Kunnskap for en bedre verden

Anvendelser av Internett -- epost og DNS

Helge Hafting/Bjørn Klefstad, Institutt for datateknologi og informatikk (IDI), NTNU
Lærestoffet er utviklet for faget <<INFT1007 Datakommunikasjon>>

Resymé: I denne leksjonen skal vi se nærmere på tjenestene epost og DNS. Vi skal se på hvilke protokoller disse tjenestene benytter seg av, og meldingene som overføres. Vi ser også på en del epost-relaterte sikkerhetstiltak. I tillegg skal vi se på de felles egenskapene til applikasjonslagets protokoller.

Innhold

2 Anvendelser av Internett -- epost og DNS

2.1 E-post og DNS kap 3.3

E-post er en tjenestene som nærmest har eksplodert de siste tiårene. En av hovedgrunnene til dette er at e-post vanligvis overføres i løpet av 10–15 sekunder. Sammenliknet med ordinær postgang som tar en til tre dager, så er dette lynraskt. Det er ikke uten grunn at ordinær post nå kalles for snailmail. Forskjellen på vanlig post og epost er at informasjonen på epost må finnes i elektronisk form.

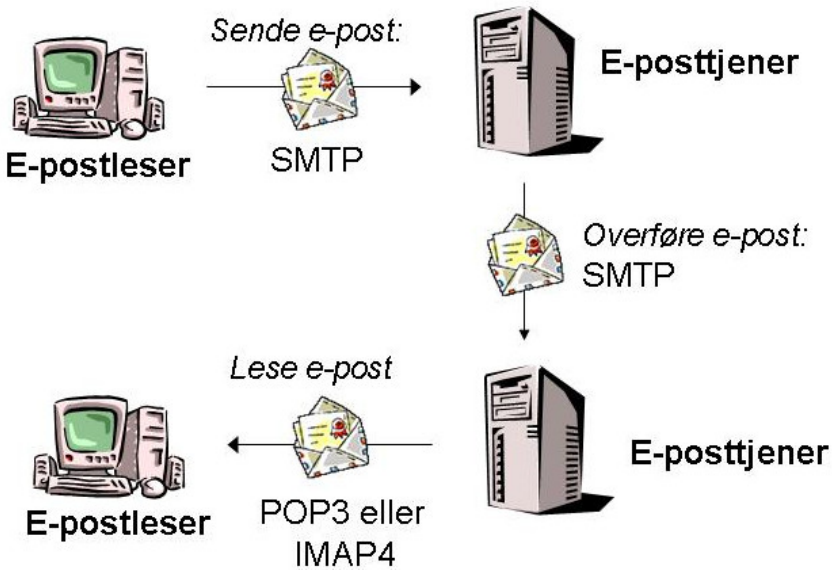
I tillegg til dette kan vi nå også overføre det meste av vedlegg så lenge de også er lagret elektronisk. Dette betyr at både dokumenter, presentasjoner, lydfiler, bilder og videoklipp kan overføres mellom to personer i løpet av svært kort tid. Spesielt for felles dokumenter som skal utarbeides så er dette hensiktsmessig. Nye avsnitt eller kommentarer kan raskt legges inn og ny versjon utveksles. Epost har blitt en meget viktig kommunikasjonskanal.

En tredje faktor som har påvirket utviklingen er tilgangen til denne tjenesten. Praktisk talt det som finnes av mobile enheter innholder funksjonalitet som kan gi tilgang til epost. Svært mange personer har en mobil eller PDA i lomma som kan benyttes til dette formålet.

Vi skal nå se nærmere på den tekniske oppbygningen av epost tjenesten. I forbindelse med overføring av epost er det flere elementer som er involvert. Figur 2.1 på neste side og figuren i boka illustrerer hva vi trenger og hvordan de ulike elementene fungerer sammen. Vi oppsummerer bare kort her de viktigste elementene. Detaljene finner du i boka.

For å skrive og sende e-post trenger vi en epost klient som er koblet opp mot en eposttjener. Denne tjeneren er vår lokale eposttjener. Vanligvis benytter vi oss av samme tjener for å sende og motta epost, men det er ikke noe i veien for å benytte en tjener for å sende epost og en annen for å motta epost.

[1] Tysk: *teergrube*, tjæregrop



Figur 2.1: Epost

En eposttjener er alltid påslått og koblet sammen i et nettverk med andre eposttjenere. Før eposten sendes fra vår lokale tjener til mottakers tjener, sjekker den om linjene er åpne og at mottakers tjener er klar til å motta epost.

Protokoller som er involvert

SMTP	Simple Mail Transfer Protocol, brukes for overføring av e-post.
POP3	Post Office Protocol, brukes til nedlasting av e-post fra tjener til mottaker.
IMAP4	Internet Mail Access Protocol, brukes til administrasjon og nedlasting av e-post fra tjener til mottaker.

Her er det viktig å merke seg forskjellen på POP3 og IMAP4. POP3 laster i utgangspunktet ned eposten til din lokale maskin. Dette betyr at nedlastet epost kun er tilgjengelig på akkurat denne maskinen. For dagens versjoner av POP3 er dette endret slik at også denne protokollen tillater at posten kan lagres på epost tjeneren slik som for IMAP4.

Dersom du benytter IMAP4 organiseres eposten på eposttjeneren slik at den kan lese fra hvilken som helst maskin som er riktig konfigurert. I dag ønsker vi gjerne å være uavhengig av hvor vi er og hvilken maskin vi benytter for å ha tilgang til epost. Derfor er IMAP4 svært utbredt. Av samme årsak er det også mange som etter hvert benytter et webgrensesnitt opp imot sin epost konto.

