# **DKO-ØVING uke 3: DNS, EPOST, WEB**

# **DNS**

Undersøke DNS-protokollen med Wireshark og Ledetekst (CMD>)

1. **DNS i lokal cache**

Undersøk hva man kan utføre fra Ledetekst-vindu, skriv **CMD> ipconfig /?** Test følgende:

* CMD>ipconfig /displaydns (som viser alle dns-oppslag du har gjort i det siste.

1. **Generer oppslag i DNS**

Oppslag skjer dersom du har et program som trenger en IP-adresse og svaret ikke ligger i egen dns-cache, men man kan også bruke kommando i ledetekst-vind **CMD>nslookup (domenenavn eller ip-adresse)**

Start wireshark og bruk ***dns*** som displayfilter (grønn bakgrunnsfarge betyr lovlig kommando)

1. Gjør DNS-oppslag cmd**>nslookup ntnu.no**

Dersom du finner dette oppslaget med en gang, dvs at du sannsynligvis sitter hjemme, skal query og answer undersøkes nå.

Dersom det er uoversiktlig mange DNS-oppslag kan du filtrere på enkeltfelt i protokoller, i dette tilfellet skriver du displayfilter ***dns.qry.name==«ntnu.no»*** og trykker enter. Man får et inntrykk av alle de oppslagene som kjører hele tiden…

1. Hvilken transportlagsprotokoll og portnummer benyttes?

UDP, srcport=64047, dstport=53

1. Hvilke to *type*-felt spørres det etter?

A og AAAA

1. Undersøk innholdet av *Data length-feltet* i svarene og sjekk om det stemmer med antall byte i de to adressene

De stemmer (A er 32 bits lang, AAAA er 128 bits lang)

1. Hva er Time-to-live for disse svarene?

28s (A) og 1min20s (AAAA)

1. **Gjør DNS-reversoppslagCMD>nslookup (IP-adresse til ntnu.no)**

Reversoppslag kan gjøres for å kontrollere hvilket domene denne IP-adressen knyttet til.

1. Hvilket domenenavn har webtjeneren for ntnu.no?

lvs160vip02.it.ntnu.no

Du vil finne et annet domenenavn enn *ntnu.no*. Forklaring: det er flere typer tjenester, dvs TYPE-records, knyttet til det felles *ntnu.no* domenenavnet. Det kan være epost, navnetjener, annet. Når man gjør et oppslag på ntnu.no vil man i første omgang få til svar at dette er et CNAME (alias), og man må spørre videre på det nye (spesifikke) domenenavnet som oppgis. Din lokale navnetjener jobber rekursivt og spør også om dette, og det nye svaret er IP-adressen som sendes til deg.

Konklusjonen på dette er at du fra din PC sender et query og får et answer på IP-adressen til ntnu.no type A, men «ute i dns» har det gått meldinger frem og tilbake mellom lokal navnetjener og andre tjenere, lokalnavnetjener jobber rekursivt (og lagrer svarene i sin lokale cache)

# **EPOST**

1. **MIME**

MIME er «Internet Mail Extension» og kom opprinnelig som en tilpansing av SMTP til utvidet tegnsett fordi SMTP opprinnelig (og fortsatt) bruker US-ASCII tegnsett med 7 bit. Men MIME har også mange andre funksjoner.

Hva er en falsk epost? Epost er epost, man noen av dem er uønsket. Uten at man bruker autentisering er det strevsomt å skille ut de uønskede eposter. Man må legge inn filter på innhold og adresser.

Hver epost har sin egen «historie» i MIME-headerfelter. Hvor kommer eposten fra, hvilke tjenere har den vært innom, hvordan er innholdet i meldingen kodet osv. Disse feltene kan man undersøke. undersøke.

Åpne en epost (dobbelklikk) og velg fil/egenskaper som vist på bildet, deretter undersøk Internett meldingshoder

Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse Et bilde som inneholder tekst, skjermbilde, Font, nummer

Automatisk generert beskrivelse

Kopier innholdet av meldingshodene og lim det inn i et word-dokument for bedre lesbarhet. Det kan fort bli flere A4-sider! Resultatet er i grunnen en overflod av informasjon som er utenfor skopet til dette emnet, men legg merke til to forhold:

* I starten er det flere felt med «received from», eposten har vært i transit gjennom flere ledd.
* Det dukker så opp noen felt som sier *X-Virus-Scanned.. og X-Spam-Score*: … Dette er informasjon om virus-søk og grad av uønsket innhold. Det er for øvrig en praksis at felter som er under utvikling bruker X- først i navnet. Mottaker kan hoppe over dette feltet dersom funksjonen ikke er implementert
* Det kommer også noen SMTP-felter med adresser fra, til og emne (subject)

# **Webtjenesten og HTTP**

Undersøke HTTP-protokollen med Wireshark

**HTTP/1.1 - studere innholdet av HTTP-meldinger**

Start Wireshark. Gjør et oppslag på en HTTP-webside, for eksempel *datakom.no*. Vi ønsker å studere headerlinjer og nyttelast i klartekst. Vi skal ikke undersøke en HTTPS-side fordi den har kryptert innhold. HTTPS vil bli undersøkt i neste øving i forbindelse med TLS.

* Tips: Sett Wireshark display-filter til *http*
* Tips: For de som ikke får opp HTTP har jeg lagt ut en pakkefangst i BB

Marker en HTTP-pakke med en GET-request og høyreklikk for å få opp muligheter. Velg Follow TCP-stream. Nå vil dialogen mellom klient og tjener vises i klartekst.

1. Studer dialogen mellom klient og tjener for å hva det spørres etter og hva som sendes til svar, for eksempel bilder eller HTML-tekst
2. Beskriv bruk av følgende headerlinjer med dialog mellom klient og tjener

* Vedvarende forbindelser
* Klient: Connection: keep-alive
* Tjener: Keep-Alive: timeout=5, max=100
* Lokalt mellomlager 🡪 (Forutsetter at man har gjort gjentatte oppslag på samme webside)
* Klient: If-Modified-since: Wed, 23 Sep 2020 13:02:56 GMT
* Tjener: (koder for enten OK eller not-modified) OK har koden 200, Not Modified har koden 304

🡪 Cookies

* Undersøk om det brukes *Set cookie/cookie* av tjener/klient. Innholdet forteller oss ikke så mye

Ikke datakom.no, men kanskje andre steder

**HTTP/3**

HTTP/3 bruker QUIC for overføring. Dette er den mest optimale webprotokollen, men det er vanskelig å bruke Wireshark til å illustrere hvordan HTTP virker fordi innholdet er kryptert.

* Hvor mye webtrafikk går med HTTP/3? Sett display-filter i Wireshark til *quic* og surfe litt rundt på nettet for å få et inntrykk.

Ganske mye

* Hvilket portnummer benyttes på tjenersiden?

443

Lag deg et sammendrag av følgende beskrivelse av hvordan HTTP/3 fungerer (nyttig lesning):   
[Lenke](https://nordvpn.com/no/blog/what-is-http3/)