LAB oppgaver**

**1) Velg den første ICMP-pakken som er sendt fra din maskin. Trykk på pluss-tegnet foran ”Internet protocol” i det midterste vinduet i Wireshark, slik at vi får utforsket denne. Hva er IP-adressen til din maskin?**

>>> ping [www.google.com](http://www.google.com)

ping -s brukes for å endre packetsize. Under endrer vi pakkestørrelsen til 3500.

>>> ping -s 3500 [www.google.com](http://www.google.com)

IP adressen vår er:

Source: 158.37.246.126 (158.37.246.126)

**2) Det er et felt som heter ”Protocol” Dette feltet har en verdi som indikerer hva slags protokoll som blir sendt over IP-laget. Hva er verdien på dette feltet og hva forteller den oss?**

Feltet gir oss ICMP (1) og det betyr at ICMP koden er 1. Typen på en ICMP pakke forteller oss hva som skjedde med pakken i prosessen.

Code 1: Destination host unreachable

Code 2: Destination protocol unrearchable

Code 3: Destination port unreachable.

Ressurs: http://en.wikipedia.org/wiki/Internet\_Control\_Message\_Protocol

Protocol: ICMP (1)

**3) Hvor mange bytes er det i IP-headeren?**

Under IPv4 i midtvinduet finner vi Header length.

Header length: 20 bytes

Størrelsen er 20 bytes.

**4) Er dette IP-datagrammet fragmentert? Forklar hvordan du kom frem til om det er fragmentert eller ikke.**

Ved å lese av Fragment Offset finner vi ut om IP-datagrammet (pakken) er fragmentert eller ikke. I vårt eksempel henter vi informasjonen:

Fragment offset: 0

Vi ser at Fragment offset = 0, derfor er den ikke fragmentert.

**Finn den første ICMP-pakken som ble sendt fra din datamaskin etter at du endret pakkestørrelsen til 3500. Pakkenummeret er antakeligvis en del høyere på disse pakkene. Merk deg nummeret og sorter på pakkenummer etterpå.**

Dette er den siste pakken før størrelsen ble endret til 3500:

146 14.667718000 64.233.161.104 158.37.246.126 ICMP 98 Echo (ping) reply id=0x0efc, seq=14/3584, ttl=47 (request in 142)

Dette er den første pakken etter at størrelsen ble endret til 3500:

444 32.647781000 158.37.246.126 64.233.161.147 ICMP 582 Echo (ping) request id=0x0f08, seq=1/256, ttl=64

**5) Denne pakken er fragmentert, men hvordan kan du se det?**

IPv4 informasjonen til pakke 444, som er den første pakken etter størrelseendring til 3500 forteller oss at:

Fragment offset: 2960

Så lenge Fragment offset >= 0 betyr det at pakken er fragmentert.

**6) Finn det første fragmentet av det fragmenterte IP-datagrammet (Det vil ha IP som protokoll i Wireshark, så da må du fjerne ”&& icmp” fra filteret hvis du satte det på)**

IP-datagrammets første fragment:

442 32.647768000 158.37.246.126 64.233.161.147 IPv4 1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=8f60) [Reassembled in #444]

**a) Hvilken informasjon i IP-headeren indikerer at datagrammet er fragmentert og hvilken informasjon i IP-headeren indikerer at dette er det første fragmentet? Hvor mange byte med data ble sendt med dette IP-datagrammet?**

“Fragmented IP protocol” forteller oss at datagrammet er fragmentert.

Flags 0x01 (More Fragments) forteller oss at det er flere fragmenter i datagrammet. Vi kan se at det første fragmentet er No. 442 som har Fragment offset: 0, mens det andre fragmentet er No. 443 som har Fragment offset: 1480.

Total Length: 1500  
Lengden på datagrammet er 1500 bytes. Dette tror vi er fordi den fragmenterte pakken er på 1480, så den har litt spillerom.

**7) Finn det andre fragmentet av det fragmenterte IP-datagrammet. Hvilken informasjon i IP-headeren indikerer at dette ikke er det første fragmentet? Hvordan vet du om det er flere fragmenter?**

IP-datagrammets andre fragment:

443 32.647777000 158.37.246.126 64.233.161.147 IPv4 1514 Fragmented IP protocol (proto=ICMP 1, off=1480, ID=8f60) [Reassembled in #444]

Fragment offset: 1480

Vi ser at Fragment offset = 1480. Dette betyr at det ikke er det første fragmentet.

Flags: 0x01 (More Fragments)

Reserved bit/Don’t fragment/More fragments er mulighetene under Flags. Vår tilbakemelding betyr at det er flere fragmenter.

**8) Hvor mange fragment ble opprettet i fra det originale datagrammet på 3500 bytes?**

Etter å ha byttet til 3500 bytes blir det opprettet 3 fragmenter. Disse 3 fragmentene oppstår når vi pinger. Den består av 2 IP pakker og 1 ICMP pakke:

442 32.647768000 158.37.246.126 64.233.161.147 IPv4 1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=8f60) [Reassembled in #444]

443 32.647777000 158.37.246.126 64.233.161.147 IPv4 1514 Fragmented IP protocol (proto=ICMP 1, off=1480, ID=8f60) [Reassembled in #444]

444 32.647781000 158.37.246.126 64.233.161.147 ICMP 582 Echo (ping) request id=0x0f08, seq=1/256, ttl=64