En må benytte flere ulike protokoller for epost fordi det er upraktisk å overføre epost direkte fra senderen til mottakeren. Dette henger sammen med at:

- Mottakermaskinen er ikke alltid koplet opp mot nettet.
- Mottakermaskinen står ikke alltid påslått.
- Sendermaskinen kan bruke lang tid dersom det er stor trafikk i nettet.

På strekningen fra e-post-tjeneren til e-postklienten hos mottaker er det flere grunner til at SMTP ikke egner seg og vi bruker POP3 eller IMAP4 isteden:

- Klientmaskinen er ikke alltid tilgjengelig.
- Brukeren bør kunne velge hvilken tid epost skal lastes ned.
- Brukeren må autentiseres (innloggingsprosedyre) for å få tilgang til postkassen.

2.2 DNS kap 3.4

DNS-tjenesten er en tjeneste for oversetting mellom domenenavn og IP-adresser (mer om IP-adresser i senere leksjoner). Denne tjenesten benyttes som regel av andre programmer og heller sjelden direkte av vanlige brukere. De ulike programmenes klienter og tjenere, som til sammen utgjør en Internetttjeneste må kommunisere ved hjelp av IP-adresser. For brukerne er det upraktisk å måtte huske IP-adresser for å kunne benytte de applikasjonstjenestene som Internett tilbyr. Vi ønsker å benytte domenenavn i stedet, for de er mye lettere å huske.

Eksempel: `www.datakom.no` eller `158.38.50.20` ?

Vi trenger derfor et verktøy som kan gjøre det mulig for brukeren å benytte domenenavn, som kan oversettes til IP-adresser for å opprette kontakt med web-tjeneren.

Hovedoppgavene for DNS er:

- Å tilby et navnesystem for domenenavn.
- Å tilby funksjoner for oversetting mellom domenenavn og IP-adresser (begge veier).

DNS er realisert ved hjelp av en *distribuert database* over domenenavn og IP-adresser. Normalt har et lokalnett sin egen DNS-tjener med informasjon som gjelder egne navn og adresser. En slik tjener utgjør på denne måten en del av den distribuerte databasen. Den har anledning til å spørre andre tjenere dersom den selv ikke kan løse et oppslag den blir forespurt om.

Merk fra boka at en tjener, som har én IP-adresse kan ha flere DNS-navn, og hva som er hensikten med dette (webhotell). Omvendt kan flere tjenere, som har flere IP-adresser ha et felles DNS-navn. Merk også hensikten med dette (lastdeling).

2.2.1 Navnerommet i DNS

Navnerommet i DNS er tredelt:

- Generiske domener – To av dem er reserverte for USA: mil og gov.
- Geografiske domener – Dette er landskoder.
- Inverse domener – Dette er et spesielt domene for å håndtere inversoppslag.

2.2.2 Generiske domener

De generiske domenene kan benyttes av organisasjoner, slik:

com	(Commercial) Kommersielle organisasjoner
edu	(Educational) Høyere utdanningsinstitusjoner
gov	(Government) Regjeringen og sambandsadministrasjonen i USA
int	(International) Internasjonale organisasjoner
mil	(Military) USAs væpnede styrker
net	(Network) Nett-organisasjoner
org	(Organisation) Organisasjoner

Se også `http://www.icann.org/en/registrars/accredited-list.html` for flere muligheter.

2.2.3 Geografiske domener

De geografiske domenene benytter en to-bokstaverskode for hvert land, for eksempel:

no	Norge
se	Sverige
dk	Danmark
de	Tyskland
ee	Estland
uk	Storbritannia
hr	Kroatia
es	Spania
us	USA (Lite benyttet; mest av skoler og offentlige institusjoner)

2.2.4 Inverse domener for IP og IPv6

Til nå finnes to inverse domene. Dette er spesialdomener som er laget for at det skal være mulig med raske inversoppslag. Alle maskiner skal være registrert i dette domenet med IP-adressen. Innholdet i databasen er blant annet domenenavn som er knyttet til den IP-adressen som det blir slått opp på.

Det inverse domenet for Internettet heter «.in-addr.arpa». Her står «in-addr» for «internet addresses», og «arpa» er «Address And Routing Parameter Area»

For å kunne delegere soner effektivt, skrives ip-adressen baklengs i inversdomenet. Så informasjonen for «158.38.48.135» er lagret på «135.48.38.158.in-addr.arpa». Verktøy som «nslookup» og «dig» ordner dette for oss, så vi slipper å skrive adresser på den måten selv. Men vi ser denne skrivemåten når programmene gir oss resultatet av oppslaget.

Det andre inverse domenet er «.ip6.arpa». Her kan vi finne ut hvilke navn som hører til ipv6-adresser.

2.2.5 Andre funksjoner i DNS

DNS dekker ikke bare hovedhensikten som er kobling mellom IP-adresser og domenenavn. Vi finner også:

- Lastdeling på webtjenere. Henvendelser til ett domene, (som www.vg.no) fordeles på flere tjenere, ved at brukerne får oppgitt ulike ip-adresser for domenet. Dermed kan flere tjenermaskiner samarbeide om å formidle den samme nettsiden.
- Dynamisk DNS. Brukes dersom tjeneren endrer IP-adresse. Det skjer hvis man f.eks. har en tjenermaskin på et hjemmeabonnement.
- Kobling mellom epostadresser og domenenavn. Vi kan f.eks. sende epost til brukernavn@ford.com. Men Ford sine eposttjener heter ikke ford.com, da det er en webserver. Dessuten har de flere tjenere, noe som sikrer 100% oppetid. Disse har ulik prioritet. Navnene (og prioritet) er:

```
ford.com mail exchanger = 10 cluster4.us.messagelabs.com.
ford.com mail exchanger = 20 cluster4a.us.messagelabs.com.
```

Post til Ford vil bli sendt til cluster4, fordi den har lavest prioritet. Hvis den er utilgjengelig, forsøkes cluster4a.

2.3 Oppslag i DNS

2.3.1 Navn til ip-adresse, A-record

Oppslag i DNS skjer for det meste automatisk. Dette skjer f.eks. hver gang vi klikker på en lenke i nettleseren. I lenken står det et navn, f.eks. www.ntnu.no, men nettleseren trenger IP-adressen til tjeneren for å kunne hente nettsiden. Dermed gjør nettleseren et dns-oppslag for å finne adressen.

Vi kan også gjøre eksplisitte oppslag i DNS. For å finne ip-adressen til www.ntnu.no bruker du denne kommandoen:

```
nslookup www.ntnu.no
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
www.ntnu.no canonical name = lvs160vip02.it.ntnu.no.
Name: lvs160vip02.it.ntnu.no
Address: 129.241.160.102
Name: lvs160vip02.it.ntnu.no
Address: 2001:700:300:6::102
```

Her fikk vi vite at den aktuelle ip-adressen var 129.241.160.102, eventuelt ipv6 2001:700:300:6::102. Vi så også at maskinens egentlige navn er lvs160vip02.it.ntnu.no. I tillegg fikk vi se hvilken navnetjener som ble spurt. (129.241.0.200) Hvis du prøver utenfor høgskolens nettverk, får du samme svar på `www.ntnu.no`, men en annen navnetjener.

Fler eksempler

```
nslookup vg.no
Non-authoritative answer:
Name: vg.no
Address: 195.88.55.16
Name: vg.no
Address: 195.88.54.16
Name: vg.no
Address: 2001:67c:21e0::16
```

Vi ser at vg.no har to vanlige ip-adresser, og en ipv6-adresse. De to ip-adressene er sannsynligvis ulike maskiner plassert på ulike steder. Dermed vil tjenesten kunne virke selv om én av dem er nede.

```
nslookup datakom.no
Non-authoritative answer:
Name: datakom.no
Address: 20.100.42.130
```

Her ser vi at datakom.no har ip-adresse 20.100.42.130.

Eksplisitt oppslag på A-record

Alle dns-oppslag vi har sett på til nå, har vært oppslag av IP-adresser. Vi sier at vi slår opp en «A-record», der A står for adresse. «nslookup» slår opp en A-record hvis vi ikke ber om noe annet. Det går an å be spesielt om en A-record også:

```
nslookup -type=A www.db.no
Non-authoritative answer:
www.db.no canonical name = db.no.
Name: db.no
Address: 87.238.33.182
Name: db.no
Address: 87.238.33.183
Name: db.no
Address: 87.238.33.180
Name: db.no
Address: 87.238.33.181
```

Vi ser at Dagbladet har flere ip-adresser, en grei måte å spre belastningen på sider som veldig mange er interessert i. Vi ser også at «www.db.no» egentlig heter «db.no». Mer om «Canonical name» senere.

2.3.2 Navn til IPv6-adresse, AAAA-record

IPv6 er på fremmasj, og vil med tiden ta over for vanlig internett. IPv6-adresser er lengre og skrives på en annen måte, ellers fungerer de på samme måte som vanlige ip-adresser. Noen organisasjoner er frempå og har allerede tatt i bruk IPv6. Avisen VG er en slik organisasjon, og vi kan finne deres ipv6-adresse slik:

```
nslookup -type=AAAA vg.no
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
Name: www.vg.no
Address: 2001:67c:21e0::16
```

2.3.3 CNAME, alias som henviser til et annet navn

CNAME (Canonical name) er en måte å la ett navn henvise til et annet. Vi ser det i blant i oppslag:

```
nslookup ftp.uninett.no
Non-authoritative answer:
ftp.uninett.no canonical name = ftp2.uio.no.
Name: ftp2.uio.no
Address: 129.240.118.47
Name: ftp2.uio.no
Address: 2001:700:100:118::47
```

Her ser vi at ftp.uninett.no henviser videre til ftp2.uninett.no, som igjen har ip-adressen 129.240.118.47. Vi ser at også denne er tilgjengelig over ipv6.

Det er praktisk for de som står for drift av maskiner som har mange navn. De kunne brukt flere A-records, som eksempelet med datakom.no og bork.iie.ntnu.no. Nytten med CNAME kommer hvis man trenger å bytte ip-adresse. Uninett ville bare trengt å endre én A-record, nemlig den for ftp2.uninett.no. De ville ikke trengt å gjøre noe med ftp.uninett.no. CNAME kan altså spare driftsarbeid, når man har mange navn/domener på samme maskin. Webhoteller kan ha veldig mange domener på én maskin.

Hvis vi slår opp «uninett.no» og «www.uninett.no» ser vi at disse to navnene også refererer til samme maskin, via CNAME. Navn med og uten «www» ordnes lett med CNAME, og fordelen er at ip-adressen bare trenger registreres én gang.

2.3.4 Reversoppslag i DNS

Reversoppslag bruker vi når vi har en ip-adresse, og vil vite hvilket navn som hører til. Slik kan vi finne ut hvilken organisasjon en ip-adresse hører til. Eksempel:

```
nslookup 195.88.54.16
Server: 158.38.50.10
Address: 158.38.50.10#53
Non-authoritative answer:
16.54.88.195.in-addr.arpa name = www.vg.no.
```

Vi ser at dette er en av tjenermaskinene til avisen VG. «nslookup» vil gjøre et reversoppslag hvis vi skriver en ip-adresse i stedet for et navn. Reversoppslag henter en post av typen «PTR». Vi kan også be om slike eksplisitt:

```
nslookup -type=ptr 129.241.0.10
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
10.0.241.129.in-addr.arpa name = ntnu-gw.nettel.ntnu.no.
```

Bak denne adressen fant vi NTNU sin gateway. Legg også merke til hvordan nslookup viser oss inversdomenet, med baklengs ip-adresse og «.in-addr.arpa»

Reversoppslag fungerer også med ipv6-adresser, men svarene kan bli veldig lange:

```
nslookup -type=ptr 2001:67c:21e0::16
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
6.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.e.1.2.c.7.6.0.1.0.0.2.ip6.arpa
name = www.vg.no.
```

2.3.5 Slå opp eposttjenere (mx-record)

Det er stort sett epost-programvare som slår opp eposttjenere. Da slår vi opp en såkalt mx-record. (MX=Mail eXchanger, har forøvrig ikke noe med Microsoft Exchange å gjøre.) Men vi kan også gjøre dette selv. For å finne hvilke maskiner som håndterer epost sendt til bruker@ford.com:

```
nslookup -type=mx ford.com
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
ford.com mail exchanger = 10 cluster4.us.messagelabs.com.
ford.com mail exchanger = 20 cluster4a.us.messagelabs.com.
```

Her ser vi de to maskinene som håndterer epost for Ford. Dessuten ser vi at de har ulik prioritet. Når tjenerne har ulik prioritet, prøver epost-programmene alltid den med lavest tallverdi først. Et slikt opplegg bruker vi når hovedtjenerne normalt tar hånd om posten, mens reservemaskiner bare er ment å trå til når hovedtjenerne er ute av drift. Flere maskiner med lik prioritet kan brukes for lastbalansering. Da vil epostprogrammene velge en av dem tilfeldig.

En maskin som sender epost til Ford vil altså gjøre et slikt oppslag. Deretter vil den velge «cluster4.us.messagelabs.com» fordi den har lavest prioritet.

Legg merke til at vi ikke fikk noen ip-adresser her. Vi er ikke alltid så uheldige, men hvis vi ikke får slik «ekstrainformasjon», blir programmet nødt til å slå opp A-records etterpå. Kommunikasjon skjer alltid ved bruk av ip-adresser, navn alene er aldri nok.

2.4 Epost, sikkerhetsproblemer og mottiltak

2.4.1 Problemene

Spam — Uønsket post som sendes til mange. Inneholder typisk annonser for tvilsomme virksomheter, eller eventuelt virus.

Virus — Virus og ormer er programmer som gjør skade. Epost er en av flere måter de spres. Spam og virus kan føre til så mye post at nettverk overbelastes og disker fylles opp. Dessuten er det irriterende å måtte slette slike meldinger hver eneste dag.

Phishing — Epost som prøver å lure folk til å oppgi informasjon, ofte kontonumre, passord, bankID og lignende. Slik post gir seg ut fra å være noen som er til å stole på, som for eksempel en bank. Ofte kombineres dette med lenker til falske nettsider.

Andre varianter av phising prøver å få folk til å sende penger. Det kan se ut som en slektning/venn har blitt ranet på ferie og trenger hjelp, eller en hastebeskjed fra sjefen om å betale en faktura for firmaet.

web bugs Epost i html-format. Kan inneholde lenker, for eksempel til bilder som ikke sendes med meldingen. I stedet hentes bildet når du ser på meldingen.

Dette kan misbrukes: Når du leser posten, får avsender vite om det fordi bildet blir lastet ned fra nettstedet hans. Dermed vet avsender at epost-adressen din er i bruk, at du har kikket på meldingen, og fra hvilken IP-adresse du gjorde det.

Selv om det er masseutsendt post, har hver melding en ulik bildelenke. Slik vet de *hvem* som lastet ned bildet. Noen ganger er dette opplagt, man ser at lenken inneholder epost-adressen.

Spammere bruker web bugs for å finne ut hvilke av epostastadressene på lista som faktisk er i bruk, noe de finner ut når klienten din laster ned bildet. Slike aktive epostadresser selger de videre til andre spammere. Dette er en god grunn for å stille inn epostklienten så den *ikke* laster ned bilder automatisk!

Avlytting Epost sendes i klartekst over nettet, konfidensiell informasjon kan fanges opp med en pakkesniffer. For å hente post til brukermaskiner hender det at passord sendes i klartekst også. Dermed kan utro tjenere eller kompromitterte maskiner sniffe opp dette. Hvis passordet er det samme som på andre tjenere, kan det bli et større problem enn epost på avveie.

Angrep Angripere vil gjerne ha kontroll over posttjenere. En slik maskin er perfekt for alle måter epost kan misbrukes. I blant kan de oppnå dette ved å trikse med protokollen (smtp) for å utnytte feil i tjenerprogramvaren.

I 1988 hadde vi den første epost-ormen, den utnyttet et problem av typen buffer-overflow i tjenerprogrammet *sendmail*. De første epostvirusene kom i 1999. Se `http://en.wikipedia.org/wiki/Timeline_of_notable_computer_viruses_and_worms` for informasjon om virus gjennom tidene.

2.4.2 Filtre på epost-tjeneren

Filtrering bør gjøres både på inn- og utgående post. Inngående for å beskytte sine klienter, utgående for å hindre at virus som kommer inn på andre vis får spredd seg ut igjen. Et enkelt filter som kutter ut eksekverbare vedlegg stopper de fleste virus. Mer avanserte filtre kan se etter kjente virus og spam-signaturer. De kan også finne virus gjemt i .zip-arkiver og .doc-dokumenter. Et problem med slike filtre er at de må oppdateres jevnlig for å ha noen effekt. Det kommer hele tiden nye virus.

Noen spamfiltre kan lære, ved at man legger spam som kommer gjennom i en egen mappe. Deretter vil filteret kjenne igjen spam-fraser neste gang de dukker opp.

2.4.3 Svartelister

Et mye brukt tiltak er svartelister. Listene inneholder navn og ip-adresser for maskiner som har sendt ut spam tidligere, og brukes til å nekte å ta imot post fra slike maskiner. Resultatet er at spammere, og ISP'er som tillater spam, ikke får sendt post til de som bruker slike svartelister. Det er for tiden så mange som bruker slike lister, at det er et reelt problem for de som havner på listene.

Det kan være både tid- og arbeidskrevende å komme *av* en slik liste igjen. Først må en oppdage at en står på en slik liste. Det er flere av dem, og det er ikke sikkert de gir deg beskjed. I det minste må en skrive et brev[1] til de som oppdaterer den aktuelle svartelista, og få med en eller annen form for bevis på at tjenermaskinen ikke lenger kan misbrukes til spam. Dette er den greieste biten. Deretter må en vente på at alle som bruker den aktuelle svartelista oppdaterer sin kopi. Det kan fort ta uker og måneder, og i mellomtiden er du utilgjengelig på epost. Konsekvensene av et sikkerhetsbrudd kan være temmelig kjedelige, spesielt hvis man har en sjef som ikke skjønner hvorfor viktig epost ikke kommer frem «nå som du har fikset feilen i epostsystemet vårt».

2.4.4 Velkonfigurert eposttjeneste

En må konfigurere egne epost-tjenere slik at de ikke kan misbrukes. En tjeneste som er åpen[2] slik at utenforstående kan bruke den til å sende post videre hvor som helst inviterer til spam. Et problem her er at noen tjenerprogrammer leveres «åpne», det er deretter opp til administratoren å sikre den. Hvis det ikke blir gjort virker tjenesten fint likevel, både for normal lovlig bruk — og for spammere. Enkelte administratorer vet ikke dette, tjeneren blir misbrukt uten at en merker det selv, og deretter havner den på de ovenfor nevnte svartelistene med de problemer det medfører. Alternativet er en tjeneste som installeres i stengt form. Problemet med det er at tjenesten ikke virker med en gang den er installert, administratoren blir nødt til å åpne tjenesten for egne brukere. Programvare basert på åpen kildekode er gjerne slik, det kan være verre med kommersielle programmer fordi all ekstra konfigurering sees på som «bry». Det er en konkurransefordel at det er «lett» å komme i gang. Problemer hemmer salget, eller fører til supporttelefoner. Spesielt gjelder dette salg til «dummies» som ikke helt vet hva de driver med. Det er imidlertid lenge siden det ble slutt på den tiden da «dummies» kunne gjøre en brukbar

[1] Papirbrev, ikke epost – siden de har blokkert eposten din…
[2] Eng: *open relay*

jobb som driftsansvarlig for en epost-tjener på Internettet. Ihvertfall hvis vi snakker om virksomhetskritisk epost.

Vær oppmerksom på at å stenge for uautorisert videresending bare beskytter omverdenen mot spam, dine egne klienter beskyttes ikke. Dette fordi utenforstående må kunne kontakte tjeneren for å sende post inn til dine klienter. Hvis ikke, får du ikke post i det hele tatt. Men sikringen gjør spammerens jobb vanskeligere – før kunne de sende ut millioner av meldinger via én tjener. Nå må de kontakte epost-tjeneren for hvert eneste domene, og det er en god del mer bry for dem. Lettere å spore dem opp blir det også.

En velkonfigurert og ikke minst oppdatert tjenermaskin står seg sannsynligvis bedre mot angrep basert på «kjente feil» også.

2.4.5 Fornuftige epost-klienter

Strengt tatt burde det ikke være nødvendig å sjekke epost med tanke på virus i det hele tatt. Dette fordi det er helt unødvendig å ha epost-klienter som er sårbare for virus. Hele problemet er kunstig, og kunne vært løst med bedre design av klientene. Dessverre er windows sårbart likevel, og en del brukere er dessuten dumme nok til å prøvekjøre tilfeldige programmer de får i posten.

Outlook er en spesielt sårbar epostklient, et enkelt tiltak som å bytte ut outlook øker sikkerheten mot epostvirus en god del. Det fins mange andre gode epostklienter, mange av dem er gratis også. Å bruke klientmaskiner basert på noe annet enn windows (f.eks. mac eller linux) hjelper enda mer, da ingen av angrepsformene som er laget for å ramme windows virker på disse plattformene.

Problemet med «web bugs» unngås ved å sette epostklienten opp til å *ikke* laste inn bilder fra eksterne kilder. Bilder som er lagt ved på normalt vis vil fortsatt synes, men ikke bilder som bare er lenket inn.

2.4.6 Sjekke avsenders DNS-informasjon

I utgangspunktet tar en smtp-tjener imot epost fra hvem som helst. Dette for at alle skal kunne sende oss (ønsket) epost.

Men det har oppstått en forskjell på spammere og andre folk. De fleste sender epost ut via posttjenesten hos ISP, eller evt. posttjenesten hos firmaet. Disse tjenerne er registrert i DNS med egen MX-record for å gjøre det lett å sende post *til* dem.

Spammere blir gjerne hindret i å bruke veldrevne tjenester. Så de setter opp sin egen midlertidige tjener. Da har de som regel hverken tid eller budsjett til å skaffe en MX-record, som regel har de ikke A-record en gang.

Dette gir eposttjeneren vår noen muligheter når omverdenen tar kontakt:

1. Slå opp navnet til IP-adressen med et reversoppslag i DNS. Sjekk videre at den har en A-record. Hvis ikke, steng forbindelsen straks.
2. SMTP begynner. Den første kommandoen er alltid «HELO/EHLO avsender-hostname». Slå opp A-record (ip-adresse) for «avsender-hostname».
   a) Hvis A-record ikke fins, avvis meldingen
   b) Hvis ingen av ip-adressen(e) matcher adressen til forbindelsen, avvis meldingen
3. Hvert brev som overføres via smtp, inneholder kommandoen «MAIL FROM bruker@epostdomene». Slå opp MX-record for «epostdomene». Hvis den ikke fins, avvis meldingen.

Med slike tiltak vil spammeren være nødt til å eie A- og MX-records i DNS, og de må peke til den offentlige ip-adressen som brukes. Det krever et visst budsjett, som går tapt når alt sammen inndras pga. misbruk. Eventuelt kan spammeren hacke noen som har alt dette, men det er ikke så lett.

2.4.7 Sender Policy Framework – SPF

SPF er et system for å validere hvilke maskiner som har lov til å sende post for et epostdomene. Det kan fort blir mange forskjellige, fordi man kan ha ha flere tjenere i reserve, og muligheten for outsourcing av epost.

Man publiserer en TXT-record for sitt epostdomene i DNS. Denne har et spesielt SPF-format, som forteller hvem som har lov til å sende post fra det aktuelle domenet. Det kan være:

- Maskiner nevnt i MX-records for domenet eller andre domener
- Maskiner med gitte navn
- Maskiner med oppgitte IPv4/IPv6 adresser
- «alle» eller «ingen»
- include/redirect som henviser til andre TXT-records. Disse må slåes opp videre.

Når tjeneren vår mottar post, kan den slå opp SPF-informasjon for avsenderdomenet. Hvis maskinen som sender ikke er lovlig sender for domenet, er det bare å avvise meldingen.

Se: `http://www.openspf.org/`

RFC4408: `https://www.ietf.org/rfc/rfc4408.txt`

Wikipedia: `http://en.wikipedia.org/wiki/Sender_Policy_Framework`

Eksempel på SPF

Avisen VG bruker spf. Vi kan sjekke hvilke tjenere som har lov til å sende post for VG med kommandoen nslookup. Kommandoen skal være tilgjengelig enten du bruker mac, linux eller windows. Den brukes slik:

```
nslookup -type=txt vg.no
;; Truncated, retrying in TCP mode.
Server: 129.241.0.200
Address: 129.241.0.200#53
Non-authoritative answer:
vg.no text = "MS=ms58603718"
vg.no text = "miro-verification=eff9412c1aa1655d0a279f7d6704841b6a02b591"
vg.no text = "fastly-domain-delegation-29tfXEmvnByNT8JY-442665-2021-10-26"
vg.no text = "google-site-verification=1pgyIHHF6q_XtdnSiBJFZiJrXtcg-vpNdfWzhJtb8pM"
vg.no text = "google-site-verification=CvJrFeKFUvWtMd11AkMfLIwBCKnMr2e4f6dzowPw32A"
vg.no text = "google-site-verification=uRwOU1tXpvCsRXnHyApK_yaMeXiTmni8cIW2KelXWO8"
vg.no
text = "v=spf1 include:_u.vg.no._spf.smart.ondmarc.com include:amazonses.com ~all"
vg.no text = "YbAM-
QjB2yzXf6DTU9dR9LYMQNGptykV9gN251w0knS5h2Iu4Nhk9kw6s1nzmgOCFnvxo/lekGs1PSCy3Z3oXAA=="
vg.no text = "wiz-domain-
verification=52455ec02f13dfe89b5827fd5085e7ab8e918767f8461a2f5b443b5e0dd6cd56"
```

Det er den lange linja med «v=spf1» som er interessant her, de andre text-linjene har andre formål enn spf. Vi ser ikke egentlig hvilke tjenere som kan sende post for VG her. I stedet ser vi to include-setninger, som henviser oss til _u.vg.no._spf.smart.ondmarc.com og amazonses.com.

Til slutt står det `~all`, som betyr «ikke alle de andre i verden». Skal vi vite mer, må vi slå opp include-stedene også:

```
nslookup -type=txt _u.vg.no._spf.smart.ondmarc.com
Non-authoritative answer:
_u.vg.no._spf.smart.ondmarc.com
text = "v=spf1 ip4:52.100.0.0/14 ip4:40.92.0.0/15 ip4:35.191.0.0/16 ip4:40.107.0.0/16
ip4:74.125.0.0/16 ip4:173.194.0.0/16 ip4:54.240.0.0/17 ip4:104.47.0.0/17
ip4:209.85.128.0/17 ip4:72.14.192.0/18 ip4:198.2.128.0/18 ip4:216.198.0.0/18
ip4:23.251.224.0/19 ip4:76.223.128.0/19 ip4:108.177.96.0/19 ip4:172.217.0.0/19
ip4:172.217.128.0/19 include:_p.1.11g95dg._u.vg.no._spf.smart.ondmarc.com ~all"
```

Her ser vi mange ip-ranges. Alle maskiner i disse seriene kan altså sende post for VG. Om vi vil ha en komplett liste, slik en eposttjener trenger, er vi langt fra ferdige. Vi fikk ennå en include som må slåes opp, og vi trenger også å slå opp amazonses.com. Det gjør jeg ikke her, men interesserte må gjerne prøve.

Det ser ut som VG har sikret seg godt med diverse alternative leverandører av eposttjenester. Likevel, dette er bare noen få av de mulige maskinene i verden. Når eposttjeneren vår får post fra noen med adresse @vg.no, kan den altså sjekke at meldingen kommer fra en av maskinene på lista. Hvis ikke, bør den ikke ta imot posten.

2.4.8 SMTP med autentisering

Autentisering brukes kun for utgående post. Brukerens epostklient må autentisere med f.eks. navn/passord for å få sende. For inngående post kan vi ikke gjøre dette, vi kan ikke opprette kontoer for alle som kan tenkes å ville sende oss post.

Dette er et alternativ til å validere ved hjelp av avsenders ip-adresse. Vi kan f.eks. la folk sende epost når de er hjemme eller på reise, uten å måtte gjøre tjeneren til open-relay. Det gir også en viss beskyttelse mot postvirus, de får ikke sendt bare fordi de har infisert en maskin med riktig ip-adresse. Viruset kjenner jo ikke passordet vårt.

Autentisering kombineres gjerne med TLS/SSL-basert kryptering, så ikke utenforstående skal få tak i passord med pakkesniffere. Vær oppmerksom på at meldingen bare er kryptert frem til den lokale eposttjeneren. Meldingen (uten autentiseringsinformasjon) sendes ukryptert videre over Internettet.

2.4.9 Grålister

Grålister[3] er en avansert teknikk som utnytter det faktum at virus og spammere ikke følger epoststandardene fullt ut. Standardene sier blant annet at en overbelastet tjener kan gi andre beskjed om å vente en stund og prøve igjen senere. Avsender vil vanligvis vente noen minutter, eller kanskje en time. For å bruke grålisting installerer en programvare som gjør følgende:

Hver gang det kommer et brev, sjekker programmet hvem posten er fra, hvem den er til, og hvilken tjenermaskin den kommer fra. Hvis kombinasjonen er ny, gir programmet avsender beskjed om å vente litt og prøve igjen senere. Etter noen tid settes fraperson-tilperson-tjener kombinasjonen på en midlertidig hvitliste slik at brevet slipper igjennom.

Effekten er at mye epost forsinkes noe, f.eks. med en times tid. Slik post lagres midlertidig på tjeneren som forsøker å sende, og den skal normalt ha ressurser nok til dette. Virus og spammere har som regel programvare eller utstyr som mangler evne eller kapasitet til å prøve omigjen senere, dermed kommer ikke uønsket post frem i noen særlig grad. Spammere er typisk interessert i å bli fort ferdig med masseutsendelsen sin, de blir nemlig sporet opp og risikerer å bli koblet fra nettet når som helst.

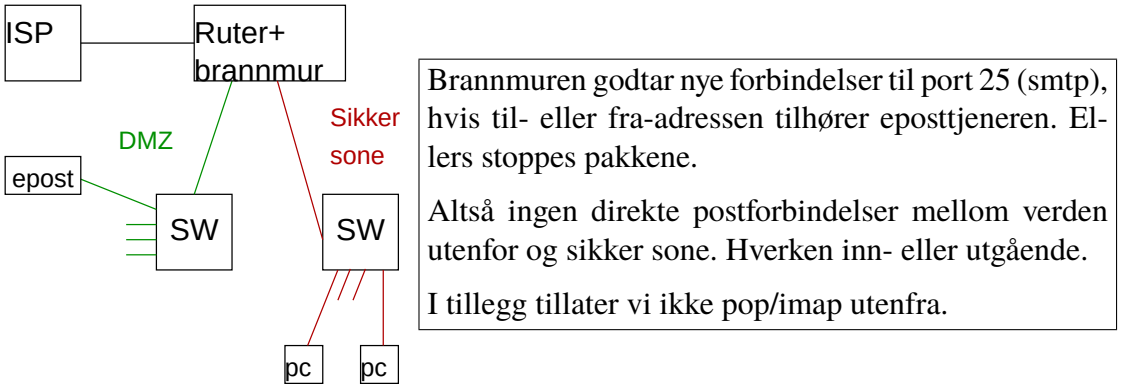
2.4.10 Brannmur

Spam kan begrenses ved konfigurere brannmurer til å bare godta SMTP-forbindelser til epost-tjeneren og ingen andre maskiner. Dermed unngår vi at andre maskiner misbrukes.

[3] Eng: *gray listing*

Noen klientmaskiner kan ha en viss SMTP-funksjonalitet, som imidlertid bare er ment å brukes lokalt.

Vi bør også stenge for utgående SMTP initiert av andre enn epost-tjeneren. Mange virus sprer seg ved at viruset inneholder en minimal epost-tjener, som sender ut nye virus til tilfeldige adresser funnet i adressebøker og på nettsider. Dette kveles ved at posten ikke kommer ut når den ikke går via tjeneren. (Post som går via tjeneren sjekkes av denne, med spam-/virusfilter.) Se figur 2.2



Figur 2.2: Brannmuroppsett for epost

2.4.11 Ikke kjøp!

Spam av markedsføringstypen brukes fordi den dessverre virker – på noen få. Derfor – om du får en epost med reklame som du ikke har bedt om – ikke kjøp, ikke engang om tilbudet er interessant. (Bruk butikker eller vanlige nettbutikker i stedet, finn en annen leverandør.)

Varer som markedsføres på tvilsomt vis, er som regel tvilsomme også. Det kan være produkter som ikke egentlig virker, falske/giftige medikamenter, aksjer som raser kort tid etter kjøp, …

2.4.12 Kryptering

Om du trenger å sende konfidensiell informasjon med epost, bruk kryptering. Et vedlegg med en godt kryptert fil kan ikke leses av andre. Husk – ikke skriv noe i epost som du ikke ville skrevet på et postkort uten konvolutt.

2.4.13 Opplæring

Det hjelper å lære opp brukerne. Få litt epostvett inn i firmahåndboka. Brukere bør være oppmerksom på ting som phishing, ikke klikke på hva som helst, og utvise sunn skepsis når gamle kjente plutselig skriver på engelsk eller «google-norsk».

En nyere utvikling er at chatGPT eller lignende kunstig intelligens brukes for å skrive lurebrev. Dermed får svindlerne et bedre språk enn de kan få med google translate. Hvis du er i tvil om et brev der noen ber om hjelp, snakk med noen som har mer erfaring – f.eks. politiet eller banken din.

2.4.14 Feller, tjæregroper[4]

Utsending av spam og virus kan forsinkes med feller. En slik felle er en maskin som tilsynelatende kjører en epost-tjeneste, som er *ekstremt* treg. Når spammeren eller viruset kontakter en slik maskin, settes overføringshastigheten ned til f.eks. ett tegn i minuttet. Dette gjøres ved at liksom-tjenesten legger kunstige begrensninger på seg selv. På dette viser tar det timer eller dager å få levert et brev, og leverandører av uønsket post blir sittende uten å få gjort noe særlig. For å lokke til seg spammere kan en f.eks ha noen epostadresser til fella gjemt i skjult tekst på nettsiden sin. Ettersom overføringskapasiteten er så lav, er det praktisk talt ingen belastning ved å kjøre en tjæregrop. På internett fins beskrivelser av vellykkede tjæregroper, og hvordan spammere brukte timer på å overføre de første ordene i en melding.

Tjæregroper kan også hindre spammere i å samle opp gyldige epostadresser for senere misbruk. Spammere prøver i blant å sende post til alle vanlige navn på en tjener. Tjeneren vil avvise ugyldige navn, det skal den jo i følge smtp-standarden. Når noen meldinger går gjennom, noteres de gyldige navnene i en database. En tjæregrop vil imidlertid legge noen minutters forsinkelse på hver feilmelding om ugyldig navn. I tillegg kan den evt. godta alle navn, så spammerens lister blir fulle av feil informasjon. En spammer som tester ut en stor mengde navn vil dermed måtte vente i månedsvis, og det har de ikke tid til.

2.4.15 Vampyrer

Ikke i ordets rette forstand. Dette er programvare som driver opp nettleiekostnader for spammere. Opplegget er enkelt. Først må spam identifiseres, enten manuelt eller via tiltak som feller og spamfiltre.

[4] Tysk: *teergrube*, tjæregrop

Spam inneholder gjerne lenker. Det kan være lenker til tvilsomme nettbutikker, eller lenker til annonsebilder de vil vise i epostleseren din. «Vampyrprogrammet» laster simpelthen ned fra disse lenkene, om og om igjen. Dette fungerer fordi du betaler fast pris for en gitt båndbredde ut mot nettet, mens webtjenere gjerne må betale proporsjonalt med hvor mye som faktisk lastes ned.

Båndbredden som trengs for å betjene noen tusen kunder koster lite, men en 10–15 spamvampyrer som fyller hver sin 50 Mbps ADSL-linje med «klikk» *sammenhengende over et par dager* er ofte mer enn et lugubert lite svindelforetak har råd til å betjene. Særlig når ingen av de kunstige «klikkene» fører til kjøp.

Se `http://spamvampire.tripod.com/spam_vampire.html`

> «When you run the page for a sufficiently long time you can actually see some of the web sites going dead.»

En bør alltid tenke over etiske problemstillinger før en går til motangrep. Et angrep over nettet gir oss ikke automatisk lov til å gå til motangrep. Men en spamvampyr er egentlig ikke et angrep i ordets rette forstand. Den misbruker ikke sikkerhetshull, og overbelaster ikke nettet heller, slik et dos-angrep gjør. Vampyren laster ned materiale som er gjort tilgjengelig, men uten å skaffe inntekter.

2.5 Om applikasjonslagets protokoller kap 3.5

Vi har nå vært igjennom 3 meget sentrale applikasjoner, som benytter klient-tjener arkitektur, og har sett på mange ulike tilhørende protokoller på applikasjonslaget. Disse protokollene brukes av applikasjonene. Merk dere at applikasjoner og protokoller begge er brukerprosesser som går på maskinene så lenge applikasjonene kjører. Dette skiller applikasjonslaget fra de andre lagene i kommunikasjonsstakken, som er en del av operativsystemets kjerneprosesser som kjører hele tiden. Se figur 3.11 i læreboka.

Vi minner igjen om at selve applikasjonene befinner seg utenfor kommunikasjonsstakken.

2.6 Tilhørende kapitler i Innføring i Datakommunikasjon

Kapittel	Pensum	Navn
3	Ja	Anvendelser av Internett kap 3.3, 3.4 og 3.5
Vedlegg A	Ja	Nslookup